Research Article GU J Sci, Part A, 12(1): 15-35 (2025) 10.54287/gujsa.1592915

Gazi University
Journal of Science
PART A: ENGINEERING AND INNOVATION
http://dergipark.org.tr/gujsa

Efficient Diagnosis of Retinal Diseases Using Convolutional Neural Networks

Mahir KAYA[1*]

[1] *Tokat Gaziosmanpaşa University, Tokat, Türkiye*

Keywords

Retinal Diseases
Deep Learning
CNN
Computer-aided Diagnosis

Abstract

The eye is a vital sensory organ that enables us to fulfill all our life's needs. Diseases affecting such a vital organ can have a detrimental impact on our lives. Although certain eye conditions are easily managed, others may result in lasting damage or loss of sight if not identified promptly. Problems within the retina or improper image focus on the retina may result in loss of eyesight. Optical Coherence Tomography (OCT) can identify diseases using retinal images taken from a side-angle view. Medical images are analyzed using Convolutional Neural Networks (CNNs) to automatically diagnose diseases. Doctors may reach varying conclusions when diagnosing diseases based on medical images. These conclusions may even contain human error. These challenges can be overcome with the use of CNNs. When creating a CNN architecture, many hyperparameter values need to be determined at the beginning before the training phase. A well-structured design is crucial for the successful performance of CNNs. The lengthy training time of CNNs makes testing every hyperparameter combination a very time-intensive process. This research determined the best hyperparameters for CNNs by means of Bayesian optimization. The study employed a dataset comprising four categories: DME, CNV, DRUSEN, and NORMAL. With Bayesian optimization, this proposed model reached an accuracy and F1 score of 99.69%, outperforming existing research findings. The proposed model will also help doctors to make decisions and speed up the decision-making process.

Cite

Kaya, M. (2025). Efficient Diagnosis of Retinal Diseases Using Convolutional Neural Networks. *GU J Sci, Part A*, *12*(1), 15-35. doi:10.54287/gujsa.1592915

Author ID (ORCID Number)

0000-0001-9182-271X Mahir KAYA

Article Process

Submission Date 28.11.2024
Revision Date 30.12.2025
Accepted Date 03.03.2025
Published Date 26.03.2025

1. INTRODUCTION

The eyes are the most important of our sensory organs. In fact, the brain is the organ that enables us to see and the eye helps us to see in this sense (Çevik et al., 2021). The eye is composed of three layers that are capable of transmitting and refracting light (Malkoç, 2006). The sclera, the first layer from the outside to the inside, is the white area where light is refracted and protects the eye from external factors. The second layer is the retina (choroid), a network of blood vessels responsible for nourishing the retina. The third layer, the retina, is located behind the eye wall and contains millions of light-sensitive nerve cells (Farsiu et al., 2014). The light coming into the eye is refracted first in the cornea and then in the lens and falls on the retina, and vision is realized by stimulating millions of nerve cells in the retina (Alqudah, 2020). Messages about vision are processed in the brain through the optic nerve, which is composed of millions of nerve fibers, to form the image (Wu et al., 2013).

*Corresponding Author, e-mail: mahir.kaya@gop.edu.tr

Vision starts in the retina. The retina also contains cells that allow us to see in the dark and in the light (Tayal et al., 2021). Numerous conditions can affect the retina, including diabetic retinopathy, retinal tears, retinal detachments, yellow spot disease, occlusions of the retinal artery, epiretinal membrane, and traumatic eye injuries (Fujimoto et al., 2000, Wu et al., 2013, Farsiu et al., 2014, Asif et al., 2022). Diseases occurring in the retina affect the visual ability of the person and the damage to the retina is irreversible (Saleh et al., 2022). Therefore, timely diagnosis and intervention are essential. Diabetic retinopathy, for example, which can cause vision loss, can be diagnosed early with a retinal examination (Çevik et al., 2021). Ultrasonography, fundus fluorescein angiography, and optical coherence tomography, in addition to retinal examination, can be used for diagnosis (Silverman et al., 2014).

Optical Coherence Tomography (OCT) is a method of creating cross-sectional images of the retina using light waves (Li et al., 2019). OCT provides a detailed view of all retinal layers and assesses their thickness. These measurements aid in the diagnosis and selection of the best treatment method for retinal diseases (Fujimoto, et al., 2000). Although retinal eye diseases are typically detected by specialist clinicians through eye examinations, their assessments may sometimes vary. Human error may also arise due to the specialists' workload or conflicting disease conditions. To address these issues, it is important to support decision-making processes with computer-aided artificial intelligence systems (Kaya & Çetin-Kaya, 2024a, Duran et al., 2025). In the last decade, deep learning-based Convolutional Neural Networks (CNNs) have been extensively utilized and successfully applied to disease diagnosis from medical images (Çetin-Kaya & Kaya, 2024). The structure of a CNN comprises convolutional layers, pooling layers, and fully connected layers, and performs end-to-end learning in the training phase, using raw images as input (Krizhevsky et al., 2012; LeCun et al., 1998). In classical machine learning, the features used as input are manually determined at the beginning of the training phase, while in deep learning-based models, feature extraction from raw images is performed automatically during the training phase.

Common problems in disease diagnosis from medical images with CNN can be listed as follows: insufficient number of labeled data, unbalanced class distributions in the datasets, noisy datasets, high similarities between dataset classes, and overfitting during training. When dealing with a dataset with limited labeled samples, underfitting occurs if CNN architectures are defined in a simple way with few layers, but overfitting occurs in the training phase if the architecture is too deep and complex (Kaya, 2024; Güneş & Çetin-Kaya, 2024). When overfitting occurs, CNN models tend to retain the training data in memory rather than generalizing from it, so their training accuracy is very high, but their performance drops significantly when encountering unfamiliar test data (Zhang et al., 2021). Therefore, determining the optimal CNN architecture is crucial for achieving high performance (Çetin-Kaya., 2024). Since CNN architectures have many hyperparameters, it is almost impossible to determine the optimal architecture manually. CNNs are also trained with medical images to diagnose diseases (Cheyi & Çetin-Kaya, 2024). In order to develop a successful model with CNNs, many hyperparameters, including the quantity and dimensions of filters within the convolutional layer, the quantity

of neurons in the densely connected layer, the dropout rate and the learning rate need to be optimally adjusted. In CNNs, hyperparameters comprise many combinations and trying all combinations is costly in terms of training time. To overcome these problems, statistical estimation methods are employed to find the best hyperparameters. In this study, Bayesian optimization method is used to determine the optimal hyperparameters in CNN architectures. Thus, four different retinal disease types were successfully classified.

1.1. Motivation

It is very important that retinal eye diseases are automatically detected by artificial intelligence systems at an early stage. Rapid treatment of early diagnosed eye diseases will accelerate the healing process. In the literature, there are various studies involving transfer learning and custom models. Since transfer learning models are complex models, it may not be possible to run them on all devices. Likewise, custom and transfer learning models have not been optimized to find the optimal hyperparameter values for eye diseases. There is a need for models with a very low error rate and a lightweight architecture for eye diseases.

1.2. Contributions

In the paper, a CNN-based model is introduced for the automated diagnosis of eye diseases using OCT images. A lightweight CNN architecture is developed using Bayesian optimization. Within the CNN structure, key settings like the quantity of filters, filter dimensions, the count of neurons in the fully connected layer, dropout percentage, and learning rate are fine-tuned for optimal performance. The proposed model achieves higher accuracy than previous studies in the literature. This lightweight model can also run quickly on resource-constrained devices.

The remainder of this document is structured as follows: Section 2 offers an overview of prior research. Section 3 describes the dataset and methodology. In Section 4, the training results of the models are presented and compared with existing literature. Section 5 presents a comparative analysis of previous research studies. Lastly, Section 6 wraps up by providing an overview of the results.

2. RELATED WORKS

Deep learning architectures are commonly used for the classification of eye diseases. One of the methods used in deep learning studies is to design a model specific to the classification problem. Thus, customized architectures are used in studies aimed at diagnosing eye diseases.

