Performance evaluation of various deep learning based models for effective glaucoma evaluation using optical coherence tomography images

Law Kumar Singh¹,² • Pooja¹ • Hitendra Garg³ • Munish Khanna²

Received: 20 August 2020 / Revised: 20 February 2022 / Accepted: 9 March 2022
Published online: 28 March 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract
Glaucoma is the dominant reason for irreversible blindness worldwide, and its best remedy is early and timely detection. Optical coherence tomography has come to be the most commonly used imaging modality in detecting glaucomatous damage in recent years. Deep Learning using Optical Coherence Tomography Modality helps in predicting glaucoma more accurately and less tediously. This experimental study aims to perform glaucoma prediction using eight different ImageNet models from Optical Coherence Tomography of Glaucoma. A thorough investigation is performed to evaluate these models’ performances on various efficiency metrics, which will help discover the best performing model. Every net is tested on three different optimizers, namely Adam, Root Mean Squared Propagation, and Stochastic Gradient Descent, to find the best relevant results. An attempt has been made to improvise the performance of models using transfer learning and fine-tuning. The work presented in this study was initially trained and tested

Law Kumar Singh
lawkumarcs@gmail.com

Pooja
pooja.1@sharda.ac.in

Hitendra Garg
Hitendra.garg@gmail.com

Munish Khanna
munishkhanna.official@rocketmail.com

¹ Department of Computer Science and Engineering, Sharda University, Greater Noida, India
² Department of Computer Science and Engineering, Hindustan College of Science and Technology, Mathura, India
³ Department of Computer Engineering and Applications, GLA University, Mathura, India
on a private database that consists of 4220 images (2110 normal optical coherence tomography and 2110 glaucoma optical coherence tomography). Based on the results, the four best-performing models are shortlisted. Later, these models are tested on the well-recognized standard public Mendeley dataset. Experimental results illustrate that VGG16 using the Root Mean Squared Propagation Optimizer attains auspicious performance with 95.68% accuracy. The proposed work concludes that different ImageNet models are a good alternative as a computer-based automatic glaucoma screening system. This fully automated system has a lot of potential to tell the difference between normal Optical Coherence Tomography and glaucomatous Optical Coherence Tomography automatically. The proposed system helps in efficiently detecting this retinal infection in suspected patients for better diagnosis to avoid vision loss and also decreases senior ophthalmologists’ (experts) precious time and involvement.

**Keywords** Glaucoma · Optical coherence tomography · Deep learning · Convolution neural network · VGG16 · Efficient net

## 1 Introduction

The very first section is composed of four subsections. The first three sub-sections are dedicated to the introduction of retinal image analysis, glaucoma, and optical coherence tomography, respectively. The last subsection is dedicated to the motivation and purpose of this study.

### 1.1 Retinal image analysis

Retinal pictures are digital images of the rear of our eyes. It displays the optic disc, the retina (where light and pictures hit), and blood vessels. The following Fig. 1 is an example of a retinal image of a healthy eye [5]. According to Abramoff et al. [1], retinal imaging has amplified during the last many years and has rapidly grown within ophthalmology in the past twenty years. It is the backbone of clinical consideration and the board of patients with retinal diseases. The availability of good quality cameras to take images of the retina makes it possible to examine the eye for the presence of many different eye diseases with a simplified, non-invasive method. To a great extent, it is utilized for historical studies, telemedicine, and research on new treatments for glaucoma.

![Eye Structure](image)

**Fig. 1** Eye Structure
1.2 Glaucoma

High intraocular pressure is most commonly associated with the development of glaucoma, which affects the eye’s optic nerve. Lucid fluid fills the front eye, known as “aqueous humor,” which continually and regularly forms on the surface of the eyeball, providing a constant level of pressure. Because of the rise in pressure, the fluid does not drain from the eye. Furthermore, this condition causes the optic nerve, which is responsible for transmitting picture information from the eye to the brain, to deteriorate with time. It has little impact on eyesight at first, so it frequently goes undetected by patients. It is the second leading reason for sightlessness and is characterised by steady damage or harm to the optic nerve and resultant loss of insight view [69]. The conventional techniques of diagnosis employed by ophthalmologists include tonometry, pachymetry, and others. Early identification and ideal treatment have appeared to satisfy the risk of visual adversity because of glaucoma. Figure 2 shows a healthy eye (left) and a glaucomatous eye (right). Imaging techniques like Optical Coherence Tomography play a crucial role in the analysis of glaucoma (Fig. 3), monitoring the progress of the disease, and evaluating structural damage [72].

1.3 Optical coherence tomography

Optical coherence tomography (OCT) is a comparatively new imaging method that gives a clear perspective on intraretinal morphology and grants non-invasive, depth-resolved, non-contact functional imaging of the retina [15]. Ophthalmologists widely accept an OCT because it provides a cross-sectional view of retinal tissues. It represents the comprehensive layer of information showing the structural damage and sectional imaging of ocular tissue [34]. It is highly dependable, and as a result, it is commonly used as a subordinate in routine glaucoma patient management. Glaucoma can also be checked with the help of the Time-domain OCT and Fourier domain OCT, which are both very helpful [56].

Glaucoma is an ophthalmic ailment distinguished by progressive structural changes, such as diminishing of the retinal nerve fibre layer (RNFL). The RNFL thickness deviation map study, a color-coded map indicating RNFL abnormality regions, detects glaucoma with high sensitivity and precision. To track and follow diffuse and focal RNFL development, pattern analysis of average and sectorial RNFL thicknesses and event analysis of the RNFL thickness maps and RNFL thickness profiles can be used. OCT-RNFL estimation of the thickness measurement is an important tool for structural analysis in current clinical glaucoma treatment [55]. OCT uses light waves to take the measurements. Cross-sectional images of the retina. Using OCT, the doctors can see every single particular layer of the retina. It allows ophthalmologists to map and quantify the thickness, which encourages the diagnosis.

![Fig. 2](image_url) Eye Images showing glaucoma and normal eye
1.4 Motivation and purpose of study

Every year, millions of citizens are hit by glaucoma, and ultimately its progression reaches irreversible blindness due to steady damage or harm to the optic nerve [24]. It is the second leading cause of blindness among human beings and the foremost reason for irreversible sightlessness worldwide. It has been observed that 13% of eye-related diseases are related to glaucoma [30]. As per prior published studies on the prediction of glaucoma patients in the world in coming years, specifically in the USA, there is a projection that there will be about 7.32 million patients in 2050 [70]. Other studies project that more than 79.6 million people will be infected by this disease in the coming years, as the population ages [55]. Significant visual impairments associated with early stages of the disease may be limited by prompt diagnosis and adequate treatment of the condition. Periodic routine screening will help to diagnose glaucoma in a timely manner while reducing the workload of specialist ophthalmologists [7]. Therefore, it is important to detect glaucoma during the initial stage in order to halt or at least mitigate its development and to maintain vision [20].

The changes in the internal structure of the retina layers caused by glaucoma need to be noticed because a timely and advanced investigation may prevent vision loss. As a result, among the researcher community, this domain is the hot cake these days. The critical objective of the latest research is to design an automated computer-aided glaucoma recognition system that takes retinal OCT images as input. As an output, it classifies the patient’s input images into two classes (glaucoma or healthy) [45]. This automated system has many benefits, such as replacing expensive advanced hardware-dependent diagnostic devices, replacing all omnipresent conventional detection methods, increasing the precision of the identification, decreasing variability, and reducing diagnosis time, thus replacing or at least minimizing the reliance on skilled ophthalmologists (required for the complex character of glaucoma disease pathology) [46]. The researchers have made several charitable attempts in the last 15 years to find an efficient approach to glaucoma recognition. Initially, research was focused on multiple image processing and feature engineering techniques. With the recent developments in artificial intelligence, various proficient state-of-art machine learning algorithms and deep learning models have come into existence [59]. With the advances in medical imaging, biomedical engineering, and the enhanced usability of these algorithms, researchers have begun concentrating on applications of these deep learning models for various human-related diseases, including glaucoma. After going through multiple studies published during the last ten years, it has been realized that practitioners have made many severe attempts at early and timely recognition of glaucoma using retina fundus images [78]. However, similar attempts to diagnose glaucoma using OCT images were not made, and the domain was not ultimately revealed [33].
Moreover, changes to published findings still have more scope, enabling the research results to be realistic for human society, which may potentially replace costly clinical instruments and minimize the reliance on ophthalmologists. This all motivates us to select this domain and to apply various of the most recent deep learning models to the early detection of glaucoma with enhanced results on various efficiency evaluation metrics. This experimental study is also a serious and successful attempt in the same direction. Through this work, an attempt has been made to design deep learning models for automated computerized glaucoma recognition which will generate high efficiency and timely results with the least amount of human intervention. This study intends to identify and implement algorithms that can learn and run through the datasets and, after that, efficiently predict glaucoma symptoms so that they can be prevented or slowed down.

The main contributions are summarized as follows:

1. Eight latest deep learning models with three different optimizers have employed for the classification of Optical coherence tomography images into two classes: normal and glaucomatous. For performance enhancement, transfer learning and fine tuning are also implemented.

2. For this empirical study, two subject datasets have been selected that are not correlated with each other. Initially, using a large dataset, models are trained, and from those, we selected the four best-performing models. Later on, exhaustive testing is performed on these four models using the second dataset only. This approach is rarely used by practitioners where training and testing are performed on different datasets; classically, part of the training dataset is selected for testing purposes.

3. While doing a comprehensive literature survey, it was found that few researchers have shown the superiority of their model’s performances using Area under Curve only. However, in this empirical study, both Area under Curve and accuracy are demonstrated in selecting the superior model(s). It is also noticed that the best performing models have achieved auspicious and highly satisfactory accuracy values.

4. As already stated, eight models with three optimizers have been analysed on various efficiency parameters like sensitivity, specificity, F1-Score, precision, Area under Curve Score, and accuracy. Based on the accuracy, it is concluded that VGG16 (using Root Mean Squared Propagation Optimizer) achieved the best accuracy of 95.68%, sensitivity of 92%, F1-Score of 93%, specificity 96%, and Area under Curve of 0.95. To our knowledge, this is a first attempt to apply an EfficientNet for Glaucoma classification from Optical coherence tomography. The results of this model are promising, but not the best.

5. The proposed computerised automated system for early and timely glaucoma identification requires the least amount of human intervention with high proficiency. The study fills up the research gap by proposing a system based on the latest deep learning-based nets and outputs high accuracy standards that are on par with various ophthalmologist societies.

The paper begins with a concise description of OCT. At the end of this segment, a detailed motivational note is also given. Section 2 explores the history and associated studies in the area of OCT in glaucoma. A comprehensive comparative table is also shown at the end of this section. In Section 3, a detailed outline is provided for the proposed methodology, and a comprehensive description of the Convolution Neural Network (CNN) is provided, along with
the architecture of various deep learning nets used in the study, a selected dataset, and preprocessing steps. Section 4 reports the performance measurement of different deep learning-based algorithms on the benchmark dataset for the mentioned purpose and extensive simulation results. Section 5 talks about and compares existing literature, and then finally Section 6 is dedicated to scope, limitations, and future guidelines.

2 Literature review

In today’s world, deep learning is one of the most prominent domains used by the research community in the medical profession. For example, researchers are employing deep learning in recognizing novel diseases like COVID-19 [25] or classical infections like Alzheimer [44]. It also helps to identify diseases like glaucoma more accurately [4]. Review studies [51] and [38] are entirely dedicated to the application of deep learning based approaches for efficient glaucoma recognition. Deep learning helps to extract complex features automatically from a variety of data. Some relevant and prominent prior published studies have been shortlisted, which are discussed below. Table 1 is also shown at the end of this section. In the table, NA means that the data is not available in the published literature, and it shows how the two groups compare.

