An Accurate Deep Learning System for the Detection of Glaucoma Using Fundus Images

Shimaa Akram[1,2], Waleed Abou Samra[3], and Ahmed H.Eltanboly[4,5], Hossam El-Din Moustafa[1]
[1]Electronics and Communications Engineering Department, Faculty of Engineering, Mansoura University, Egypt
[2]Electronics and Communications Engineering Department, Faculty of Engineering, Horus University, Egypt
[3]Department of Ophthalmology, Mansoura Ophthalmic Center, Faculty of Medicine, Mansoura University, Egypt
[4]Department of Mathematics and Engineering Physics, Faculty of Engineering, Mansoura University, Egypt
[5]Department of Mathematics, Faculty of Basic Sciences, Galala University, New Galala City, Egypt

Article Info

Article history:

Receivedmonth dd, yyyy
Revised month dd, yyyy
Accepted month dd, yyyy

Keywords:

Glaucoma Disease (GD)
Retinal images
Automated Central Retinal Image Analysis(ACRIMA)
Deep Learning
Convolutional Neural Network

ABSTRACT

Glaucoma disease (GD) is a rapidly growing consequence of Glaucoma estimates globally. The importance of accurate GD diagnosis in improving patient care and treatment outcomes has seen a significant increase in research interest in recent years. The significant advancement in Deep Learning (DL) approaches has proven to be superior to traditional detection methods. In this paper, we proposed a Deep Learning constructed using a convolutional neural network (CNN) for the automated detection of glaucoma from fundus images to distinguish between Glaucoma -affected and healthy. The important features from the input data are extracted using a CNN that has been built and trained to do so. On the ACRIMA dataset, these frameworks have been tested and trained, which contains a total of 705 images. We also evaluated how well our proposed CNN-based system performed compared to those other pretrained models (EfficientNetB0, ResNet101, Resnet50, VGG16, and InceptionV3). The aim of this paper is to comparative analysis of the performance obtained from different configurations with CNN architectures and hyper-parameter tuning. Among the considered deep learning models, the EfficientNetB0 model showed the highest accuracy of 98% for the ACRIMA fundus image dataset. We accomplished the best performance using our proposed CNN network, achieving 98.1% accuracy, 98.31% sensitivity, and 97.85% specificity for the ACRIMA dataset. Additionally, this study presents a comparative analysis of how changes in the hyperparameter of the model can affect classification performances. As a result, the proposed method is more efficient and robust compared to that described in the literature.

Corresponding Email:***shimaaakram1@gmail.com***

1. INTRODUCTION

The eye is the most important sensory organ for detecting light and transmitting signals to the brain via the optic nerve to create images, thereby providing vision and the capability to separate between colors and depth [1]. A group of progressive optic neuropathies known as Glaucoma Disease (GD) affects the optic nerve and causes a variety of structural and visual field impairments [2]. According to estimates, cataract, Glaucoma, corneal scarring (including trachoma), age-related macular degeneration, and diabetic retinopathy are the main causes of blindness, with ratios of 48%, 12%, 9%, 9%, and 5%, respectively [3]. According to another estimate, there will be 111.8 million cases worldwide by 2040 [4]. Hence, it is an issue of concern because most Glaucoma patients are unaware they have the disease, so accurate detection and treatment can help avoid visual defects and blindness [5]. Since there is still no reliable test to confirm a Glaucoma diagnosis, the treatment includes a number of hard-working tests [6]. The old diagnosis techniques performed by ophthalmologists are manual and time-consuming. Therefore, it was essential to create automated techniques for accurate Glaucoma diagnosis in order to prevent retinal damage that could lead to blindness in the future. Experts use imaging modalities such as retinal fundus photography (FP) and optical coherence tomography (OCT) [7].

Artificial Intelligence-based technologies have been rapidly blending into our daily lives. Various efforts are also being made in the medical and healthcare fields to use Artificial Intelligence (AI) developments for real-time medical treatment and diagnosis. As a result, accurate detection of Glaucoma disease using technologies for non-invasive computer-aided diagnosis (CAD) that use images in conjunction with Deep Learning (DL) is very challenging since it indicates a later stage [8]. The detection of

Glaucomatous symptoms using color fundus images is a challenging task that requires years of practice and specific expertise. To help in the detection of eye diseases, color fundus imaging has frequently been used in combination with image processing techniques [9].

Utilization of CNN, one of the most widely used Deep Learning techniques for image analysis, has grown quickly for processing medical images. CNN inputs data directly from two-dimensional images and extract discriminative features in those images and combine these features for classification purposes.

The major objective of the study is to create a completely automated technique for classifying Glaucoma using fundus images. The fundamental objective of the proposed technique is to reduce the dependency on large datasets. The network is trained on the ACRIMA dataset to verify its performance, and the results are contrasted with CNN architectures that are discussed in the literature review. The proposed method significantly improves Glaucoma accurate detection. The contributions of this paper are as follows:

1) We offered a robust CNN model to handle an unbalanced dataset.
2) We propose a supervised DL method to classify normal and Glaucoma -affected fundus images using five different pre- trained CNN models.
3) We performed extensive experiments for hyperparameter selection in order to achieve the best performance out of our proposed model.
4) We performed in-depth studies and contrasted our approach for GD detection with several existing approaches.
5) We presented a systematic review that provides a comprehensive overview of the implementation of a Deep Learning-based approach for automatic detection of Glaucoma through fundus imaging that helps ophthalmologists diagnose Glaucoma.

2. RELATED WORK

In recent years, there has been a significant amount of research done to classify the image data of Glaucoma patients. For classifying fundus images, several researchers have presented various Machine Learning (ML) and Deep Learning (DL) models. CNN is now a reliable and efficient method for Glaucoma detection. For example, as shown in Table I which presents a review of the literature on Glaucoma fundus imaging modalities, start with a Glaucoma evaluation from color fundus images using CNN was proposed by Elangovan et al. [10]. In their research, a DL network architecture (CNN) is used for the classification between images of the fundus with and without Glaucoma. Their proposed strategy for the diagnosis of Glaucoma, included an 18-layer CNN to extract the discriminative features, and they also used data augmentation. Their network was tested on five datasets separately. The highest results of the five datasets indicated that binary classification accuracy, sensitivity, specificity, and precision were 96.64%, 96.07%, 97.39%, and 97.74%, respectively, utilizing the ACRIMA dataset.

Singh et al. [11] suggested using deep convolutional neural networks for automated detection of glaucoma. Two convolutional neural network models, namely modified VGG16 and modified ResNet-50, were used for automated feature extraction and classification in the diagnosis of glaucoma. The modified ResNet-50 model obtained 93% accuracy, 85.71% specificity, and 94.94% sensitivity on the ACRIMA dataset compared to the modified VGG16 model's 94% accuracy, 80.95% specificity, and 97.47% sensitivity.