Alqudah (2020) proposed a custom CNN model for the classification of eye diseases from spectral-domain OCT images. The proposed model includes 4 Convolutional layers and consists of 19 layers in total. In the study, classification was performed for 5 classes: diabetic macular edema (DME), age-related macular degeneration (AMD), Drusen, choroidal neovascularization (CNV), and Normal. Training of the suggested model was conducted over 100 epochs, with the utilization of the Adam optimization algorithm. In the test

using 1250 images, the general accuracy rate of the model is 97.12% and the accuracy values on a class basis are between 97.84% (Drusen) and 100% (AMD). In this study, no optimization algorithm was applied for the filter count and kernel dimension in the model. For hyperparameters such as epoch and learning speed, the trial and error method was applied. For testing the proposed approach, the test data specified in the dataset was used. A successful result was obtained with a small model with fewer parameters than transfer learning models. Tayal et al. (2021) proposed three custom CNN architectures for classification of eye disease. The models differ in the number of CNN layers they have (five, seven and nine layers). Before the images were transmitted to the models, they were first preprocessed and then different image enhancement techniques were applied. All three models were trained for 15 rounds and the Adam optimizer is utilized with a learning rate of 0.001 and a batch size of 84. As a result of the tests, the best result was obtained with the 7-layer model with 96.5% accuracy and 95.33% F1-score. High performance was achieved by enhancing the images with image processing techniques. The effect of 3 different architectures with different number of layers was analyzed. The study did not use an optimization algorithm to find the optimum architecture. Berrimi and Moussaoui (2020) proposed two CNN architectures for multiple classification and compared their performance with transfer learning models. The initial CNN design integrates three convolutional layers, with each layer's output subsequently processed through maximum pooling. Subsequently, a fully connected layer and an output layer, comprising four neurons that align with the class count, are appended. The second CNN architecture includes a dropout layer and a batch normalization layer in addition to the first CNN architecture. The first CNN model achieved 95.60% accuracy and the second CNN model achieved 98.75% accuracy.

As the model depth increases, it involves a significant the duration required to train a model from end to end. In addition, many images are required to achieve good classification performance. Access to both medical images and competent experts to take part in the labeling process can be difficult. To overcome these problems, rather than training the model from scratch to completion, it is preferred to train a certain number of layers using the transfer learning method, thus creating deep learning models with higher performance with fewer images. Saleh et al. (2022) performed a transfer learning study on classifying retinal weaknesses from OCT images. The images were preprocessed in two stages. First, contrast enhancement was performed and then anisotropic diffusion filtration algorithm was applied. In the study, The SqueezeNet, a customized version of SqueezeNet, and the InceptionV3 models were implemented using a transfer learning technique. Regarding accuracy, the Modified SqueezeNet model (98%) outperformed the original SqueezeNet model (96.85%). The InceptionV3 model achieved the highest accuracy of 98.4%. Li et al. (2019) presented a deep learning model for the determination of retinal eye illnesses, which combines deep features with handcrafted features. Handcrafted features are derived from Sift and Gabor filters. The study was carried out with three frameworks. In the first one, images and handcraft features are combined and sent to the deep architecture, while in the second one, images and handcraft features are sent separately to the deep architecture and the results are combined for classification. In the third framework, images and handcrafted features are combined after each convolutional block and RCNet model is used to realize this. With the dataset, three groups with different

distributions were created and training and testing were performed. The best performance for all three groups was obtained by adding the RCNet model and Gabor features. The authors explored combining manually designed features with those derived from deep learning methods. In order to examine the effect of different datasets, they divided the dataset into three different groups. Especially in group 2, a total of 1000 images were employed for training, and another 1000 images were reserved for testing. In this case, they showed that deep learning models integrated with handcrafted features perform better on small datasets. In Group 1, 1000 images of the dataset were used for testing and the rest for training. In Group 3, 50% of the dataset was reserved to training and the other 50% part to testing. Asif et al. (2022) presented a model for identifying eye illnesses from OCT images, utilizing transfer learning as a key component. In the study, ResNet50 architecture is used by modifying the last layer and training the entire system as a whole. A fully connected block including three dense layers, Relu activation function, three BatchNormalization layers, Dropout layer and L2 regularization is added to the RESNet50 model. The model was tested with 242 images for each class and 968 images in total, and an accuracy of 99.48% was obtained. Zheng et al. (2020) undertook an investigation on the use of synthetic images generated with GAN in the detection of retinal diseases. They used Kermany dataset consisting of original images and a synthetic dataset. The synthetic dataset was created from 130 urgent and 148 nonurgent images using GAN. This dataset contains 100,456 OCR images, of which 48,751 are urgent and 51,705 are nonurgent. The Inception V3 model was trained with both the original and synthetic dataset using transfer learning. The performance of the models was assessed across the actual image data and the fabricated image data. In the test phase using the original dataset, the effectiveness of the model trained using an artificially generated dataset was slightly lower than the model trained with the original images. Kermany et al. (2018) introduced a transfer learning model that utilizes the InceptionV3 architecture for classifying eye diseases in OCT images. Training of the model was performed across 100 epochs, utilizing a total of 108,312 images. Then, it was tested with 1000 images and a specificity of 97.4%, a sensitivity of 97.8%, and an accuracy of 96.6% was observed. The authors also investigated whether there would be any performance loss if the model underwent training using a reduced dataset. Training of the model utilized a dataset containing 1000 images for each class, and an accuracy rate of 93.4% was obtained as a result of the test. Tuncer et al. (2021) employed advanced, already trained models to perform the classification of OCT images into various categories. In the feature extraction phase of the proposed architecture, they utilized AlexNet, GoogleNet and ResNet18 architectures. Accuracy values of 96.88%, 97.40% and 95.36% were obtained respectively. To maximize the architectures' functional output, they used the SVM algorithm in the classification phase. When the results obtained are analyzed, Alexnet-SVM, Resnet18-SVM and Googlenet-SVM architectures obtained accuracy values of 98.96%, 95.36% and 98.2% respectively. Kim and Tran (2021) proposed two models in their study. In the first model, three binary CNN classifiers are used. These classifiers can be thought of as sequential, first reducing from 4 classes to 2 classes and then performing binary classification in the remaining classes. The first classifier (ResNet152) was used to classify the images into DME and CNV and Normal and Drusen. Then, the second classifier (InceptionV3) is used for CNV and DME discrimination and the third classifier (VGG19) is used for Drusen and Normal discrimination. Four distinct binary CNN classifiers are

employed by the second model. Each classifier detects a specific disease (Classifier1 (VGG16)-CNV vs. Others, Classifier2 (VGG16)-DME vs. others, Classifier3 (VGG19)-Drusen vs. others and Classifier4 (IncepitonV3)-Normal vs. others). Each classifier works only to detect the disease of interest and groups the rest of the classes as "other". Model 1 achieved 98.1% accuracy while model 2 achieved 98.7% accuracy. İncir and Bozkurt (2024a) used K-means clustering algorithm to segment hard exudates, which are important lesions of the disease, and increased the effect of these regions in the original image. Thus, the importance of data preprocessing is emphasized. In addition, they utilized ResNet50, MobileNet, DenseNet121 and EfficientNetV2-M architectures in the feature extraction phase. The extracted features are fed separately as input to the Global Average Pooling layer and then forwarded to the dense and dropout layers. At the end of the study, ResNet50, MobileNet, DenseNet121 and EfficientNetV2-M models achieved 91.07%, 88.62%, 91.87% and 94.36% accuracy on the original data set, respectively. On the preprocessed dataset, the accuracy values were 92.18%, 90.70%, 93.30% and 95.16%, respectively. İncir and Bozkurt (2024b) created a meaningful and sufficient dataset for diabetic retinopathy classification with the aid of well-designed data preparation and alteration methodologies. A selection of pre-trained models, specifically EfficientNetV2-M, MobileNet, VGG16, Inception-V3, Xception, DenseNet-121, and ResNet-50, was employed by them to extract features. As a result, EfficientNetV2-M architecture achieved the highest accuracy value with 97.65%.

3. MATERIAL AND METHOD

3.1. Dataset

A dataset consisting of four different classes of diseased and healthy, namely CNV, DME, DRUSEN, and NORMAL was used in this work (Kaggle, 2018). In this dataset, there are 37205 CNV diseased eye OCT images, 11348 DME diseased OCT images, 8616 DRUSEN diseased OCT images and 26315 NORMAL healthy OCT images to be used in training, while there are 242 OCT images in each class in the test. 8 images have been allocated to each class for validation. Since this number is quite small for validation, we rearranged the dataset by reserving 5% of the training dataset for validation. Figure 1 displays examples of images from this dataset.