Maetschke et al. [33] have introduced a profound training technique that uses 3-D CNN and separates eyes from the raw un-segmented OCT volume of the optic nerve head (ONH) directly into nonglaucoma or glaucoma eyes. They applied deep learning techniques and obtained a high AUC of 0.94, which provides more information regarding OCT volume regions, which is majorly significant for glaucoma diagnosis. An et al. [3] developed machine learning algorithms for glaucoma detection in open-angle glaucoma patients. This approach was based on using OCT methods and on coloured fundus pictures. CNN was used for the disc fundus graphic, disc RNFL thickness map, disc GCC thickness map, and disc RNFL deviation map. Knowledge augmentation and dropout were used to train CNN. The process hit 0.963 area under the curve (AUC). Naveed et al. [43] discussed various techniques of research for more accurate glaucoma screening. The most commonly used methods for glaucoma detection using OCT were attached, and using fundus images was also described. They discussed that the increment in cup-to-disc ratio (CDR) is the primary factor in classifying and sorting out glaucoma patients from other patients. Lee et al. [28] proposed a method for automatically segmenting the rim in spectral-domain 3-D OCT volume and optic disc images. Four surfaces that are intra-retinal were segmented using a rapid multi-scale 3-D graph search algorithm. Kansal et al. [23] presented an investigation to compare the OCT device’s glaucoma detection precision and accuracy. An electronic data extraction search strategy was used for the relevant study. The parameters of the study were glaucoma patients, parametric glaucoma, mild, pre-parametric glaucoma, myopic glaucoma, and moderate-to-severe glaucoma. The result of the study proves that OCT devices, when demonstrated, give excellent detection accuracy. They conclude that all five OCT devices have comparable categorization abilities. Patients with more severe glaucoma had better AUROCs. Bussell et al. [6] offer a comparison of Spectral-domain OCT (SD-OCT) and Time-domain OCT (TD-OCT) for glaucoma prediction. With SD-OCT there are advantages in assessing glaucoma due to faster scanning speeds, increased axial resolution that leads to lower susceptibility to eye movement artifacts, and improved reproducibility while having similar detection accuracy. For glaucoma testing, SD-OCT has better axial resolution, quicker scanning rates, and better repeatability than TD-OCT. In a
| Year | Author and References | Method(s) | Techniques (including preprocessing) | Dataset | Accuracy /AUC |
|------|-----------------------|-----------|--------------------------------------|---------|--------------|
| 2019 | An et al.[3]          | Machine learning and Transfer learning of CNN (VGG19) Segmentation of Optic disc cup and rim in spectral-domain 3D OCT (SD-OCT) volumes. | Subject and extracted images,CNN; Random forest. Intraretinal surface segmentation using multi-scale 3D graph; Retinal flattening; Convex Hull-based fitting. | 208 glaucoma eyes and 149 healthy eyes | AUC-0.94 |
| 2009 | Lee et al.[28]        | Segmentation of Optic disc cup and rim in spectral-domain 3D OCT (SD-OCT) volumes. | Retinal nest; Convex Hull-based fitting. | 27 spectral-domain OCT scans (14 right eye scans and 13 left eye scans) | ACC-NA | AUC-NA |
| 2018 | Kansal et al.[23]     | Optical coherence tomography for glaucoma diagnosis: An evidence-based meta-analysis. Glaucoma diagnostic accuracy of commercially available OCT devices. | Meta-analysis; Focus on AUROC curve; Macula ganglion cell complex (GCC). | 16,104 glaucomatous and 11,543 normal control eyes | AUROC-for RNFL 0.897 GCC 0.885 GCIP 0.858 |
| 2011 | Xu et al.[74]          | 3D Optical Coherence Tomography Super Pixel with Machine Classifier Analysis for Glaucoma Detection | Machine Classifier. | 192 eyes of 96 subjects having 44 normal, 59 glaucoma, 89 glaucomatous eyes. | ACC-NA | AUC-0.855 |
| 2014 | Busell et al.[6]      | OCT for glaucoma diagnosis, screening and detection of glaucoma progression | Focus on SD-OCT, TD-OCT, RNFL thickness, GCC, GCIP and AUC | 100 healthy subjects for 30 months of longitudinal evaluation. | ACC-NA | AUC-NA |
| 2019 | Asoka et al.[4]       | CNN and Transfer Learning | Image Resize and Image Augmentation | Training 94 Open Angle Glaucoma (OAG) and 84 normal eyes. For testing 114 OAG eyes and 82 normal eyes | AUCROC: DL Model 0.937, RF 0.820, SVM 0.674 |
| 2017 | Muhammad et al.[39]   | CNN (AlexNet)+ Random Forest | Border segmentation, Image Resize | 57 image taken from Glaucoma and 45 images from Normal | Best Accuracy reported 93.31% |
| 2020 | Lee et al.[27]        | NASNET Deep Learning Model | Image pixel value normalized between 0 and 1 and the image size 331 x 331 matrix. | 86 Glaucoma and 196 healthy | Deep Learning achieve AUC 0.990 |
| 2020 | Thompson et al.[68]   | CNN | N A | 612 Glaucomatous eyes and 542 normal eyes | Deep Learning Probability: ROC Curve Area 0.96 |
recently published study on glaucoma identification, authors selected OCT data (Singh et al. [58]). Forty-five vital features were extracted using two approaches. Five machine learning algorithms were employed for the categorization of OCT images into the glaucomatous and non-glaucomatous classes. K Nearest Neighbour (0.97) achieved the highest AUC.

| Year | Author and References | Method(s) | Techniques (including preprocessing) | Dataset | Accuracy /AUC |
|------|-----------------------|-----------|--------------------------------------|---------|---------------|
| 2020 | Wang et al. [71]      | CNN with semi-supervised learning | N A | 2669 Glaucomatous eyes and 1679 normal eyes | AUC:0.977 |
| 2019 | Maetschke et al. [33] | 3D CNN | Augmentation and image resizing | 847 volumes from glaucoma and 263 volumes from healthy | DL model AUC 0.94 |
| 2019 | Ran et al.[50]        | 3D CNN (ResNet) | Data Preprocessing | 1822 glaucomatous eyes and 1007 normal eyes | Primary validation dataset accuracy 91% |
| 2019 | Russakoff et al. [54] | gNet3D | N A | 667 eyes as referable and 428 eyes as non-referable | AUC:0.88 |
| 2019 | Medeiros et al. [37]  | CNN | Image downsampled to 256×256 pixels | 699 glaucoma eyes and 476 normal eyes | Predicted RNFL thickness: 0.944 |
| 2019 | Thompson et al. [67]  | CNN | For ensuring the quality of image, manually reviewed and signal strength of at least 15 dB | 549 glaucoma eyes and 166 normal eyes | DL prediction AUC: 0.945 |
| 2019 | Fu et al. [14]        | CNN (Multi level Deep Learning) | N A | 1102 images with open angle glaucoma and 300 images with angle-closure | AUC:0.95 CI |
| 2019 | Xu et al. [73]        | CNN (ResNet18) | Image Resize | 1943 images with open angle, 2093 images with closed angles | AUC:0.933 |
| 2019 | Hao et al. [21]       | Multi-Scale Regions Convolutional Neural Networks (MSRCNN) | Three scale regions with different sizes are resized to 224×224 | 1024 with open angle glaucoma, 1792 images with narrowed angle subjects for glaucoma and 146 normal subjects | AUC: 0.9143 |
| 2011 | Mwanza et al. [41]    | Statistical Software are used for statistical analysis | Image resized 200×200 pixels | 73 subjects for glaucoma and 146 normal subjects | AUC:0.96 |
| 2012 | Sung et al. [61]      | Statistical analysis variance test and pearson correlation | Image resized 200×200 pixels | 405 glaucoma patients, and 109 healthy individuals | AUC: 0.957 |
| 2012 | Kotowski et al. [26]  | Statistical analysis | OCT image segmentation | 51 healthy, 49 glaucoma suspect and 63 glaucomatous eyes | AUC: 0.913 |
Xu et al. [74] presented an analysis of various ocular diseases using a 3-D standard quantitative SD-OCT. Authors presented data analysis techniques and the use of the 3-D dataset. Superpixels are grouped from adjacent pixels to detect glaucoma damage; machine learning classifiers analysed the proposed algorithm that uses SD-OCT images. In the recently published study on glaucoma, the authors (Ajesh et al. [2]) aimed to boost the sensitivity of glaucoma diagnosis through a novel classification focused on efficient optimisation. The Jaya-chicken swarm optimization (Jaya-CSO) proposal combines the Jaya algorithm with the CSO system to change the RNN classifier weights. Recent research on the use of Deep Learning (DL) on OCT for glaucoma evaluation is summarised in this review, along with the potential clinical implications of developing and using Deep Learning models (Ran et al. [52]). This research describes a method for diagnosing glaucoma using a B-scan OCT picture (Raja et al. [49]). The suggested approach uses a CNN to automatically partition retinal layers. The inner Limiting Membrane (ILM) and retinal pigmented epithelium (RPE) were used to compute the CDR for glaucoma diagnosis. The suggested approach extracts candidate layer pixels using structure tensors and then classifies them using CNN. A VGG16 architecture was used to extract and classify retinal layer pixels. The resultant feature map was combined into a SoftMax layer for classification, producing a probability map for each patch’s centre pixel. The authors described a deep learning architecture for glaucoma screening using OCT images of the optic disc (Wang et al. [71]). The authors used structural analysis and function regression to successfully discriminate glaucoma patients from normal controls. The technique works in two steps: first, they used a semi-supervised learning strategy with a smoothness assumption to give missing function regression labels. The proposed multi-task learning network may then simultaneously explore the structure and function link between OCT image and visual field measurement, improving classification performance.

The authors present and evaluate CNN models for glaucoma detection using OCT RNFL probability maps (Thakoor et al. [66]). Attention-based heat maps of CNN areas of interest imply that adding blood vessel location information might improve these models. They created purely OCT-trained (Type A) and transfer-learning based (Type B) CNN architectures that detected glaucoma using OCT probability map pictures with good accuracy and AUC. Using Grad-CAM attention-based heat maps to highlight locations in pictures helps us learn what causes uncertainty in false positives and false negatives. In this study, Fu et al. [13] presented an automated anterior chamber angle (ACA) segmentation and measuring approach for AS-OCT images. To get clinical ACA measurements, this study introduced marker transfer from labelled exemplars and corneal border and iris segmentation. These include clinical anatomical examination, automated angle-closure glaucoma screening, and massive clinical dataset statistical analysis. They designed a Multi-Context Deep Network (MCDN) architecture in which parallel CNNs were applied to certain picture areas and scales proven to be helpful in identifying angle-closure glaucoma clinically. The purpose of this study was to evaluate the diagnostic capability of swept-source OCT (DRI-OCT) and spectral-domain OCT (Cirrus HD-OCT) for glaucoma (Lee et al. [29]). This study used two OCT systems to measure PP-RNFL, whole macular, and GC-IPL thickness. They measured the PP-RNFL using three-dimensional DRI-OCT scanning with 12 clock-hour sectors. The authors’ objective was to develop a segmentation-free deep learning technique for glaucoma damage assessment utilising SD-whole OCT’s circle B-scan image (Thompson et al. [68]). This single-institution cross-sectional investigation employed SD-OCT images of glaucoma (perimetric and preperimetric) and normal eyes. A SD-OCT circular B-scan without
segmentation lines was used to train a CNN to distinguish glaucoma from normal eyes. The DL algorithm’s estimated likelihood of glaucoma was compared to SD-OCT software’s standard RNFL thickness parameters. They proved that the DL algorithm had larger AUCs than RNFL thickness at all stages of illness, notably preperimetric and moderate perimetric glaucoma. The aim of this study was to evaluate the diagnostic accuracy of two well-known modalities, OCT and fundal photography, in glaucoma screening and diagnosis (Murtagh et al. [40]). A meta-analysis of diagnostic accuracy was done using the AUROC. The meta-analysis included 23 studies. This includes 10 OCT articles and 13 fundus photography papers. The pooled AUROC for fundal pictures was 0.957 (95% CI = 0.917–0.997) and for the OCT cohort was 0.923 (95% CI = 0.889–0.957). OCT pictures help clarify changes in the retina caused by glaucoma, particularly in the RNFL layer and the optic nerve head. In this study, practitioners presented a method for assessing glaucoma using OCT volumes (Gaddipati et al. [16]). This research proposed a deep learning strategy for glaucoma classification using a capsule network directly on 3-D OCT data. The proposed network beats 3-D CNNs while using fewer parameters and training epochs. The reason is that clinical diagnosis requires several criteria, which are difficult to infer from segmented areas. Biomarkers such as RNFL thinning, ONH rim thinning, and Bruch’s membrane opening (BMO) occur in distinct places. As compared to typical convolutional networks, the capsule network integrates geographically scattered information better and requires less training time. The authors attempted to compare OCT and visual field (VF) detection of longitudinal glaucoma progression (Zhang et al. [77]). OCT was utilised to map the thickness of the peripapillary retinal nerve fibre layer and ganglion cell complex (GCC). For the NFL or GCC, OCT-based progression detection was characterised as a trend. They support that in early glaucoma, OCT is more sensitive than VF in detecting progression. While NFL is less effective in advanced glaucoma, GCC appears to be a useful advancement detector. This research presented an algorithmic approach to find major retinal layers in OCT pictures (del Amor et al. [8]). This approach uses an encoder-decoder FCN architecture with a robust post-processing mechanism. This facilitates the investigation of retinal illnesses like glaucoma, which causes RNFL loss. A method for segmenting rodent retinal layers was proposed here. By segmenting the RNFL+GCL+IPL layer, glaucoma can be diagnosed.

OCT is frequently used in clinical practise, but its ideal use is unknown (Tatham et al. [65]). What is the optimal measurement structure? What is a substantial change? affect the patient’s structural alterations. How is ageing affecting longitudinal measures? How can ageing changes be distinguished from real progression? How often and how well should OCT be used in visual fields? These questions were recently studied. The purpose of this study was to evaluate the diagnostic accuracy of circumpapillary retinal nerve fibre layer (cRNFL), optic nerve head, and macular parameters for the detection of glaucoma using Heidelberg Spectralis OCT (McCann et al. [36]). Participants were clinically examined and tested for glaucoma using full-threshold visual field testing. Asymmetric macular ETDRS scans, Bruch’s membrane opening minimum rim width (BMO MRW) scans, and Glaucoma Module Premium Edition (GMPE) cRNFL Anatomie Positioning System scans were used as index tests. This research proposed two deep-learning based glaucoma detection methods using raw circumpapillary OCT images (García et al. [17]). The first involves building CNNs from scratch. The second is to fine-tune some of the most popular modern CNN designs. When dealing with tiny
datasets, fine-tuned CNNs outperform networks trained from scratch. Visual Geometry Group networks show the best results, with an AUC of 0.96 and an accuracy of 0.92 for predicting the independent test set for the VGG family of networks.

The objective of this study was to independently validate the UNC OCT Index’s effectiveness in identifying and predicting early glaucoma (Mwanza et al. [42]). The University of North Carolina Optical Coherence Tomography (UNC OCT) Index technique was used to calculate the average, minimum, and six sectoral ganglion cell-inner plexiform layer (GCIPL) readings from CIRRUS OCT. They support that the UNC OCT Index may be a better tool for early glaucoma identification than single OCT characteristics. The purpose of this study is to find out what causes peripapillary retinal splitting (schisis) in glaucoma sufferers and those at risk (Grewal et al. [19]). The result was measured using OCT raster images that revealed peripapillary retinal splitting. A majority of the affected eyes had adherent vitreous with traction and peripapillary atrophy. To determine whether this is a glaucoma-related condition, an age and axial length-matched cohort is required. The goal of this work was to create a 3-D deep learning system using SD-OCT macular cubes to distinguish between referable and nonreferable glaucoma patients using real-world datasets (Russakoff et al. [54]). To achieve consistency, the cubes were first homogenised using Orion Software (Voxeleron). The system was validated on two external validation sets of Cirrus macular cube scans of 505 and 336 eyes, respectively. Researchers conclude that retinal segmentation preprocessing improves 3-D deep learning algorithms’ performance in detecting referable glaucoma in macular OCT volumes without retinal illness. This study proposed a hybrid computer-aided-diagnosis (H-CAD) system that combines fundus and OCT imaging technologies (Shehryar et al. [57]). The OCT module computes the cup-to-disc-ratio (CDR) by studying the retina’s interior layers. The cup shape was recovered from the ILM layer using unique approaches for calculating cup diameter. Similarly, to calculate disc diameter, a variety of novel methods have been developed to determine the disc boundary. A novel cup edge criteria based on the mean value of RPE-layer end points has also been developed. A novel approach for reliably extracting the ILM layer from SD-OCT pictures has been presented. In this recently published study, the authors’ objective was to detect glaucoma from raw SD-OCT volumes using a Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) method (Sunija et al. [62]). An SD-OCT-based depthwise separable convolution model for glaucoma detection was proposed. An innovative new slide-level feature extractor (RAGNet) was suggested. A novel method for extracting the region of interest from each B-scan of the SD-OCT cube was proposed. An efficient, redundant-reduced 2D depthwise separable convolution algorithm was suggested for glaucoma diagnosis using retinal OCT. For the first time, authors suggested a deep-learning approach based on the spatial interdependence of features taken from B-scans to examine the hidden knowledge of 3-D scans for glaucoma identification (García et al. [18]). Two distinct rounds of training were proposed: a slide-level feature extractor and a volume-based prediction model. The proposed model predicted volume-level glaucoma by taking features from B-scans of volumes and combining them with information about the hidden space around them.

