




# DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases

Akash Tayal<sup>1</sup> · Jivansha Gupta<sup>1,2</sup> · Arun Solanki<sup>2</sup> · Khyati Bisht<sup>1</sup> · Anand Nayyar<sup>3</sup>  · Mehedi Masud<sup>4</sup>

Received: 15 October 2020 / Accepted: 24 February 2021

© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

## Abstract

Artificial intelligence has the potential to revolutionize disease diagnosis, classification, and identification. However, the implementation of clinical-decision support algorithms for medical imaging faces challenges with reliability and interpretability. This study presents a diagnostic tool based on a deep-learning framework for four-class classification of ocular diseases by automatically detecting diabetic macular edema, drusen, choroidal neovascularization, and normal images in optical coherence tomography (OCT) scans of the human eye. The proposed framework utilizes OCT images of the retina and analyses using three different convolution neural network (CNN) models (five, seven, and nine layers) to identify the various retinal layers extracting useful information, observe any new deviations, and predict the multiple eye deformities. The framework utilizes OCT images of the retina, which are preprocessed and processed for noise removal, contrast enhancements, contour-based edge, and detection of retinal layer extraction. This image dataset is analyzed using three different CNN models (of five, seven, and nine layers) to identify the four ocular pathologies. Results obtained from the experimental testing confirm that our model has excellently performed with 0.965 classification accuracy, 0.960 sensitivity, and 0.986 specificities compared with the manual ophthalmological diagnosis.

**Keywords** Biomedical imaging · Deep learning · Disease detection · Image processing · Optical coherence tomography (OCT)

✉ Anand Nayyar  
anandnayyar@duytan.edu.vn

Akash Tayal  
akashtayal@yahoo.com

Jivansha Gupta  
jivanshagupta@gmail.com

Arun Solanki  
ymca.arun@gmail.com

Khyati Bisht  
khyatibisht4998@gmail.com

Mehedi Masud  
mmasud@tu.edu.sa

<sup>1</sup> Indira Gandhi Delhi Technical University for Women, New Delhi, India

<sup>2</sup> Gautam Buddha University, Greater Noida, India

<sup>3</sup> Duy Tan University, Da Nang, Viet Nam

<sup>4</sup> Department of Computer Science, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia

## 1 Introduction

In the past few decades, we have seen the advent of computer science in disease detection and diagnosis for biomedical sciences. Artificial intelligence (AI) has revolutionized the disease diagnosis and the anatomization process by performing the classification steps, which were time-consuming and tedious for the experts [1–3]. The medical field has been accepting and adopting AI because of the rampant increase in applications employing AI-based technologies in recent times and the physicians' demand to operate with reduced errors, mishaps, and misdiagnosis. Many AI and subset DL networks are useful in medical image processing for prognosis and diagnosis of various ailments (e.g., breast cancer, lung cancer, and brain tumor), which are tedious and prone to human error if manually performed. Medical images are processed using these DL methods to solve various tasks, such as prediction, segmentation, and classification, consequently accurately bypassing human abilities.

The scope of AI is significant in retinal disease diagnosis and procedure. The mechanism requires precise, correct

identification, and extraction of ocular layers, making it easy for ophthalmologists to focus on the treatment. In this study, the benefits of AI have been leveraged to classify and identify an ocular disease; the retina's structural complexity makes it inconvenient and time-consuming for accurate evaluation by the expert. The retina is situated on the inside back wall of the eye and is responsible for sending light and images back to the brain. When the light focuses on the retina instead of elsewhere, normal vision is observed. A person with normal vision can see objects at near and far distances. Vision loss, myopia, and macular degeneration can occur if the retinal layer is affected. Some commonly known retinal diseases are choroidal neovascularization (CNV), drusen, diabetic retinopathy, and diabetic macular edema (DME).

Employing advanced AI techniques in medical diagnosis and image detection has brought the much-needed headway in medical science. The automated detection of retinal diseases involves a preprocessing with image quantization, segmentation and sampling procedures, training of neural networks with the vast data, and analysis of statistics. The researchers are currently focusing on improving the accuracy of classification and identification of the disease, reducing the computational time and memory utilization, proper segmentation of ocular layers, and minimizing computational complexity.

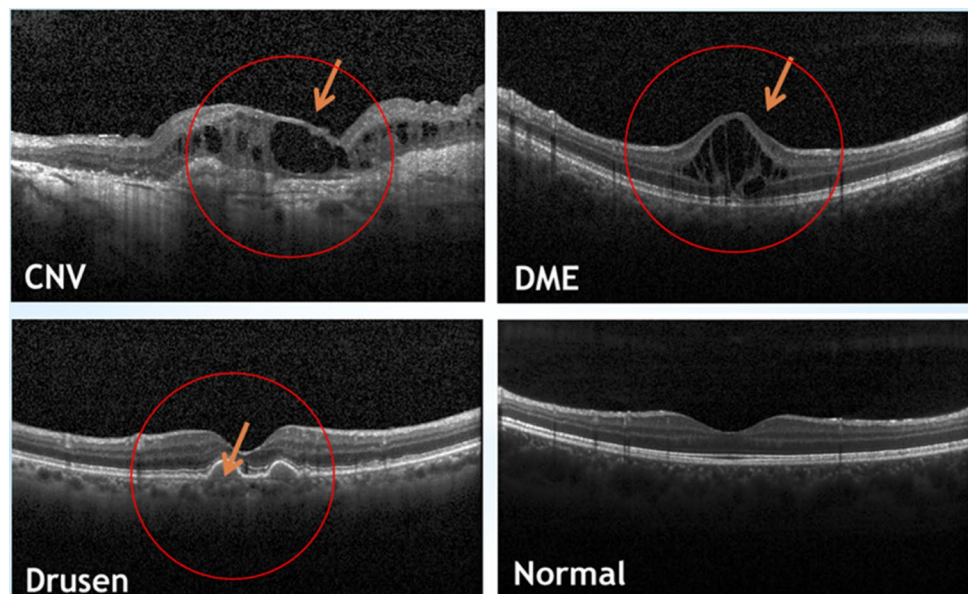
Optical coherence tomography (OCT) is a modern non-invasive imaging technique built on low coherence interferometry. This technique can reconstruct (tomographic) sectional images with a high depth resolution of the object under study using the projected light beams. The measured dimension of the thickness of the retinal layers helps early detection of pathologies and disease diagnosis. The two

types of OCT are time-domain (TD) OCT and spectral-domain (SD) OCT. The TD-OCT is used to produce the 2D scans of the given sample internal edifice. The SD-OCT is said to be 50 times quicker than the conventional TD-OCT technique. Furthermore, SD-OCT is 100 times faster than the ultra-high-resolution OCT. The SD-OCT scan has more clarity and high quality compared with the TD-OCT systems.

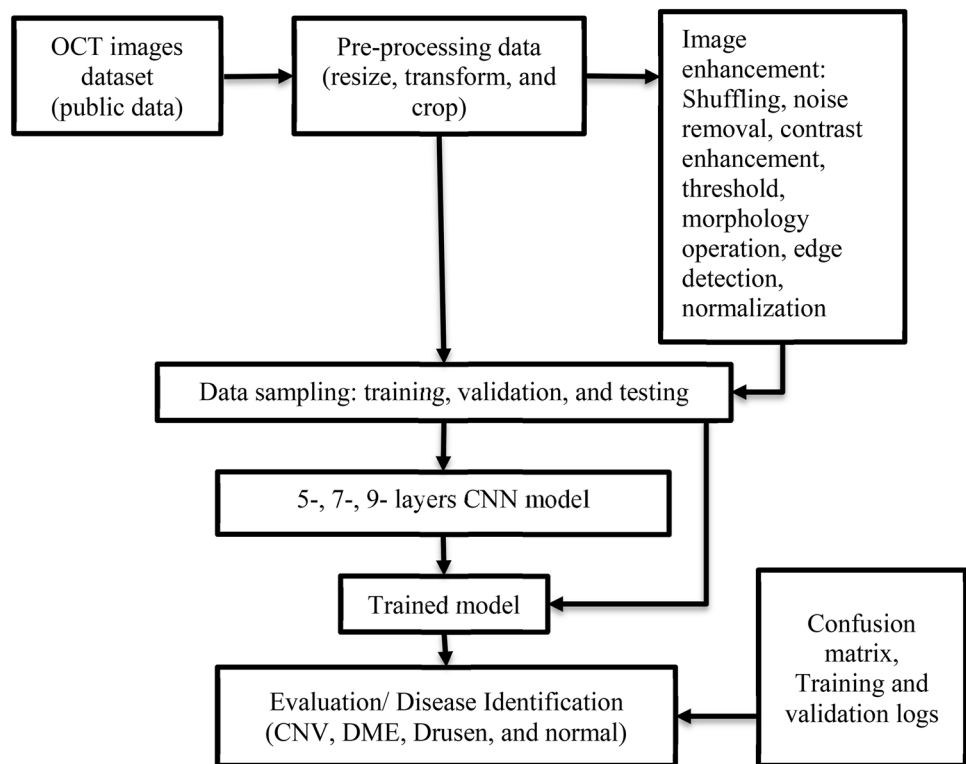
Figure 1 shows the OCT scans of CNV, drusen, DME, and normal retina. The growth of new blood vessels through the Bruch membrane in the choroid layer describes CNV, and it causes vision loss. The accretion of fluid in the retina's macula part forms DME. The yellow deposits composed of lipids, protein, and calcium salts under the retina characterizes drusen. The risk of developing age-related macular generation (AMD) increases in drusen.

This research has evaluated and analyzed a large dataset of retinal OCT images available in the public domain to classify the normal retina and three ocular pathologies (CNV, drusen, and DME) for accurately detecting results of significant pathological structures. Figure 2 describes the framework of the proposed methodology for detecting ocular degeneration using OCT images of the retina. The OCT images of the retina are preprocessed and enhanced for noise removal using a median filter. Contrast Limited Adaptive Histogram Equalization (CLAHE) for contrast enhancements is used, followed by morphological operation, thresholding, and contour-based edge detection for retinal layer extraction. This image dataset is analyzed using three different Convolution Neural Network (CNN) models (of five, seven, and nine layers) to identify the four ocular pathologies.

**Fig. 1** OCT scans of diseased and normal retina



**Fig. 2** Block diagram of the CNN-based approach with image processing for retinal disease detection



The proposed approach has an accuracy of 96.5%. The primary goal is to help the patients and eye specialists make an automated and fast diagnosis. Another goal is to increase the analytical performance by improving the accuracy and assisting ophthalmologists in making quicker and efficient detection, which can be of enormous benefit for the patients.

The paper is organized as follows: Sect. 2 elaborates the literature review; Sect. 3 presents the methodology; Sects. 4 and 5 provide the experimental setup and analysis of the results, respectively; Sect. 6 discusses the conclusion; Sect. 7 presents the future scope and limitation.

## 2 Literature review

Deep learning techniques have advanced the state-of-the-art in medical image analysis. However, the application of DL in retinal diseases is relatively recent. Numerous ocular diseases, such as DME, drusen, and CNV, can be captured using OCT scans of the human eye and analyzed using DL techniques. This section elaborates the research on the automation of ocular pathology using AI, machine learning, and DL. The tomography process involves the reintegration of different cross-sectional images of the subject using various projections. Țălu et al. [4] and Schmidt-Erfurth et al. [5] stated that OCT is a high-resolution imaging technique classified as SD-OCT and TD-OCT. The SD-OCT results provide a cross-sectional and volumetric view of the retina

in high resolution. TD-OCT provides a 2D image of the given structure of the internal part of the retina. TD-OCT is ineffective because it only includes thickness analysis of the macula, while SD-OCT enables the monitoring and measurement of various characteristic features. The study found that OCT is a useful technique in analyzing, monitoring, and assessing AMD's different stages. Moreover, drusen could be analyzed using various characteristics of its structure.

Srinivasan et al. [6] presented a classification method based on support vector machine (SVM) classifiers and Histogram of Oriented Gradients (HOG) descriptors and obtained successful results to detect dry AMD and DME using the OCT imaging technique. Their proposed method did not involve the segmentation of inner retinal layers. The SD-OCT datasets consisted of 45 volumetric scans, 15 normal, 15 AMD, and 15 DME. The algorithm achieved the highest specificity and perfect sensitivity in detecting 100% of AMD cases, 100% of DME cases, and 86.67% of normal cases.