Data partitioning results for the OCT data are presented in Table 1. The 8 images reserved for validation in the Kaggle data were added to the training dataset and 5% of each class was reorganized for validation based on this result.

3.2. Convolutional Neural Network

CNNs are deep learning architectures used in image processing for image recognition and segmentation that take raw images as input (Litjens et al., 2017, Kaya & Çetin-Kaya, 2024b). This algorithm, which captures features in images in different processes, consists of different layers (LeCun et al., 2015; O'Shea & Nash, 2015). CNNs perform feature extraction automatically from raw input images in the training phase, instead of using manual feature extraction in the training phase in classical machine learning.

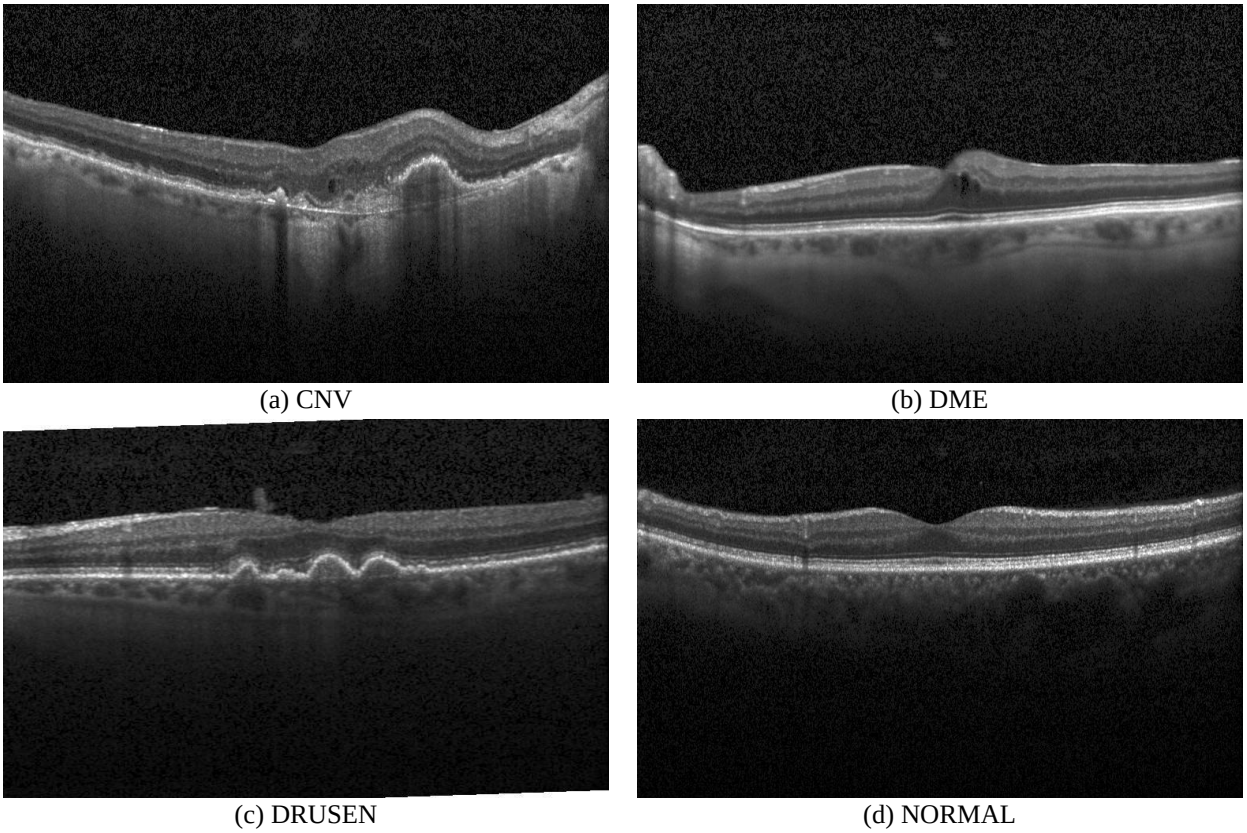


Figure 1. *Retinal Disease Images*

Table 1. *Data partitioning of the OCT dataset*

	Original dataset			After reserving the validation data (%5) from train		
	Train	Validation	Test	Train	Validation	Test
CNV	37205	8	242	35352	1861	242
DME	11348	8	242	10788	568	242
DRUSEN	8616	8	242	8193	431	242
NORMAL	26315	8	242	25007	1316	242

A CNN model is composed of three main components: initial feature determination via several sequential convolution layer, followed by a pooling layer, and concluding with a fully connected layer for network classification. (LeCun et al., 2015). Architecturally, a CNN is designed as a feed-forward network, including layers dedicated to normalization, feature extraction, and pooling. (LeCun et al., 2015; O'Shea & Nash, 2015). Neurons sharing identical filters are exclusively linked to localized image segments, maintaining the spatial arrangement, and their weights are shared to minimize the model's parameter count (Zeiler & Fergus, 2014). In the CNN architecture, convolutional layers in the early stages learn general features about the image such as color blobs, edges, and lines. Subsequent layers learn special forms specific to the dataset (O'Shea & Nash, 2015; Lu et al. 2017).

A significant benefit of CNNs is their capacity to reduce the parameter number within Artificial Neural Networks (ANN). CNNs use shared filter weights. CNNs can also shrink the dimension of the feature map after each sequential convolution operation through pooling mechanisms. Unlike ANNs, CNNs help to extract features spatially through filters that are moved over the image. Therefore, CNN empowers developers to address intricate challenges beyond the capabilities of traditional ANN and to construct more extensive models. As the layers deepen, more abstract features are obtained from the data presented as input to the CNN. This is critical for object detection (Schulz et al., 2018; Mascarenhas & Agarwal, 2021). Overfitting is one of the most serious issues in CNN architectures (O'Shea & Nash, 2015; Litjens et al., 2017). In CNNs, the number of parameters increases as the network gets deeper. In the case of limited labeled data, CNNs have high training accuracy because they can memorize the training data, they struggle with test data that has not been seen before (Zhang et al., 2021). Hence, it is crucial to ascertain the ideal quantity of filters within the convolutional layer in CNN designs.

3.3. Bayesian Optimization

Bayesian optimization involves a step-by-step, repetitive process most commonly used in hyperparameter optimization problems. In this method, compared to other hyperparameterization techniques, it determines the next evaluation points based on the results obtained previously. Bayesian optimization uses two basic components for this process; the acquisition and the surrogate functions (Frazier, 2018).

The surrogate function places all evaluated points into the objective function. The goal function determines the utilization of different points according to the link between discovery and utilization through the acquisition function, after calculating the probability through Bayes' theorem (Frazier, 2018). Bayes model is faster than other hyperparameter optimization techniques (Snoek et al., 2012). Because the optimized hyperparameter combinations can be determined with pre-tested values (Frazier, 2018; Fernandes et al., 2021).

Tree-structured Parzen estimator is used in this study. In each iteration, new observations are identified and tested by deciding the optimal hyperparameter result at the end of the iteration. The test results are added to the dataset and the iteration is continued. The Bayes formula in Equation 1 is applied (Brochu, et al., 2010; Fernandes et al., 2021). Performance comparison of the trial results is based on the Expected Improvement formula in Equation 2 (Brochu, et al., 2010; Fernandes et al., 2021).

$$p(y|x): p(x|y) = \frac{p(x|y) * p(y)}{p(x)} \tag{1}$$

$$EI_{f^*}(x) = \frac{\gamma f^* I_{(x)} \int_{-\infty}^{f^*} p(f)df}{\gamma I_{(x)} + (1-\gamma) g_{(x)}} \propto (\gamma + \frac{g_{(x)}}{I_{(x)}} (1-\gamma))^{-1} \tag{2}$$

The acquisition function chosen for this research is the Expected Improvement (EI). EI is a function used in the optimization process to select the best among candidate solutions. EI focuses on exploiting the current best solution by balancing between exploration and exploitation, i.e. the ability to search near the best value, and exploring new values in new search spaces.