After going through the comprehensive literature survey presented above, not many studies are still published directly by OCT on the use of image networks for glaucoma detection; meanwhile, in the first section, several benefits of OCT have already been shared. This identified research gap motivates us to present an experimentally based study for early and timely glaucoma detection in patients with the help of the latest image nets. The best performing net(s) can be suggested for its practical usability, minimizing intra-observer variability and reducing the burden of overloaded expert ophthalmologists.
3 Proposed framework

The section is bifurcated into three subsections. The first sub-section is dedicated to the proposed methodology. In the next subsection, a brief note on preprocessing is presented; finally, a detailed description of the various selected architectures is offered in the last subsection.

3.1 Proposed methodology

Deep Learning is a subset of machine learning that includes multiple hidden layers made up of neurons. It is based on the human brain’s functions and structure, which is termed an “artificial neural network” (ANN). ANN recognizes patterns and takes decisions like humans. CNN is an approach to deep learning that solves complex problems (towardsdatascience.com/the-most-intuitive-and-easiest-guide-for-convolutional-neural-network-). The CNN consists of layers such as convolutional layers, fully connected layers, max-pooling layers, dropout layers, and activation functions. The sequential filters conducting 2D convolution on the input picture are the coevolutionary layers. The characteristic map is the output of the coevolutionary sheet. The dropout layer eliminates waste. Neurons are fully interconnected with each other in the final layer, known as the “Fully Connected Layer.” ReLU (Rectified linear unit) and Softmax are used as activation functions after convolution layers. The cross-entropy loss function trains the CNN. Deep Learning CNN uses a large number of images for training.

3.1.1 Data set

The shortlisted data set consists of 2 categories of OCT images (Glaucoma and Normal OCT Images). A private dataset, approved by expert ophthalmologists, used in this study consists of a total of 4200 images (Table 2). The dataset consists of 4000 images as a training dataset and 220 images for testing purposes. The dataset is used to train and test the different convolutional neural networks using TensorFlow. The dataset is used on eight different Image-Net (VGG16, VGG19, InceptionV3, DenseNet, Efficient Net (B0 and B4), Xception Net, and Inception-ResNet-V2) trained models.

3.2 Preprocessing

The images are processed in a preprocessing step to a common format and scaled depending on the various CNNs (224 × 224, 299 × 299) to convert them into the homogeneous format. In preprocessing, we use rescaling, rotation range, height shift range, and horizontal flip. All the 4220 images are preprocessed evenly.

| Data Set                | Format | Glaucoma | Normal |
|-------------------------|--------|----------|--------|
| Train (Private Dataset) | JPG    | 2000     | 2000   |
| Test (Private dataset)  | JPG    | 110      | 110    |
| **Total**               | **JPG**| **2110** | **2110** |
3.3 Architectures used

For this experimental study, eight different CNNs were preferred for detecting glaucoma automatically. Fig. 4 demonstrates the workflow of the proposed method for efficient glaucoma detection. Three separate optimizers, stochastic gradient descent (SGD), Root Mean Squared Propagation (RMSProp), and Adam, are used to train CNN. For the detection of glaucoma, transfer learning and fine-tuning approaches have also been implemented. Transfer learning is a method of taking functionalities learnt from a problem and applying them to a different and related problem. The system is performed when the dataset is too small to train a full-scale model. The workflow of transfer learning consists of taking a layer from a previously learned model and then freezing it, and then placing new layers on top of it. After applying new layers to our dataset, the training processes were completed. Fine-tuning means rerunning the whole model we got with extra care to make sure it works with the new data.

3.3.1 VGG16

VGG16 is a CNN with pooling layers, max-pooling layers, and completely linked layers. The 16 in VGG stands for the 16 layers that hold weights. The input has a set scale. It passes through a convolutional layer, as seen in Fig. 5, where each filter is used with a very limited receptive field. Convolutional layers (max pooling) are accompanied by five spatial pooling layers. Max-pooling is done on a two-by-two pixel window, followed by three entirely linked layers. The final layer is softmax. The totally linked layers are all the same in all networks.

![Fig. 4 A proposed deep learning-based framework, for performance evaluation of various image nets, to aid early glaucoma detection from OCT images.](image-url)
Four layers, including the Flatten layer, two Dense Layer, and Dropout layer, were used as the softmax layer in this proposed method. The summary of fine-tuned is depicted below (Table 3).

### 3.3.2 InceptionV3

InceptionV3 is a pre-trained neural network model for image recognition which usually consists of two different parts:

a) Feature Extraction part within CNN, and  
b) Classification part including fully connected and softmax layer

InceptionV3 consists of features from both V1 and V2, and includes batch normalization processing. This network consists of 48 deep layers. This model is used as a multi-level feature extractor (pyimagesearch.com/2017/03/20/imagenet-vggnet-resnet-inception-xception-keras/).

A simple architecture is shown in Fig. 6. It is shown that the Inception Net is the combination of $1 \times 1$ convolutional layer, $3 \times 3$ convolutional layer, and $5 \times 5$ convolutional layers with their output concatenated into a single output vector, resulting in the input for the next level [11]. This net input is $(229 \times 229)$, followed by a combination of convolutional layers using ReLU as its activation function. The output of these layers is passed to the transition layer, which concatenates all the convolutional layers. The weights of VGG and ResNet are greater than in Inception V3. The below-shown Table 4 depicts the summary of fine-tuned InceptionV3 [63].

#### Table 3 Summary of Fine-tuned VGG16

| Layer          | (type)      | Output Shape     | Param #   |
|----------------|-------------|------------------|-----------|
| VGG16          | (Model)     | (None, 7, 7, 512)| 14,714,688|
| flatten_1      | (Flatten)   | (None, 25,088)   | 0         |
| dense_1        | (Dense)     | (None, 1024)     | 25,691,136|
| dropout_1      | (Dropout)   | (None, 1024)     | 0         |
| dense_2        | (Dense)     | (None, 2)        | 2050      |
| **Total parameters** |           |                  | **40,407,874** |
| **Trainable parameters** |           |                  | **25,693,186** |
| **Non-trainable parameters** |           |                  | **14,714,688** |
3.3.3 VGG19

VGG19 is a publicly available CNN model. It includes five stacks, and every stack consists of between two and four convolutional layers; then it is followed by the max-pooling layer [18]. In the end, there are three fully connected layers. This architecture improves the depth of the neural network and is used in $3 \times 3$ convolutional filters. Figure 7 shows the basic architecture of VGG19. This model consists of 19 layers, including 16 convolutional layers, three fully connected layers, five max-pooling layers, and one softmax layer at the end of the network. VGG16 (16 layers) and VGG19 (19 layers) have worked with minimal $5 \times 5$ and $7 \times 7$ scale kernels in convolutional layers, rather than providing a large number of hyperparameters. In the centre, the convolutional net has two entirely connected layers, followed by a softmax. When the two VGG models are compared, it is observed that the current model includes an extra layer in each of its three convolutional blocks. The below-shown Table 5 depicts the summary of fine-tuned VGG19.

![InceptionV3 architecture](image)

**Table 4** Summary of Fine-tuned InceptionV3

| Layer       | (type)                     | Output Shape       | Param #       |
|-------------|-----------------------------|--------------------|---------------|
| InceptionV3 | (Model)                     | (None, 8, 8, 2048) | 21,802,784    |
| Avg_pool    | (GlobalAveragePooling)      | (None, 2048)       | 0             |
| dropout_1   | (Dropout)                   | (None, 2048)       | 0             |
| dense_1     | (Dense)                     | (None, 2)          | 4098          |
| Total parameters |                          |                    | 21,806,882    |
| Trainable parameters |                  |                    | 4098          |
| Non-trainable parameters |                |                    | 21,802,784    |
3.3.4 Inception-ResNet-V2

Inception-ResNet-V2 is a convolutional neural network trained in the ImageNet database on more than a million images. This network contains 164 neural network deep layers and can classify images into over 1000 categories of objects. The input size of the images used in this network is 299 × 299. Due to the vast range of differentiation in the categories of image classification (keras.io/applications/#Inception-ResNet-V2), the network has mastered the rich feature of representation for a wide variety of pictures [11]. Figure 8 shows the simple architecture of Inception-ResNet-V2. It can be observed that the input layer takes an input of size 299 × 299. This model is better because it combines the advantages of both residual connections and the inception model [31]. This model uses convolutional filters used in parallel and that are of different sizes on the same input map. Their results are connected to the paired output. It is also known as the scaling model because it integrates the information into a multi-scale level and ensures the model’s better performance. The input layer is followed

Table 5  Summary of Fine-tuned VGG19

| Layer     | (type)   | Output Shape       | Param #   |
|-----------|----------|---------------------|-----------|
| VGG19     | (Model)  | (None, 7, 7, 512)   | 20,024,384|
| flatten_1 | (Flatten) | (None, 25,088)     | 0         |
| dense_1   | (Dense)  | (None, 1024)       | 25,691,136|
| dropout_1 | (Dropout)| (None, 1024)       | 0         |
| dense_2   | (Dense)  | (None, 2)          | 2050      |
| Total parameters |           |                     | 45,717,570|
| Trainable parameters |   |                     | 32,772,610|
| Non-trainable parameters |  |                     | 12,944,960|
by the Steam block (towardsdatascience.com/a-simple-guide-to-the-versions-of-the-inception-network-7fc52b863202) which means that before the introduction of the inception block, some initial operation has to be performed, and those operations are labelled as Steam. InceptionV3 is an updated version of Inception V1 and Inception V2 with more parameters; it differs from ordinary CNN in the layout of the original blocks, requiring the input tensor to be lapped with several filters and its effects to be concatenated. In particular, it has a block of parallel convolutionary layers with 3 different filter sizes (1 × 1, 3 × 3, and 5 × 5) [12]. In addition, 3 × 3 max pooling is also done. Outputs are concatenated and forwarded to the next inception module.

As shown in the fig below, there are 3 Inception ResNet blocks labelled as A, B, and C. The Reduction Blocks A and B are used to reduce the input size coming to the block from 35 × 35 to 17 × 17 in Reduction Block A and from 17 × 17 to 8 × 8 in Reduction Block B. An average pooling layer is used for dimension reduction. Once it is done, a dropout layer is added to weight it has done, and the output is passed to a Softmax layer. Table 6 represents a summary of the fine-tuned Inception-ResNet-V2.

Table 6 Summary of Fine-tuned Inception-ResNet-V2

| Layer         | (type)  | Output Shape          | Param #       |
|---------------|---------|-----------------------|---------------|
| Inception_resnet_v2 | (Model) | (None,5, 5, 1536)    | 20,024,384    |
| flatten_2     | (Flatten) | (None,38,400)     | 0             |
| dense_3       | (Dense)  | (None, 256)         | 9,830,656     |
| dense_4       | (Dense)  | (None, 2)           | 514           |
| Total parameters |         |                      | 64,167,906    |
| Trainable parameters |     |                      | 64,107,362    |
| Non-trainable parameters | |                      | 60,544        |
3.3.5 Xception

Xception, Chollet towardsdatascience.com/the-most-intuitive-and-easiest-guide-for-convolutional-neural-network-, created by Google, stands for the Extreme Version of Inception, is a deep-separable, updated convolution, which is much stronger than Inception-v3. Depthwise separable convolutions will replace the inception units. Xception is a deep CNN architecture containing depth-wise separable convolutions. A Changed Depth Wise Separable Convolution is a point-wise convolution preceded by a deep-wise convolution. Xception, a form of convolutional neural network, is an enhanced version of the Inception architecture and includes deep separable convolutions and 71 layers deep. It replaces regular Inception modules with depth-separable convolutions [47]. In classical classification problems, the model yielded favourable results. As seen in Fig. 9 below, Separable Conv2d is a modified depth of wise separable convolution. Separable Convolution is viewed as Inception Modules with a maximum number of towers. It consists of 71 layers, consisting of 36 convolutionary layers, forming the foundation of the extraction features of the network and organised into 14 modules, which are linear residual links.

Fine-tuning is applied to Xception Net by adding the new layers at the end of the standard Xception Net and training them for ten epochs to get an accuracy of 90% and an AUC score of 0.9 RMSProp optimizers. The network has an input size of 224 × 224 with a global average pooling layer and a dense layer added to the fine-tuned Xception Net. Xception Net is loaded from the Keras library (https://keras.io/guides/). The details of the layers added, their output shape and total parameters are listed in Table 7.