Ajitha et al. [12] created a 13-layer convolutional neural network (CNN) algorithm. In their work, 1113 fundus images from four databases have been used, divided into 70%, 20%, and 10% for train, validate, and test, respectively. Their model is implemented with two classifiers: SoftMax achieved an accuracy of 93.86%, a sensitivity of 85.42%, a specificity of 100%, and a precision of 100%. The SVM classifier achieved an accuracy of 95.61%, a sensitivity of 89.58%, a specificity of 100%, and a precision of 100%.

Ayesha et al. [13] proposed a Deep Learning model to detect Glaucoma at its initial stage. Two datasets, DRISHTI-GS and G1020 fundus images, were used. They applied some filters to improve the input data. The EfficientNet architecture was used in the proposed model. Techniques for validation were performed using the DRISHTI-GS dataset, and the accuracy, sensitivity, and specificity were 98%, 95.19%, and 94%, respectively.

Memon et al. [14] obtained a feature extraction and classification system for Glaucoma by combining all three releases of RIMONE datasets; they proposed a Deep Learning model based on a five-layer convolutional neural network with two fully connected layers. They created augmented reality by applying an affine transform to the input dataset. They evaluated their model using five-fold experiments. The average results show an overall classification accuracy of 85%, a sensitivity of 80%, and a specificity of 88%.

Around the world, Glaucoma affects millions of people. The number of people who suffer from Glaucoma -related visual impairment can be decreased with accurate Glaucoma detection. Various researchers presented various models for automated Glaucoma detection. However, as noted in the literature review, the majority of them used segmentation and feature extraction in their frameworks, which made their techniques labour-intensive and complex. This study addressed this problem and suggested a simple, reliable method that

produced promising results without depending on segmentation. We studied how various architectures performed as well as certain transfer learning techniques using pre-trained CNN models. The performance of the CNN model is evaluated using a variety of performance metrics, including sensitivity, specificity, precision, and accuracy. This inspired us to produce a CNN model for the classification of Glaucoma using public databases. The proposed CNN is trained entirely from scratch, and the ACRIMA database is used to evaluate how effectively the model performs. The results are compared with those obtained by EfficientNetB0, ResNet101, ResNet50, VGG-16, InceptionV3, and further CNN architectures described by Elangovan et al., Sanli Yi et al., Ajitha et al., Ayesha et al., and Memon et at.

TABLE I: Comparison of our proposed model's sensitivity, specificity, precision, and accuracy with existing literature techniques.

Author	Methods	Sensitivity	Specificity	precision	Accuracy
Elangovan et al. [10]	CNN	96.07%	97.39%	97.74%	96.64%
Singh Yi et al. [11]	modified VGG16	97.47%	80.95%	-	94%
	modified ResNet-50	94.94%	85.71%	-	93%
Ajitha et al. [12]	CNN with SoftMax classifier	85.42%	100%	100%	93.86%
	CNN with SVM classifier	89.58%	100%	100%	95.61%
Ayesha et al. [13]	EfficientNet	95.19%	94%	-	98%
Memon et al. [14]	CNN	80%	88%	-	85%
Proposed	**CNN**	**98.3%**	**97.85%**	**98.31%**	**98.1%**

3. METHOD AND MATERIAL

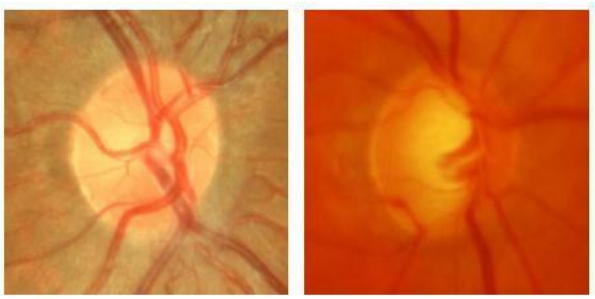
3.1. Study data set

This experimental study examines fundus images that are available to the public. The dataset is collected in Spain under the Automated Central Retinal Image Analysis project, which aims to develop an automated central retinal image analysis method. Both training and testing fundus images used for our experiments are from the ACRIMA project [15]. There are 705 fundus images in the entire dataset. The IMAGEnet ˆA capturing system and A Topcon TRC retinal camera are used to capture fundus images. Most fundus images have a 35° field of view and are centered on the optic disc. Two Glaucoma specialists with eight years of expertise annotated fundus images. Depending on the severity of the Glaucoma, the dataset is split into two classes. There are 396 Glaucoma images and 309 non- Glaucoma fundus images in each class. Figure 1 shows samples of fundus images from each class. Table II depicts the categorization of the dataset in each class.The sizes of the images range from 178 x 178 to 1420 x 1420 pixels.

TABLE II: Categorization of the Dataset

Dataset	Glaucoma	Normal	Total	Train	Test

ACRIMA	396	309	705	Glaucoma: 278 Normal: 216 Total: 494	Glaucoma: 118 Normal: 93 Total: 211



(a) (b)

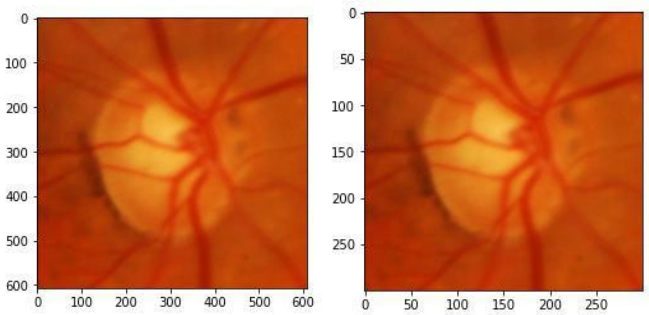
Figure1: For the ACRIMA dataset,fundusimages(a)Normal eye (b)Glaucoma effected eye.

3.2. Preprocessing and Data Augmentation

Fundus images are captured under varying conditions of deterioration, such as low variation brought on by insufficient illumination of the optical equipment. For the enhancement of fundus images, image enhancement techniques are frequently utilized to improve or correct the pixel distribution across a wide range of intensities. So, we initially adopt the min-max normalization method [16] to remove the influence of illumination and reduce machinery and impulse noise to modify the values of the image's pixel intensities. For image normalization, the pixel values are rescaled to [-1, 1] using a pixel-wise multiplication factor of 0/255, which is applied to rescale the pixel values as follows:

$$P\hat{}N = \left(1 - p_{Min}^{\hat{}}\right) + \frac{p_{Max}^{\hat{}} - p_{Min}^{\hat{}}}{Max - Min} p_{Min}^{\hat{}} \quad (1)$$

where P and P^N indicate input and the normalized fundus picture , respectively, Max=255; and Min=0 indicate the pixel intensity spectrum of the input fundus picture , and p_Max^T^ and p_Min^T^ are the intensity range of normalized picture. Before training, the grayscale pictures are resized to 300x300 pixels after normalization to fit the input layer's dimensions for the model. Since the images were grayscale, more channels were created by tripling the values of each pixel. Figure 2 shows the sample fundus images before and after resizing.