A transfer learning method was described by Karri et al. [7] to identify retinal pathologies on the basis of the inception network using the retinal OCT images. The dataset consisted of OCT images with dry AMD, DME, and normal subjects. Their study demonstrated that the fine-tuned CNN was able to effectively identify pathologies compared with classical learning methods. The classifying OCT algorithm shown with limited training data and trained with the use of non-medical images can be fine-tuned. The mean prediction

accuracies were 99% for normal, 89% for AMD, and 86% for DME.

Wang et al. [8] proposed a model for the detection of AMD, DME, and healthy macula using OCT images. Classification algorithms are proven to be necessary for training a classification model. The quadratic programming-based algorithm, kernel-based algorithm, linear regression-based classification algorithm, neural network algorithm, Bayesian algorithm, tree-based algorithm, and ensemble forest algorithm are the various classification algorithm groups. The dataset was tested using one representative from each classification algorithm group. The SMO-sequential minimal optimization-based model was the best with an accuracy of 99.3%. Their experimental procedure involved four steps, namely OCT image preprocessing, feature extraction and selection, classification model building, and predicting results.

Alsaih et al. [9] presented an automated classification framework to detect DME for SD-OCT imaging data volumes. Their method included the general steps of preprocessing, feature detection, feature representation, and classification. The SVM and principal component analysis resulted in a sensitivity of 87.5% and specificity of 87.5%. LBP-ri vectors contributed to the most successful result in the classification of disease.

Choi et al. [10] applied CNN-based deep learning technique for fundus photography analysis and classification of various retinal diseases. Fundus photographs were taken from the structured analysis of the retina database for automated detection of numerous retinal diseases, based on DL-CNN using MatConvNet. The dataset was built by including 10 different categories of retinal images, including normal retina images. The classification results varied as per the number of categories. These results were obtained using the random forest transfer learning method, which was based on VGG-19 architecture.

Choi et al. [10] stated that other retinal diseases can eventually lead to irreversible loss of vision. The causes of vision impairment may include retinal vessel occlusion, retinitis, and hypertensive retinopathy. Previous studies focused on glaucoma, DMR, AMD, and other eye pathologies using fundus photographs. A more effective detection method is necessary to reduce vision loss caused by retinal diseases. The DMR screening was initially adopted for diabetic patients, and it used fundus photographs as inputs.

Hussain et al. [11] proposed the automated identification of AMD and DME using SD-OCT images. The thickness of the retina or individual retinal layer and the volume of pathologies, such as drusen, were some of the retinal features used in the techniques' classification methodology. The SD-OCT images were segmented to extract critical retinal features. A dataset of 251 subjects, 59 normal, 15 DME, and 177 AMD is evaluated for effectiveness by training the

system as a two-class problem of a diseased and healthy retina using a random forest classifier. The methodology had an accuracy of more than 96%.

Kermany et al. [12] achieved 96.6% accuracy along with 97.8% sensitivity and 97.4% specificity. The AUC was 0.999 in distinguishing the retinal diseases (CNV, drusen, and DME) from normal subjects. A variation in the number of images was observed in each category because the training dataset consisted of 37,206 images of CNV, 11,349 images of DME, 8617 images of drusen, and 51,140 normal images. The model's performance was biased because the validation dataset acted as the testing dataset, while only 250 images were chosen for testing and validation from each retinal class. The result analysis was affected due to the imbalance of the number of images in each class.

Schlegl et al. [13] proposed a DL-based technique for the detection of different types of fluids in the retina across various macular diseases using OCT images. Their dataset consisted of OCT images of 1200 patients, including 400 patients with AMD, 400 patients with DME, and 400 patients with RVO (retinal vein occlusion). This fully automated method achieved a mean accuracy of 0.94 with 0.91 precision and 0.94 recall value and was developed to quantify and detect subretinal fluid and intraretinal cystoid.

Das [14] surveyed the diagnosis of retinal diseases, such as retinal tear, retinal detachment, glaucoma, macular hole, and macular degeneration, using various machine learning techniques. The study of healthcare analytics and implementation of deep learning-based pathology detection is a work by Hossain and Muhammad [11, 15]. Some commonly used machine techniques in ocular diagnosis are logistic regression, Naive Bayes, KNN algorithm, and SVM classifier. The implementation of machine learning techniques can be studied from [16–20].

Lemaître et al. [21] addressed the problem of classification of SD-OCT data for automated detection of patients affected by DME. Tsanim et al. [22] employed four CNN models, namely, Vanilla CNN, MobileNetV2, ResNet50, and Xception network, to detect the category of diseases from the retinal OCT scanned images.

Feng et al. [44] focused their study on a four-class retinal disease classification problem for the detection of drusen, DME, CNV, and normal retina using optical coherence tomography images. They proposed a novel classification model for the automated detection of most common blinding diseases and prepared a big dataset of retinal OCT images. The model was based on improved ResNet50. Their approach achieved an accuracy of 0.973, a sensitivity of 0.963, a specificity of 0.985, and an AUC of 0.995 at the B-scan level.

OCT images' automated layering with a blurred layered structure and low contrast is considered challenging or difficult. Xiaoming et al. [23] solved this problem using a new

OCT detection method. Their methodology was based on complex Shearlet transforms. The test dataset consisted of OCT images of dry AMD, Stargardt disease, and retinal macular area with normal condition. The results proved that the complex Shearlet transform method was an effective measure because more layers of OCT images could be detected using this method.

The recent work on CNN and its application in image processing can be studied from Bhatt et al. [24]. They have presented the prevalent DL models, their architectures, related pros and cons, and their medical diagnosis and healthcare system prospects. Kim [25] proposed a model that applies an image super-resolution method to an algorithm that classifies emotions from facial expressions using deep learning. Nie et al. [26] discussed convolution deep learning models for 3D object retrieval. Zhao et al. [27] reviewed the state-of-the-art blood vessel segmentation methods by dividing them into two categories, rule-based and machine-learning-based. Rajalingam et al. [28] presented an image fusion algorithm to visualize and analyze the MRI-CT-PET medical images better. A recent paper by Xi et al. [29] discusses multiscale CNNs for the segmentation of CNV from the OCT data. Table 1 illustrates a summary of critical papers for OCT analysis.

The research gap identified from the literature review are as follows:

1. Most papers have used a pre-trained model, which has fixed biases and weights for retinal disease classification;
2. The researchers have not studied the effect of image enhancement or segmentation over feeding raw images; and
3. Existing research models have low specificity and sensitivity values, which are considered essential parameters for evaluating the performance for medical diagnosis.

This paper addresses the research gaps mentioned above.

### 3 Methodology

Figure 3 demonstrates the process flow of the OCT image data analysis for the detection of four retinal diseases classification using DL-CNN models. Each step is explained in the sections below.

#### 3.1 Data collection

The images of the retinal OCT scans for DME, drusen, CNV, and normal retina are taken from the public (Mendeley database) dataset published in Kermany et al. [12]. The images were taken from the dataset and partitioned

into training, testing, and validation folders, each of which has subfolders for four model classes (CNV, DME, drusen, and normal), having a total of 84,495 b-scan views of OCT images in .jpeg format.

#### 3.2 Preprocessing data

The first step is to obtain uniform-sized normalized images; the dataset images are read, transformed, resized, and cropped. The image sampling is performed into training, validation, and testing in the ratio of 90.16:1.84:8.00. Out of the 83,484 images (dataset), 75,270 images were used for training, 1536 images as validation data, and 6678 images as testing data. Table 2 shows the number of images of each class type in the respective data loaders, and the distribution of the dataset is given in terms of percentages. The images were shuffled to reduce the biases during training to obtain improved results, and they were loaded into different data loaders in a batch size of 84. The data loading was performed in uniformly sized batches because the entire dataset processing in a single step would have resulted in computation memory overload and system crash.

Figure 4 shows different samples of images from training, validation, and testing datasets post-preprocessing. From these samples, the nine-layered retinal structure of the normal eye retina for the given samples is visible. The samples with CNV show proliferation of blood vessels in the choroid layer of the retina, causing ruptures in the Bruch's membrane. These samples are visible as hollow cavity-like structures in the retinal scans in CNV. The DME results from the accumulation of fluids in the macula in the retina resulting from leaky blood vessels, which causes fovea swelling and is visible as tiny holes in the image. The build-up of small yellow/white extracellular material aggregates between the retina pigment epithelium of the eye and the Bruch's membrane causes drusen, visible as dome-like elevations. Meanwhile, the normal retinal structure is seen with clear and continuous membrane boundaries with a deep cut fovea valley with almost a uniform thickness across the structure.

Steps for preprocessing of data:

1. Read files from the directory.
2. Apply resizing of each image to  $150 \times 150$  pixels.
3. Apply CentreCrop operation with final dimensions of  $128 \times 128$  pixels to each image.
4. Convert the image to the tensor data type for compatibility with the model.
5. Normalize the image by subtracting the mean from each pixel value and dividing the result by standard deviation using standard transform.

**Table 1** Literature review summary related to OCT analysis

References	Database	Type of dataset	Target diseases	Approach	Description	Performance
Feng Li et al. [1]	Proprietary	OCT	CNV, DME, druse, and normal	Multi-ResNet 50 Ensembling	The improved ResNet50 mainly consisted of convolutional layers, pooling layers, and fully connected layers	Accuracy: 97.9%; sensitivity: 96.8%; specificity: 99.4%; AUC: 0.998; kappa: 0.969
Kermary et al. [6]	Public (Mendeley database)	OCT	CNV, DME, drusen, and normal	Inception V3 Architecture	The network is composed of 11 inception modules of five types, and each module is designed with convolutional layer, activation layer, pooling layer, and batch normalization layer	Accuracy: 96.6%; sensitivity: 97.8%; specificity: 97.4%; AUC: 0.999
Hussain et al. [2]	Public (DHU and Tian et al. images) and proprietary	OCT	AMD, DME, and normal	Random forest algorithm	Creates a set of decision trees from a subset of training set and then decides the final class of test object based on voting	Av. accuracy: 96%; av. sensitivity: 94%; av. specificity: 85%; mean AUC: 0.99
Alsaith et al. [3]	Proprietary	OCT	DME and normal	Linear SVM	The algorithm creates a line or a hyperplane that separates the data into classes	Sensitivity: 87.5%; specificity: 87.5%
Srinivasan et al. [4]	Public (DHU)	OCT	AMD, DME, and normal	HOG descriptors and SVM classifiers	This algorithm divides the image into connected regions, called cells, and the shape of the local objects is described by counting the strength and orientation of the spatial gradients in each cell	Accuracies of AMD, DME, and normal were 100%, 100%, and 86.67%, respectively
Karri et al. [5]	Public (DHU)	OCT	AMD, DME, and normal	Inception network	This approach fine-tunes a pre-trained CNN to improve its prediction capability and identifies salient responses during prediction to understand learned filter characteristics	Mean accuracies: 99%, 89%, and 86%