3.3.6 EfficientNet

Neural architecture is used to build a modern baseline network and is further scaled to create a family of models known as Productive Nets. Effective Nets by Tan and Le [64] give better accuracy and reliability than convolutional nets. The goal of deep learning architectures is to reveal more effective approaches to smaller models, and EfficientNet is one of them, using a new activation function called Swish instead of the Rectifier Linear Unit (ReLU) Activation function. EfficientNet achieves more efficient performance by scaling depth, width, and resolution equally when scaling down the model. The first step in the compound scaling process is to look for a grid to find a connection between the different scaling dimensions of

![Fig. 9 Xception architecture](image-url)
the baseline network under a fixed resource constraint [35]. An effective scaling factor for depth, width, and resolution measurements is calculated in this manner. These coefficients are then used to scale the baseline network to the optimal target network. The key building block for EfficientNet is the inverted bottleneck MBConv; blocks consist of a layer that first extends and then compresses channels, so that direct connections are used between bottlenecks that bind far fewer channels than expansion layers [35]. This design has in-depth separable convolutions that minimise the measurement by almost k compared to conventional layers, where k is the size of the kernel that denotes the width and height of the 2D convolution window. It consists of eight B0 to B7 models. EfficientNetB7 gives ImageNet the best precision. It’s 8.4x smaller and 6.1x faster than the top convolutional net. The architecture of EfficientNet is shown in Fig. 10. The input size of 224 × 224 is used. The MBConv Block is an inverted Residual Block (which is also used in MobileNetV2). In this proposed work, EfficientNetB0 and EfficientNetB4 are used with all three optimizers, mainly SGD, RMSProp, and Adam. It was observed that Efficient Net B4 gave a much higher accuracy than EfficientNetB0. Fine-tuning is applied to EfficientNetB0 and EfficientNetB4 by adding the new layers at the end of the standard EfficientNetB0 and freezing all the standard layers. EfficientNetB0 is the baseline model. The fine-tuned EfficientNetB0 is trained over 20 epochs to get higher accuracy and to minimize loss. An accuracy of 77.5% and an AUC score of 0.77 were achieved with EfficientNetB0 using the Adam optimizer. The layers’ details, output shape, and total parameters used in EfficientNetB0 are listed below in Table 8. EfficientNetB4 is fine-tuned by adding new layers at the end of the standard EfficientNetB0 and freezing all the standard layers and training the newly added layers. Layers added include Global Pooling, Dense, and Drop Out Layers. EfficientNetB0 is the baseline model through which Efficient Net B4 is developed. The fine-tuned EfficientNetB4 is trained over 20 epochs to get higher accuracy and to minimize loss. An accuracy of 88.5% and an AUC score of 0.87

![EfficientNet architecture](image-url)

**Table 7** Summary of Fine-tuned Xception

| Layer       | (type)           | Output Shape       | Param #  |
|-------------|------------------|--------------------|----------|
| Xception    | (Model)          | (None, 10, 10, 2048) | 20,861,480 |
| Avg_pool    | (GlobalAveragePool) | (None, 2048)    | 0        |
| dropout_1   | (Dropout)        | (None, 2048)      | 0        |
| dense_1     | (Dense)          | (None, 2)         | 4098     |
| Total parameters |                |                    | **20,865,578** |
| Trainable parameters |          |                    | 4098     |
| Non-trainable parameters |           |                    | **20,861,480** |
were achieved with EfficientNetB4 using the Adam optimizer. The layers added, their output shapes, and total parameters used in EfficientNetB4 are listed below in Table 9 in detail.

### 3.3.7 DenseNet

A Dense Convolutional Network (DenseNet), developed by Huang et al. [22], is a CNN that connects each layer to every other layer in a feed-forward fashion. DenseNets are different from ResNets as they do not sum output feature maps of the layer with incoming feature maps instead of concatenating them. Each layer has access to the preceding feature maps. Each layer adds new information to feature maps. DenseNets consist of DenseBlocks. The dimensions of the feature map remain constant in a denseblock, but the number of filters is variable. The layers between dense blocks are known as transition layers. These layers apply downsampling with batch normalization, 1 × 1 convolution, and 2 × 2 pooling layers, as depicted in Fig. 11. DenseNet alleviates the vanishing-gradient problem, strengthens feature propagation, encourages feature reuse, and uses fewer parameters, which are a few advantages. DenseNet201 (Dense Convolutional Network) is a convolutionary neural network with a depth of 201 layers [32]. It is an upgrade of the ResNet that involves dense links between layers. It binds each layer to the other in a feed-forward manner. Unlike typical L-layer convolution networks with L-connections, DensNet201 has L (L + 1)/2 direct connections. Indeed, relative to conventional networks, DenseNet can boost efficiency by increasing the computing criteria, reducing the number of parameters, promoting the re-use of features, and improving the dissemination of features [32]. The network has an input size of 224 × 224. The summary of the fine-tuned DenseNet is shown below (Table 10).

### Table 8 Summary of Fine-tuned EfficientNetB0

| Layer        | (type)               | Output Shape   | Param #   |
|--------------|----------------------|----------------|-----------|
| EfficientNetB0 | (Model)              | (None, 7, 7, 1280) | 4,049,564 |
| Avg_pool     | (GlobalAveragePool)  | (None, 1280)   | 0         |
| dropout_1    | (Dropout)            | (None, 1280)   | 0         |
| dense_1      | (Dense)              | (None, 2)      | 2562      |
| Total parameters |                    |                | 4,052,126 |
| Trainable parameters |            |                | 784,002   |
| Non-trainable parameters |            |                | 3,268,124 |

### Table 9 Summary of Fine-tuned EfficientNetB4

| Layer        | (type)               | Output Shape   | Param #   |
|--------------|----------------------|----------------|-----------|
| EfficientNetB4 | (Model)              | (None, 7, 7, 1792) | 17,673,816 |
| Max_pool     | (GlobalAveragePool2D)| (None, 1792)   | 0         |
| dropout_1    | (Dropout)            | (None, 1792)   | 0         |
| Fc_out       | (Dense)              | (None, 2)      | 3586      |
| Total parameters |                    |                | 17,677,402 |
| Trainable parameters |            |                | 809,986   |
| Non-trainable parameters |            |                | 16,867,416 |
4 Experimental results

This is the most extensive section, divided into seven sub-sections (4.1–4.7). Each sub-section is dedicated to one of the major parameters related to simulated results, like hyperparameters used (4.1), evaluation of performance metrics (4.2), training and testing datasets (4.3), computed results on various CNNs (4.4–4.5), and finally dataset influence (4.6) and expected computation time (4.7).

4.1 Hyperparameters used

The hyperparameters are tunable and set manually at the beginning of the learning process. Many hyperparameters that are used in the study, like learning rate, loss function, and optimizer, are listed in detail in Table 11. The validation set is used to set the number of epochs for each network used in this methodology. The models are trained and fine-tuned during the implementation. Three different optimizers are used on each net: Adam, SGD, and RMSprop. All three optimizers are trained at different learning rates depending upon the accuracy they receive through the networks. The loss function remains constant, i.e., the categorical cross-entropy loss function. The batch size of 100 for the training dataset and 10 for the validation dataset remains throughout the transfer learning networks. The Adam optimizer is used with a decay rate of 0.1 in every selected model. The following hyperparameters (batch size, learning rate, loss function, training size, validation size, image size, epochs, momentum,

Table 10  Summary of Fine-tuned DenseNet

| Layer                      | (type)          | Output Shape     | Param #          |
|----------------------------|-----------------|------------------|------------------|
| DenseNet201                | (Model)         | (None, 7, 7, 1920) | 18,321,984       |
| Global_average_pooling2d_5 | –               | (None, 1920)     | 0                |
| Dense_7                    | (Dense)         | (None, 1024)     | 1,967,104        |
| Dropout_3                  | (Dropout)       | (None, 1024)     | 0                |
| Dense_8                    | (Dense)         | (None, 2)        | 2050             |
| Total parameters           |                 |                  | **20,291,138**   |
| Trainable parameters       |                 |                  | **1,972,994**    |
| Non-trainable parameters   |                 |                  | **18,318,144**   |
| Model         | Batch Size | Loss function                  | Adam Epochs | Adam Learning Rate | T-Acc | SGD Epochs | SGD Learning Rate | T-Acc | RMSprop Epochs | RMSprop Learning Rate | T-Acc |
|--------------|------------|--------------------------------|-------------|--------------------|------|------------|-------------------|------|----------------|------------------------|------|
| VGG16        | 10         | Categorical cross-entropy      | 5           | 0.0001             | 92.2 | 10         | 0.0001            | 88.9 | 5              | 0.0001                 | 99.5 |
| VGG19        | 10         | Categorical cross-entropy      | 10          | 0.0001             | 99.1 | 10         | 0.0001            | 76.0 | 10             | 0.0001                 | 99.3 |
| DenseNet     | 10         | Categorical cross-entropy      | 20          | 0.0001             | 83.0 | 20         | 0.0001            | 83.6 | 20             | 0.0001                 | 96.8 |
| Inception    | 10         | Categorical cross-entropy      | 40          | 0.0001             | 94.0 | 30         | 0.0001            | 94.5 | 15             | 0.0001                 | 98.5 |
| Xception     | 10         | Categorical cross-entropy      | 10          | 0.0001             | 60.5 | 20         | 0.0001            | 98.5 | 10             | 0.0001                 | 98.9 |
| Inception-ResNet-v2 | 10   | Categorical cross-entropy      | 30          | 2e-3               | 87.9 | 30         | 2e-3              | 84.7 | 30             | 2e-3                   | 94.1 |
| EfficientNetB0 | 10         | Categorical cross-entropy      | 20          | 0.0001             | 98.3 | 20         | 0.0001            | 94.7 | 20             | 0.0001                 | 93.2 |
| EfficientNetB4 | 10         | Categorical cross-entropy      | 20          | 0.0001             | 96.1 | 20         | 0.0001            | 94.2 | 20             | 0.0001                 | 94.9 |

*T-Acc means Training Accuracy
dropout value, and activation function) are fine tuned using a hit-and-trial approach to generate the best performance.

The validation accuracy of 99.09% is the highest when the VGG16 model uses RMSprop as an optimizer. In the case of DenseNet, using Adam optimizer with a learning rate of 0.0001, the model’s validation accuracy gradually increased from 72.73% to 85.45% in 20 epochs. The model is then implemented with the SGD optimizer and RMSprop gives a validation accuracy of 80.91% and 85.91% in 20 epoch runs. It is found that the best validation accuracy is received through the RMSprop optimizer. InceptionNet uses the Adam optimizer with 40 epochs run and receives a validation accuracy of 75.91%. The model is trained with the SGD optimizer, which gives a validation accuracy of 75% after 30 epochs. Finally, the model uses the RMSprop optimizer, which, after 15 epochs, receives a 72.2% validation accuracy. After plotting the respective graphs for all the optimizers, it is established that InceptionNet is best trained on the Adam optimizer. When trained with the SGD optimizer, XceptionNet gives a validation accuracy of 72.27% after a 20-epoch run. When trained on Adam and RMSprop optimizers, the model after ten epochs gives validation accuracy of 92.73% and 90.45%, respectively. The model gives the best validation accuracy with the Adam optimizer. Inception-ResNet-v2 is trained with 30 epochs run on every optimizer with a learning rate of $2e^{-3}$ for better accuracy. The validation accuracy is 84% with Adam, 81% with SGD, and 91% with RMSprop after 30 epochs. The experiment shows that RMSprop performance is best with Inception-ResNet-v2. EfficientNetB0 is trained on all three optimizers, which gives 77% validation accuracy and 98% training accuracy on the Adam optimizer, which is sighted as the best accuracy of all the three optimizers with 20 epochs. It is implemented, giving 89%
validation accuracy with the Adam optimizer, with a total of ten epochs run, the highest among all the three optimizers. 81% accuracy with the SGD optimizer and 87% accuracy with the RMSProp optimizer with a total of 20 epochs is achieved.

4.2 Performance metrics

The confusion matrix is a performance calculation used to solve classification problems in deep learning where output can be two or more classes (refer Table 12). It is a table composed of four distinct variations of predicted and real values. The confusion matrices are on the order of 2 x 2. The matrix of uncertainty consists of four values: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). In this analysis, various normative performance assessment metrics such as accuracy, precision, recall, F-score, AUC, sensitivity, and specificity have been computed. These are based on four words, i.e., TP, FP, TN, and FN. TP applies to patients who have a disorder and the result is positive, while FP is positive for patients who do not have a disease but the test is positive. Similarly, TN applies to people who do not have an illness, and the result is negative, while FN is a disease patient, but the test is negative.

Sensitivity (or recall or a true positive rate) is the number of true positives separated by the sum of the true positives and false negatives and is one of the best measures of diagnostic precision. The evaluation shows us the frequency at which CNN accurately predicts glaucoma. The sum of sensitivity and false negative incidence is the number of true negatives divided by the total number of checks. The accuracy is defined as the number of true negatives divided by the sum of the true negatives and false positives. The sensitivity measure calculates just how many of the positive cases were correctly predicted, while the specificity measure determines just how many of the negative cases were correctly predicted. The research shows that CNN’s medical reporting is reliable for individuals who don’t have glaucoma. The recall is the number of accurate outcomes. Precision is how close to the target the sample comes. The word F1-score tests the accuracy of a source. The F1-score is a metric used to gauge model accuracy. The method incorporates the accuracy and recall of the model in order to determine a single value, the harmonic mean. Precision is the amount of significance the measurement has on the outcome. Accuracy is the most important metric used to measure the efficiency of machine learning classifiers. The algorithm measures the ratio of correctly defined images to the overall number of images in the dataset. Sensitivity and specificity scores help assess the accuracy of performance. A receiver operating characteristic curve (ROC curve) is a graph of a classification model which shows how well a classification performs at decreasing classification thresholds. A ROC curve maps true and false positives against true positives at differing degrees of classification. The ROC study shows a reliable calculation of glaucoma classification. The region under the ROC curve indicates the sensitivity of the measurement. The field

| Table 12 | Confusion matrix |
|----------|------------------|
|          | Predicted cases  |
|          | Glaucoma         | Healthy (Non-Glaucoma) |
| Actual cases | True Positive (TP) | False Negative (FN) |
| Positive    | False Positive (FP) | True Negative (TN) |
| Negative    |                   |                       |

MultiMedia Tools and Applications (2022) 81:27737–27781
under the curve is a partially-independent vector. It explores how our CNN is able to discriminate between regular and glaucoma patients Fig. 12.