3.4. Proposed Method

The model proposed in this paper consists of eight convolutional layers. Following the initial convolutional layer, a Batch Normalization layer and then a 2x2 max pooling layer were added. This initial structure was repeated once more. Then, after two convolutional and Batch Normalization layers, a max pooling layer was added. This last structure was repeated three times. A Batch Normalization layer was incorporated following every convolutional layer. Thus, eight convolutional layers were used in total. After the convolutional layer, which is the feature extraction layers, the flatten layer was added to make the data suitable for the densely connected layer. Two dense layers were implemented in the proposed design. After the flatten and fully connected layers, a dropout layer was added. In the last layer, since there are 4 classes in our dataset, we created an output layer with 4 neurons using softmax. The CNN architecture we used to achieve the best performance is shown in Figure 2.

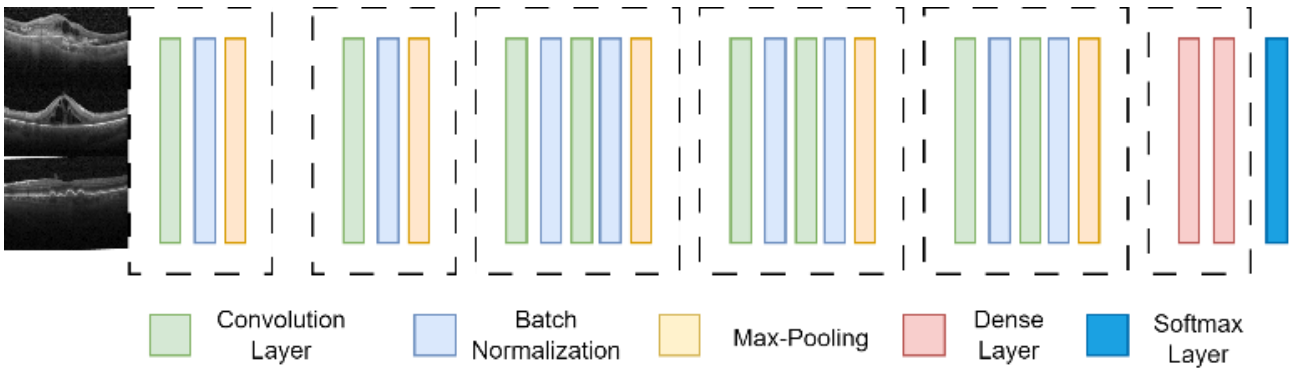


Figure 2. *Proposed CNN Architecture*

Utilizing Bayesian optimization, the best values were found for the dropout rate, the filter count in the convolutional layers, the quantity of neurons within the densely connected layer, learning rate, and filter kernel sizes. In fact, the number of filters is expected to be increased for more robust feature extraction in later layers. Since overfitting is often encountered in such cases, finding the best values is time-consuming when performed manually.

The procedure for Bayesian optimization is detailed in Algorithm 1. First, a dataset is created for several randomly selected combinations of hyperparameters. The surrogate model is trained with this dataset. From the candidate hyperparameter combinations, the one with the highest expected improvement is selected and tested on the real function.

Algorithm 1*. Bayesian Optimization*

Define surrogate model and acquisition function

Find initial dataset with random hyperparameter combinations

While i < maxIteration **do**:

-»Train the surrogate model with the dataset

-»Calculate expected improvement for candidate hyperparameter sets

-»Choose hyperparameter combination with high expected improvement

-»Find accuracy value for selected hyperparameter combination

-»Update the dataset with new hyperparameter combination

i++

end while

4. RESULTS

The experimental studies and all operations were performed on a regular PC configuration, which consisted of 16 gigabytes of RAM, an NVIDIA GeForce GTX 1080 Ti GPU with 11 gigabytes of memory, and an Intel i5-8400 processor. In this research, we designed the most successful model by optimizing the optimal filter count, the quantity of neurons within the fully connected layer and other hyperparameters for an architecture consisting of eight convolution layers and 2 densely connected layers. For the model, we set the epochs value to 50 after resizing the images to 224x224x3 and started training.

CNN models are generally an end-to-end learning architecture that automatically extracts features from the raw image. CNN models should have enough data for effective learning. Therefore, training times depend on the size of the dataset and are quite costly. There are many hyperparameters in CNN models, and optimizing all hyperparameters will extend the optimization process considerably. For this reason, the number of filters, kernel dimension, neuron count in the fully connected layer and dropout percentage, which are generally considered as the most important hyperparameters in the literature, are taken into consideration. For CNN models to be robust, feature extraction needs to be done in detail. In general, the number of filters and kernel size are effective at this point. The neuron count in the densely connected layer is important for classification accuracy, so classification performance after feature extraction is highly dependent on the neurons in the fully connected layer. The biggest problem in the training phase is overfitting. To avoid this, it is important to determine the appropriate dropout rate. Moreover, the hyperparameter ranges were determined based on high performance studies in the literature, state-of-the-art CNN model architectures and trial-and-error method.

Table 2 shows the hyperparameters to be optimized and their values. Table 3 shows the optimum hyperparameters for the proposed CNN architecture. The results of the two best CNN architectures can be seen comparatively in Table 3. The optimization process was terminated upon reaching 50 iterations, which was the pre-defined limit. Since the training of CNN models is very costly, the number of iterations should not be kept too high, but since it is thought that the desired performance value cannot be achieved at a small iteration value such as 20, the number 50 was chosen. This value was chosen to keep the training cost low and to achieve high performance.

Table 2. *Hyperparameters values (range)*

Hyperparameters	Value
Filter number of conv layers	from 16 to 256, step: 16
Kernel size	3x3, 5x5, 7x7
Dropout	from 0 to 0.8, step: 0.1
Dense neuron number	from 16 to 256 , step :16
Learning rate	0.00001; 0.0001; 0.001

The results of the performance evaluation are presented in Table 4, which includes the accuracy, precision, recall, F1-score, and AUC for each model. CNN models obtained as a result of Bayesian optimization gave the highest accuracy. In cases where the dataset is unevenly distributed, precision, recall and F1-score values should also be considered for comparison. The Bayesian model searches for the CNN model with the most optimal parameters for 50 iterations. As a result, it obtained the best performing models that do not fall into overfitting in the training phase. Existing state-of-the-art CNN models were also used for comparison. In these models with pre-trained weights, the final layers are removed up to the last convolution layer, after which the GlobalAveragePooling layer is added. After that, a dense layer with 512 neurons and a layer with a dropout percentage of 0.5 were added. Finally, a layer with four neurons was added for classification. Only the last layers of these models were trained with transfer learning. Among the transfer learning models, DenseNet201 was the best model with 95.87% accuracy.

As a result of the test data, the Confusion Matrix values for each class are shown in Figure 3. The classes in the confusion matrix are as follows: CNV(0), DME(1), DRUSEN(2), and NORMAL(3). Actual labels are represented along the horizontal axis, while the model's predicted values are displayed on the vertical axis. All of the images that belonged to the CNV and Normal classes were correctly classified by the proposed model 1. One image in the DME class and two images in the DRUSEN class were incorrectly classified. According to proposed model 2, the difference here is that 3 images that should be DRUSEN are incorrectly classified as CNV. Performance metrics were calculated as described in (Kaya et al., 2023).