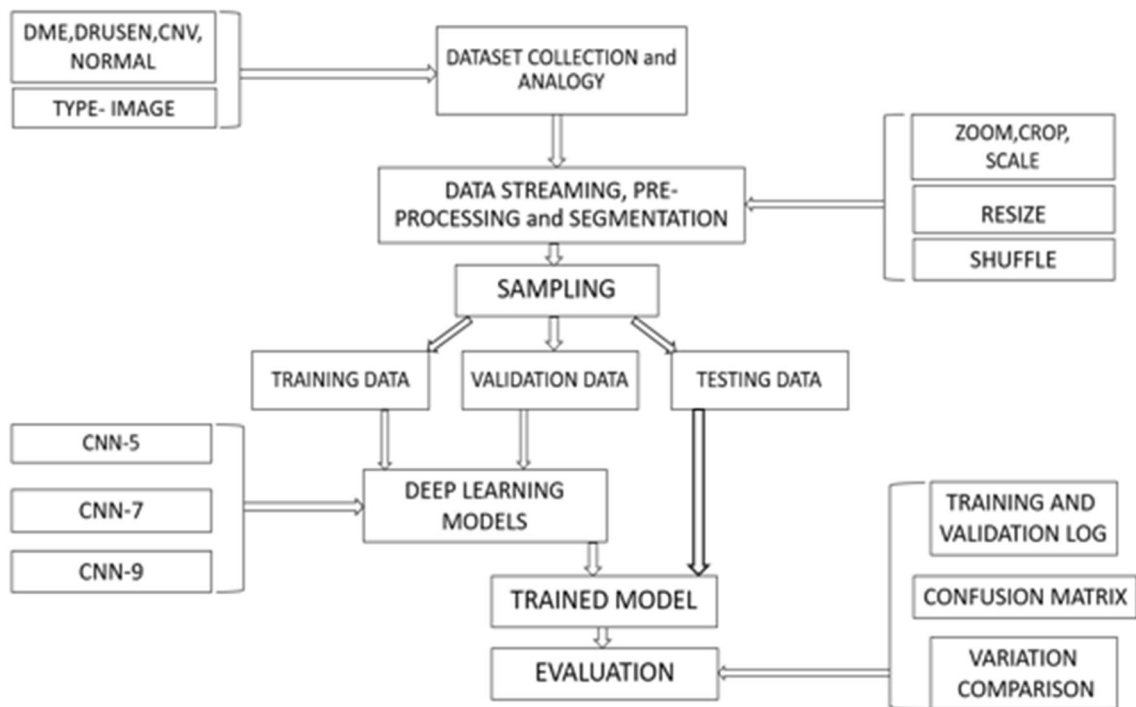


Fig. 3 Process flow diagram

Table 2 Dataset distribution table

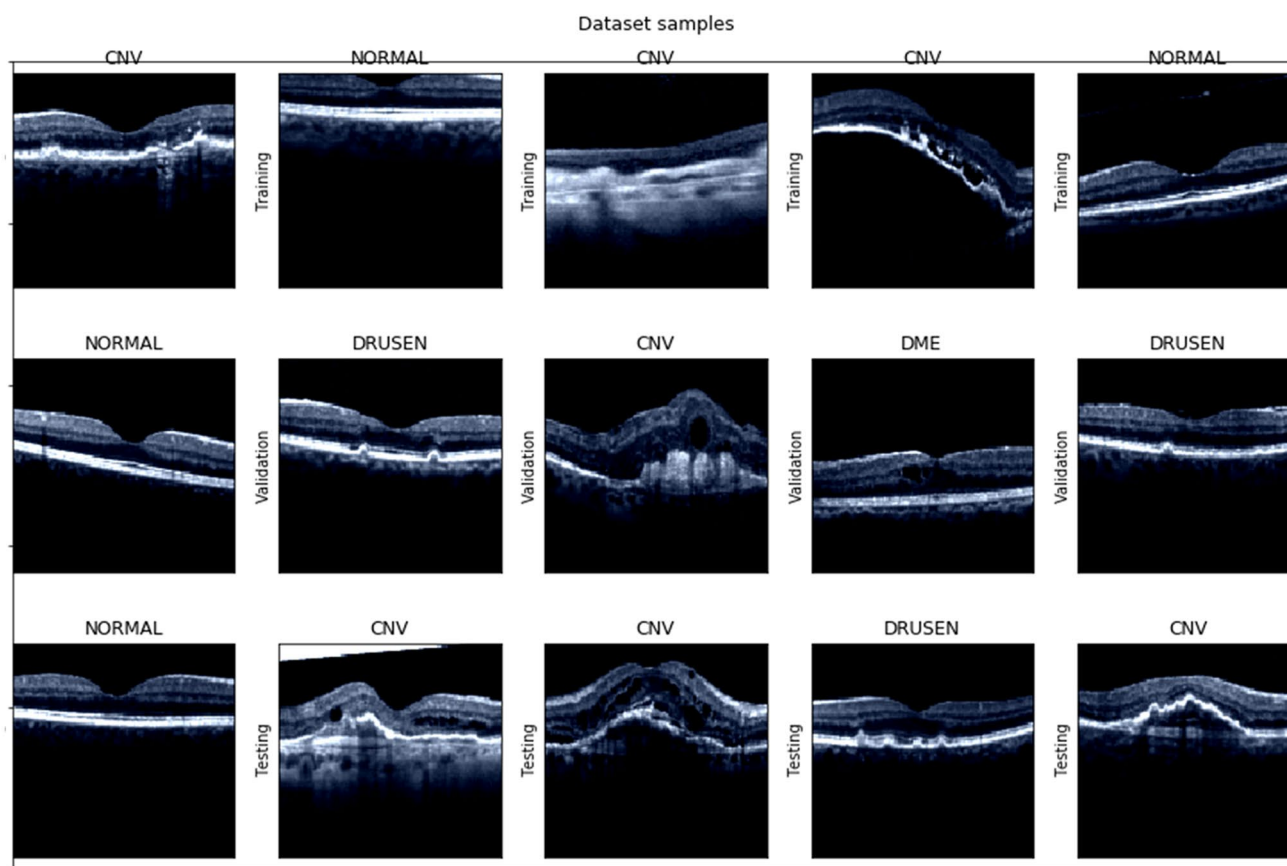
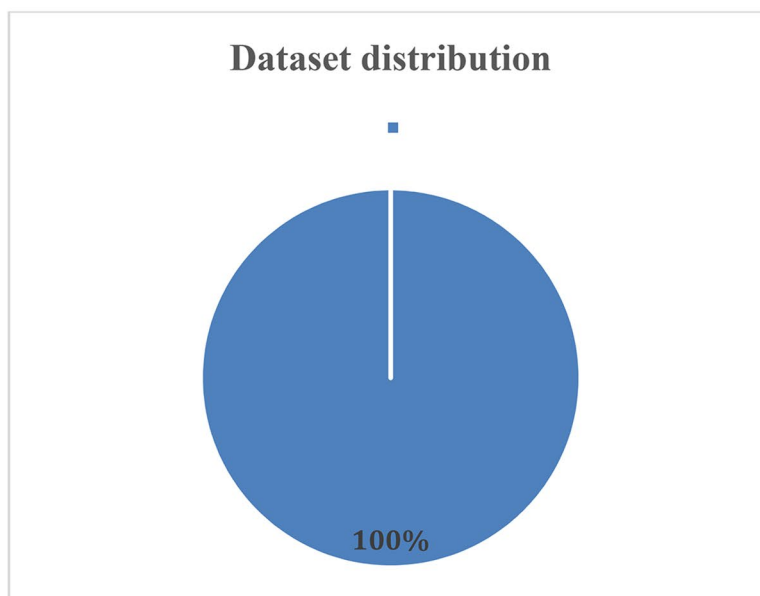
	CNV	DME	Drusen	Normal	Total	Percentage
Total	37,205	11,348	8616	26,315	83,484	100.00
Training	33,509	10,197	7753	23,811	75,270	90.16
Testing	2951	936	693	2098	6678	8.00
Validation	745	215	170	406	1536	1.84

### 3.3 Image enhancement

The image features, retinal structure edge, and retinal layer are improved using various image processing methodologies. Image enhancement helps remove background noise; thus, the model training in Step 3.4 becomes less laborious and complicated and achieves greater efficiency. The images obtained from OCT are low in contrast and blurred as they are plagued with speckle noise. Speckle noise is mainly due to eye movements and blinking during image capturing. The other reasons may include camera noise, pixel value distortion, and random diffuse scattering resulting from interfering ultrasound pulses [30]. Multiple studies to reduce the effects of speckle noise have been conducted, which use different filtering techniques. Shaw et al. [31] mentioned noise filtering algorithms, such as a medium filter, mean filter, Gaussian filters, Fourier, and Butterworth filters for image smoothing for noise removal. Their studies showed that Gaussian filters are best suited for OCT

scans. Kalyanakumar et al. [32] showed how homomorphic wiener filters perform better, followed by Gaussian filters for speckle noise removal from OCT images. They used mean square error, signal-to-noise ratio (SNR), peak SNR, and visual inspection as evaluation parameters. In another method, Canny edge detectors based on Gaussian filtering are used to remove noise, but it damaged the image’s edge structure, resulting in an exceptional edge loss. Xiaoming et al. [23] presented edge detection using Shearlet transformation, which uses the BM3D algorithm for speckle noise removal, followed by the complex Shearlet-based algorithm to layer the retinal OCT image.

This research uses a medium filter after experimenting with other filters available in the literature because it provides excellent results and optimum speed performance. Medium filter considers each neighboring pixel value to decide whether a pixel represents continuity with its surrounding pixel and updates the noise pixel’s value by the medium of surrounding pixel values.



**Fig. 4** Images after preprocessing in training, validation, and testing datasets (without image enhancement, with normalization, and resizing operations)

After the speckle noise is removed, the next step is to improve the contrast of the scans. Low contrast is generally due to poor illumination conditions, capturing devices, and

inexperienced technicians. Nandani et al. [33] showcased a comparison of different contrast improving algorithms over OCT scan images. They observed that the CLAHE method



outperforms other techniques. Setiawan et al. [34] proposed using CLAHE in Green (G) channel to improve the color retinal image quality.

The CLAHE algorithm is an enhanced version of adaptive histogram equalization used to reduce the noise amplification in regions of homogeneity. This algorithm is widely used in medical images and ophthalmology. In this method, the image is divided into subsections, and equalization is performed for each area. This situation results in flattening the division of gray levels and increasing the visibility of the image's hidden features. Thus, we applied and compared Histogram Equalization and CLAHE to improve the gray-scale image's contrast and enhance the edges.

The next image enhancement step is edge detection. Xiaoming et al. [23] used a complex-Shearlet-based method with properties of adequate space, multiscale, frequency domain localization, multidirection, and contrast invariance. They compared their work with other algorithms and precisely extracted edges with strong anti-noise robustness. Dodo et al. [35] worked upon the level set method for separating retinal layers into seven non-overlapping layer structures. They started by selecting a region of interest and obtained gradient edges from it, and these were used to initialize curves for the layers. A different approach in Pekala et al. [36] showed a deep learning-based model based on fully convolutional networks with a Gaussian process coupled with regression-based post-processing to segment the images. Luo et al. [37] used the popular two-pass method, Canny edge detector, and the edge-flow technique for edge detection and found the two-pass method's performance promising over the others. They also found that intensity-based edge detectors, such as the Canny edge detector, and the two-pass method outperformed the texture-based edge-flow method for OCT retinal image analysis. The Canny edge detector algorithm observed fine edge losses in the edge structure due to Gaussian filtering used in the algorithm. Similar results are observed when this algorithm was used on our research data for edge detection.

A contour-based algorithm was applied to detect edges and fine details in the scan images. Finding contours is essential for shape analysis and feature/object detection and recognition. Contour joins all the continuous points along a boundary with the same intensity. It is an outline of the feature to be extracted in a binary image using gradient operations. Contour overlaying is performed to enhance the boundary quality as breaks occur in the edges after segmentation and morphological operations. These steps were preceded by binary image thresholding and morphological transformation for noise removal to successfully find contours. The present studies have employed active contour-based segmentation in their work (González-López et al. [38], Somfai et al. [39], Perez-Cisneros et al. [40]. Perez-Cisneros et al. [40] used active contour models and

estimation of distribution algorithms to generate contours by a prior step of the reference shape's alignment process, which increased the exploration and exploitation capabilities. Mishra et al. [41] improved the active contour model using an efficient two-step kernel-based optimization scheme that first identified the individual layers' approximate location and refined the results using an active contour model.

In our training of the model shown in Figs. 5, 6, and 7, only edges or segmented structures were not suitable inputs for our model because they did not account for fine details in the membrane structure. The edge detection results in the collection of edge segments or contours encompassing the whole image. The images lacked useful information, such as layered structures of the retina and cavity within these layers by only extracting edges, thus, the model could not learn much from this information. The segmentation was conducted on the retina structure samples, which is the extraction of the coherent region of interest isolated from the background. Originally, segmentation is a low-level image processing technique where the image is divided on the basis of the regions of importance into many segments separated by boundaries. These mechanisms produced better results but still could not perform and with successive processing steps. Moreover, these mechanisms lacked the fine layer structural details but were able to include cavity structures to a certain extent. However, combining both methods allowed us to enhance the edge and fine details in these images. The former method obtained a testing accuracy of up to 90.20%, while the latter with segmented output achieved up to 94.47%. Finally, geometric transformations, image resizing, zooming, cropping, and normalizing are performed, and images of  $128 \times 128$  pixels were obtained. Normalization helps in obtaining the data within a range, which helps CNN in performing better and make training faster. Figure 8 shows the final enhanced image sample used in Step 3.4 for the training of the model.