Raising the True Positive rate would boost both the False Positive rate and the True Positive rate. The area under the receiver operating characteristic curve (AUC) is used to assess the model’s performance; the area is the entire two-dimensional area under the receiver operating characteristic curve from (0,0) to (1,1). This is an overall indicator of success across every conceivable degree of identity.

\[
\text{Sensitivity (or Recall)} = \frac{\text{True positives}}{\text{True Positives + False Negatives}}
\]

\[
\text{Specificity} = \frac{\text{True negative}}{\text{True Negatives + False positives}}
\]

\[
\text{Precision} = \frac{\text{True positives}}{\text{True Positives + False Positives}}
\]

\[
\text{Accuracy} = \frac{\text{True positives + True Negatives}}{\text{True Positives + False Positives + False Negatives + True Negatives}}
\]

### 4.3 Training and testing

During this study, in its initial phase, work was performed on 4220 OCT images for the detection of glaucoma. The training dataset consists of 2000 glaucoma images and 2000 normal images, while the testing set consists of 110 images of glaucoma and 110 normal OCT images. Data is to be categorized into two classes: glaucoma and normal images. To attain enhanced performance, a different number of epochs is chosen for the training process in each model, which is shown in Table 13 for fine-tuned models. All the models are evaluated on three different optimizers, including Adam, RMSprop, and SGD. A loss function is selected by experimenting between binary cross-entropy and categorical cross-entropy. Categorical cross-entropy is used for training the models because it provides better results.

In Table 13, the number of epochs used for each CNN with different optimizers is shown. A cross-entropy loss function with a batch size of 10 images has been used for evaluating the various selected CNN’s. In the case of EfficientNet B0 with SGD optimizer, the average running time is 55.2 s per epoch, and in the case of Xception with RMSProp optimizer, it is 134.3 s per epoch.

### 4.4 Comparing CNN algorithms

After the selection of various hyper-parameters used in the study, evaluation of all the architectures (VGG16, VGG19, InceptionV3, Xception, DenseNet, Inception-ResNet-V2, Efficient Net B0, Efficient Net B4) in terms of the evaluation metrics is performed, using all the available data for training and testing. On the selected hyperparameters mentioned in
Tables 11 and 13, each model is trained and evaluated based on its performance metrics. It is observed that by using the “imagenet” weights, there is an improvement in results.

Fig. 13 demonstrates the confusion matrix of all the trained models, with all three optimizers, implemented in the proposed work. Based on these confusion matrices, the performance metrics of each net are calculated, which is shown in Table 14 (consisting of 3 sub-Tables 14A, 14B, and 14C). From keenly observing this table, it is derived that the most promising fine-tuned models on the training and testing data are VGG16, Xception, Inception-ResNet-V2, and Efficient Net. The sensitivity and specificity were also calculated using a confusion matrix. Tables 15 and 16 prove that the best performing models for our work are VGG16, Inception-ResNet-V2, and Xception using the RMSProp optimizer with an accuracy of 99.09%, 91.81%, and 90.45%, and an AUC score of 0.99, 0.91, and 0.90, respectively.

With the help of Tables 14 and 15, it is derived that VGG16 with RMSProp optimizer with SGD optimizer gave the highest AUC value of 0.99. It is also observed that the highest sensitivity and specificity are shown by the VGG16 Image Net using RMSProp Optimizer. The specificity is 0.9818, sensitivity is 1.0, Precision is 0.98, Recall 1.0 and the F1 score is 0.99. To recognize glaucoma, high specificity and high sensitivity are required. The higher the AUC score of an Image Net, the better the accuracy of the CNN to identify glaucoma from OCT images. Table 14A, B, and C are compiled to depict the performance of shortlisted models on different selected optimizers when a private dataset was on hand. Table 14A clearly indicates that in the case of the ADAM optimizer, VGG16 presents the best show on 4 out of 7
parameters with the highest accuracy of 98.18%. Similarly, 14.2 shows that VGG16 outperforms others in 4 parameters again (out of 7). Finally, in the case of the RMSProp optimizer, VGG16 again comes out to be the best, outperforming others in 5 out of 7 parameters. It is also observed that the InceptionV3 model presents the weakest show in all three cases. The difference between the best and the weakest model is more than 30%, specifically in the case of the accuracy parameter; the approximately same pattern is also observed in Tables 14B and C, where the difference between the best and the weakest model is not less than 30%, in the case of accuracy as the performance metric. In terms of sensitivity, EfficientNetB0 performs the worst of all models across all optimizers, while VGG16 performs the best in all cases (optimizers). The difference in sensitivity between the best and worst performing models is also significant (more than 50%). Table 16 is composed to show the AUC score (scale-invariant, which measures how well predictions are ranked rather than their absolute values) with all the three optimizers. The table clearly indicates that VGG16 shows incredible performance again by scoring the top rank with a value of more than 0.97 in all three optimizer cases.

In all three cases, the difference between the best and weakest models is greater than 35%; the best model remains constant while the weakest performer changes in different optimizers; a pictorial representation of this is shown in Fig. 14. Table 17 depicts the performance of the best 4 models (selected on the basis of performance shown in Table 15) on the standard benchmark public dataset. Here in this case too, VGG16 steals the show by showing an overwhelming response of more than 95% accuracy and a 0.95 AUC score. In the case of the accuracy metric, the difference between the VGG16 and the weakest performer is approximately 15% (a significant difference), and in the case of AUC, another significant performance evaluation metric, the difference between the best performer and the weakest performer (Inception-ResNet-V2) is more than 30%. However, in terms of precision and specificity, Inception-ResNet-V2 achieves the full score. Regarding other significant parameters (F1-Score, recall, and sensitivity), the VGG16 again outperforms other significant models. During overall

![Fig. 13 Confusion Matrix for: a)VGG16 (SGD), b)VGG16 (RMSProp), c) VGG16 (Adam), d)VGG19 (SGD), e) VGG19 (RMSProp), f) VGG19 (Adam), g) InceptionV3(SGD), h) InceptionV3(RMSProp), i) InceptionV3(Adam), j)Xception(SGD), k)Xception(RMSProp),l)Xception(Adam), m)DenseNet(SGD), n)DenseNet (RMSProp), o)DenseNet(Adam), p)Inception-ResNet-V2(SGD), q) Inception-ResNet-V2 (RMSprop), r) Inception-ResNet-V2(Adam), s)Efficient Net B0 (SGD), t)EfficientNetB0(RMSProp), u)EfficientNetB0(Adam), v)EfficientNetB4(SGD), w)EfficientNetB4(RMSProp), x)EfficientNetB4 (Adam)
### Table 14 Performance Metrics of all CNN’s evaluated

| A: Performance Ratios using ADAM optimizer on the various CNN’s |  |  |  |  |  |  |  |
|---------------------------------------------------------------|---|---|---|---|---|---|---|
| Model             | Accuracy(%) | Precision | F1-score | AUC  | Sensitivity | Specificity | P value |
| VGG16             | 98.18       | 0.97      | 0.98     | 0.98 | 0.990       | 0.9727      | 0.3370 |
| VGG19             | 87.72       | 0.97      | 0.86     | 0.88 | 0.781       | 0.9727      | 0.1556 |
| DenseNet          | 85.45       | 0.87      | 0.85     | 0.85 | 0.836       | 0.8727      | 0.1388 |
| InceptionV3       | 75.90       | 1.00      | 0.68     | 0.76 | 0.518       | 1.00        | 0.0114 |
| Xception          | 92.72       | 0.98      | 0.92     | 0.93 | 0.872       | 0.9818      | 0.2224 |
| Inception-ResNet-v2| 83.63      | 0.95      | 0.81     | 0.84 | 0.951       | 0.7681      | 0.1164 |
| EfficientNetB0    | 77.72       | 1.00      | 0.71     | 0.77 | 0.55        | 1.00        | 0.0032 |
| EfficientNetB4    | 88.63       | 1.00      | 0.87     | 0.88 | 0.77        | 1.00        | 0.1310 |

| B: Performance Ratios using SGD optimizer on the various CNN’s |  |  |  |  |  |  |  |
|---------------------------------------------------------------|---|---|---|---|---|---|---|
| Model             | Accuracy | Precision | F1-score | AUC  | Sensitivity | Specificity | P value |
| VGG16             | 98.18     | 0.97      | 0.98     | 0.98 | 0.990       | 0.9727      | 0.3370 |
| VGG19             | 83.63     | 0.96      | 0.81     | 0.84 | 0.700       | 0.9727      | 0.1260 |
| DenseNet          | 80.90     | 0.97      | 0.77     | 0.81 | 0.636       | 0.9818      | 0.0013 |
| InceptionV3       | 75.00     | 1.00      | 0.67     | 0.75 | 0.500       | 1.0000      | 0.0007 |
| Xception          | 72.27     | 0.72      | 0.72     | 0.72 | 0.727       | 0.7181      | 0.0011 |
| Inception-ResNet-v2| 81.81     | 0.97      | 0.78     | 0.82 | 0.972       | 0.7397      | 0.0574 |
| EfficientNetB0    | 73.63     | 1.00      | 0.64     | 0.73 | 0.47        | 1.00        | 0.0012 |
| EfficientNetB4    | 80.91     | 1.00      | 0.76     | 0.81 | 0.62        | 1.00        | 0.0051 |

| C: Performance Ratios using RMSProp optimizer on the various CNN’s |  |  |  |  |  |  |  |
|---------------------------------------------------------------|---|---|---|---|---|---|---|
| Model             | Accuracy | Precision | F1-score | AUC  | Sensitivity | Specificity | P value |
| VGG16             | 99.09     | 0.98      | 0.99     | 0.99 | 1.00        | 0.9818      | 0.4770 |
| VGG19             | 90.00     | 0.97      | 0.89     | 0.90 | 0.827       | 0.9727      | 0.2187 |
| DenseNet          | 85.90     | 0.95      | 0.84     | 0.86 | 0.754       | 0.9636      | 0.1361 |
| InceptionV3       | 72.27     | 1.00      | 0.62     | 0.72 | 0.445       | 1.0000      | 0.0010 |
| Xception          | 90.45     | 0.99      | 0.90     | 0.90 | 0.818       | 0.9909      | 0.2113 |
| Inception-ResNet-v2| 91.81     | 0.95      | 0.92     | 0.92 | 0.951       | 0.8898      | 0.2187 |
| EfficientNetB0    | 75.00     | 1.00      | 0.67     | 0.75 | 0.5        | 1.00        | 0.0011 |
| EfficientNetB4    | 86.82     | 0.99      | 0.85     | 0.86 | 0.75        | 0.99        | 0.1260 |
analysis of the table, in the case of recall (sensitivity and F1-score), the Inception-ResNet-V2 comes out to be the weakest model among all competitors.

Table 15 shows the Convolutional Net’s accuracy, and Table 16 shows the AUC score of all the CNNs used. VGG16 depicts the highest accuracy of 99.09% with RMSProp Optimizer followed by Xception Net with 92.72% accuracy with Adam optimizer and finally Inception-ResNet-V2 with an accuracy of 91.81% using RMSProp Optimizer.

The above Fig. 14 shows the best five imagenets that have received the best accuracy on the different optimizer. Table 15 proves that VGG16 with the RMSprop optimizer computes the best accuracy, which is 99.09%, followed by XceptionNet, which generates an accuracy rate of 92.72% with the Adam optimizer. Inception-ResNet-V2 with RMS prop generates 91.81% accuracy on the validation dataset used during training of the model. VGG19 produces the fourth-best accuracy with the RMSprop optimizer, which is 90%, followed by EfficientNetB4, which receives 88.63% accuracy with the Adam optimizer. A combined ROC (Fig. 14) is plotted, which shows the difference in sensitivity and specificity between these five models (VGG16 (RMSprop), VGG19 (RMSprop), XceptionNet (Adam), Inception-ResNet-V2 (RMSprop), EfficientNetB4 (Adam)).

### 4.5 CNN performance on testing dataset

After evaluating all the models mentioned previously, based on accuracy and AUC score, the best four models are selected for testing on the Mendeley dataset by Raja et al. [48]. The best models selected, on the basis of Table 15, for the testing process are Inception-ResNet-V2, Xception, VGG16, and EfficientNetB4. These best CNN Nets were tested for glaucoma detection on a Mendeley dataset (consisting of a total of 139 images, 40 of which are classified as glaucoma images and 99 normal images). The optimizers used for testing are VGG16 with

---

### Table 15 Accuracy of all the CNN’s evaluated

| GROUPS       | SGD   | RMSProp | Adam  |
|--------------|-------|---------|-------|
| VGG16        | 98.18 | 99.09   | 98.18 |
| VGG19        | 83.63 | 90.00   | 87.72 |
| DenseNet     | 80.90 | 85.90   | 85.45 |
| InceptionV3  | 75.00 | 72.27   | 75.90 |
| Xception     | 72.27 | 90.45   | 92.72 |
| Inception-ResNet-V2 | 81.81 | 91.81 | 83.63 |
| EfficientNetB0 | 73.63 | 75.0   | 77.72 |
| EfficientNetB4 | 80.91 | 86.82  | 88.63 |

### Table 16 AUC score of all the CNN’s evaluated

| GROUPS       | SGD   | RMSProp | Adam  |
|--------------|-------|---------|-------|
| VGG16        | 0.98  | 0.99    | 0.98  |
| VGG19        | 0.84  | 0.90    | 0.88  |
| DenseNet     | 0.81  | 0.86    | 0.85  |
| InceptionV3  | 0.75  | 0.72    | 0.76  |
| Xception     | 0.72  | 0.90    | 0.93  |
| Inception-ResNet-V2 | 0.82 | 0.92 | 0.84 |
| EfficientNetB0 | 0.73  | 0.75    | 0.77  |
| EfficientNetB4 | 0.81  | 0.86    | 0.88  |
RMSProp Optimizer, Inception-ResNet-V2 with RMSprop, XceptionNet with Adam Optimizer, and Efficient Net with Adam Optimizer.

Using Fig. 15, examples of the classification of the VGG16 network with the RMSProp optimizer from the test dataset are portrayed. Figure 15(a, b) are true negative examples because VGG16 correctly identified the glaucoma images as glaucoma. The false positive is 0 because CNN correctly identified all glaucoma images as glaucoma. Figure 15(c, d) are false negative examples because VGG16 correctly identified them as glaucoma. Figure 15(e, f) are true positive examples because VGG16 correctly identified them as non-glaucom.

Fig. 16 shows the examples of classification of the Inception-ResNet-V2 network with the RMSProp optimizer from the test dataset. Figure 16(e and f) are true positive examples as the model correctly identified all normal images as normal. Inception-ResNet-V2 identified all the 70 normal images as normal. Therefore, false negative examples are 0. Figure 16(a and b) shows the images that are correctly predicted and show true negative results as the model predicted glaucoma images as glaucoma. The rest of the two images that are in Fig. 16(c and d) show the result of false-positive, which was found to be 26. That means that the model predicted the wrong result by labelling glaucomatous images normal.