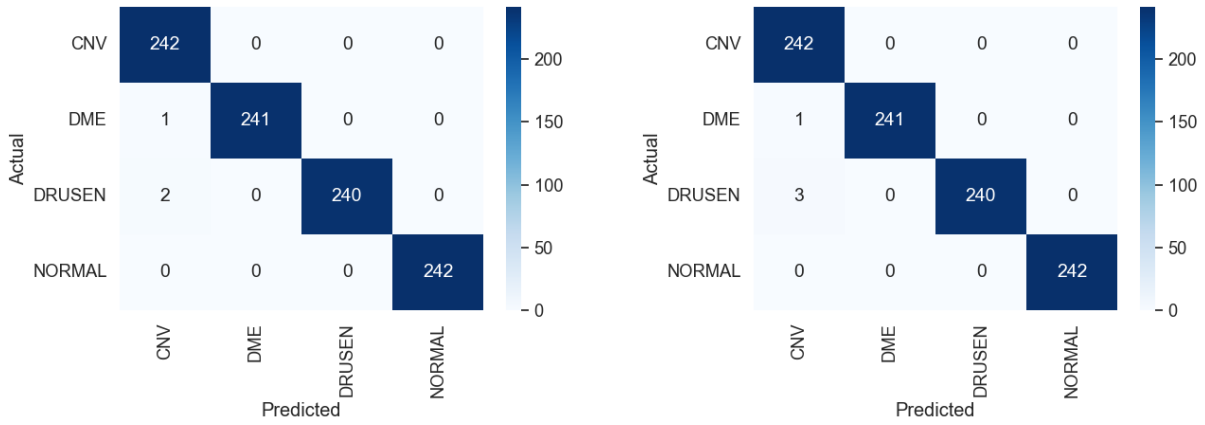
Table 3. *Best hyperparameters for two models*

Hyperparams	Model-1(A,BxB)	Model-2(A,BxB)

Conv1	112, 3x3	112, 3x3
Conv2	112, 5x5	112, 5x5
Conv3	112, 5x5	16, 3x3
Conv4	48, 3x3	48, 3x3
Conv5	240, 3x3	208, 5x5
Conv6	144, 3x3	48, 3x3
Conv7	112, 3x3	112, 5x5
Conv8	112, 5x5	208, 3x3
Dropout 1	0	0
Dense 1	128	96
Dropout 2	0.2	0.4
Dense 2	96	80
Dropout 3	0	0
Learning rate	0.0001	0.001
(A, BxB) , A stands for the number of filters, B is the kernel size of the filter		

Table 4. *Performance metrics of proposed models and transfer learning based models*

Models	Accuracy(%)	Precision(%)	Recall(%)	F1-Score(%)	AUC(%)
DenseNet121	93.29	94.25	93.29	93.31	95.52
DenseNet169	94.63	95.01	94.53	94.62	96.42
DenseNet201	95.87	96.08	95.87	95.87	97.25
VGG19	87.50	90.14	87.50	87.20	91.67
InceptionV3	94.01	94.48	94.01	93.96	96.01
InceptionResNetV2	93.18	93.96	93.18	93.15	95.45
Xception	90.39	92.32	90.39	90.32	93.60
MobileNetV2	94.42	94.86	94.42	94.42	96.28
Proposed Model-1	99.69	99.69	99.69	99.69	99.79
Proposed Model-2	99.59	99.59	99.59	99.59	99.72



Proposed Model 1 Proposed Model 2

Figure 3. *Confusion Matrix for Proposed Models*

Figure 4 presents the training-validation accuracy and loss graphs of the proposed models during the training. When these graphs are analyzed, it can be said that the models generally do not fall into an overlearning situation. If the models had fallen into a state of overlearning, for example in the accuracy graph, after a certain epoch value, the training accuracy would start to improve, that is, to increase or continue stably, while the validation graph would start to fall downward after this epoch value and the validation performance would start to decrease. However, while the training accuracy tended to increase over the epoch, the validation accuracy did not increase in the same way and generally followed a near-parallel trend with a slight gap between them. The validation graph shows a sharp drop in the dropout cases, but quickly recovers in the following epochs.

Figure 5 and 6 show the training-validation accuracy and loss graphs of the 10-fold cross validation method in each fold respectively. For 10-fold cross validation, the training and test dataset are combined and the whole dataset is divided into 10 folds. In each training phase, the previously unused part of the 10 partitions will be used for testing and the remaining part will be used for training. In this way, the model will have tried all the samples in the dataset for testing. In this way, the accuracy of the models in each fold will be considered together and the average accuracy will be the model's total efficacy. The accuracy obtained in this way will better reflect the dataset. At this stage, the best CNN architectures found by Bayesian optimization were reconstructed with the optimum parameters found and retrained and then tested on the 10-fold cross validation dataset. If we consider the accuracy graphs at this stage, the training and validation accuracy graphs continued in an increasing and mostly overlapping manner throughout the epochs. This means that during training and validation, the models learned at full capacity without overlearning. Averaging all test accuracies in each fold yields an average accuracy of 97.03% for Bayesian model 1 and 96.73% for Bayesian model 2. Validation methods such as 10-fold cross validation usually yield the most reliable model accuracies. However, none of the existing studies applied k-fold-cross-validation. This dataset is partitioned into training, validation and testing on the Kaggle website. For each class, 8 images are allocated as validation. In our study, 5% of the training dataset is reserved for validation. In the dataset, 968 images are allocated for testing and this partition

is used only for testing purposes after the model training is finished. Since the majority of current studies typically report the test image accuracies based on these 968 images, it becomes easier to make comparisons in this way.

