A Comprehensive Study of DCNN Algorithms-based Transfer Learning for Human Eye Cataract Detection

Omar Jilani Jidan[1], Susmoy Paul[2], Anirban Roy[3], Sharun Akter Khushbu[4], Mirajul Islam[5], S.M. Saiful Islam Badhon[6]
Department of Computer Science and Engineering, Daffodil International University, Dhaka 1341, Bangladesh[1, 2, 3, 4, 5, 6]
Faculty of Graduate Studies, Daffodil International University, Dhaka 1341, Bangladesh[5]

***Abstract*—This study presents a comparative analysis of different deep convolutional neural network (DCNN) architectures, including VGG19, NASNet, ResNet50, and MobileNetV2, with and without data augmentation, for the automatic detection of cataracts in fundus images. Utilizing hybrid architecture models, namely ResNet50-NASNet and ResNet50+MobileNetV2, which combine two state-of-the-art DCNNs, this research demonstrates their superior performance. Specifically, MobileNetV2 and the combined ResNet50+MobileNetV2 outperform other models, achieving an impressive accuracy of 99.00%. By emphasizing the efficacy of diverse datasets and pre-processing techniques, as well as the potential of pretrained DCNN models, this study contributes to accurate cataract diagnosis. Furthermore, the proposed system has the potential to reduce reliance on ophthalmologists, decrease the cost of eye check-ups, and improve accessibility to eye care for a wider population. These findings showcase the successful application of deep learning and image processing techniques in the early detection and treatment of various medical conditions, including cataracts, addressing the needs of individuals with diminished vision through ocular images and innovative hybrid architectures.**

Keywords—Cataract detection; eye disease; ocular images; deep convolutional neural network (DCNN); hybrid architecture

I. INTRODUCTION

The eye is a crucial component of the human body, but conditions such as cataracts can impede its function. Cataracts are characterized by the formation of a cloudy, impenetrable layer on the eye, typically as a result of aging. This can lead to a variety of vision issues, including difficulties with tasks such as reading, driving, and recognizing people. According to a report by the World Health Organization (WHO), a staggering 2.2 billion individuals will be affected by blindness or visual impairments by the year 2025, with 1 billion of these cases predicted to be preventable. Cataracts are a significant contributor to this trend, with an estimated 40 million people expected to lose their sight due to this condition.

In Bangladesh, cataracts account for 80% of eye problems affecting individuals over 30 years old, with approximately 120,000 new cases reported each year. This issue is particularly prevalent in rural areas where access to medical services and ophthalmologists is limited, resulting in a high number of cases of blindness. Early detection of cataracts can prevent complete blindness, but current detection methods are costly and require an ophthalmologist. Therefore, there is a need to develop an automated cataract diagnosis system to reduce the burden on medical professionals, lower costs, and make eye care more accessible to a wider population.

Automated systems for identifying and assessing cataracts have been a subject of research for many years. Despite previous studies analysing fundus images for cataract detection and classification, their suboptimal performance was attributed to limitations in feature extraction and pre-processing methods. To enhance the precision of deep learning models, an extensive literature review was conducted, evaluating various aspects including datasets, pre-processing techniques, feature extraction methods, feature selection criteria, classifiers, and models. Our findings revealed the potential of deep CNN architecture-based models using image processing methods for detecting cataracts on fundus images. Surprisingly, few studies have utilized DCNN architecture-based models with transfer learning for cataract detection. Many researchers have investigated the use of deep convolutional neural network-based architectures, including VGG19 [1-2] and ResNet [3], for the purpose of detecting cataracts. These architectures, which fall under the category of deep neural network-based architectures [4], have gained widespread recognition in the fields of computer vision and imaging due to their ability to efficiently tackle tasks such as segmentation, detection, classification, and image analysis [5]. They comprise multiple layers of information processing stages. To identify cataracts from fundus images, researchers have utilized pre-trained deep neural network models via transfer learning techniques [6]. Moreover, other models such as MobileNetV2 and NASNet can be employed for image classification tasks, and are also suitable for the purpose of cataract classification.

This study aims to bridge a research gap in the field of AI-assisted medical diagnosis by focusing on the application of deep learning and image processing techniques for the automatic detection of cataracts in ocular images. Unlike previous studies, this research provides a comparative analysis of model performance, specifically examining the impact of data augmentation on various Deep Convolutional Neural Networks (DCNNs). By employing the ResNet50-NASNet and ResNet50+ MobileNetV2 ensemble model, which combines two state-of-the-art DCNNs, this study achieves a higher accuracy rate in detecting cataracts while maintaining efficient image recognition and classification. Furthermore, the proposed system has the potential to reduce the dependence on ophthalmologists, decrease the cost of eye check-ups, and enhance accessibility to eye care for a wider population. These significant contributions shed light on the successful application of deep learning and image processing techniques for the early detection and treatment of not only cataracts but also various other medical conditions.