Steps for image enhancement:

1. Read files from the directory.
2. Apply medium blur filter for smoothening.
3. Convert to grayscale for future operations.
4. Apply CLAHE over image for low contrast improvement.
5. Image thresholding by suitable threshold cut limits.
6. Remove further noise and breaks in structure by morphology operation.
7. Extract contours from the above output to extract retinal layer edges (the other edge detection techniques were not useful as discussed).
8. Draw contours to the original image to allow edges and layer structures.
9. Apply further transforms as in the previous step, including resizing, center crop, and normalization.

Fig. 5 Edge detection results

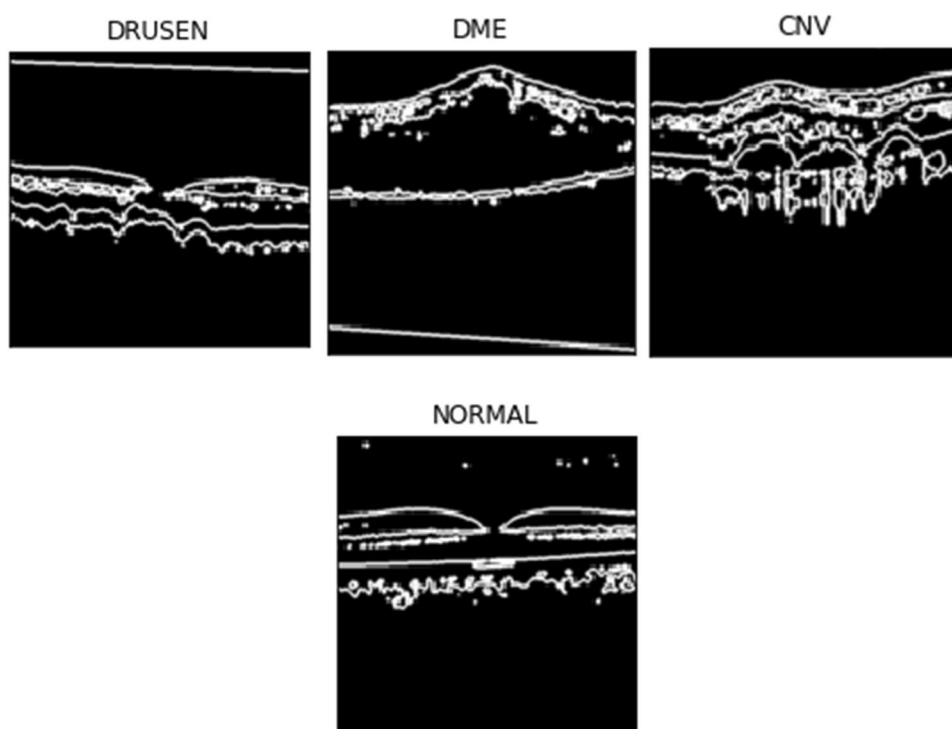
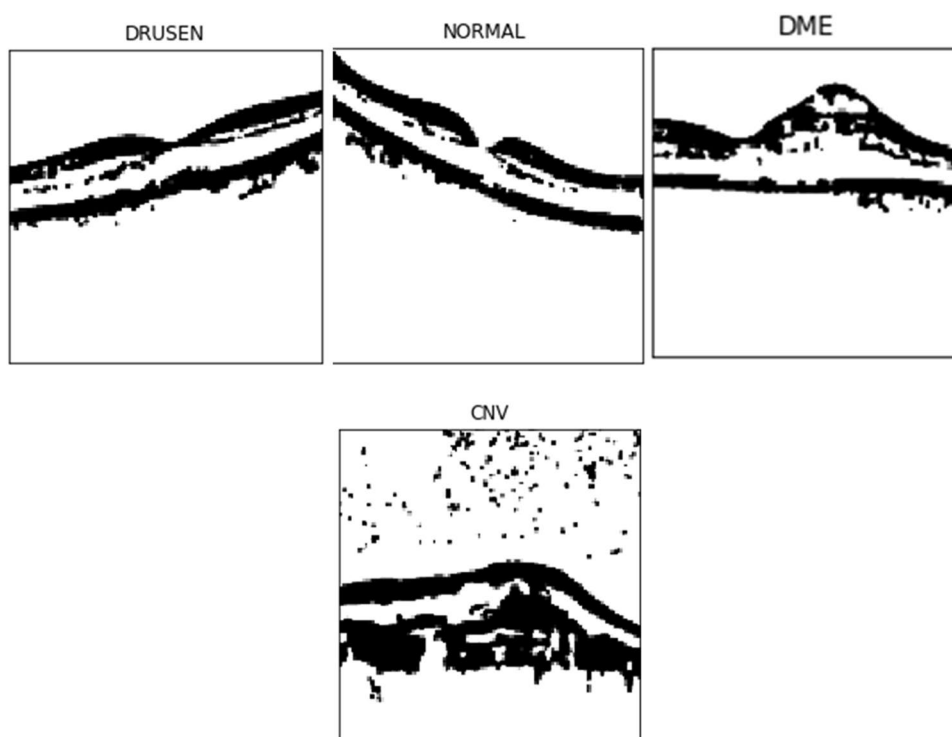


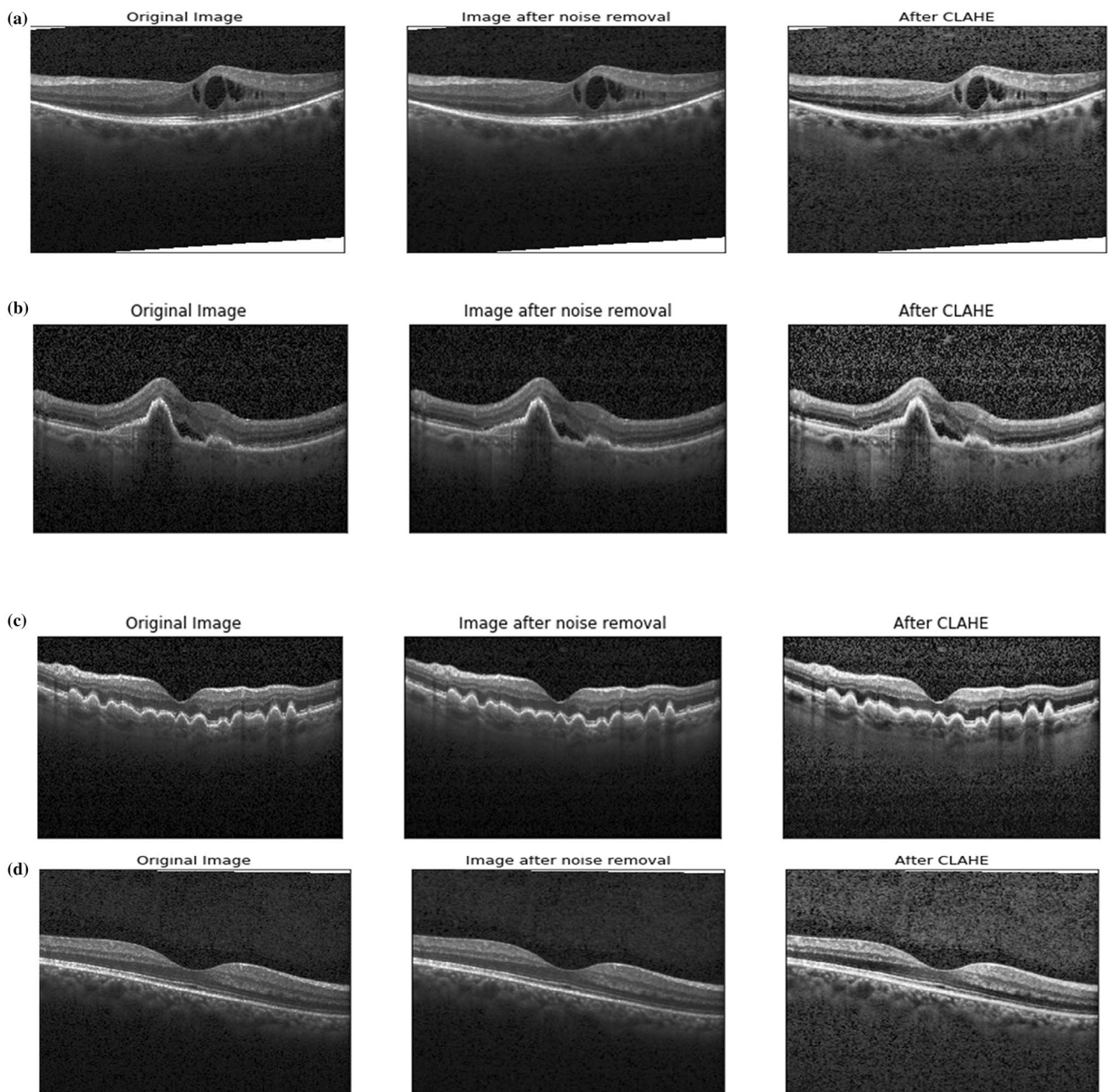
Fig. 6 Segmented retinal structure results



### 3.4 Deep learning models

Our research has used three different CNN-based model architecture and compared the results of these model architectures on the selected dataset. The CNN-

based architecture was chosen for this problem because it demonstrates excellent performance and accurate results in computer vision problems and image classification among the other deep neural network architectures [42]. The benefits of CNN-based models over



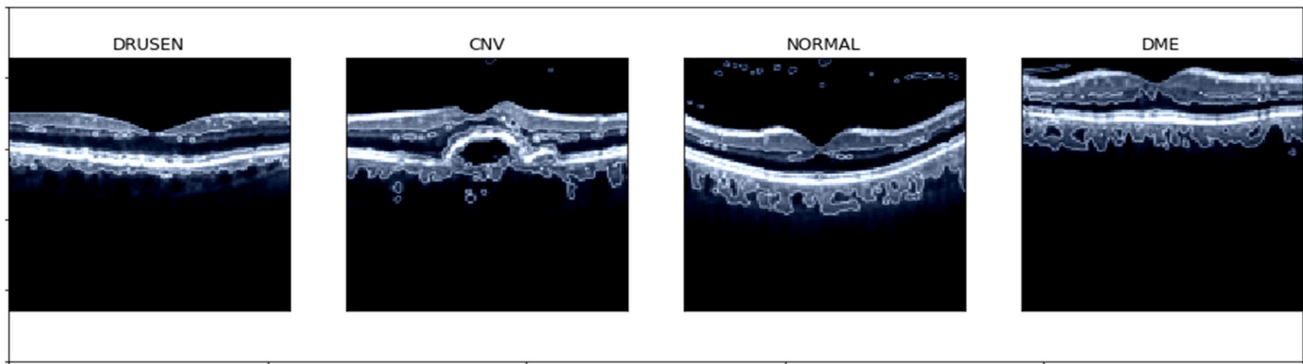
**Fig. 7** Image processing step outcomes for the four classes of diseases: **a** DME, **b** CNV, **c** drusen, and **d** normal

conventional feed-forward neural network models include lesser parameters and connections and faster training [43].

In Fig. 9, the three models are based on different numbers of convolutional layers, max-pooling, and fully connected dense layers and explained below:

1. *Five-layered CNN model* This model has five CNN layers, one input CNN layer with three input channels, and four hidden CNN layers, all with ReLU (rectifier linear

unit) activation. The first, second, fourth, and fifth CNN layers outputted were fed to the max pool layer with a filter size of  $2 \times 2$ . The kernel size applied to the image has a dimension of  $3 \times 3$ . The required padding and stride were set to one. The final output of the 2D CNN layers was flattened, and the features extracted were fed to a block of three fully connected layers with ReLU activation. Finally, the log-softmax probability was calculated and used for further computations. A dropout with a probability of 0.4 was used to avoid overfitting.



**Fig. 8** Final processed images after retinal structure edge and layer enhancement

2. *Seven-layered CNN model* The second model was developed following a similar architecture as in the previous one with an increased number of hidden CNN layers. This model has used CNN-based blocks for better feature extraction and the applied max pool layer to the output of convolutional blocks. Four CNN blocks exist; the first and second blocks consist of only a single CNN layer. The third block consists of three CNN layers, while the fourth one has two CNN layers. Each block has an output max pool layer with  $2 \times 2$  filter dimensions. The output of these CNN blocks has dimensions of  $48 \times 8 \times 8$ , which is fed to a block of three fully connected dense layers.
3. *Nine-layered CNN model* The third model has nine CNN layers compared with the previous models. The block model architecture was used for training the dataset. The first and second blocks consist of a single CNN layer. The third block consists of three CNN layers, while the fourth and fifth blocks have two CNN layers each. The five-block architecture generates an output with dimensions of  $64 \times 4 \times 4$  fed to fully connected dense layers.