(a) (b)

Figure2.SamplesofafundusimageBeforeandafterresizingare shown in(a)and(b), respectively.

Then for the Data Augmentation, In order to make a better generalized DL, the size of the data used for training the model should increase to avoid model overfitting problems. The availability of a large number of scans and the unbalanced dataset, specifically for Glaucoma diagnosis, are significant issues in research that lead to overfitting issues that influence the model's accuracy. Therefore, data augmentation [17] is the technique used for reliable predictions. The target is to reproduce the variations seen in the captured images by applying a few transformations to the training data. In our model, we use the augmentation approach in each fundus image of the training set to generate new ones. Some popular augmentation techniques that are frequently used are grayscales, horizontal flips, vertical flips, random crops, brightness, contrast, zoom, translations, rotations, and much more. We can rapidly increase the number of training images by using just a few of these transformations on our training data, and we can build a very robust model. These transformations were tried, but they were ineffective for our proposed model and reduced the number of findings. As a result, we were satisfied to use only rotation, the nearest-neighbour fill mode, random zoom, and horizontal and vertical shift augmentation. For the data augmentation, we choose to randomly rotate some training images by 30 degrees, randomly zoom some training images by 20%, randomly shift images horizontally by 20% of the width, and randomly shift images vertically by 20% of the height. Figure 3 demonstrates samples of data-augmented images.

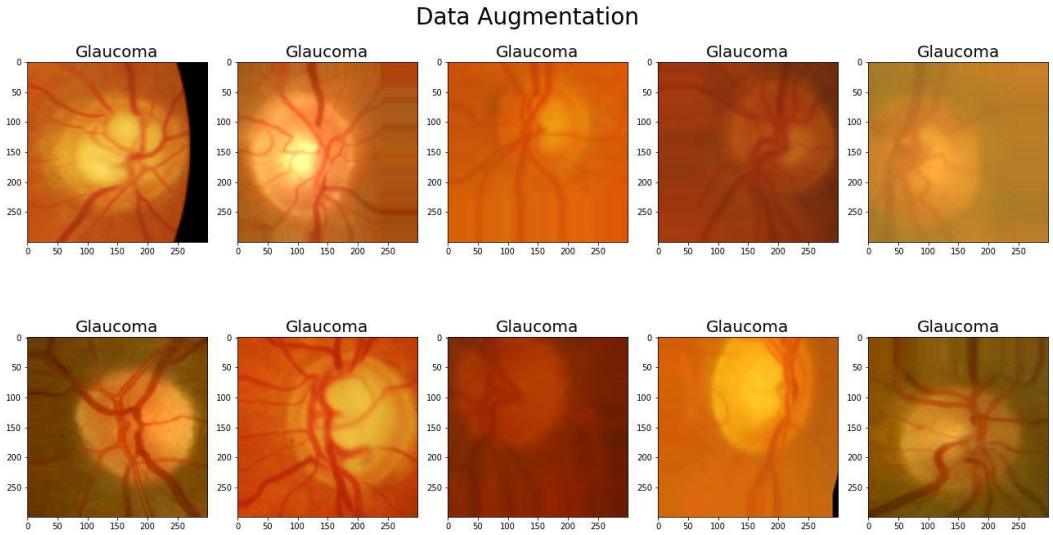


Figure3.Examplesof train data-augmentedimages.

All preprocessing operations were carried out in Python (Python Software Foundation; version 2.3), utilizing the Tensor- Flow backend and Scikit-Learn and Keras libraries, which are available at https://scikit-Learn.org/stable/ and https://keras.io, respectively.

3.3. Proposed Deep Learning Architecture

The proposed approach for diagnosing Glaucoma from fundus images using CNN is shown in Figure 4. It consists of two main blocks: preprocessing and classification. Image resizing, normalization, and data augmentation are employed in the preprocessing stage. Following the preprocessing stage, the images are resized to a 300 x 300 matrix size and fed into CNN's proposed model for testing and training.

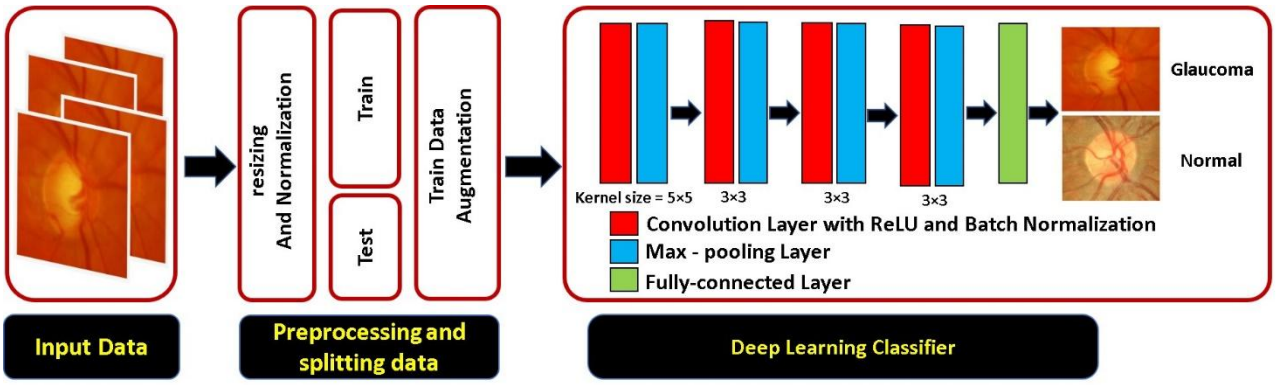


Figure4. CNN's proposedmodelarchitectureusedforGlaucomadetectionfromfundusimages

This input matrix size was chosen primarily to decrease computing complexity and speed up the training process. Table III includes each component of the proposed CNN architecture in detail. Input layers, convolutional layers, fully connected layers, and output layers form the four main parts of our model. The input layers of the model receive the tuned data set. CNN's proposed model uses a variety of layers, including convolution, ReLU, batch normalization, and max pooling, to extract the required features out of the fundus images. In this research, the convolutional (Conv2D) layer is used as the first. 128 filters are specified for the first conv2D layer, 64 filters for the second, 32 filters for the third, and 16 filters for the last layer, the pooling (MaxPool2D) layer of CNN is the second essential layer. Basically, this layer acts as a down sampling filter. It examines the two neighboring pixels and selects the highest value. We chose the pooling size to be 2×2 to increase the model's computational efficiency, as well as we also used a Batch Normalization (BN) layer. The fundamental concept behind the Deep CNN architecture is convolution operation. The convolution layer's primary function is to extract unique features from fundus images. The convolution layer is referred to in these equations [18]:

$$X_j^l = f\left(\sum_{i \in M_j} X_i^{l-1} * K_{ij}^l + b_j^l\right) \quad (2)$$

Where,X_j^l representsj^{th}activation mapofthecurrentl^{th}layer, X_i^{l-1}isthej^{th}activation mapforthelayerbefore$(l-1)^{th}$,whereM_j representsaselectionofinput activation maps.$b_j^l$and K_{ij}^l arebiasandweight vectors. Convolution operationsareperformedusingthe$*$operator;theinputmapswillbe convolved with various kernels foraparticularoutputmap,and$f$ denotestheactivationfunction.