The key contribution of this study can be summarized as follows:

1) Demonstrated the effectiveness of diverse datasets and pre-processing techniques in improving the accuracy of deep learning models in medical image analysis and classification.

2) Highlighted the potential of pre-trained DCNN models in accurately identifying cataracts in fundus images.

3) Showcased the benefits of hybrid architectures in enhancing the accuracy of medical image analysis and classification.

4) Provided insights into the impact of data augmentation on the performance of deep learning models in medical image analysis and classification.

The structure of this paper consists of four sections. Section I serves as an introduction, while Section II includes a review of related work and a literature review of the challenges. Section III provides an outline of the methodology and materials utilized in the research. In Section IV, the results and discussion of the study are presented. Section V lists the limitations and future work. Finally, Section VI contains the conclusion along with a list of references.

II. Literature Review

The following section provides a comprehensive literature review on the subject matter, examining relevant studies and scholarly works. Weni et al. [7] proposed a CNN-based system for cataract classification achieving an accuracy of 97% in 50 epochs using a basic CNN architecture with ReLU and SoftMax activation. Hossain et al. [8] developed a ResNet50-based method for cataract detection in fundus images. Zhou et al. [9] used the DST method with residual network to minimize overfitting and memory shortage problems and employed the ResNet architecture for cataract detection and grading. Zhang et al. [10] provided a multi-model ensemble method for automatic cataract detection using ultrasound images and achieved the highest accuracy of 97.5%. Nihal et al. [11] addressed the issue of overfitting in large datasets by utilizing residual network architecture. VGG19, ResNet50, and Resnet101 models were employed in [12] and [13]. Neha et al. [14] proposed a multi-class, multi-label cataract detection system utilizing fundus images from publicly available and private datasets. Similarly, Lakshmi et al. [15] and Nadim et al. [16] utilized VGG16 for cataract detection. Conversely, Khan et al. [17] utilized transfer learning approach based on a VGG19-based CNN achieving an accuracy of 97.47%.

Table I provides a comprehensive overview of relevant studies, their methods, models and datasets utilized in the context of eye disease. Several studies have recently explored cutting-edge technologies aimed at improving accuracy and efficiency in different fields. Among them, four papers [28-31] stand out for their innovative approaches showcase the latest technologies, such as saliency detection networks, EMG signals, ant colony optimization, and graph-based extreme learning machines. In recent years, there has been a growing interest in combining machine learning algorithms with optimization techniques to enhance the accuracy and efficiency of models. Several recent studies [32-37] have investigated the potential of such combinations, as highlighted in the literature.

TABLE I. Literature Review of Relevant Studies

Paper	Objectives	Method	Dataset Type	Source
[18]	Detect Cataract and Identify the Severity	DCNN	Fundus Image	ODIR, Kaggle
[19]	Detect Cataract and Non-Cataract	ResNet50	Fundus Image	ODIR, Kaggle
[20]	Detect Cataract and Non-Cataract	Resnet with DST	Fundus Image	Private
[21]	Detect Cataract and Identify the Severity	DCNN	Fundus Image	Private
[22]	Eye Disease Detection	VGGNet	Fundus Image	ODIR, Kaggle
[23]	Detect Cataract and Non-Cataract	CNN, VGGNet	Fundus Image	Private
[24]	Eye Disease Detection	VGGNet. ResNet, AlexNet	Fundus Image	Private
[25]	Detect Cataract and Identify the Severity	VGGNet. ResNet, AlexNet	Fundus Image	Private
[26]	Eye Disease Detection	VGGNet	Fundus Image	ODIR, Kaggle
[27]	Eye Disease Detection	VGG16, EfficientNet, ResNet, MobileNetV2	Fundus Image	ODIR, Kaggle

III. Methodology

This section describes the tools utilized in the data collection process, data analysis, and the proposed deep learning method for classifying cataracts, specifically a multi-layer neural network architecture. The model is capable of distinguishing cataracts from fundus images. The architecture of the proposed system is illustrated in Fig. 1. The framework of the cataract detection system comprises several steps, including image acquisition, preprocessing, model implementation, and performance analysis. These steps are thoroughly described in the subsequent sections.