The first convolutional layer acts as the input layer and converts each image into a vector. The convolutional layers extract spatial and temporal features by applying different filter kernels over the entire image. These filters slide around the image doing element-wise multiplication of filter weights with image pixel values. These values are then summed up for each filter stride and generate a new activation or feature map, which is inputted to the hidden CNN layers. The hidden CNN layers then improvise on the feature extraction and increase the depth of activation maps. The output is fed through ReLU activation to introduce nonlinearity for better performance. Then, the output is fed to the pooling layer (max pool) to reduce the feature map's dimensionality. The last layer of the model comprises fully connected dense neural networks that use these generated features and classify them. The log-softmax probability final output is used

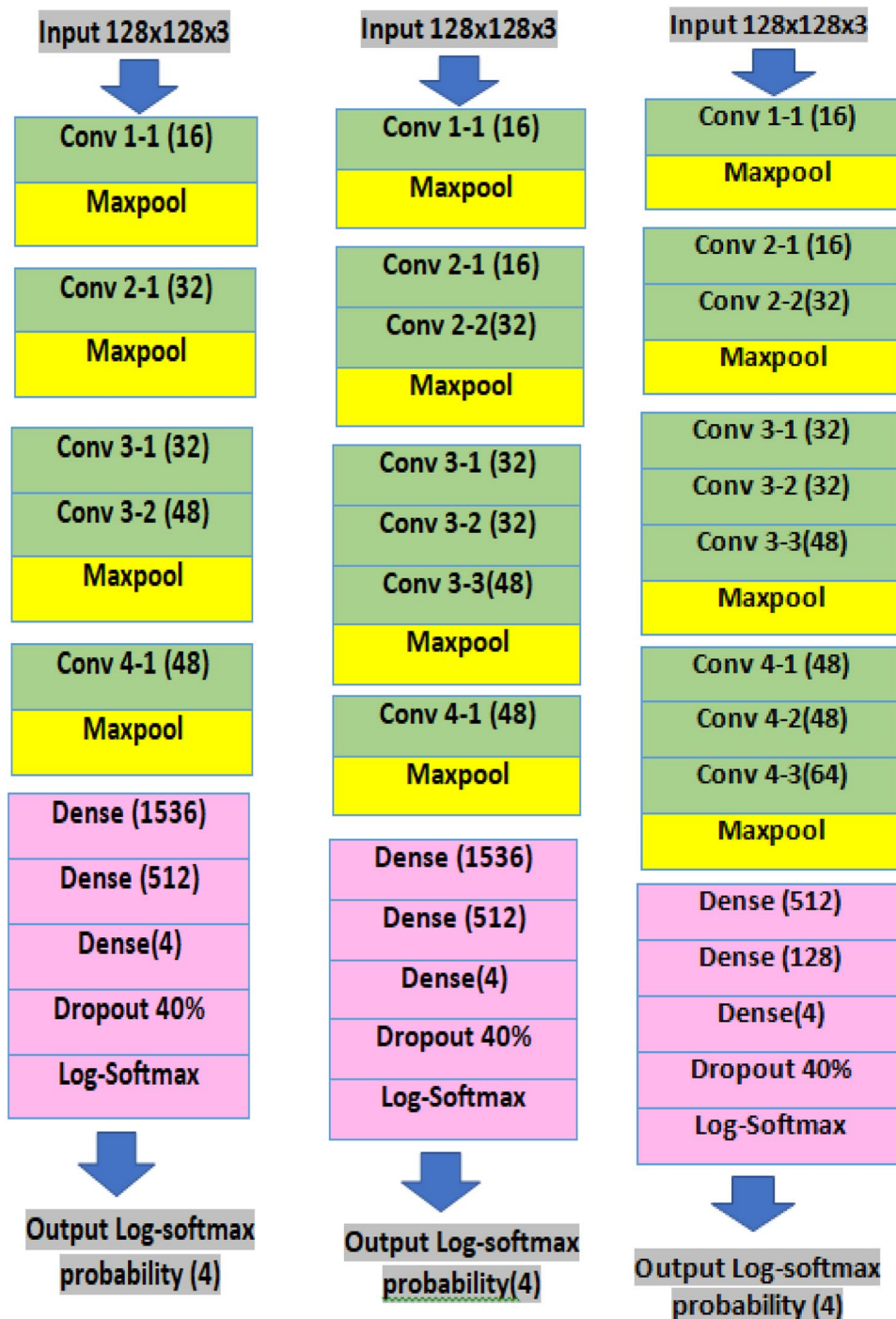
to compute the error in prediction using the defined NLL-Loss (negative log-likelihood) criterion and backpropagate the error through the network for gradient weight tuning of CNN and fully connected layers with an Adam optimizer with a learning rate of  $10^{-3}$ .

In our model, the dropout regularization technique and early stopping algorithm are used to avoid overfitting of results. Table 3 provides the CNN parameters for initialization. Fifteen epochs are used to train the model with a batch size of 84. A total of 880 steps exist in each epoch and a total of 13,440 steps for training. The current step model parameters were tested on the validation dataset after every 20th step during training for the top-1 class prediction to evaluate the model's performance. The finally trained models were then used to evaluate the testing dataset, and statistical analysis was conducted on the observed results.

In CNN, a set of inputs from the training data is mapped to a set of outputs. Many unknown weights exist for a neural network; therefore, the perfect weights for it are impossible to calculate. The problem of learning is seen as a search or optimization problem, and the model may use an algorithm to navigate the space of possible sets of weights to make useful predictions.

Optimizers are algorithms used to modify the neural network parameters, such as weights and learning rate, to reduce losses. Gradient descent is the popularly used optimization algorithm. The term "gradient" refers to an error gradient. The model is used to make predictions with a given set of weights and the error for the calculated predictions. The gradient descent algorithm makes changes in the weights; accordingly, the next evaluation reduces the error. This notion means that the optimization algorithm is navigating down the gradient (or slope) of error. This algorithm is employed in linear regression and classification algorithms. ADAM is among the most efficient optimizer algorithms that find the learning rate for each model attribute. The learning rate is the parameter that defines how the model responds to error estimated after the weights are

**Fig. 9** **a** Five-layered CNN model architecture; **b** seven-layered CNN model architecture; **c** nine-layered CNN model architecture



updated. ADAM considers the exponentially decaying average of gradients (such as momentum), which are termed the first moment, and squared gradients termed as the second moment. Hence, the model is named ADaptive moment. The past and squared gradients are calculated, which are then biased toward zero. The bias updated gradient and squared gradients are calculated. Finally, the weights are updated.

The function, which is minimized or maximized, is referred to as a criterion. This function can be referred to as the cost function, error function, or loss function while minimizing it. The training loss, which can be defined as the error or difference between true and predicted values used, is the NLL. We pass in the raw output from the model’s final layer because the NLL loss in PyTorch expects log probabilities, which are useful to obtain predictions

**Table 3** Parameter initialization for CNN

Parameter	Value/type
Weight initialization	Random
Image input size	128 × 128
Loss criterion	NLLLoss
Optimizer	ADAM
Learning rate	0.001
Batch size	84
Network output	Softmax probability of CNV, DME, drusen, and normal

for a classification model with a Softmax output, represented by

$$L(\hat{h}) = -\frac{1}{M} \sum_{m=1}^M \log(\hat{h}_m), \quad (1)$$

where  $M$  is the number of classes ( $=4$ ) and  $\hat{h}$  is the model current prediction outcome.

The logarithm ensures that the maximum value of the log of probability occurs at the same point as the original probability function because the logarithm is a monotonically increasing curve. Hence, maximizing the log of the probability function works similar to maximizing the probability function. The weights are updated as given below:

$$\text{weight} = \text{weight} - (\text{learningrate} \times \text{gradient}). \quad (2)$$

The neural network model suffers from the problem of overfitting. The model performs well on the training dataset but does not work well on the testing dataset. Regularization is applied to the model to minimize overfitting. In this technique, we modify the existing model and the learning algorithm to perform well on both datasets. Various regularization techniques are available in machine learning, such as L1 and L2 regularization, data augmentation, dropout, and early stopping.

Our research aims to reduce overfitting conditions during training and testing of the model using a dropout technique. In the training phase, random nodes are selected with a probability “pi”, and their activations are made zero for each hidden layer and training input for every iteration. In the testing phase, all the activations are considered, but reduced by a factor “pi” to account for the missing activations.

Steps for model training:

1. Shuffle dataset. Load processed images to data loaders (train, validation, and test). Define batch size and split ratio.
2. Define model architecture by defining different layers, activations, and input and output dimensions.

3. Define loss criterion, optimizer, learning rate, and number of epochs.
4. For each epoch:
  - Take a batch of training dataset.
  - (a) Initialize optimizer, input, and labels
  - (b) Pass input image to model
  - (c) Compute the training loss and backpropagate loss to update weights
  - (d) After each  $k$ th step, validate the trained model using the validation dataset. Compute validation loss and accuracy
  - (e) Store results in an array.
5. Visualize results.
6. Once the test model is trained on the test dataset, plot the confusion matrix and compute the accuracy, precision, sensitivity, specificity, kappa score, F1 value, and test loss.

## 4 Experimental setup

Google Kaggle is used for accessing the data and training and testing the model architecture. Kaggle consists of Nvidia Tesla P100.1xsingle core hyperthreaded Xeon 2 GHz processors, 46 MB cache, 13 GB RAM, and 220 GB disk space. Python v3 and PyTorch v1.4.0 were used. The accelerators supported include TPU and GPU, and our project was trained on the GPU environment.

## 5 Results

In this research, the following evaluation criteria are computed to identify the accuracy of the CNN model for identifying/classifying ocular diseases.

### 5.1 Evaluation criteria and definitions

1. Confusion matrix: It helps in exploring the details necessary to diagnose the performance of our model. A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions is summarized with count values and broken down by each class. The confusion matrix shows the ways in which our classification model is confused when it makes predictions. This mechanism provides insight into not only the errors being made by our classifier but also the types of errors that are being made by the models. The confusion matrix is also useful for measuring recall, precision, specificity, and accuracy, which we also have considered in this

study while evaluating the performance of our different models and used confusion matrix to compute the same.

The value of true positive (TP) and true negative (TN) and false positive (FP) and false-negative (FN) can be derived from the confusion matrix and are explained below:

- TP: correctly predicted positive class;
- FP: incorrectly predicted positive class;
- FN: incorrectly predicted negative class;
- TN: correctly predicted negative class.