In Fig. 17, examples of the Efficient Net B4 network classification with the Adam optimizer from the test dataset are shown. Figure 17(e and f) are true positive examples as the net correctly identified no glaucoma images as no glaucoma images. Figure 17(a and b) show the correctly predicted images and show true negative results as the model predicted glaucoma images as glaucoma. The other two images, that is, Fig.17(c and d), show the result of false-positive, which was found to be 34. That means that the model predicted the wrong result by

| GROUPS          | Optimizer | Accuracy (%) | Precision | Recall | F1 score | AUC  | Sensitivity | Specificity |
|-----------------|-----------|--------------|-----------|--------|----------|------|-------------|-------------|
| VGG16           | RMSProp   | 95.68        | 0.93      | 0.93   | 0.93     | 0.95 | 0.92        | 0.96        |
| XCEPTION NET    | Adam      | 83.45        | 0.71      | 0.71   | 0.72     | 0.80 | 0.75        | 0.87        |
| Inception-Resnet-V2 | RMSProp   | 83.45        | 1.0       | 0.42   | 0.60     | 0.71 | 0.43        | 1.0         |
| Efficient Net B4 | Adam      | 85.61        | 0.95      | 0.53   | 0.68     | 0.76 | 0.53        | 0.98        |

Table 17 Performance metric for the best CNN on the test dataset
Fig. 15  Testing of VGG16 network with RMSProp optimizer for images from the test dataset

Fig. 16  Testing of Inception-Resnet-V2 network with RMSProp optimizer on images from the test dataset
labelling glaucomatous images as normal, and (g) is a false-negative, which is an image classified as glaucoma but is non-glaucoma (normal). EfficientNetB4 identified all the 70 normal images as normal. Therefore, false negative examples are 0.

In Fig. 18, examples of the classification of the Xception network with the Adam optimizer from the test dataset are shown. Figure 18(e and f) are true positive examples, as the net
correctly identified no glaucoma images as no glaucoma images. Figure 18(a and b) show the correctly predicted images and show true negative results as the model predicted glaucoma images as glaucoma. The next two images, Fig. 18(c and d), show the result of a false-positive, which was found to be 27. That means that the model predicted the wrong result by labelling glaucomatous images as normal, and Fig. 18(g and h) are false negatives, images classified as glaucoma but are non-glaucoma. Xception identified all the 70 normal images as normal. Therefore, false negative examples are 0.

4.6 Data set influence

As depicted in Table 15, the best performing four models are tested on the Mendeley dataset [48], and the performance can be evaluated with the help of Table 17. Figure 19 shows the ROC curves and confusion matrix of Inception-ResNet-V2, Xception, VGG16, and EfficientNet B4. It can also be concluded from the same figure that the VGG16 model has the best performance among all the nets. As already mentioned, the training dataset was totally separated from the testing dataset; thus, the best CNNs were evaluated on the separate dataset.
CNN is enormously successful in analyzing medical images and provides a much more accurate result than clinical diagnosis. Therefore, VGG16, Xception, InceptionResnet, and EfficientNetB4 are evaluated on different testing datasets Fig. (20). Observing Table 18 (and also Table 19), it is concluded that VGG16 with the RMSProp optimizer gave an accuracy of 95.68% and an AUC score of 0.95. Xception Net was evaluated with the Adam optimizer gave an accuracy of 83.45% and an AUC score of 0.80. Inception-ResNet-V2 was assessed with an RMSProp optimizer and passed with an accuracy of 83.45% with an AUC score of 0.71. EfficientNetB4 was evaluated on the Adam optimizer with an accuracy of 85.61% and an AUC score of 0.76. It shows how well each of the best-performing CNNs did in a combined ROC curve.

A P-test has also been performed to validate the computed results (refer to Tables 14a, b and c). The P-Test is distinct from the test performance of an alternative model and is used to investigate the test performance of one machine learning model. To obtain a p value for checking whether one model has a substantially different AUC than another model, DeLong’s test can be performed. A method in which an empirical AUC is determined is defined by DeLong et al. [9]. Suppose VGG16 predicts glaucoma disease with an AUC of 0.98 and VGG19 predicts glaucoma disease with an AUC of 0.88, then DeLong’s test can be used to show that VGG16 has a substantially different AUC with p < 0.05 compared to VGG19. The Empirical AUC is equivalent to the Mann-Whitney U-statistic. The Mann-Whitney statistics estimate the probability that a randomly chosen observation from the population represented by healthy individuals would be less than or equal to a randomly chosen observation from the population represented by glaucoma affected individuals. A nonparametric null test says that a randomly chosen value from one population is just as likely to be less than or greater than a randomly chosen value from the other population.

\[
\text{Mann–Whitney statistic} = \text{empirical AUC} = \hat{\theta} = \frac{1}{mn} \sum_{j=1}^{n} \sum_{i=1}^{m} \psi(X_i, Y_j)
\]  

Whether model A or model B is better in terms of AUC is the goal of DeLong’s test, where theta-hat (A) is the AUC of model A, and theta-hat (B) is the AUC of model B. Here, A and B represent two separate models that are proposed. To calculate the Z-score, one needs to calculate the empirical AUCs, the variance V and the covariance C to calculate the Z-score. If Z deviates too far from zero, then it can be inferred that model A has a statistically different AUC with p < 0.05 from model B. A “two-tailed test” is also conducted after measuring the Z-score in order to say that model A’s AUC is different from model B’s AUC.

Unlike other loss functions, such as Cross-Entropy Loss or Mean Square Error Loss, whose objective is to learn to predict a label or a value given an input, the objective of Ranking Losses is to predict relative distances between inputs [76].

### Table 18: Accuracy score of the best CNN on the test dataset

| GROUPS           | OPTIMIZER   | Accuracy (%) | AUC  |
|------------------|-------------|--------------|------|
| VGG16            | RMSProp     | 95.68        | 0.95 |
| XceptionNet      | Adam        | 83.45        | 0.80 |
| Inception-ResNet-V2 | Adam    | 83.45        | 0.71 |
| EfficientNetB4   | RMSProp     | 85.61        | 0.76 |
\[ RL = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{|\mathcal{Y}_i|} \left| \{ (y_a, y_b) : r_i(y_a) > r_i(y_b), (y_a, y_b) \in \mathcal{Y}_i \} \right| \]  

(6)

Where \( r_i \) is the ranking function.

Here in Table 19, the ranking loss for EfficientNetB4 is higher than the VGG16, XceptionNet, and Inception-ResNet-V2, which measure the irrelevant labels as well as similar scenarios in testing ranking loss.

### 4.7 Computational complexity

For the implementation of this study, the Keras library with a tensor flow background on a Google Colab-based Titan Xp graphics processing unit (GPU) has been used. The Titan Xp GPU is used to run the various ImageNets. As previously discussed in detail, the configuration parameters that consume the most time and are taken into account are regularisation technique, batch size, number of images for training, number of fine-tuned layers, and so on. There are other parameters of the models which are equally vital and need to be discussed; computational complexity is one of them. Computational complexity is broadly discussed in terms of time complexity (time required) of the architecture of the CNN model, and that can be computed in terms of either actual running time or theoretical running time, separately. The real running time is determined by factors that are different from the running time that was predicted.

There are multiple factors that one should consider while computing the running time. Sometimes it is also called “execution time” and “the associated cost of the model.” One has to consider factors like the cost of performing the training and hyperparameter settings. While predicting the execution time, one has to find the total number of epochs required to reach the level of best performance (desired level of accuracy) and the execution time of a single epoch (single forward and backward pass through all of the training data). Here we are not considering whether the GPU is optimally performing due to data size or the ability to make full use of all GPU cores simultaneously. Moreover, for the sake of simplicity, we are assuming our considered deep learning parameters are equally applicable to the central processing unit (CPU), tensor processing unit (TPU), or even the intelligence processing unit (IPU). Batch size contributes a lot to computing overall execution time as the time per batch is multiplied by the number of batches. Another significant factor is the number of layers, which is increasing day by day in the upcoming models (AlexNet was proposed in 2012 with 8 layers, while ResNet has 152 layers in 2015). Furthermore, with new and more complex problems, we are passing much more data to process on each layer, resulting in a sharp increase in the execution time required to train the network, regardless of whether one is working on a local GPU or executing on a global GPU. Factors like electricity, air-conditioning cost, per hour payment of global GPU cost, and training a deep network on

| GROUPS              | Optimizer  | Training Ranking Loss | Testing Ranking Loss |
|---------------------|------------|-----------------------|----------------------|
| VGG16               | RMSProp    | 0.0137                | 0.3941               |
| XceptionNet         | Adam       | 0.3863                | 0.4792               |
| Inception-ResNet-V2 | RMSProp    | 0.4213                | 0.3348               |
| EfficientNetB4      | Adam       | 0.5240                | 0.3894               |
GPUs also need attention. The choice of optimal hardware for predicting the execution time of a particular architecture is still almost unexplored. Features (layer features, layer-specific features, implementation features, and hardware features) also play an influential role in the prediction of execution time during the training phase. Going into details of the features, layer features include activation function, optimizer, and batch size. The layer-specific features include multilayer perceptron (MLP) features (number of inputs and number of neurons), convolutional features (matrix size, kernel size, input depth, output depth, stride size and input padding), pooling features (kernel size, stride size and input padding) and recurrent features (recurrence type and bidirectional). Hardware features of a graphics card include the technology used, the number of graphics cards in a pack, the number of graphics cards in each pack, how many graphics cards can connect to each other, and how much memory each card has.

Operations are typically performed using forward or forward and backward transfers, and execution times are reported from several executes. The function set for service, along with execution timings, is then used to train a fully linked feed-forward network. This network will then be used to estimate the time of execution for a new procedure based on a series of features. The deep learning network’s average performance will be estimated after predicting the individual operations that can be combined across the layer(s). The other parameters that can be part of the concentration are dropout rate, loss function, model optimizer, layers, and neurons in each layer.

The time required for the forward and backward passes on a single batch is computed as follows: Eq. 7 again helps predict the computational time for a single epoch.

\[ T_b = \sum_{i=0}^{l} b_{M(i)} \]  

(7)

Where \( l \) is the number of layers in the deep neural network and \( b_{M(i)} \) is the batch execution time estimate for \( i^{th} \) layer and \( M(i) \) is the type of layer \( i \).

Finally, to compute the total execution time for the single epoch, the following equation can be used.
\[ E = pT_b \]

where \( p \) is the number of batches required to process the data.

As previously stated, the Keras Library is used in this work with a TensorFlow background. Xp GPU is used to run the various ImageNets. The time for each architecture (for all the optimizers and batch size 10) was obtained by the summation of the time consumed, which is 1202 s in the case of VGG16, 2414 s in the case of VGG19, 6955 s for InceptionV3, 5347 s for DenseNet, 3854 s for Xception, 8154 s for Inception-ResNet-V2, 2781 s for EfficientNetB0, and 3775 s for EfficientNetB4 (Refer Table 13). When the model was fine-tuned, 100 ms was required to assign a glaucoma probability for each OCT.

Another aspect of measuring the computational complexity is the theoretical time complexity (including training time and testing time), which is broadly computed in terms of the order (Big-Oh) of a given Eq. 8.

\[
O\left( \sum_{i} d_i f_i - 1 \cdot s_i^2 \cdot n_i \cdot m_i^2 \right)
\]

In this equation \( i \) is the index of a convolution layer and component \( d \) denotes the number (depth) of convolution layer, \( n_i \) is the width (number of filters) in the \( i^{th} \) layer. \( n_{i-1} \) is also known as the number of input channels of the \( i^{th} \) layer, \( s_i \) is the spatial size of the filter, \( m_i \) is the spatial size of the output feature map. In the above equation, the time factor related to fully connected layers and pooling layers is not considered (these layers may take 5–10% computational time). During this analysis, the input/output dimensions of the fully connected and pooling layers in all candidate models have been fixed. Prior literature has discussed that training time per image is approximately three times the testing time per image (one for forwarding propagation and two for backward propagation).

5 Discussion and comparison with published literature

In this section, the simulation results of the proposed study are compared with 17 prominent previous works. Table 20 shows a comparison between recently published studies (published on or after 2017) and our work in the detection of glaucoma. Our proposed work provides improvements in results as compared to previous work in terms of accuracy, AUC, sensitivity, and specificity.