Weuse2Dconvolutionallayerswithakernelsizeof5×5forthefirstblockand3×3fortheotherblocks.And the Rectified Linear Unit (ReLU) activation function for all blocks, the rectifier activation function, is used to add nonlinearity to the network; it is considered one of the most popular and efficient activation functions that is frequently used in DL. A2DconvolutionallayerandaMax-poolinglayerarethe following two layers.Thepooling methodthat ismostfrequentlyusedismaxpooling. By convolutionally layeringfiltersoverit,itaccumulatestheconvolutionallayer'sfeatures.Asthenumberofparametersisminimized,th ecomputationalcostisdecreased,aidingintheavoidanceofoverfitting.Theconvolutionallayers'outputissubseque ntlytransformedintoalong1-dimensionalfeaturevectorbyaflatteninglayer.It combines all the found local features of the previous convolutional layers. Thisflattenlayeroutputisfedtothefullyconnecteddenselayerwith ReLUactivation,followedbythedropoutlayer(p =0.5).Eachactivationunitinthelayeraboveisconnectedtoeachinputneuroninalayerthatisfullyconnected.Conside ringthelabelsthathavebeengiventotheimages,thislayercategorizesthem.Thefinalneuronisamultidenseneuron. Whensigmoidactivationisprovidedasthemodeloutput,basedontheclassificationlabelassignedtotheneurons,thes igmoidactivationfunctionclassifiestheneurons.Finally,whethertheinputfundusimageisGlaucomapositiveornor malisindicatedinthefinallayer.Duringthetraining,thebinary-cross-

entropylossfunction,whichiscommonlyusedforbinaryclassclassificationtasks,wasusedfor100epochsatalearnin grateof0.0001andabatchsizeof32.

TABLE III: Layer-by-layer description of the suggested CNN architecture

Layer	Size	Kernels Size	Activation	Padding
Convolution Maxpooling Batch Normalization	128 (2,2)	(5,5)	'relu'	'Same'
Convolution Maxpooling Batch Normalization	64 (2,2)	(3,3)	'relu'	'Same'
Convolution Maxpooling Batch Normalization	32 (2,2)	(3,3)	'relu'	'Same'
Convolution Maxpooling Batch Normalization	16 (2,2)	(3,3)	'relu'	'Same'
Optimizer	Rmsprop			
Learning Rate	0.0001			
Batch Size	32			
No. of Epochs	100			
Total Parameters	10,138,593			
Trainable Parameters	10,138,145			
Non-Trainable Parameters	448			

For comparative analysis, more pre-trained models are also used to see their performance with Glaucoma image classification. Each one of the five pre-trained models (EfficientNetB0, ResNet101, Resnet50, VGG16, and InceptionV3) are tuned and developed to detect Glaucoma cases from the same ACRIMA dataset separately. The pre-trained model of these CNN architectures is used to extract features and outputs are fed to a 2D Global average Pooling layer, two Batch Normalization layers, a Dense layer, and two dropout layers (p = 0.5). With the same architecture as our proposed model, the output is feed to the fully connected layer with dropout. Figure 5 depicts the architecture of each one of the five pre-trained models used for Glaucoma detection from fundus images.

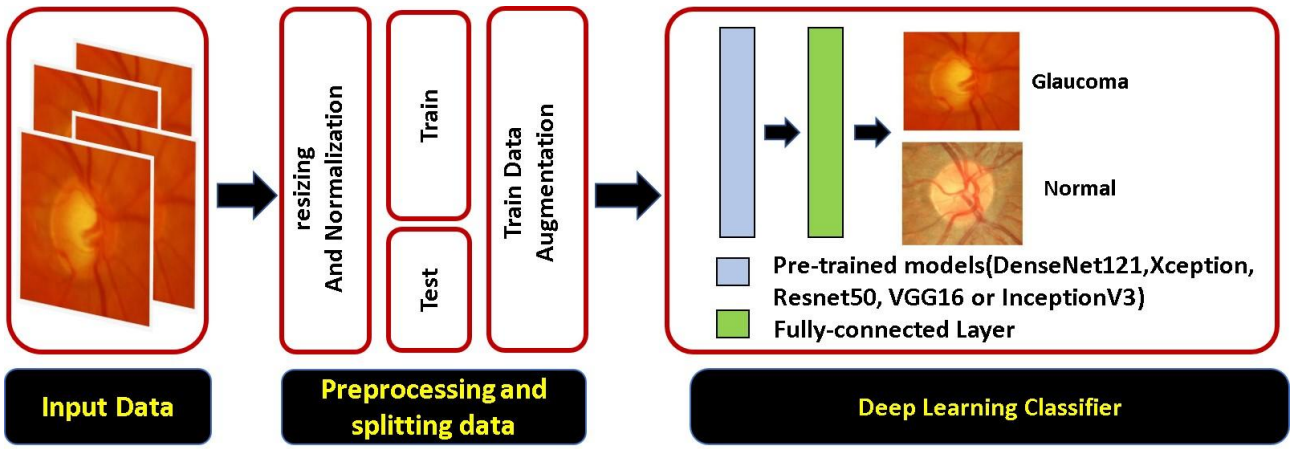


Figure 5.Thearchitectureofpre-trainedmodelusedforGlaucomadetectionfromfundusimages

4. RESULTS AND DISCUSSION

4.1. Hyperparameter selection

The number of epochs, starting learning rate, batch size, input image size, and number and size of the filters in the convolutional layer must all be carefully selected for the model to perform better. The initial learning rate, the batch size, and the size of the input image all have a big effect on how efficiently the network performs [19]. First, some parameters are fixed such as split ratio, image size, number of epochs, and optimizer as 30%–70% for network testing and training, 300 × 300, 100, and Adam, respectively. After a large no of iteration we fixed these values, then batch size and initial learning rate are variable ,where the number of images utilised in the gradient estimation procedure is known as the batch size [20] and smaller values of learning rate result in longer training time [21], then, to set the range of parameters for learning rate and batch size that provide better classification accuracy, Some iterations are performed, The learning rate begins at 0.01 and gradually drops to 0.0001 and the batch size is started at 16 and then gradually increases to 100. While the splitting ratio is 30:70, the image size is 300 × 300, 100 epochs, and the Adam optimizer is used.