2. Accuracy: It is the measure of how accurately the classifier can classify the data. The following equation provides the accuracy:

$$\text{Accuracy}(\text{range } 0 - 1) = \frac{\text{TP} + \text{TN}}{\text{FP} + \text{FN} + \text{TP} + \text{TN}}. \quad (3)$$

3. Precision: It defines the relation of total positive results that are correct to the classifier's total positive results, as provided by

$$\text{Precision} (\text{range } 0 - 1) = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (4)$$

4. Sensitivity (or recall): It corresponds to the TP rate of the considered class and is computed using

$$\text{Sensitivity} (\text{range } 0 - 1) = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (5)$$

5. Specificity: It corresponds to TN rate of the considered class (i.e., the proportion of negatives that have been correctly identified). The following equation provides the specificity:

$$\text{Sensitivity} (\text{range } 0 - 1) = \frac{\text{TN}}{\text{TN} + \text{FP}}. \quad (6)$$

6. F1 score: It considers the precision and recall values to calculate the weighted average and is computed using the following equation:

$$\text{F - 1 Score} = \frac{2 \times \text{TP}}{\text{FP} + \text{FN} + (2 \times \text{TP})}. \quad (7)$$

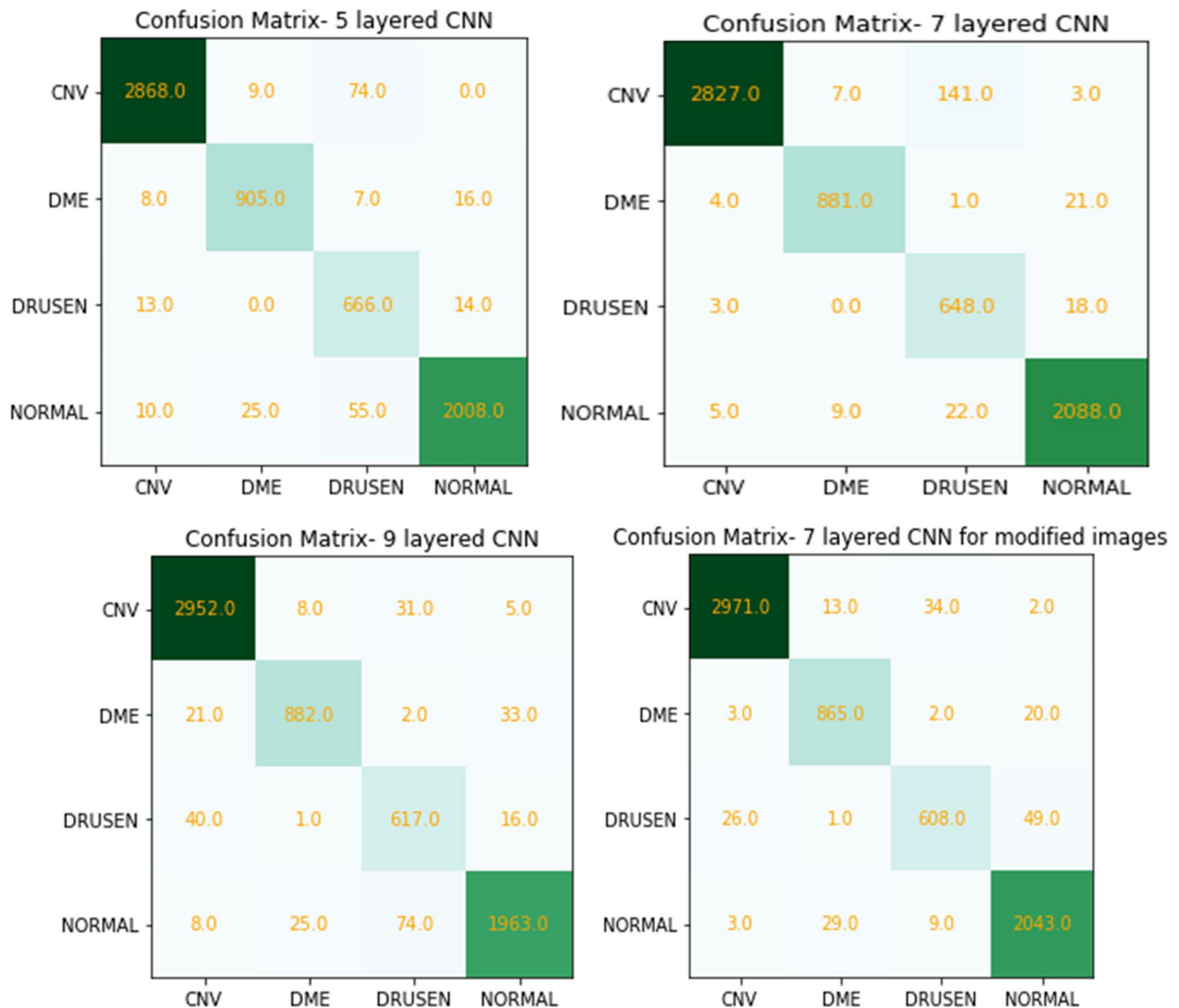
7. Minimum training loss: It is the minimum amount of error on the training set of data during the training steps.
8. Minimum validation loss: It is the minimum error after running the validation set of data through the trained network.
9. Maximum validation accuracy: It is the measure of how accurate the model's prediction is compared with the true data after running the validation set of data.

10. Minimum testing loss: It is the minimum loss calculated on the testing dataset using the trained model.
11. Model size and training time: The time taken by the program to train the model and test results defines the training time in minutes. The size of memory used by the CPU to store model weights defines the model size in megabytes.
12. Kappa value: Cohen's kappa statistic is an important overall accuracy measurement parameter for multiclass classification problems with data imbalance. In this case, other measures may not provide a complete performance picture of the classifier by considering the possibility of the outcomes occurring by chance.

## 5.2 Parameter-based evaluation

The system is trained on three machine learning models, namely, five-layer CNN model, seven-layer CNN model, and nine-layer model. Figure 10 illustrates the confusion matrix for each of the CNN models. The comparison aims to find the most suitable and efficient model for our dataset by comparing the performance metrics and visualization results from all the CNN layers. The parameter-based model evaluation's essential measures are precision, accuracy, F1 score, kappa value, losses, and confusion matrix. Table 4 reports the outcomes of our study.

The epochs at 15/13,440, learning rate at 0.001, and input size at 0.19 MB are maintained for all three models. The minimum training loss is 0.039854, the minimum validation loss is 0.149069, and the minimum test loss is 0.005284 in the five-layer CNN that is similar to the losses seen in other layered models. The nine-layer CNN model is memory-optimized. The overall accuracy is high for the five- and seven-layer CNN models at 96.50% and 96.54%, respectively. The maximum validation accuracy used to estimate the model's prediction capability for the five-layer CNN at 96.30% is higher than that of other models. The F1 score, which signifies the accuracy by finding the balance between precision and recall value, is considerably low for the five-layer CNN, but it is best for the seven-layer CNN. The trainable parameters increase with the increase in the number of layers. Hence, considerable differences exist between the nine-layer CNN and the rest of the models. The kappa coefficient identifies the relationship between the expected accuracy and the observed accuracy in a confusion matrix. The value is high for the five-layer CNN (0.949) and seven-layer CNN (0.948). The precision is calculated for all four classes, and the nine-layer CNN has the highest precision for CNV at 98.53%. The seven-layer CNN has the highest precision for DME, drusen, and normal at 97.13%, 96.86%, and 98.31%, respectively. The recall value indicates the accuracy with which the model detects all the classes in the dataset. The recall accuracy is highest in the seven-layer



**Fig. 10** Confusion matrices for different models

CNN in CNV and normal detection at 99.58% and 98.03%, respectively, whereas the five-layer CNN performs best in DME and drusen with accuracy rates of 98.93% and 96.38%, respectively.

The model training time is about the same for all three model structures and almost double for the image-enhanced dataset. In the five-layer CNN model, the total trainable parameters were 55,58,116, which took 233.4027 min of training time. The model's estimated total size was 26.01 MB (0.19 MB input size, 4.62 MB forward/backward pass size, and 21.2 MB parameter size). In the seven-layer model, the total trainable parameters were 55,69,684, which took 238.8658 min of training time. The model's total estimated size was 25.61 MB (0.19 MB input size, 4.18 MB forward/backward pass size, and 21.25 MB parameter size), which is

slightly less than that of the five-layer model. The last model with nine layers has total trainable parameters of 7,00,636, which took 230.1585 min of training time. The model's total estimated size was 7.08 MB (0.19 MB input size, 4.22 MB forward/backward pass size, and 2.67 MB parameter size). This model may be best suited to run in environments with memory limitations with some trade-off with performance. In the three models devised, a high specificity value and sensitivity value of 0.98 and 0.96 were accomplished. The F1 score observed in the seven-layer model was highest among the three models.

The training log shows that the model training was significant for ten epochs, and beyond that, it started to overfit (Fig. 11). The increase in accuracy was also reduced. Thus, epochs were reduced in the final model; hence, computations



**Table 4** Performance of different models at the B-scan level for detection of four classes

Comparison parameters	Five-layer CNN	Seven-layer CNN	Nine-layer CNN	With image processing
Epochs/steps	15/13,440	15/13,440	15/13,440	10/8960.0
Learning rate	0.001	0.001	0.001	0.001
Min training loss	0.039854	0.05164	0.084115	0.085634
Min validation loss	0.149069	0.14923	0.158438	0.154837
Max validation accuracy	96.30%	95.36%	95.55%	95.14%
Accuracy (overall)	96.54%	96.50%	96.05%	97.14%
Precision (CNV)	97.19%	94.93%	98.53%	98.34%
Precision (DME)	96.69%	97.13%	94.03%	97.19%
Precision (drusen)	96.10%	96.86%	91.54%	88.89%
Precision (normal)	95.71%	98.31%	94.83%	98.03%
Recall accuracy (CNV)	98.53%	99.58%	97.72%	98.93%
Recall accuracy (DME)	98.93%	98.22%	96.29%	95.26%
Recall accuracy (drusen)	96.38%	79.80%	85.22%	93.11%
Recall accuracy (normal)	83.04%	98.03%	97.32%	96.64%
Sensitivity	0.9649	0.9605	0.9654	0.9447
Specificity	0.9883	0.9868	0.9884	0.9816
F1 score	89.7995	95.33464	94.43444	95.80451
Kappa	0.949	0.948	0.941	0.957
Model training time (minutes)	233.4027	238.8658	230.1585	420.3642
Min test loss	0.005284	0.005602	0.008608	0.006861
Total trainable parameters	55,58,116	55,69,684	7,00,036	55,69,684
Input size (MB)	0.19	0.19	0.19	0.19
Forward/backward pass size (MB)	4.62	4.18	4.22	4.18
Params size (MB)	21.2	21.25	2.67	21.25
Estimated total size (MB)	26.01	25.61	7.08	25.61

and time also decreased. Figure 11 illustrates that the five-layer CNN model is slightly overfitting.

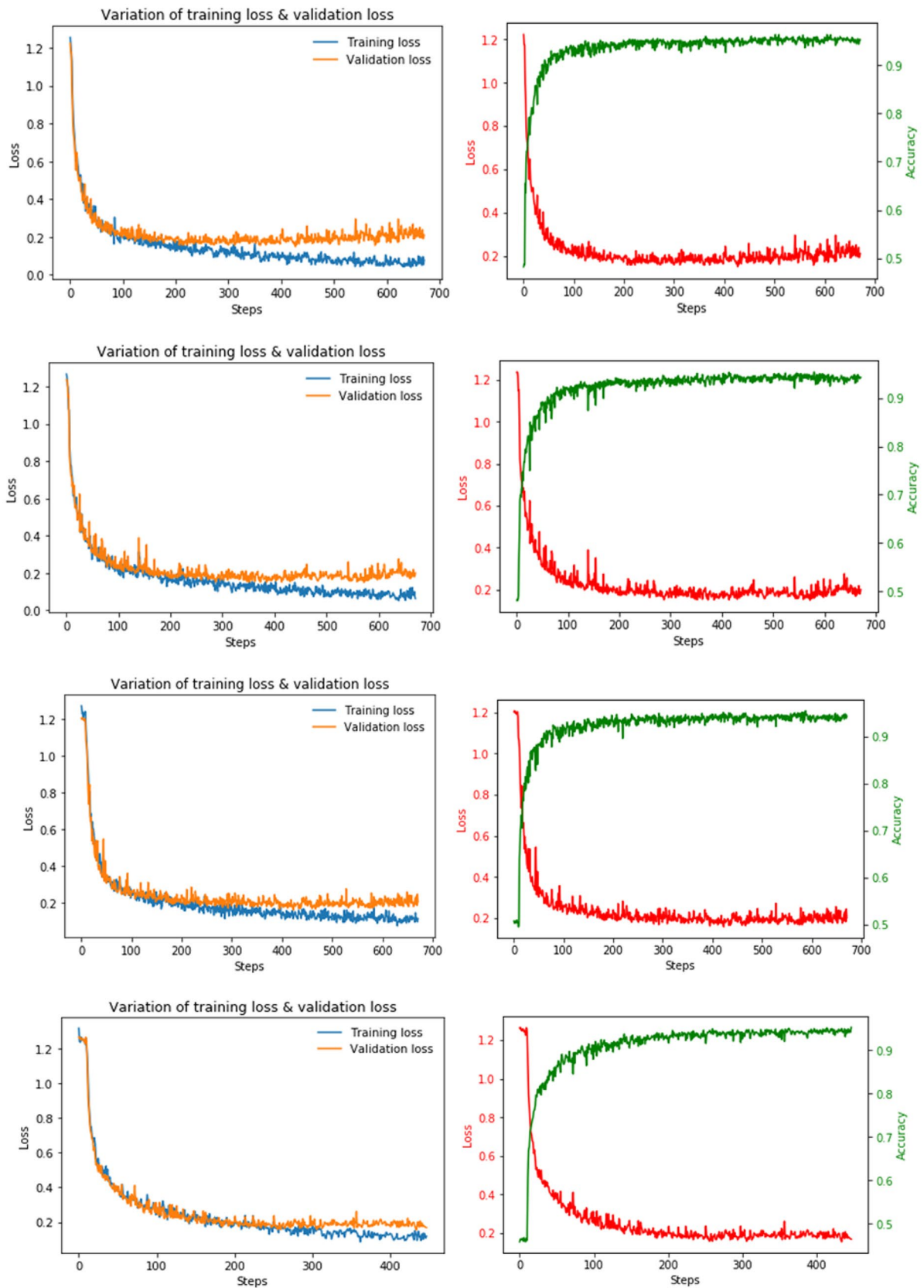
### 5.3 Visualizing model performance

The deep learning-based classification performed is a black box AI system for automated decision-making, which uses machine learning techniques to map feature data into class without uncovering the reasons. Different visualization techniques are used to analyze the performance and understand the decision-making of these models. For this purpose, popular open-source matplotlib and OpenCV libraries were utilized. The variation of training loss and validation loss (NLLLoss with softmax envelope) over successive steps during training was plotted. This situation demonstrates how the two losses vary with each other. Both the losses decrease with successive steps, reflecting that predictions are becoming increasingly accurate, and updating weights results in the movement of losses in the direction of minima. This phenomenon can also show if the model is overfitting or not.