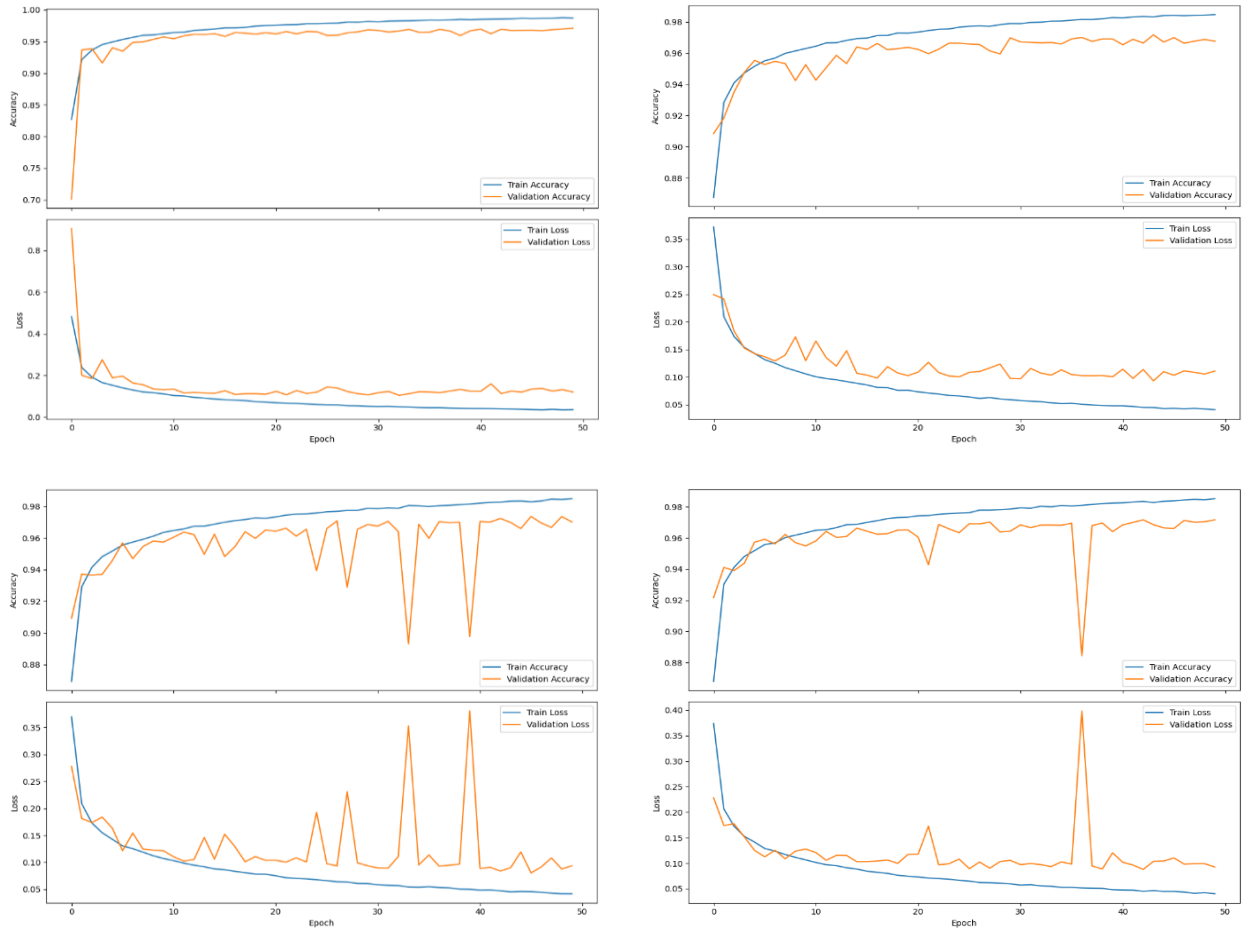


Figure 4. *Training-validation accuracy and loss graphs for Proposed Models*

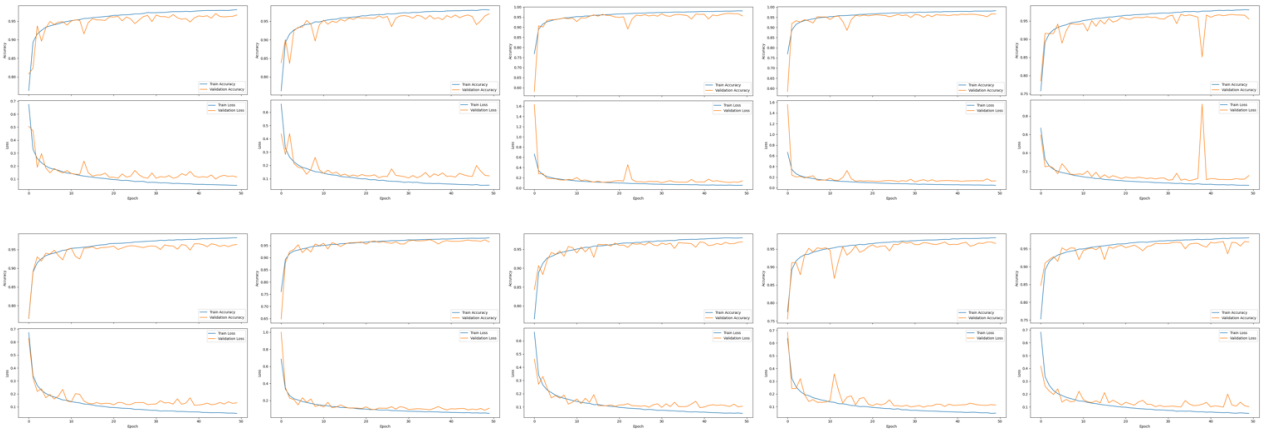


Figure 5. *Training-validation accuracy and loss graphs for 10-fold cross validation of Bayes Model 1*

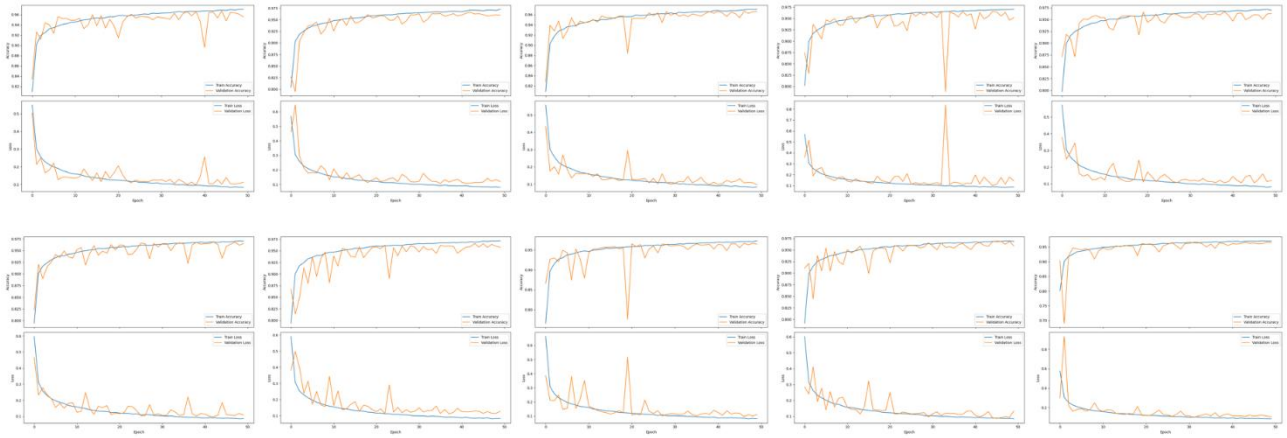


***Figure 6**. Training-validation accuracy and loss graphs for 10-fold cross validation of Bayes Model 2*

Figure 7 presents the training-validation accuracy and loss graphs of state-of-the-art CNN models. Considering the loss graph, the training-validation loss graphs are declining over the epochs and the important point is that the validation loss graph follows the training loss graph from the top and since there is some space between them, these models have the capacity for improvement. Figure 8 presents the confusion matrix tables of popular pre-trained CNN models. When the confusion matrices of the models are analyzed, it is seen that most of the errors are due to the misclassification of the DRUSEN class as CNV. Then the DME class is misclassified as CNV and finally the DME class is misclassified as NORMAL. The data volume for the most frequently misclassified categories can be augmented.

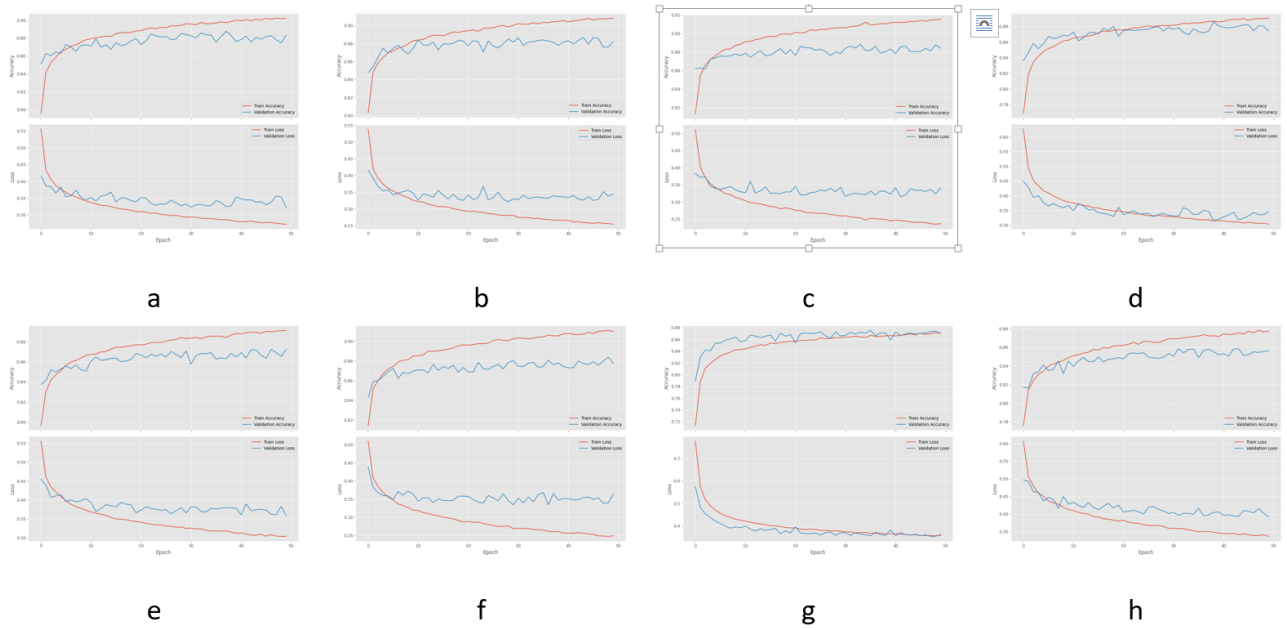


***Figure 7**. Training and validation accuracy and loss graphs for state-of-the-art CNN models a) DenseNet121 b) DenseNet169 c) DenseNet201 d) InceptionResNetV2 e)InceptionV3 f)MobileNetV2 g)VGG19 h)Xception*