Table. IV shows the variation in classification sensitivity, specificity, precision, recall, and accuracy with different learning rates and batch sizes at fixed value of split ratio, image size, number of epochs, and optimizer as 30%–70% for network testing and training, 300 × 300, 100, and Adam, respectively for the ACRIMA dataset. It is observed that the highest classification result is obtained when the initial learning rate is 0.0001 and the batch sizes is 32, as shown in Figure 6.

TABLE IV: Performance variation with batch size and learning rate for Adam optimizer

	Batch Size	Learning Rate	sensitivity	specificity	precision	recall	accuracy
70%:30% 300 × 300 100 epochs Adam	16	0.01	0.9322	0.9677	0.9735	0.9322	0.95
		0.001	0.9068	0.957	0.964	0.9068	0.93
		0.0001	0.9661	0.9785	0.9828	0.9661	0.97
	32	0.01	0.9576	0.9462	0.9576	0.9576	0.95
		0.001	0.9322	0.9785	0.9821	0.9322	0.95
		0.0001	0.9746	0.9677	0.9746	0.9746	0.97
	64	0.01	0.6949	0.9892	0.988	0.6949	0.82
		0.001	0.9407	0.9677	0.9737	0.9407	0.95
		0.0001	0.9576	0.9677	0.9741	0.9576	0.96
	100	0.01	0.9407	0.4624	0.6894	0.9407	0.73
		0.001	0.9237	0.9785	0.982	0.9237	0.95
		0.0001	0.9831	0.8387	0.8855	0.9831	0.92

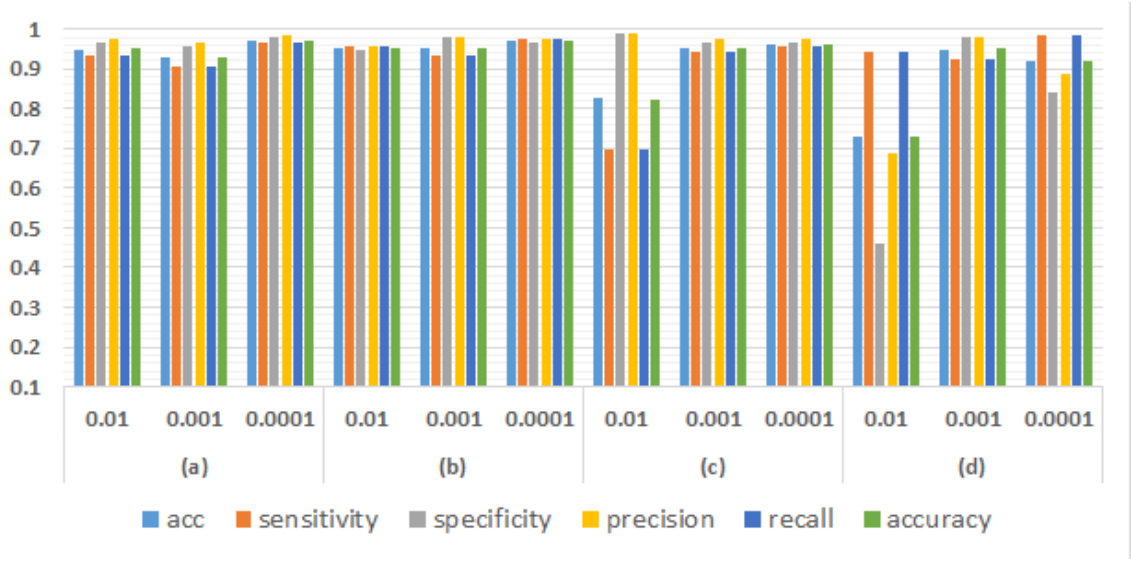


Figure6.Comparisonofclassification accuracy forthe ACRIMA datasetwithbatchsizeandinitiallearningrate,where(a),(b),(c),and(d)are values ofbatchsizeof16,32,64,and100, respectively.

In the next stage, fine tuning of the optimizer is performed at a 16, 32 batch size, and the initial learning rate has two values: 0.001 and 0.0001, we do some iteration with different optimizer types. Different optimizer types are used in this experiment, such as Adam, Rmsprop, and Adagrad. It is noted the classification's accuracy for each initial learning rate by varying the optimizer type among Adam, Rmsprop, and Adagrad. Table. V describes the obtained results. It is observed that for ACRIMA dataset, when the batch size is 32, the initial learning rate is 0.0001, and optimizer is Rmsprop the greatest classification accuracy is obtained. Figure 7 show the accuracy of the proposed model for (0.001 and 0.0001) learning rate, and 32 batch size for each optimizer (Adam, Rmsprop, and Adagrad).

TABLE V: Performance variation with batch size, learning rate and optimizer

	optimizer	Batch Size	Learning Rate	sensitivity	specificity	precision	recall	accuracy
70%:30% 300 × 300 100 epochs	Adam	16	0.001	0.9068	0.957	0.964	0.9068	0.93
			0.0001	0.9661	0.9785	0.9828	0.9661	0.97
		32	0.001	0.9322	0.9785	0.9821	0.9322	0.95
			0.0001	0.9746	0.9677	0.9746	0.9746	0.97
	Rmsprop	16	0.001	0.9407	0.9677	0.9737	0.9407	0.95
			0.0001	0.9661	0.957	0.9661	0.9661	0.96
		32	0.001	0.9576	0.9677	0.9741	0.9576	0.96
			0.0001	0.9831	0.9785	0.9831	0.9831	0.981
		16	0.001	0.9492	0.9677	0.9739	0.9492	0.96

	Adagrad		0.0001	0.9746	0.9677	0.9746	0.9746	0.97
		32	0.001	0.9492	0.9677	0.9739	0.9492	0.96
			0.0001	0.9237	0.957	0.9646	0.9237	0.94