It is observed that the majority of the above mentioned studies were conducted on private datasets. Initially, work is performed on private datasets to select the best ones, which is then applied to public datasets at a later stage for evaluation purposes to find the best one. In this paper, eight well-known and widely accepted deep learning-based models among the researcher community are selected that have demonstrated their efficiency in solving image-based classification problems (InceptionV3 [11] Inception-ResNet-V2 [12] Xception [47], EfficientNet [35] DenseNet [75], VGG16 [60], VGG19 [10]). Among these eight models, three optimizers have been applied to find the best combination of model and its optimizer. Genuine effort has been made to identify the best suitable optimizer for each of Imagenet’s to provide higher accuracy for detecting glaucoma from OCT. Extensive experimentation has also been performed to fine-tune hyperparameters and test them at various epochs.
| YEAR | AUTHOR(S) | DATASET | METHOD USED | PERFORMANCE |
|------|-----------|---------|-------------|-------------|
| 2020 | Russakoff et al.[54] | Training on 1095 eyes and testing on 505 and 336 eyes. | gNet3d(CNN) | AUROC ranging from 0.78 to 0.95 |
| 2019 | Thompson et al.[67] | 549 glaucoma eyes and 166 normal eyes | CNN was trained to predict the SDOCT BMO-MRW global and sector values when evaluating optic disc photograph. | Deep Learning(DL) prediction AUC 0.945 |
| 2019 | Xu et al. [73] | 1943 images with open angle;300 images with angle-closure | ResNet-18(CNN) | AUC 0.933 |
| 2019 | Hao et al. [21] | 1024 with open angle glaucoma,1792 images with narrowed angle | Three scale regions with different sizes are resized to 224x224. Multi-Scale Regions Convolutional Neural Networks (MSRCNN). | AUC 0.9143 |
| 2018 | Lee et al. [29] | 91 healthy images and 58 glaucomatous eyes | Measuring peripapillary retinal nerve fiber layer(PP-RNFL) thickness,full macular thickness, and ganglion cell-inner plexiform layer (GC-IPL) thickness. Three dimensional optic disc scanning of DRI-OCT. | AUC upto 0.968 |
| 2020 | Thomson et al. [68] | 612 glaucomatous images and 542 normal images | CNN | ROC curve area 0.96 |
| 2019 | Ran et al. [50] | 1822 glaucomatous images and 1007 normal images | Data preprocessing and then 3D CNN(ResNet) | Primary validation dataset accuracy 91% |
| 2019 | Maetschke et al. [33] | 847 volumes of glaucoma and 263 volumes of healthy | Unsegmented OCT volumes of the optic nerve head using a 3D-CNN | AUC 0.94 |
| 2020 | Wang et al. [71] | 2669 infected cases and 1679 normal cases | CNN with semi-supervised learning | AUC from 0.933 to 0.977 |
| 2020 | Shehryar et al. [57] | 22 images of healthy eyes and 28 images of glaucoma eyes. | Hybrid computer-aided-diagnosis (H-CAD) system CNN | Accuracy upto 96%; Specificity upto 95%; Sensitivity upto 94%; AUC scores ranging from 0.930 to 0.989 |
| 2019 | Thakoor et al. [66] | 737 RNFL probability map images from 322 eyes of 322 patients and 415 eyes of 415 healthy controls | | |
| 2020 | Raja et al. [49] | 50 B-scan OCT (Local Dataset) | VGG-16 architecture for feature extraction and classification of retinal layer pixels. | Accuracy 94%; Sensitivity 94.4%; Specificity 93.75% |
| 2017 | Muhammad et al. [39] | 57 Glaucoma images and 45 Normal images | Border Segmentation, Image Resize. CNN(AlexNet)+ Random Forest | Accuracy upto 93.31% |
| 2018 | Kansal et al. [23] | 16,104 glaucomatous and 11,543 normal eyes. | Meta Analysis; | AUROC of glaucoma diagnosis for RNFL average for all glaucoma |
have been taken to showcase the comparison on the basis of selected deep-net models and their results. This is all being implemented to ensure that the proposed method generates the best results. To the best of our knowledge, such a type of evaluation, exploration, intensive analysis, and tuning is rarely performed by researchers and has been observed in prior published studies on glaucoma detection from OCT, further supporting the work’s originality. The best performing model is compared with seventeen prior published, well-recognized studies; the comparative table is compiled and shown, to correlate our performance with these studies. The comparative Table 20 clearly justifies the superiority of our proposed method. The accuracy generated by our proposed system (0.9568) is among the best in the class. However, in [29] and [57], the researchers have shown some better results than ours. It is found that the number of subject images on which they have tested their approach is much less than the number of subject images on which the presented approach is tested. The value generated for other medically significant metric specificity (0.96) is also the best in the class. The remaining efficiency measuring parameters (precision, recall, F1-Score, AUC, and sensitivity) also achieve an incredible show (more than 90%) through our approach. When one carefully examines Table 20, one can see that none of the other comparable studies have done as much analysis and parameter calculation as we have to demonstrate and justify the efficacy of our proposed approach. People can be sure that they have a condition if their diagnostic test gives them a positive result, and our proposed system is good at that, too. Because there are relatively limited publicly available datasets for glaucoma and the published research linked to automatic detection of glaucoma from OCT is based on their own privately obtained

| YEAR | AUTHOR(S) | DATASET | METHOD USED | PERFORMANCE |
|------|-----------|---------|-------------|-------------|
| 2019 | An et al.[3] | Private dataset (208 images) | Transfer Learning of CNN with VGG19 | AUC 0.963 |
| 2019 | Asaoka et al. [4] | Training 94 Open Angle Glaucoma (OAG) and 84 normal eyes. For testing 114 OAG eyes and 82 normal eyes | Image Resize, Image Augmentation,CNN and Transfer Learning | AUCROC: DL Model 0.937,RF 0.820,SVM 0.674 |
| 2021 | García et al. [18] | 176 healthy and 144 glaucomatous SDOCT volumes centred on the optic nerve head (ONH) | Combination of CNN and LSTM networks | AUC 0.8847 |
| PROPOSED METHODOLOGY | Private Dataset for training and Mendeley Dataset for proof testing (4220 images and 139 images respectively) | Imagenet Models used-VGG16, VGG19, DenseNet, InceptionV3, XceptionNet, Inception-ResNet-v2, EfficientNetB0, EfficientNetB4 | VGG16 with RMSProp Accuracy=0.9568, AUC=0.95, Specificity=0.96 |

The performance received is presented in above shown tables.
datasets, meaningful comparison of different automated algorithms proposed to detect glaucoma is challenging. Thus, one can realize that a fair comparison cannot be made because of different subject datasets in some cases. The presented study simulation results are highly promising and better than almost all the papers discussed in Table 20.

An evaluation of the work presented in this empirical study can be made in two verticals. The first vertical comparison that can be made in-between the models selected for this study is the price. While in the second one, a comparison can be made between our best performing model and other prior previous studies published in reputed journals. Regarding the first vertical, we have selected eight well-known and well-accepted deep learning-based models among the researcher community that have demonstrated their efficiency when solving image-based classification problems. And among these models, three optimizers have been applied, in all eight models, to find the best combination of model and its optimizer. We have also performed extensive experimentation to fine-tune hyperparameters and tested them on various epochs also. Such a type of evaluation and exploration is rarely performed by any researcher and has not been observed in prior published studies on glaucoma detection from OCT images. This is all being implemented to ensure that the proposed method generates the best results. To the best of our knowledge, very few studies have been published on glaucoma detection from OCT images, and none of the previous reputed papers has performed such intensive analysis and tuning as compared to ours. Regarding the second vertical, we have compared our best performing model with 17 recently published, well-recognized studies; the comparative table is compiled and shown to correlate our performance with these studies. The comparative tables (18 and 20) clearly justify the superiority of our proposed method. It has been discovered that the internal architecture of our shortlisted models differs due to the various architectures. Some models perform well, while others are not satisfactory. This may be due to the fact that the other architectures have larger and more complicated structures and more training parameters than the best performing ones. More tracing parameters and less training data would produce an over-fitting phenomenon, which may yield a less inaccurate classification performance.

OCT is currently one of the most frequently done imaging techniques, with 5.35 million OCT scans conducted in 2014 alone in the US Medicare population. This is a progressive imaging technique that ophthalmologists often employ. These images provide precise information about the layers of the retina. OCT’s cross-sectional image of retinal tissues was previously frequently employed to diagnose and treat eye disorders. OCT technologies have shown a comparable ability to differentiate between normal and glaucomatous eyes in important thickness measurement sectors for general population glaucoma diagnosis. The use of Deep Learning (DL) on OCT images to evaluate glaucoma has been shown to be efficient, accurate, and promising. Together with the findings of prior research, our findings demonstrate the utility of a DL-based technique for diagnosing glaucoma in a real-world setting. As a result, the proposed network effectively learns highly discriminative features from OCT. The observed data validated that our approach is successful in detecting this illness. The use of DL on OCT to evaluate glaucoma has been shown to be efficient, accurate, and promising. As shown in the preceding tables, the suggested framework outperforms current methodologies in terms of accuracy, specificity, and AUC. Our findings are more than 95% in all medical efficiency measuring criteria (accuracy, specificity, and AUC), resulting in relevant results that are comparable to the accuracy of senior (or expert) ophthalmologists. The
suggested strategies achieve resilience by minimizing reliance on picture quality and the impact of additional artifacts. The suggested strategy showed a considerable improvement in performance when compared to earlier state-of-the-art techniques. As a result, our suggested approach will aid in accurately predicting glaucoma using an OCT picture. The approach given here is not restricted to glaucoma; rather, it may be used to diagnose various ocular disorders, particularly in the optic nerve head area. This technology may be applied to various eye disorders by training on disease-specific data and making minor adjustments. These CNN models have the ability to operate in conjunction with human specialists to improve general eye health and speed up the identification of blinding eye illnesses. As previously shown, the system’s operating time (training or testing time) is likewise rather outstanding for outputting the final screening result for a single OCT picture. Our proposed Human-Centered health-care recognition system has trustworthiness, explainability, accountability and is ready for social acceptance. As a result, it may be used independently in remote hospitals. In remote locations where ophthalmologists are few, the suggested technique may be used to recognize glaucoma, and only true cases can be sent to experts for further investigation. The suggested solution could be put in mobile devices or connected to IoT-based devices, and it could be used to help people all over the world or in places where there aren’t enough expert eye doctors or detection equipments.

6 Conclusion, limitations and future work

Every year, millions of people are affected by glaucoma, and its progression can cause permanent blindness in humans. Therefore, it must be diagnosed and prevented at an early stage. At its early stages, glaucoma does not affect vision and hence usually goes unnoticed by patients. That is why some researchers have called this disease a “snake thief of sight”. The only way to stop glaucoma from progressing and losing your vision is to get it diagnosed early and get the right treatment right away. Till now, ophthalmologists were practising the customary way of detecting glaucoma by just looking at the OCT, which is sometimes unreliable, dependent on intra-observer variability, sometimes inaccurate, and time-consuming. Ideally, only highly experienced ophthalmologists should inspect these images to assure the accuracy of the performed diagnosis. However, their availability round the clock and across the globe is a severe issue due to the low number of experts and the heavy overload of periodic manual screening on them. Other than this, certain patient images, most of which have no signs of illness, must also be checked. Automatic scanning and screening devices have the ability to solve these drawbacks by remotely collecting and manipulating retinal photographs, and only people with glaucoma symptoms have been recommended to seek medical attention. Hence, it is essential to design an automated computer-based diagnosis method that is fast, reliable, and highly accurate for detecting glaucoma. This proposed empirical approach’s primary objective is to evaluate the automation of the detection of glaucoma through OCT scans with the help of advanced deep learning algorithms. The OCT used in our research was taken from private hospitals around the city and then passed through eight different pre-trained architectures. Based on the results, the four best-performing models were shortlisted. Later, these models are tested on the well-recognized standard public Mendeley dataset. The influence of the architecture, data size, and training strategy has been thoroughly investigated. The CNN evaluated excellent performance measurements, including accuracy, sensitivity, specificity, precision, F1-score, AUC value, and ROC
curve. To assess these pre-trained nets, Keras libraries on Google Colaboratory are used with TensorFlow as the backend. The obtained final data from all these nets offered promising results in terms of accuracy. The best options are VGG16 with RMSProp optimizer with an accuracy of 95.68% and EfficientNetB4 using Adam optimizer with an accuracy of 85.61%, followed by Xception Net using Adam optimizer with an accuracy of 83.45% and Inception-ResNet-v2 using RMSProp optimizer with an accuracy of 83.45%. These CNN generated auspicious results can be a valuable alternative to experienced ophthalmologists and expensive devices to classify glaucoma OCT. Because the American Academy of Ophthalmology has set standards for how good a job should be, it is likely that this work will be good enough to meet those standards.

Some of the limitations of this study include a focus on the classification of input images into two classes only. The larger the dataset on which models can be trained, the better the performance of the trained models will be. The available benchmark dataset for OCT is limited. Hence, the performance may be influenced by the limited available public benchmark dataset of OCT on which this study was performed. Glaucoma prediction (recognition) is the primary objective of the study, and no consideration is given in the case of any other disease present in the patient’s eye. Novel CNN models are proposed by researchers at a fast pace; it is not possible to compare the performance of all the available models. There may be some models that compute better results than our shortlisted models. Performance optimization, for performance improvement, is unexplored in our study and can be incorporated into the model as a future direction. Lastly, focus on the progression of glaucoma growth has not been considered. As a future direction, “Low-light image enhancement via a deep hybrid network” can be implemented to observe the improvement in the results [53].

Funding  No funding for this study.

Declarations

Human and animal rights  This article does not contain any studies with human or animal subjects performed by any of the authors.

Conflict of interest  On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

1. Abràmoff MD, Garvin MK, Sonka M (2010) Retinal imaging and image analysis. IEEE Rev Biomed Eng 3:169–208
2. Ajesh F, Ravi R (2020) Hybrid features and optimization-driven recurrent neural network for glaucoma detection. Int J Imaging Syst Technol 30(4):1143–1161
3. An G, Omodaka K, Hashimoto K, Tsuda S, Shiga Y, Takada N, Kikawa T, Yokota H, Akiba M, Nakazawa T (2019) Glaucoma diagnosis with machine learning based on optical coherence tomography and color fundus images. J Healthcare Eng:2019
4. Asaoka R, Murata H, Hirawaka K, Fujino Y, Matsura M, Miki A, Kanamoto T, Ikeda Y, Mori K, Iwase A, Shoji N, Inoue K, Yamagami J, Arie M (2019) Using deep learning and transfer learning to accurately diagnose early-onset glaucoma from macular optical coherence tomography images. Am J Ophthalmol 198:136–145
5. Bock R, Meier J, Nyüü LG, Horneck J, Michelson G (2010) Glaucoma risk index: automated glaucoma detection from color fundus images. Med Image Anal 14(3):471–481