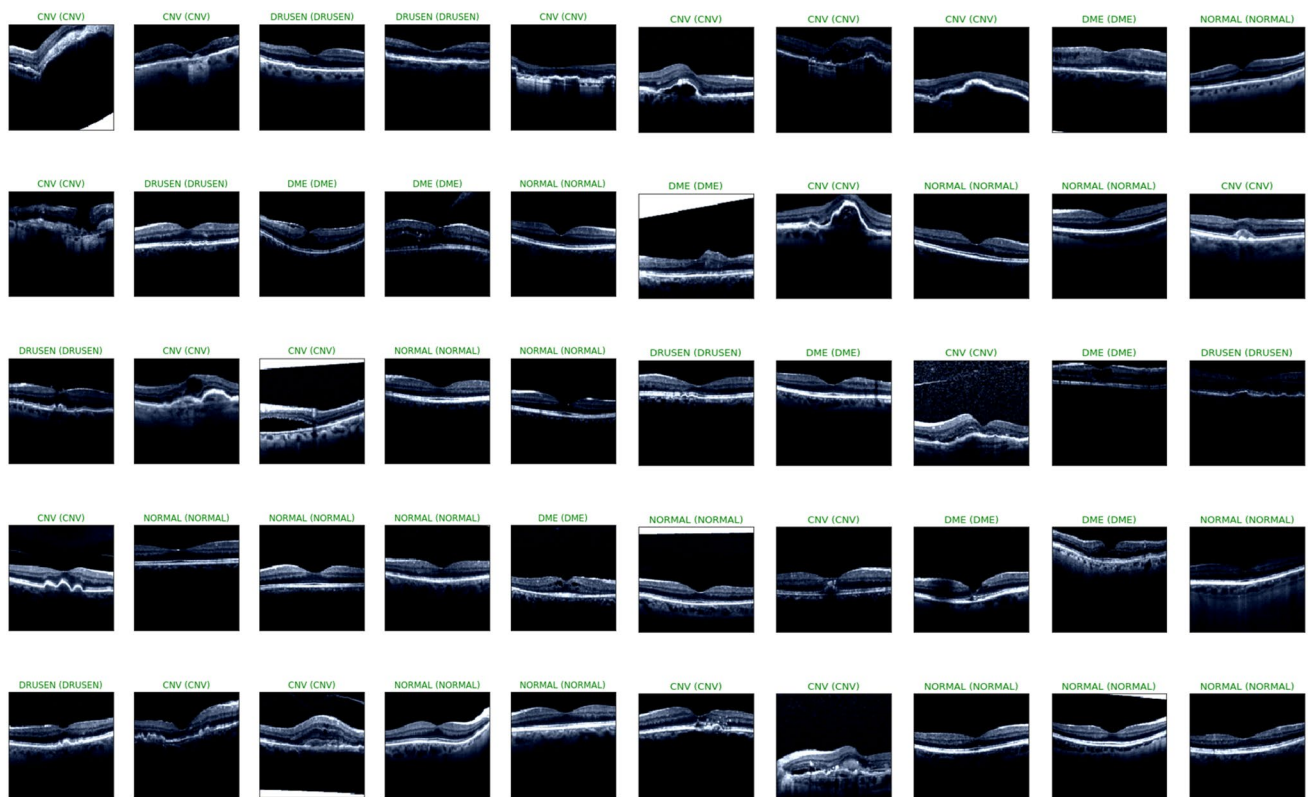
Next, a comparison of the variation of validation loss and accuracy over the training steps is performed (Fig. 11). We can see a subsequent increase in the validation accuracies

with a decrease in validation loss. The filter outputs of all the CNN layers and fully connected layers show how the model views the image internally as it passes down through multiple layers. The confusion matrix is plotted to calculate multiple parameters and evaluate the model performance. Finally, 50 random example images from the test dataset were taken, and their prediction for each model was observed (Fig. 12). Each correct prediction was marked with a green and an incorrect prediction with a red. Running on different samples multiple times for each model showed that most predictions were correct.

This research focuses on three CNN models with five, seven, and nine layers. Lower numbers, such as a three- or four-layer model, were analyzed, which showed poor performance in extracting fine features because the input OCT images are similar with subtle differences in structure. The result showed that the higher layered models performed better in identifying the information, such as layered structures of the retina and cavity. We also wanted to study and explore memory and time-efficient solution. We added layers to our model and observed that performance decreased for layers greater than nine to our model. We observed a decrease in the gap between the training and the validation loss with



**Fig. 11** Performance of five, seven, nine-layer models and with image enhancement CNN model: **a** training and validation losses over successive training steps; **b** validation loss and validation accuracy over successive training steps



**Fig. 12** Predictions on random OCT samples using a seven-layer model

the increase in the number of CNN layers in the successive models (Fig. 11). The saturation in losses is achieved at later stages with the increase in the number of layers. The slope is more gradual when two models are compared in Fig. 13, which helps the model learn over greater epochs and requires more features for classification.

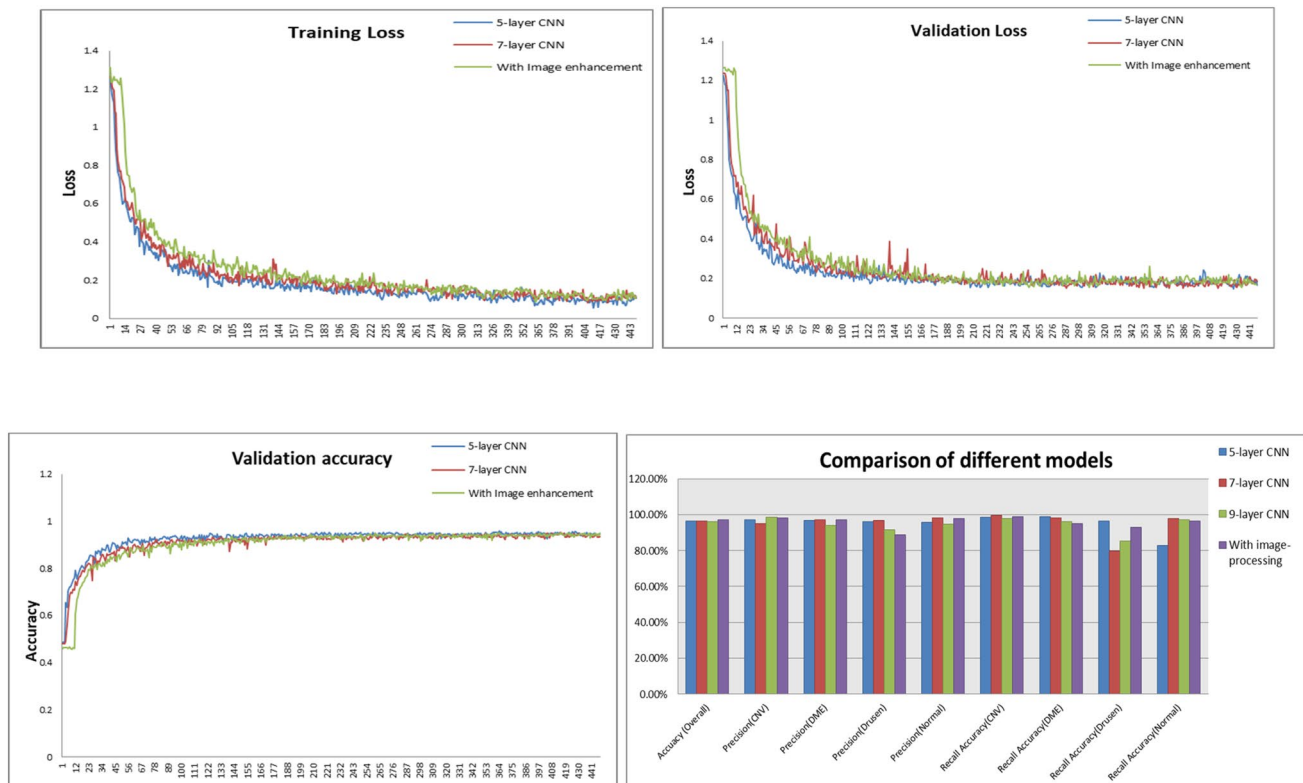
#### 5.4 Variations due to image enhancement

In the final model, the performance was evaluated using the seven-layer CNN model with an image-enhanced input (Fig. 13 and Table 4). During the model's training, 8960 steps with each batch size of 84 scans per step and 0.001 learning rate were performed. In a multiclass comparison of CNV, drusen, DME, and normal, the model attained an overall accuracy of 97.14%, F1 score of 95.8045, and kappa value of 0.957. These values show a slight improvement in performance measures by advancing our preprocessing algorithm and better results than all the three models. The minimum training, validation, and testing losses are similar to the other models. However, the training time has increased due to an increase in the model's processing steps and complexity. Figure 11 shows a decrease in losses, and the increase in accuracy becomes more gradual than the other models. In this model, higher classification accuracy

for CNV, DME, and normal classes can be achieved than others with decreased value for drusen. This model also had poor performance in terms of sensitivity, which decreased to 0.94 from 0.96 compared with the other models. The training time increased due to the additional preprocessing of the image.

## 6 Conclusion

This study presents a comprehensive and systematic implementation of deep learning techniques (CNN) for accurately classifying and identifying ocular pathological structures for CNV, drusen, and DME versus normal. The framework utilizes OCT images of the retina, which are preprocessed and processed for noise removal, contrast adaptation for the edge, and layer structure for the retinal structure edge and layer enhancement. This image dataset is analyzed using three different CNN models (of five, seven, and nine layers) with an ADAM optimizer to classify and identify the four ocular pathologies. The output results can show the distinction between drusen, CNV, DME, and normal scans with very high F1 score, precision, accuracy, and sensitivity with a considerable decrease in time taken for detection and epochs. After the different layered CNN models



**Fig. 13** Comparison of different models

are evaluated, we could identify the detrimental parameters affecting the algorithm's operations. The seven-layer CNN model is the one with balanced statistics and is suggested by our proposed work for use. The proposed approach has an accuracy of 96.5%. The primary goal is to help the patients and eye specialists in making an automated and fast diagnosis with increased accuracy, performance, and quicker and efficient detection, which can greatly benefit the patients.

## 7 Limitation and future scope

This research successfully demonstrated the detection of four ocular diseases from the OCT images with an accuracy of 96%. Certain limitations of this study are as follows: (1) the dataset which we had selected for the project had scans collected from a single demographic region and did not contain diversity in terms of eye structure observed in people of different races; (2) the images taken for this project specifically included the OCT scans, while for other diseases, the scans may not be OCT but fundus photographs or angiographic pictures, which may require the project to be trained again for such types images; (3) the scans taken from the dataset consisted of all of them in the same scanning

settings or techniques. Therefore, the efficacy of this model for different systems is still not fully established.

We can further improve this work by exploring various options for dimensional reduction. In this work, we reduced the size of the input images to  $128 \times 128$  pixels to minimize the input parameters and employed max pool layers in the models, which also decreases the dimensions of the feature matrix over successive steps.