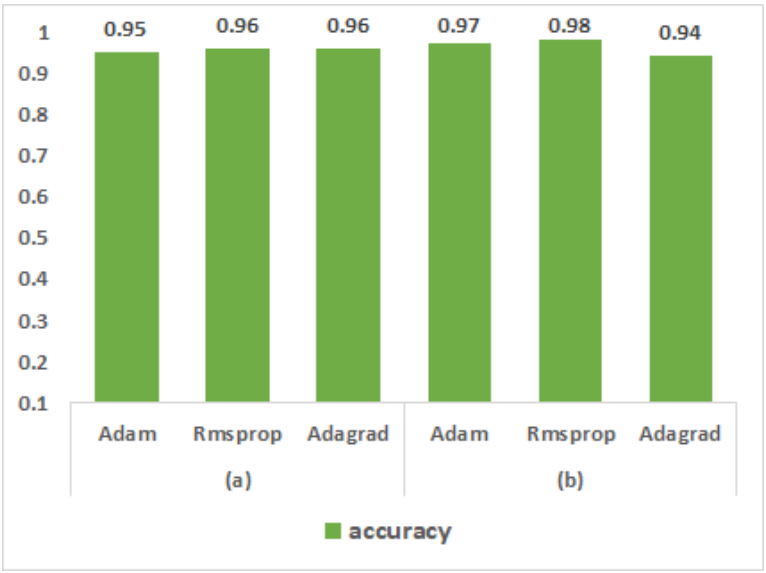


Figure7. Variation ofperformancewith(a)0.001LearningRate,32Batchsizebatchsize(b)0.0001LearningRate,32Batchsizeandoptimizer

RMSProp(rootmeansquarePropogation)isoneofthemostinfluentialadaptivealgorithmsfortrainingDeepneuralnetworks,itwascreatedbyGeoffreyHinton[22].TheRMSpropupdateisdoneasfollows:

$$v_t = \rho * v_{t-1} + (1-\rho) * g_t^2 \quad (3)$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{v_t+\epsilon}} * g_t \quad (4)$$

Where $v_t$isanexponentialaverageofthesquareofthegradient,g_t isSumofthesquaresofthepreviousgradientswithrespecttoallparameters$w_t$alongitsdiagonal,ρisUsuallyaround0.9andηisLearningrate,usuallyaround0.001.

4.2. Performance evaluation

This work implements and compares several CNN and pre-trained architectures for Glaucoma detection using fundus images. The CNN architecture is trained using data collected over 100 epochs with 32 batch sizes per epoch. Data for network testing and training is split into 30% and 70% of all experiments, respectively. The Keras framework is used in the implementation. Deep neural network training requires parallel processing. Thus, to implement the classifier's testing and training, we used the open-source packages Python version 3.0 and Kaggle (with 16 GB of RAM and NVIDIA TESLA P100 GPUs). We utilized modules from the Tensorflow library called Keras to develop our proposed model.
Different metrics, such as classification accuracy, sensitivity, specificity, precision, and recall, are used to evaluate the networks' performance. The model evaluation determines how effectively a proposed model generalizes to the data. To assist us in discriminating between several pretrained models. Below is a description of the metrics' equations.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (5)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (6)$$

$$Specif\ icity = \frac{T\,N}{T\,N + F\,P} \quad (7)$$

$$Precision = \frac{T\,P}{T\,P + F\,P} \quad (8)$$

$$Recall = \frac{T\,P}{T\,P + F\,N} \quad (9)$$

Where" true positives"(TP)referstothetotalnumberofGlaucomaimagesthatthemodelaccuratelypredicted.Thetotalnumberofnormalimagesthatthemodelcorrectlyidentifiedasbeing" truenegatives"(TN).Thetotalnumberofnormalimagesthatthemodelincorrectlyidentifiedas Glaucomaisknownas" false positives"(FP).ThetotalnumberofGlaucomaimagesthatthemodelincorrectlyidentifiedasnormalisreferredtoas" falsenegatives"(FN).

Inadditiontoquantitativeparameters,accuracyandlossgraphsagainstthenumberofepochsareusedtoqualitativelyevaluatenetworkperformance.AsshownFigure8 theoverallaccuracyofourproposedCNNmodelis98.1%whereasthepretrainedmodelsachievedtheaccuracyof98%,97%,93%,95%and96%respectively,byEfficientNetB0,ResNet101,ResNet50,VGG16andInceptionV3.Itdemonstratesclearlythattheproposedmodeloutperformstheotherpretrainedmodelsintermsofaccuracy.Figure9compresthelossforeachmodelusingtheACRIMADataset.Metricslikeaccuracy,sensitivity,specificity,precision,andrecallprovideaclearerunderstandingofthemodels'performance.Thetestdatasetthattheconfusionmatrixgeneratesisusedtocalculatetheseperformancemetrics.Figure10displaystheproposedmodelconfusionmatricesinadditiontothoseofotherpretrainedmodels.Thisishighlyusefulinidentifyingwhichclassesaremislabeledmorefrequently.Confusionmatricesareusedformodelmanagementandmonitoringinadditiontomodelevaluation.Table.VIcomparesandcontrastsoursuggestedmodelwiththeotherpretrainedmodels.