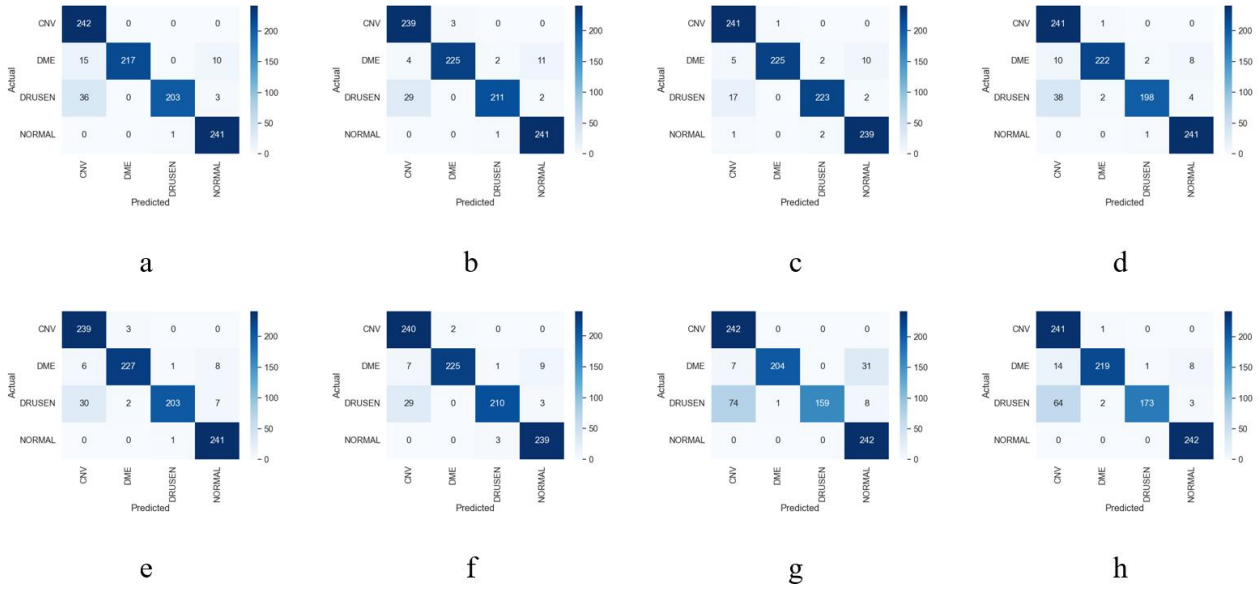


Figure 8*. Confusion Matrix for state-of-the-art CNN models a) DenseNet121 b) DenseNet169 c) DenseNet201 d) InceptionResNetV2 e)InceptionV3 f)MobileNetV2 g)VGG19 h)Xception*

Table 5 summarizes the existing studies, the architectures they use and their results. The proposed CNN architecture gave better results than the current studies.

Table 5. *Proposed CNN model comparison with existing studies*

Reference	Model	Number of class	Performance Metrics	
			F1-Score	Accuracy
Kermany et al. (2018)	InceptionV3	4 classes	0,9760	0.966
Li et al. (2019)	RCNET	4 classes	0,9819*	0,988
Kim and Tran (2021)	Transfer Learning	2 classes	0,99	0,987
Alqudah (2020)	Custom CNN	5 classes	0,9819*	0.9712
Tayal et al. (2021)	Custom CNN	4 classes	0,95	0,9649
Saleh et al. (2022)	InceptionV3	4 classes	0,96	0,9840
Asif et al. (2022)	Resnet50	4 classes	0,99	0,9948
Proposed Model	Custom CNN	4 classes	0,9969	0,9969

*Calculated from the confusion matrix.

5. DISCUSSION

This study proposes a lightweight CNN design for high accuracy classification of eye diseases. Since the CNN architecture contains many hyperparameters, it is almost impossible to tune them manually to find the optimum architecture. Therefore, an optimal architecture is obtained with a Bayesian optimization based algorithm. On the Kaggle web page, the dataset is set as training, validation and test. In our wok, 5% of the training dataset is allocated as validation since there is very little validation-only dataset. In the literature, k-fold cross validation is generally not used. Data is commonly partitioned into the training, validation, and test sets. Generally, different test datasets were created and used than the test dataset shared on the Kaggle website. This makes it difficult to compare with many existing studies. The dataset is large enough, but there is an imbalance in the number of images in each class.

Li et al. (2019) investigated the integration of handcrafted features with different datasets into deep learning architecture. In this study, it is seen that when the data is organized into training and test by 50%, the accuracy value drops significantly compared to the test case with approximately 1000 images. Asif et al. (2022) used the original dataset and 968 test image data in the same way as us. They added a fully connected layer and dropout to the ResNet50 architecture for transfer learning. They found the same test accuracy as our work with a more complex architecture and more parameters. In addition, they increased the original dataset by approximately 8 times. This process increases the training cost considerably. Alqudah (2020) proposed a model with 4 convolution layers and found a 5-class classification with 97.12% accuracy. 1250 images were used for testing. The remaining dataset was split, with 70% used for training and 30% used for validation. Although it is unclear how the four-layer architecture and the number of filters in each layer were determined, the trial and error method was used for hyperparameters such as learning rate, batch size and epochs. Tayal et al. (2021) divided the dataset into training, validation and testing. About 8% of the dataset was used for testing. Three different custom CNN architectures were proposed. The highest accuracy was 96.5%. Since authors selected the test data more than the one given on the Kaggle web page, it is not possible to make an accurate comparison. Saleh et al. used transfer learning methods such as SqueezeNet, modified SqueezeNet and Inception V3. They divided the dataset into parts to keep the training time short. For each model, they divided these parts into training validation and testing. They used 2700 images for testing in their best model with 98.4% accuracy.

After a review of the relevant literature, research using transfer learning models have generally achieved high accuracy values. Since these models are trained from the beginning with fine tuning, their training is more costly. They also have high parameter numbers. Other custom models have fewer parameters than state-of-the-art transfer learning models, but their overall test accuracy has been limited. Considering all these evaluations, our proposed model reached higher accuracy than the current studies.

6. CONCLUSION

Early diagnosis plays a key role in managing retinal eye diseases. Retinal eye disease classification is often conducted through the analysis of OCT images; however, expert-based manual classification can sometimes be inaccurate. For this reason, automatic classification of retinal diseases by computer-aided systems is gaining importance. CNN architectures have recently demonstrated successful performance in disease detection from medical images. For successful image classification, CNN architectures need to be well designed. In CNN architectures, hyperparameters such as the number of filters in each layer, kernel dimension, learning rate, dropout and the count of nodes within the fully connected layer should be optimally determined. Since there are many combinations of hyperparameters, it is very difficult to perform this process manually. Therefore, we have determined the optimum hyperparameters with a Bayesian optimization based algorithm. High performance metrics (99.69% accuracy and F1 score) were achieved with the proposed model. These results are better than the existing research findings in the literature. The intention of the proposed model is to streamline the decision-making process and lessen the workload of expert personnel. In future studies, optimization will be performed on different architectures.

CONFLICT OF INTEREST

The author declares no conflict of interest.

REFERENCES

Alqudah, A. M. (2020). AOCT-NET: a convolutional network automated classification of multiclass retinal diseases using spectral-domain optical coherence tomography images. *Medical & Biological Engineering & Computing*, 58, 41-53. https://doi.org/10.1007/s11517-019-02066-y

Asif, S., Amjad, K., & Qurrat-ul-Ain (2022). Deep residual network for diagnosis of retinal diseases using optical coherence tomography images. *Interdisciplinary Sciences: Computational Life Sciences*, *14*(4), 906-916. https://doi.org/10.1007/s12539-022-00533-z

Berrimi, M., & Moussaoui, A. (2020). *Deep learning for identifying and classifying retinal diseases*. In 2020 2nd International Conference on computer and information sciences (ICCIS) (pp. 1-6). IEEE. https://doi.org/10.1109/ICCIS49240.2020.9257674

Brochu, E., Cora, V. M., & De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. arXiv preprint arXiv:1012.2599. https://doi.org/10.48550/arXiv.1012.2599

Cheyi, J., & Çetin-Kaya, Y. (2024). Advanced CNN-Based Classification and Segmentation for Enhanced Breast Cancer Ultrasound Imaging. *Gazi University Journal of Science Part A: Engineering and Innovation*, *11*(4), 647-667. https://doi.org/10.54287/gujsa.1529857

Çetin-Kaya, Y. (2024). Equilibrium Optimization-Based Ensemble CNN Framework for Breast Cancer Multiclass Classification Using Histopathological Image. *Diagnostics*, *14*(19), 2253. https://doi.org/10.3390/diagnostics14192253

Çetin-Kaya, Y., & Kaya, M. (2024). A Novel Ensemble Framework for Multi-Classification of Brain Tumors Using Magnetic Resonance Imaging. *Diagnostics*, *14*(4), 383. https://doi.org/10.3390/diagnostics14040383

Çevik, İ., Çakmak, H., Çelik, Ö., & Okyay, P. (2021). Yaşam Boyu Göz Sağlığı: "2020 Vizyonu: Görme Hakkı". *ESTÜDAM Halk Sağlığı Dergisi*, 6(3), 310-321. https://doi.org/10.35232/estudamhsd.891156