Further extension of the model may include analyzing the other ocular pathology class, such as diabetic retinopathy, AMD, and glaucoma. The model currently operates on the OCT images for the classification; however, it would be beneficial to modify the model to operate on the OCT angiography and fundus photographs. Models can be developed or explored to consider the biological variations in eye and retina structure.

**Acknowledgements** This study was supported by Taif University Researchers Supporting Project (number: TURSP-2020/10), Taif University, Taif, Saudi Arabia.

## References

1. Yang, X., et al.: Deep relative attributes. *IEEE Trans. Multimedia* **18**(9), 1832–1842 (2016)

2. Hossain, M.S., Muhammad, G., Alamri, A.: Smart healthcare monitoring: a voice pathology detection paradigm for smart cities. *Multimedia Syst.* **25**(5), 565–575 (2019)
3. Hossain, M.S., Amin, S.U., Muhammad, G., Sulaiman, M.: Applying deep learning for epilepsy seizure detection and brain mapping visualization. In: *ACM Trans. Multimedia Comput. Commun. Appl. (ACM TOMM)*, vol. 15(1s) (2019)
4. Țălu, S.-D., Țălu, Ș.: Use of OCT imaging in the diagnosis and monitoring of age related macular degeneration, age related macular degeneration—the recent advances in basic research and clinical care, Gui-Shuang Ying. *IntechOpen* (2012). <https://doi.org/10.5772/33410>
5. Schmidt-Erfurth, U., Klmscha, S., Waldstein, S.M., Bogunović, H.: A view of the current and future role of optical coherence tomography in the management of age-related macular degeneration. *Eye (Lond.)* **31**(1), 26–44 (2017). <https://doi.org/10.1038/eye.2016.227>
6. Srinivasan, P.P., Kim, L.A., Mettu, P.S., Cousins, S.W., Comer, G.M., Izatt, J.A., Farsiu, S.: Fully automated detection of diabetic macular edema and dry age-related macular degeneration from optical coherence tomography images. *Biomed. Opt. Express* **5**(10), 3568–3577 (2014). <https://doi.org/10.1364/BOE.5.003568>
7. Karri, S.P., Chakraborty, D., Chatterjee, J.: Transfer learning based classification of optical coherence tomography images with diabetic macular edema and dry age-related macular degeneration. *Biomed. Opt. Express* **8**(2), 579–592 (2017). <https://doi.org/10.1364/BOE.8.000579>
8. Wang, Y., Zhang, Y., Yao, Z., Zhao, R., Zhou, F.: Machine learning based detection of age-related macular degeneration (AMD) and diabetic macular edema (DME) from optical coherence tomography (OCT) images. *Biomed. Opt. Express* **7**(12), 4928–4940 (2016). <https://doi.org/10.1364/BOE.7.004928>
9. Alsaih, K., Lemaitre, G., Rastgoo, M., Massich, J., Sidibé, D., Meriaudeau, F.: Machine learning techniques for diabetic macular edema (DME) classification on SD-OCT images. *Biomed Eng Online.* **16**(1), 68 (2017). <https://doi.org/10.1186/s12938-017-0352-9>
10. Choi, J.Y., Yoo, T.K., Seo, J.G., Kwak, J., Um, T.T., Rim, T.H.: Multi-categorical deep learning neural network to classify retinal images: a pilot study employing small database. *PLoS ONE* **12**(11), e0187336 (2017). <https://doi.org/10.1371/journal.pone.0187336>
11. Hussain, A., Bhuiyan, A., Luu, C.D., Theodore Smith, R., Guymer, R.H., Ishikawa, H., et al.: Classification of healthy and diseased retina using SD-OCT imaging and Random Forest algorithm. *PLoS ONE* **13**(6), e0198281 (2018). <https://doi.org/10.1371/journal.pone.0198281>
12. Kermany, D.S., Goldbaum, M., Cai, W., Valentim, C., Liang, H., Baxter, S.L., McKeown, A., Yang, G., Wu, X., Yan, F., Dong, J., Prasadha, M.K., Pei, J., Ting, M., Zhu, J., Li, C., Hewett, S., Dong, J., Ziyar, I., Shi, A., et al.: Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell* **172**(5), 1122–1131.e9 (2018). <https://doi.org/10.1016/j.cell.2018.02.010>
13. Schlegl, T., Waldstein, S.M., Bogunovic, H., Endstraßer, F., Sadeghipour, A., Philip, A.M., Podkowinski, D., Gerendas, B.S., Langs, G., Schmidt-Erfurth, U.: Fully automated detection and quantification of macular fluid in OCT using deep learning. *Ophthalmology* **125**(4), 549–558 (2018). <https://doi.org/10.1016/j.ophtha.2017.10.031>
14. Das, S., Malathy, C.: Survey on diagnosis of diseases from retinal images. *J Phys Conf Ser.* **1000**, 012053 (2018). <https://doi.org/10.1088/1742-6596/1000/1/012053>
15. Hossain, M.S., Muhammad, G.: Deep learning based pathology detection for smart connected healthcares. *IEEE Netw.* **34**(6), 120–125 (2020)
16. Pandey, S., Solanki, A.: Music instrument recognition using deep convolutional neural networks. *Int. J. Inf. Technol.* **13**(3), 129–149 (2019)
17. Rajput, R., Solanki, A.: Real-time analysis of tweets using machine learning and semantic analysis. In: *International Conference on Communication and Computing Systems (ICCCS2016)*, Taylor and Francis, at Dronacharya College of Engineering, Gurgaon, 9–11 Sept, vol 138 issue 25, pp. 687–692 (2016).
18. Ahuja, R., Solanki, A.: Movie recommender system using K-Means clustering and K-Nearest Neighbor. In: *Accepted for Publication in Confluence-2019: 9th International Conference on Cloud Computing, Data Science & Engineering*, Amity University, Noida, vol. 1231, no. 21, pp. 25–38 (2019).
19. Tayal, A., Kose, U., Solanki, A., Nayyar, A., Saucedo, J.A.M.: Efficiency analysis for stochastic dynamic facility layout problem using meta-heuristic, data envelopment analysis and machine learning. *Comput. Intell.* **36**(1), 172–202 (2020)
20. Singh, T., Nayyar, A., Solanki, A.: Multilingual opinion mining movie recommendation system using RNN. In: Singh, P., Pawłowski, W., Tanwar, S., Kumar, N., Rodrigues, J., Obaidat, M. (eds.) *Proceedings of First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019)*. Lecture Notes in Networks and Systems, vol 121. Springer, Singapore (2020). [https://doi.org/https://doi.org/10.1007/978-981-15-3369-3\\_44](https://doi.org/https://doi.org/10.1007/978-981-15-3369-3_44).
21. Lemaitre, G., Rastgoo, M., Massich, J., Cheung, C.Y., Wong, T.Y., Lamoureux, E., Milea, D., Mériaudeau, F., Sidibé, D.: Classification of SD-OCT volumes using local binary patterns: experimental validation for DME detection. *J. Ophthalmol.* **2016**, 3298606 (2016). <https://doi.org/10.1155/2016/3298606>
22. Tasnim, N., Hasan, M., Islam, I.: Comparative study of Deep Learning approaches on Retinal OCT Image (2019).
23. Xiaoming, L., Ke, Xu., Peng, Z., Jiannan, C.: Edge detection of retinal OCT image based on complex shearlet transform. *IET Image Process.* **13**(10), 1686–1693 (2019). <https://doi.org/10.1049/iet-ipr.2018.6634>
24. Bhatt, C., Kumar, I., Vijayakumar, V., et al.: The state of the art of deep learning models in medical science and their challenges. *Multimedia Syst.* (2020). <https://doi.org/10.1007/s00530-020-00694-1>
25. Kim, P.W.: Image super-resolution model using an improved deep learning-based facial expression analysis. *Multimedia Syst.* (2020). <https://doi.org/10.1007/s00530-020-00705-1>
26. Nie, W., Cao, Q., Liu, A., et al.: Convolutional deep learning for 3D object retrieval. *Multimedia Syst.* **23**, 325–332 (2017). <https://doi.org/10.1007/s00530-015-0485-2>
27. Zhao, F., Chen, Y., Hou, Y., et al.: Segmentation of blood vessels using rule-based and machine-learning-based methods: a review. *Multimedia Syst.* **25**, 109–118 (2019). <https://doi.org/10.1007/s00530-017-0580-7>
28. Rajalingam, B., Al-Turjman, F., Santhoshkumar, R., et al.: Intelligent multimodal medical image fusion with deep guided filtering. *Multimedia Syst.* (2020). <https://doi.org/10.1007/s00530-020-00706-0>
29. Xi, X., Meng, X., Yang, L., et al.: Automated segmentation of choroidal neovascularization in optical coherence tomography images using multi-scale convolutional neural networks with structure prior. *Multimedia Syst.* **25**, 95–102 (2019). <https://doi.org/10.1007/s00530-017-0582-5>
30. Baroni, M., Fortunato, P., Torre, A.: Towards quantitative analysis of retinal features in optical coherence tomography. *Med. Eng. Phys.* **29**, 432–441 (2007). <https://doi.org/10.1016/j.medengphy.2006.06.003>

31. Shaw, P.R., Manickam, S., Burnwal, S., Kalyanakumar, V.: Study of removal of speckle noise from OCT images. *35*, 313–317 (2015).
32. Kalyanakumar, V., Manickam, S.: Performance evaluation of speckle reduction filters for optical coherence tomography images. *Int J Pharm. Bio. Sci* **6**, B837–B845 (2015)
33. Saya Nandini Devi, M., Santhi, S.: Improved oct image enhancement using Clahe. In: *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* ISSN: 2278–3075, vol. 8, issue 11 (2019)
34. Setiawan, A., Mengko, T., Santoso, O., Suksmono, A.: Color retinal image enhancement using CLAHE, pp. 1–3 (2013). <https://doi.org/https://doi.org/10.1109/ICTSS.2013.6588092>.
35. Dodo, B.I., Li, Y., Liu, X., & Dodo, M.I.: Level Set Segmentation of Retinal OCT Images (2019).
36. Pekala, M., Joshi, N., Liu, T., Bressler, N.M., DeBuc, D.C., Burlina, P.: Deep learning based retinal OCT segmentation. *Comput. Biol. Med.* **114**, 103445 (2019). <https://doi.org/10.1016/j.compbiomed.2019.103445>
37. Luo, S., Yang, J., Gao, Q., Zhou, S., Zhan, C.A.: The edge detectors suitable for retinal OCT image segmentation. *J. Healthc. Eng.* **2017**, 1 (2017)
38. González-López, A., de Moura, J., Novo, J., Ortega, M., Penedo, M.G.: Robust segmentation of retinal layers in optical coherence tomography images based on a multistage active contour model. *Heliyon*. **5**, e01271 (2019). <https://doi.org/10.1016/j.heliyon.2019.e01271>
39. Somfai, G.M., Jozsef, M., Chetverikov, D., DeBuc, D.: Active contour detection for the segmentation of optical coherence tomography images of the retina. *Invest. Ophthalmol. Vis. Sci.* **55**(13), 4793 (2014)
40. Perez-Cisneros, M., Cruz-Aceves, I., Avina-Cervantes, J.G., Lopez-Hernandez, J.M., Garcia-Hernandez, M.G., Torres-Cisneros, M., Estrada-Garcia, H.J., Hernandez-Aguirre, A.: Automatic image segmentation using active contours with univariate marginal distribution. *Math. Probl. Eng J.* (2013). <https://doi.org/10.1155/2013/419018>
41. Mishra, A., Wong, A., Bizheva, K., et al.: Intra-retinal layer segmentation in optical coherence tomography images. *Opt. Express* **17**(26), 23719–32372 (2009)
42. Sultana, F., Sufian, A., Dutta, P.: Advancements in image classification using convolutional neural network. In: *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pp. 122–129 (2018)
43. Hossain, M.S., Muhammad, G.: Cloud-based collaborative media service framework for healthcare. *Int. J. Distrib. Sens. Netw.* **2014**, 11 (2014)
44. Feng, L., Chen, H., Liu, Z, et al.: Deep learning-based automated detection of retinal diseases using optical coherence tomography images. *Biomed. Opt. Express* **10**, 6204–6226 (2019)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.