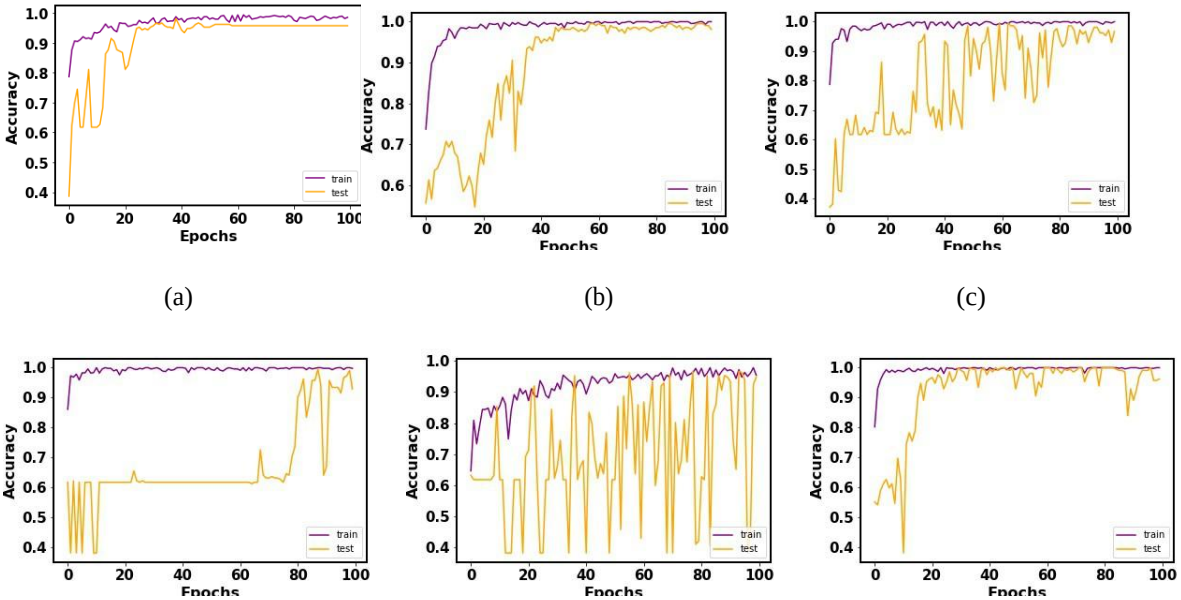
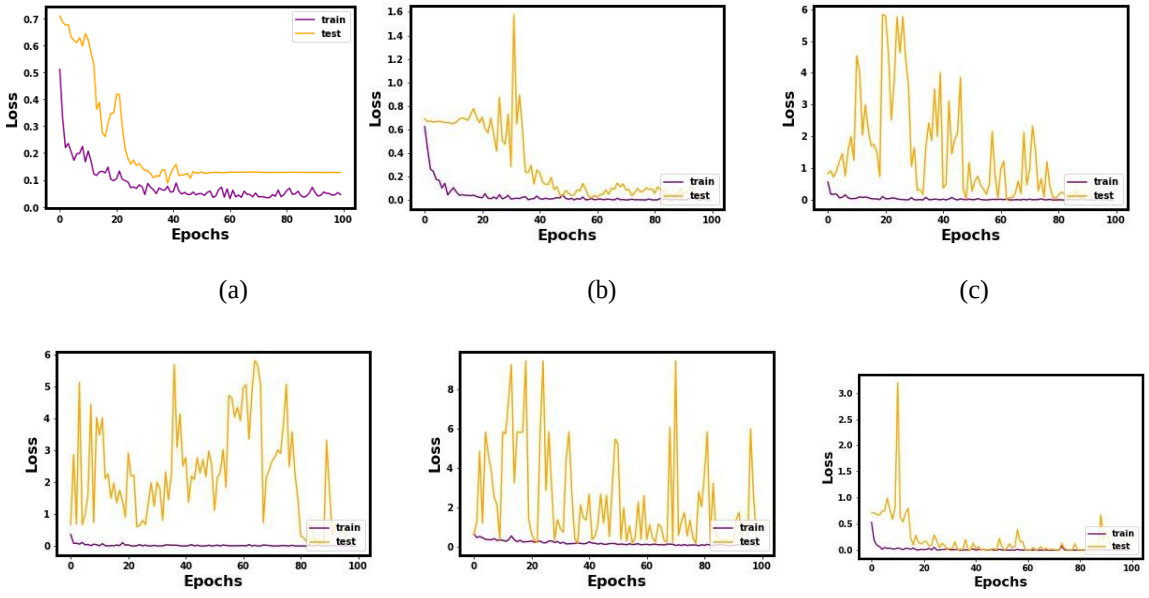


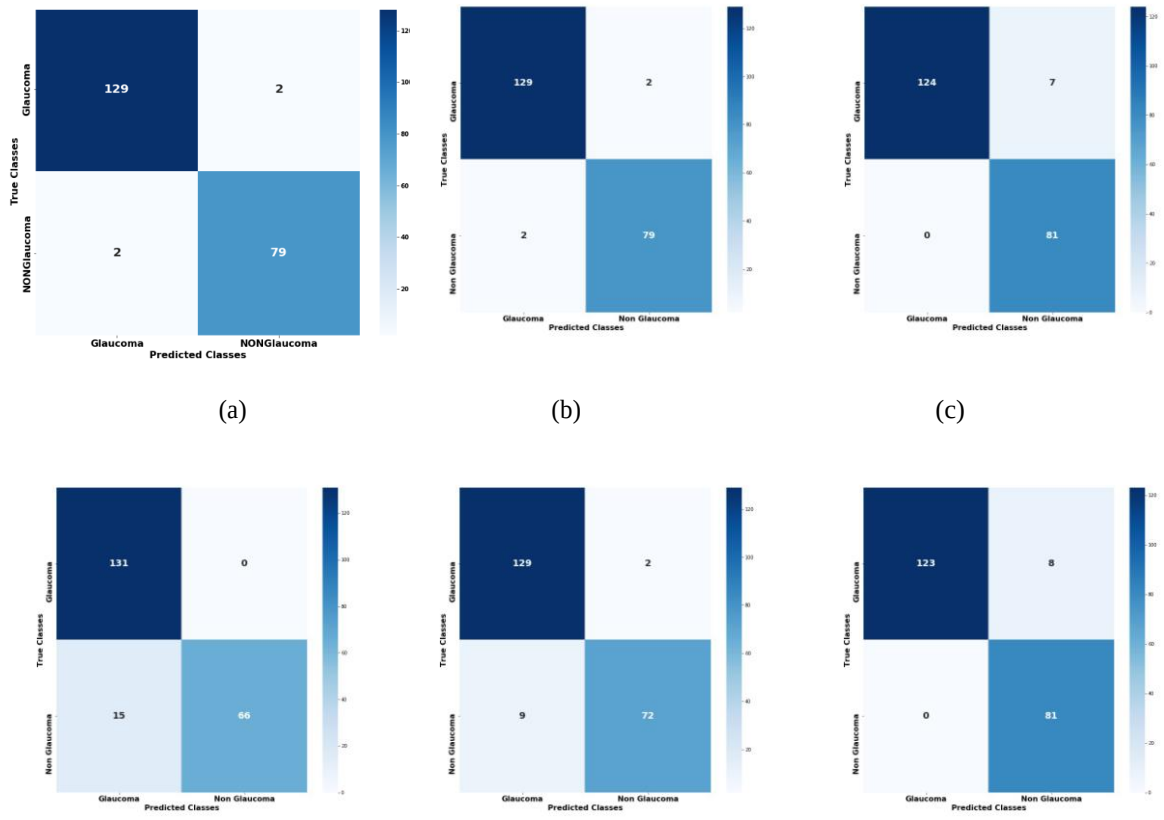
Figure8. Accuracy fortrainand test inrelationtothenumberofepochsforthe following models(a)proposedmodel (b) EfficientNetB0 (c) ResNet101(d)Resnet50(e) VGG16, and(f)InceptionV3models.



(a) (b) (c)

(d) (e) (f)

Figure9.Lossfortrainandtestinrelationtothenumberofepochsforthe following models (a)proposedmodel(b) EfficientNetB0 (c)ResNet101(d)Resnet50(e) VGG16, and(f)InceptionV3models.



(a) (b) (c)

(d) (e) (f)

Figure10.Confusionmatricesforthe following models(a)proposedmodel(b) EfficientNetB0 (c)ResNet101(d)Resnet50(e) VGG16, and(f)InceptionV3models.

TABLE VI: Comparison of the proposed model to previous CNNs that use transfer learning for data split by 30%-70% test and train, respectively.