Springer
6. Bussel IL, Wollstein G, Schuman JS (2014) OCT for glaucoma diagnosis, screening and detection of glaucoma progression. British J Ophthalmol 98(Suppl 2):ii15–ii19 IEEE
7. Charlson ES, Sankar PS, Miller-Ellis E, Regina M, Fertig R, Salinas J, … O’Brien JM (2015) The primary open-angle african american glaucoma genetics study: baseline demographics. Ophthalmology 122(4):711–720
8. del Amor R, Morales S, n Colomer A, Mossi JM, Woldbye D, Klemp K, … & Naranjo V (2019). Towards Automatic Glaucoma Assessment: An Encoder-decoder CNN for Retinal Layer Segmentation in Rodent OCT images. In 2019 27th European Signal Processing Conference (EUSIPCO) (pp. 1–5). IEEE.
9. DeLong ER, DeLong DM, & Clarke-Pearson DL (1988). Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. Biometrics, 837-845.
10. Dey N, Zhang YD, Rajinikanth V, Raja NSM (2021) Customized VGG19 architecture for pneumonia detection in chest X-rays. Pattern Recogn Lett 143:67–74
11. Dong N, Zhao L, Wu CH, Chang JF (2020) Inception v3 based cervical cell classification combined with artificially extracted features. Appl Soft Comput 93:106311
12. Ferreira CA, Melo T, Sousa P, Meyer MI, Shakibapour E, Costa P & Campilho A (2018). Classification of breast cancer histology images through transfer learning using a pre-trained inception resnet v2. In international conference image analysis and recognition (pp. 763-770). Springer, Cham.
13. Fu H, Xu Y, Lin S, Zhang X, Wong DWK, Liu J, … Aung T (2017) Segmentation and quantification for angle-closure glaucoma assessment in anterior segment OCT. IEEE Trans Med Imaging 36(9):1930–1938
14. Fu H, Baskaran M, Xu Y, Lin S, Wong DWK, Liu J, … Aung T (2019) A deep learning system for automated angle-closure detection in anterior segment optical coherence tomography images. Am J Ophthalmol 203:37–45
15. Fujimoto JG, Pitris C, Boppart SA, Brezinski ME (2000) Optical coherence tomography: an emerging technology for biomedical imaging and optical biopsy. Neoplasia 2(1–2):9–25
16. Gaddipati DJ, Desai A, Sivaswamy J & Vermeer KA (2019). Glaucoma assessment from oc t images using capsule network. In 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC) (pp. 5581-5584). IEEE.
17. García G, del Amor R, Colomer A & Naranjo V (2020). Glaucoma detection from raw Circumpapillary OCT images using fully convolutional neural networks. In 2020 IEEE international conference on image processing (ICIP) (pp. 2526-2530). IEEE
18. García G, Colomer A, Naranjo V (2021) Glaucoma detection from raw SD-OCT volumes: a novel approach focused on spatial dependent features. Comput Methods Prog Biomed 200:105855
19. Grewal DS, Merlau DJ, Giri P, Munk MR, Fawzi AA, Jampol LM, Tanna AP (2017) Peripapillary retinal splitting visualized on OCT in glaucoma and glaucoma suspect patients. PLoS One 12(8):e0182816
20. Haleem MS, Han L, Van Hemert J, Li B (2013) Automatic extraction of retinal features from colour retinal images for glaucoma diagnosis: a review. Comput Med Imaging Graph 37(7–8):581–596
21. Hao H, Zhao Y, Fu H, Shang Q, Li F, Zhang X & Liu J (2019y) Anterior chamber angles classification in anterior segment oc t images via multi-scale regions convolutional neural networks. In 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC) (pp. 849-852). IEEE.
22. Huang G, Liu Z, Van Der Maaten L & Weinberger KQ (2017). Densely connected convolutional networks. In proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4700-4708).
23. Kansal V, Armstrong JJ, Pintwala R, Hutnik C (2018) Optical coherence tomography for glaucoma diagnosis: An evidence based meta-analysis. PLoS One 13(1):e0190621
24. Kass MA, Heuer DK, Higginbotham EJ, Johnson CA, Keltner JL, Miller JP, … & Gordon MO (2002) The ocular hypertension treatment study: a randomized trial determines that topical ocular hypotensive medication delays or prevents the onset of primary open-angle glaucoma. Arch Ophthalmol 120(6):701–713
25. Khanna M, Agarwal A, Singh LK, Thawkar S, Khanna A & Gupta D (2021). Radiologist-level two novel and robust automated computer-aided prediction models for early detection of COVID-19 infection from chest X-ray images. Arab J Sci Eng. 1-33
26. Kotowsky J, Folio LS, Wollstein G, Ishikawa H, Ling Y, Bilonick RA, Kagemann L, Schuman JS (2012) Glaucoma discrimination of segmented cirrus spectral domain optical coherence tomography (SD-OCT) macular scans. Br J Ophthalmol 96(11):1420–1425
27. Lee J, Kim YK, Park KH, Jeoung JW (2020) Diagnosing glaucoma with spectral-domain optical coherence tomography using deep learning classifier. J Glaucoma 29(4):287–294
28. Lee K, Niemeijer M, Garvin MK, Kwon YH, Sonka M, Abramoff MD (2009) Segmentation of the optic disc in 3-D OCT scans of the optic nerve head. IEEE Trans Med Imaging 29(1):159–168
29. Lee SY, Bae HW, Seong GJ, Kim CY (2018) Diagnostic ability of swept-source and spectral-domain optical coherence tomography for glaucoma. Yonsei Med J 59(7):887–896
30. Leske MC, Heijl A, Hussein M, Bengtsson B, Hyman L, Komaroff E (2003) Factors for glaucoma progression and the effect of treatment: the early manifest glaucoma trial. Arch Ophthalmol 121(1):48–56
31. Li H, Hu M, Huang Y (2019) Automatic identification of overpass structures: a method of deep learning. ISPRS Int J Geo Inf 8(9):421
32. Li Y, Xie X, Shen L, Liu S (2019) Reverse active learning based atrous DenseNet for pathological image classification. BMC Bioinform 20(1):1–15
33. Maetschke S, Antony B, Ishikawa H, Wollstein G, Schuman J, Garnavi R (2019) A feature agnostic approach for glaucoma detection in OCT volumes. PLoS One 14(7):e0219126
34. Mansberger SL, Menda SA, Fortune BA, Gardiner SK, Demirel S (2017) Automated segmentation errors when using optical coherence tomography to measure retinal nerve fiber layer thickness in glaucoma. Am J Ophthalmol 174:1–8
35. Marques G, Agarwal D, de la Torre Dietz I (2020) Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network. Appl Soft Comput 96:106691
36. McCann P, Hogg RE, Wright DM, McGuinness B, Young IS, Kee F, Azauga-Blanco A (2019) Diagnostic accuracy of spectral-domain oct circumpapillary, optic nerve head, and macular parameters in the detection of perimetric glaucoma. Ophthalmol Glaucoma 2(5):336–345
37. Medeiros FA, Jammal AA, Thompson AC (2019) From machine to machine: an OCT-trained deep learning algorithm for objective quantification of glaucomatous damage in fundus photographs. Ophthalmol 126(4):513–521
38. Mirzania D, Thompson AC, Muir KW (2020) Applications of deep learning in machine to detect: a systematic review. Eur J Ophthalmol 1120672120977346
39. Muhammad H, Fuchs TJ, De Cuir N, De Moraes CG, Blumberg DM, Liebmann JM, … Hood DC (2017) Hybrid deep learning on single wide-field optical coherence tomography scans accurately classifies glaucoma suspects. J Glaucoma 26(12):1086–1094
40. Murtagh P, Greene G, O’Brien C (2020) Current applications of machine learning in the screening and diagnosis of glaucoma: a systematic review and meta-analysis. Int J Ophthalmol 13(1):149–162
41. Mwanza JC, Oakley JD, Budenz DL, Anderson DR, Cirrus Optical Coherence Tomography Normative Database Study Group (2011) Ability of cirrus HD-OCT optic nerve head parameters to discriminate normal from glaucomatous eyes. Ophthalmology 118(2):241–248
42. Mwanza JC, Lee G, Budenz DL, Warren JL, Wall M, Artes PH, Callan TM, Flanagan JG (2018) Validation of the UNC OCT index for the diagnosis of early glaucoma. Transl Vision Sci Technol 7(2):16–16
43. Naveed M, Ramzan A, & Akram MU (2017). Clinical and technical perspective of glaucoma detection using OCT and fundus images: a review. In 2017 1st international conference on next generation computing applications (NextComp) (pp. 157-162). IEEE.
44. Nawaz H, Maqsood M, Afzal S, Aadir F, Mehmood I, Rho S (2021) A deep feature-based real-time system for Alzheimer disease stage detection. Multimed Tools Appl 80(28):35789–35807
45. Pachiyappan A, Das UN, Murthy TV, Tatavarti R (2012) Automated diagnosis of diabetic retinopathy and glaucoma using fundus and OCT images. Lipids Health Dis 11(1):1–10
46. Quigley HA, Broman AT (2006) The number of people with glaucoma worldwide in 2010 and 2020. Br J Ophthalmol 90(3):262–267
47. Rahimzadeh M, Attar A (2020) A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. Inform Med Unlocked 19:100360
48. Raja H, Akram U, Muhammad; Ramzan A; Khalil T ; Nazid N (2020), “Data on OCT and Fundus Images”, Mendeley Data, v2. https://doi.org/10.17632/2mn5zn74.2.
49. Ran AR, Cheung CY, Wang X, Chen H, Luo LY, Chan PP, … Tham CC (2019) Detection of glaucomatous optic neuropathy with spectral-domain optical coherence tomography: a retrospective training and validation deep-learning analysis. Lancet Digital Health 1(4):e172–e182
50. Ran AR, Cheung CY, Wang X, Chen H, Luo LY, Chan PP, … Tham CC (2019) Detection of glaucomatous optic neuropathy with spectral-domain optical coherence tomography: a review. Eye 35(1):188–201
51. Ran AR, Tham CC, Chan PP, Cheng CY, Tham YC, Rim TH, Cheung CY (2021) Deep learning in glaucoma with optical coherence tomography: a review. Eye 35(1):188–201
52. Ren W, Liu S, Ma L, Xu Q, Xu X, Cao X, Du J, Yang MH (2019) Low-light image enhancement via a deep hybrid network. IEEE Trans Image Process 28(9):4364–4375
53. Russakoff DB, Mannil SS, Oakley JD, Ran AR, Cheung CY, Dasari S, Riyazzuddin M, Nagaraj S, Rao HL, Chang D, Chang RT (2020) A 3D deep learning system for detecting referable glaucoma using full OCT macular cube scans. Transl Vision Sci Technol 9(2):12–12
54. Schmitt JM (1999) Optical coherence tomography (OCT): a review. IEEE J Selected Topics Quantum Electron 5(4):1205–1215
56. Sehi M, Grewal DS, Sheets CW, Greenfield DS (2009) Diagnostic ability of Fourier-domain vs time-domain optical coherence tomography for glaucoma detection. Am J Ophthalmol 148(4):597–605
57. Sheryar T, Akram MU, Khalid S, Nasreen S, Tariq A, Perwaiz A, Shaukat A (2020) Improved automated detection of glaucoma by correlating fundus and SD-OCT image analysis. Int J Imaging Syst Technol 30(4):1046–1065. https://doi.org/10.1002/ima.22413
58. Singh LK, Garg H, Khanna M (2021) An artificial intelligence-based smart system for early glaucoma recognition using OCT images. Int J E-Health Med Commun (IJEHMC) 12(4):32–59
59. Singh LK, Pooja GH, Khanna M, Bhadoria RS An enhanced deep image model for glaucoma diagnosis using feature-based detection in retinal fundus. Med Biol Eng Comput 1–21
60. Sitaula C & Hossain MB (2020). Attention-based VGG-16 model for COVID-19 chest X-ray image classification. Appl Intell. 1-14
61. Sung KR, Na JH, Lee Y (2012) Glaucoma diagnostic capabilities of optic nerve head parameters as determined by cirrus HD optical coherence tomography. J Glaucoma 21(7):498–504
62. Sunija AP, Gopi VP, Palanisamy P (2022) Redundancy reduced depthwise separable convolution for glaucoma classification using OCT images. Biomed Signal Process Control 71:103192
63. Szegedy C, Ioffe S, Vanhoucke V, Alemi AA (2017) Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. AAAI 4:12
64. Tan M & Le QV (2019). Efficientnet: rethinking model scaling for convolutional neural networks. arXiv preprint arXiv:1905.11946.
65. Tatham AJ, Medeiros FA (2017) Detecting structural progression in glaucoma with optical coherence tomography. Ophthalmol Physiol Opt 124(12):S57–S65
66. Thakoor KA, Li X, Tsamis E, Saja DA & Hood DC (2019). Enhancing the accuracy of glaucoma detection from OCT probability maps using convolutional neural networks. In 2019 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC) (pp. 2036-2040). IEEE.
67. Thompson AC, Jammal AA, Medeiros FA (2019) A deep learning algorithm to quantify neuroretinal rim loss from optic disc photographs. Am J Ophthalmol 201:9–18
68. Thompson AC, Jammal AA, Berchuck SI, Mariottoni EB, Medeiros FA (2020) Assessment of a segmentation-free deep learning algorithm for diagnosing glaucoma from optical coherence tomography scans. JAMA Ophthalmol 138(4):333–339
69. Tuck MW, Crick RP (1992) Screening for glaucoma: age and sex of referrals and confirmed cases in England and Wales. Ophthalmic Physiol Opt 12(4):400–404
70. Vajaranant TS, Wu S, Torres M, Varma R (2012) A 40-year forecast of the demographic shift in primary open-angle glaucoma in the United States. Invest Ophthalmol Vis Sci 53(5):2464–2466
71. Wang X, Chen H, Ran AR, Luo L, Chan PP, Tham CC, Chang RT, Mannil SS, Cheung CY, Heng PA (2020) Towards multi-center glaucoma OCT image screening with semi-supervised joint structure and function multi-task learning. Med Image Anal 63:101695
72. what is glaucoma? On https://www.webmd.com/eye-health/glaucoma-eyes#1
73. Xu BY, Chiang M, Chaudhary S, Kulkarni S, Pardeshi AA, Varma R (2019) Deep learning classifiers for automated detection of gonioscopic angle closure based on anterior segment OCT images. Am J Ophthalmol 208:273–280
74. Xu J, Ishikawa H, Wollstein G, & Schuman JS (2011). 3D optical coherence tomography super pixel with machine classifier analysis for glaucoma detection. In 2011 annual international conference of the IEEE engineering in medicine and biology society (pp. 3395-3398). IEEE.
75. Zhang J, Lu C, Li X, Kim HJ, Wang J (2019) A full convolutional network based on DenseNet for remote sensing scene classification. Math Biosci Eng 16(5):3345–3367
76. Zhang M, Zhi-Hua Z (2014) A review on multi-label learning algorithms. Knowledge Data Eng, IEEE Trans 26(8):1819–1837
77. Zhang X, Dastiridou A, Francis BA, Tan O, Varma R, Greenfield DS, … Advanced Imaging for Glaucoma Study Group (2017) Comparison of glaucoma progression detection by optical coherence tomography and visual field. Am J Ophthalmol 184:63–74
78. Zheng C, Johnson TV, Garg A, Boland MV (2019) Artificial intelligence in glaucoma. Curr Opin Ophthalmol 30(2):97–103

Publisher’s note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.