Duran, O., Turan, B., & Kaya, M. (2025). Machine-learning-based ensemble regression for vehicle-to-vehicle distance estimation using a toe-in style stereo camera. *Measurement*, 240, 115540. https://doi.org/10.1016/j.measurement.2024.115540

Farsiu, S., Chiu, S. J., O'Connell, R. V., Folgar, F. A., Yuan, E., Izatt, J. A., ... & Age-Related Eye Disease Study 2 Ancillary Spectral Domain Optical Coherence Tomography Study Group. (2014). Quantitative classification of eyes with and without intermediate age-related macular degeneration using optical coherence tomography. *Ophthalmology*, *121*(1), 162-172. https://doi.org/10.1016/j.ophtha.2013.07.013

Fernandes, V., Junior, G. B., de Paiva, A. C., Silva, A. C., & Gattass, M. (2021). Bayesian convolutional neural network estimation for pediatric pneumonia detection and diagnosis. *Computer Methods and Programs in Biomedicine*, 208, 106259. https://doi.org/10.1016/j.cmpb.2021.106259

Frazier, Peter I., A Tutorial on Bayesian Optimization, arXiv:1807.02811v1, 2018, doi: https://doi.org/10.48550/arXiv.1807.02811

Fujimoto, J. G., Pitris, C., Boppart, S. A., & Brezinski, M. E. (2000). Optical coherence tomography: an emerging technology for biomedical imaging and optical biopsy. *Neoplasia*, *2*(1-2), 9-25. https://doi.org/10.1038/sj.neo.7900071

Güneş, A., & Çetin-Kaya, Y. (2020). Evrişimsel Sinir Ağları ile Görüntülerde Gürültü Türünü Saptama. *Bilgisayar Bilimleri ve Mühendisliği Dergisi*, *17*(1), 75-89. https://doi.org/10.54525/bbmd.1454595

İncir, R., & Bozkurt, F. (2024a). *A Study on the Segmentation and Classification of Diabetic Retinopathy Images Using the K-Means Clustering Method*. In: 32nd Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE. https://doi.org/10.1109/SIU61531.2024.10600987

İncir, R., & Bozkurt, F. (2024b). A study on effective data preprocessing and augmentation method in diabetic retinopathy classification using pre-trained deep learning approaches. *Multimedia Tools and Applications*, *83*(4), 12185-12208. https://doi.org/10.1007/s11042-023-15754-7

Kaggle (2018). Retinal OCT Images. (Accessed:17/03/2024). URL

Kaya, M. (2024). Feature fusion-based ensemble CNN learning optimization for automated detection of pediatric pneumonia. *Biomedical Signal Processing and Control*, 87, 105472. https://doi.org/10.1016/j.bspc.2023.105472

Kaya, M., & Çetin-Kaya, Y. (2024a). A Novel Deep Learning Architecture Optimization for Multiclass Classification of Alzheimer's Disease Level. *IEEE Access*, 12, 46562-46581. https://doi.org/10.1109/ACCESS.2024.3382947

Kaya, M., & Çetin-Kaya, Y. (2024b). A novel ensemble learning framework based on a genetic algorithm for the classification of pneumonia. *Engineering Applications of Artificial Intelligence*, 133, 108494. https://doi.org/10.1016/j.engappai.2024.108494

Kaya, M., Ulutürk, S., Çetin-Kaya, Y., Altıntaş, O., & Turan, B. (2023). Optimization of Several Deep CNN Models for Waste Classification. *Sakarya University Journal of Computer and Information Sciences*, 6(2), 91-104. https://doi.org/10.35377/saucis...1257100

Kermany, D. S., Goldbaum, M., Cai, W., Valentim, C. C., Liang, H., Baxter, S. L., ... & Zhang, K. (2018). Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell, 172(5), 1122-1131. https://doi.org/10.1016/j.cell.2018.02.010

Kim, J., & Tran, L. (2021). *Retinal disease classification from oct images using deep learning algorithms*. In 2021 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB) (pp. 1-6). IEEE. https://doi.org/10.1109/CIBCB49929.2021.9562919

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *Imagenet classification with deep convolutional neural networks*. In Advances in Neural Information Processing Systems (NIPS) (pp. 1097-1105). https://doi.org/10.1145/3065386

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436-444. https://doi.org/10.1038/nature14539

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). *Gradient-based learning applied to document recognition*. Proceedings of the IEEE, 86(11), 2278-2324. https://doi.org/10.1109/5.726791

Li, X., Shen, L., Shen, M., & Qiu, C. S. (2019). Integrating handcrafted and deep features for optical coherence tomography based retinal disease classification. *IEEE Access*, 7, 33771-33777. https://doi.org/10.1109/ACCESS.2019.2891975

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... & Sánchez, C. I., (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60-88, https://doi.org/10.1016/j.media.2017.07.005

Lu, Y., Yi, S., Zeng, N., Liu, Y., & Zhang, Y., (2017). Identification of rice diseases using deep convolutional neural networks. *Neurocomputing*, 267, 378-384. https://doi.org/10.1016/j.neucom.2017.06.023

Malkoç, İ. (2006). Göz Küresinin Tabakaları: Anatomik ve Histolojik Bir Derleme. *Eurasian J Med*, 38, 124-129.

Mascarenhas, S., & Agarwal, M. (2021). *A comparison between VGG16, VGG19, and ResNet50 architecture frameworks for image classification*. 2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON). https://doi.org/10.1109/CENTCON52345.2021.9687944

O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458. https://doi.org/10.48550/arXiv.1511.08458

Saleh, N., Abdel Wahed, M., & Salaheldin, A. M. (2022). Transfer learning-based platform for detecting multi-classification retinal disorders using optical coherence tomography images. *International Journal of Imaging Systems and Technology*, *32*(3), 740-752. https://doi.org/10.1002/ima.22673

Schulz, E., Speekenbrink, M., & Krause, A. (2018). A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions. *Journal of Mathematical Psychology*, 85, 1-16. https://doi.org/10.1016/j.jmp.2018.03.001

Silverman, A. L., Tatham, A. J., Medeiros, F. A., & Weinreb, R. N. (2014). Assessment of optic nerve head drusen using enhanced depth imaging and swept source optical coherence tomography. *Journal of neuro-ophthalmology: the official journal of the North American Neuro-Ophthalmology Society*, *34*(2), https://doi.org/198. 10.1097/WNO.0000000000000115

Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. arXiv:1206.2944v2. https://doi.org/10.48550/arXiv.1206.2944

Tayal, A., Gupta, J., Solanki, A., Bisht, K., Nayyar, A., & Masud, M. (2021). DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases. *Multimedia Systems*, 28(4), 1-22. https://doi.org/10.1007/s00530-021-00769-7

Tuncer, S. A., Çınar, A., & Fırat, M. (2021). Hybrid CNN Based Computer-Aided Diagnosis System for Choroidal Neovascularization, Diabetic Macular Edema, Drusen Disease Detection from OCT Images. *Traitement du Signal*, *38*(3). https://doi.org/10.18280/ts.380314

Wu, Z., Ayton, L. N., Guymer, R. H., & Luu, C. D. (2013). Relationship between the second reflective band on optical coherence tomography and multifocal electroretinography in age-related macular degeneration. *Investigative ophthalmology & visual science*, *54*(4), 2800-2806. https://doi.org/10.1167/iovs.13-11613

Zhang, C., Bengio, S., Hardt, M., Recht, B., & Vinyals, O. (2021). Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3), 107-115. https://doi.org/10.1145/3446776

Zheng, C., Xie, X., Zhou, K., Chen, B., Chen, J., Ye, H., ... & Liu, J. (2020). Assessment of generative adversarial networks model for synthetic optical coherence tomography images of retinal disorders. *Translational Vision Science & Technology*, *9*(2), 29-29. https://doi.org/10.1167/tvst.9.2.29

Zeiler, M. D., & Fergus, R. (2014). *Visualizing and understanding convolutional networks*. In Computer Vision–ECCV 2014: 13th European Conference on Computer Vision, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I (pp. 818-833). Springer International Publishing. https://doi.org/10.1007/978-3-319-10590-1_52