Model	Sensitivity	Specificity	precision	recall	accuracy
EfficientNetB0 [23]	0.9847	0.9753	0.9847	0.9847	0.98
ResNet101 [24]	0.9466	1.0000	1.0000	0.9466	0.97
Resnet50 [25]	1.0000	0.8148	0.8973	1.0000	0.93
VGG16 [26]	0.9847	0.8889	0.9348	0.9847	0.95
InceptionV3 [27]	0.9389	1.0000	1.0000	0.9389	0.96
Proposed	0.9831	0.9785	0.9831	0.9831	0.981

4.3. CNN algorithm comparison

Number of different CNN architectures are found in the literature. Table I compares the performance of the suggested model with other well-known models developed by other researchers. Although this is a promising outcome for our suggested methodology, some researchers might be able to obtain even better outcomes with binary classification. On the other hand, this work demonstrates how the initial learning rate, batch size, and type of optimizer affect the effectiveness of the model's performance.

5. CONCLUSION

The prevention of blindness is significantly assisted by the accurate detection of Glaucoma. The three main considerations in any disease detection process, particularly Glaucoma, are time, cost, and accuracy. In order to solve these problems, a CNN-based model for the detection of Glaucoma from fundus images is suggested in this paper. The ACRIMA dataset is used to train the CNN models and contains a total of 705 fundus images split into two classes. The proposed model outperforms the other five pretrained models discussed. The proposed model has a 98.1% accuracy. This model also evaluates the results against other well-known studies in the area. Last but not least, CNN has excellent prospects of detecting Glaucoma with a minimum of effort, time, resources, and costs. The suggested model may definitely significantly contribute to the accurate and quick detection of Glaucoma, thereby reducing overall testing costs and time due to its high accuracy. In the future, performance improvement can be achieved by exploring more efficient data augmentation approaches and training the model on larger datasets. Further, research can be focused towards the development of a multistage glaucoma classification model, as it would enable better risk assessment

REFERENCES

[1] R. L. Gregory, Eye and brain: The psychology of seeing. McGraw-Hill, 1973.
[2] J.Kountouras, C. Zavos, and D. Chatzopoulos, "Primary open-angle glaucoma: pathophysiology and treatment," The Lancet, vol. 364, no. 9442, pp.1311–1312, 2004.
[3] A. Foster and S. Resnikoff, "The impact of vision 2020 on global blindness," Eye, vol. 19, no. 10, pp. 1133–1135, 2005.
[4] Y.-C. Tham, X. Li, T. Y. Wong, H. A. Quigley, T. Aung, and C.-Y. Cheng, "Global prevalence of glaucoma and projections of glaucoma burden through 2040: a systematic review and meta-analysis," Ophthalmology, vol. 121, no. 11, pp. 2081–2090, 2014.

[5] D. L. Budenz, K. Barton, J. Whiteside-de Vos, J. Schiffman, J. Bandi, W. Nolan, L. Herndon, H. Kim, G. Hay-Smith, J. M. Tielsch, et al., "Prevalence of glaucoma in an urban west african population: the tema eye survey," JAMA ophthalmology, vol. 131, no. 5, pp. 651–658, 2013.
[6] D. Gupta, Glaucoma diagnosis and management. Lippincott Williams & Wilkins, 2005.
[7] M. D. Abra`moff, M. K. Garvin, and M. Sonka, "Retinal imaging and image analysis," IEEE reviews in biomedical engineering, vol. 3, pp. 169–208, 2010.
[8] A.Shoukat and S. Akbar, "Artificial intelligence techniques for glaucoma detection through retinal images: State of the art," Artificial Intelligence and Internet of Things, pp. 209–240, 2021.
[9] P. M. Burlina, N. Joshi, M. Pekala, K. D. Pacheco, D. E. Freund, and N. M. Bressler, "Automated grading of age-related macular degeneration from color fundus images using deep convolutional neural networks," JAMA ophthalmology, vol. 135, no. 11, pp. 1170–1176, 2017.
[10] P.Elangovan and M. K. Nath, "Glaucoma assessment from color fundus images using convolutional neural network," International Journal of Imaging Systems and Technology, vol. 31, no. 2, pp. 955–971, 2021.
[11] Singh, Sukhpal, NitigyaSambyal, and Ashutosh Aggarwal. "Automated Glaucoma Detection Using Deep Convolutional Neural Networks." (2023).
[12] S.Ajitha, J. D. Akkara, and M. Judy, "Identification of glaucoma from fundus images using deep learning techniques," Indian Journal of Ophthalmology, vol. 69, no. 10, p. 2702, 2021.
[13] A.Shoukat, S. Akbar, and K. Safdar, "A deep learning-based automatic method for early detection of the glaucoma using fundus images," in 2021 International Conference on Innovative Computing (ICIC). IEEE, 2021, pp. 1–6.
[14] I. Memon, A. A. Ursani, M. A. Bohyo, and R. Chandio, "Automated diagnosis of glaucoma using deep learning architecture," Eng. Sci. Technol. Res.J, vol. 3, pp. 58–62, 2019.
[15] A. Diaz-Pinto, S. Morales, V. Naranjo, T. Ko¨hler, J. M. Mossi, and A. Navea, "Cnns for automatic glaucoma assessment using fundus images: an extensive validation," Biomedical engineering online, vol. 18, no. 1, pp. 1–19, 2019.
[16] D.Borkin, A. Ne´methova´, G. Michal'cˇonok, and K. Maiorov, "Impact of data normalization on classification model accuracy," Research Papers Faculty of Materials Science and Technology Slovak University of Technology, vol. 27, no. 45, pp. 79–84, 2019.
[17] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," Journal of big data, vol. 6, no. 1, pp. 1–48, 2019. [18] J.Bouvrie, "Notes on convolutional neural networks," 2006.
[19] Y.Bengio, "Practical recommendations for gradient-based training of deep architectures," in Neural networks: Tricks of the trade. Springer, 2012, pp.437–478.
[20] I. Kandel and M. Castelli, "The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset," ICT express, vol. 6, no. 4, pp. 312–315, 2020.
[21] F. He, T. Liu, and D. Tao, "Control batch size and learning rate to generalize well: Theoretical and empirical evidence," Advances in Neural Information Processing Systems, vol. 32, 2019.
[22] T. Tieleman, G. Hinton, et al., "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," COURSERA: Neural networks for machine learning, vol. 4, no. 2, pp. 26–31, 2012.
[23] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in International conference on machine learning. PMLR, 2019, pp. 6105–6114.
[24] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in European conference on computer vision. Springer, 2016, pp.630–645.
[25] ——, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp.770–778.
[26] K.Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.
[27] C.Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826.