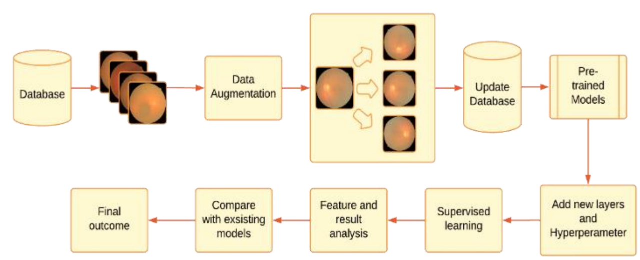


Fig. 1. Workflow of eye cataract detection.

A. Dataset Properties

The present study utilized data collected from publicly available datasets, namely, the Ocular Disease Recognition [43] and Eye Diseases Classification [44], as well as datasets from Kaggle. Fig. 2 provides an overview of the dataset images used in the study. The figure presents a visual representation of the dataset, showcasing examples of fundus images included in the research. The aforementioned datasets comprised images of normal and cataract eye conditions that were labeled by qualified authorities. The combined dataset utilized in this study included two classes, cataract and normal, and a total of 2000 images. The cataract class comprised 1000 images, while the normal class contained 1000 images that were appropriately labeled and split in a suitable ratio. In order to increase the amount of data in the dataset, image augmentation was applied. After augmentation, the total number of images in the dataset was increased to 5000. During the training phase, 80% of the data was utilized, with the remaining 20% reserved for the testing session. The combination of datasets and augmentation techniques utilized in this study provided a diverse and extensive dataset, which was utilized for the development and evaluation of an automated system for the recognition of cataracts in eye images.

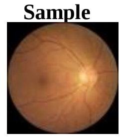
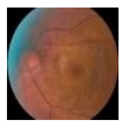
Category	Source	Sample	Category	Source	Sample
Cataract	ODIR [43]		Normal	ODIR [43]	
Cataract	EDC [44]		Normal	EDC [44]	

Fig. 2. Dataset image overview.

B. Data Pre-Processing

Multiple fundus image datasets were acquired, and a pre-processing pipeline was implemented to ensure consistency and enhance the quality of the data. Fig. 3 presents a step-by-step visualization of the data pre-processing methods used in the study. The images were selectively filtered to include only cataract and normal images, while excluding others such as diabetic retinopathy, hypertension, pathological myopia, glaucoma, age-related macular degeneration, among others.

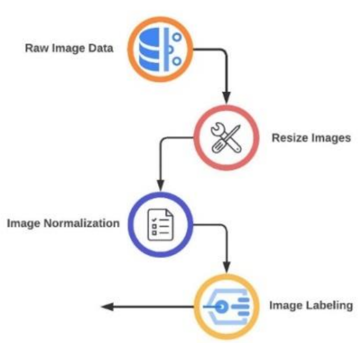


Fig. 3. Data pre-processing methods step by step.

To standardize the image sizes, the OpenCV library was utilized for resizing and normalization, involving subtracting the mean value from all pixels. Each image was appropriately labelled as either cataract or normal, and the dataset was then converted into an array format using NumPy, facilitating the subsequent training process.

To enhance the model's capacity to generalize to new images, augmentation techniques were applied, including rotation and flipping. These techniques were implemented using the Keras framework [38]. Fig. 4 depicts the dataset splitting process employed in the study. The random augmentation process was applied to both the training and testing data to enhance the model's ability to learn with a larger dataset. The arguments used in the augmentation process are provided in Table II.

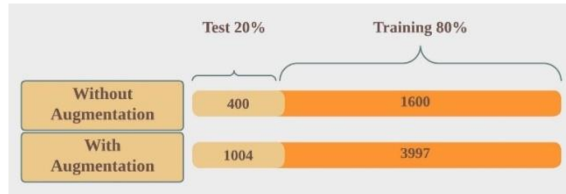


Fig. 4. Dataset splitting.

TABLE II. ARGUMENTATION FUNCTIONAL VALUE

Augmentation	Value
Rescale	1/0.255
Rotation	20
Flip	True

Rescale: Rescaling is a method of adjusting the range of values of a dataset by multiplying or dividing each value by a constant factor. It is commonly used in image processing, to standardize the data where pixel values are often represented as integers between 0 and 255, and rescaling the data can help to prevent overfitting and improve model's performance. Data augmentation rescaling by a factor of 1/.255 can be mathematically represented as:

Suppose, a 2D matrix

A = [[1, 0, 0], [0, 1, 0], [0, 0, 1]]

$$A\_scaled = A^* (1/.255) \tag{1}$$

Rotation: Data augmentation rotation by 20 degrees can be mathematically represented as a matrix multiplication of the 2D matrix with the rotation value. Here, θ is the data point after rotation.

Rotation value,

R = [[cos(20), -sin(20)], [sin(20), cos(20)]]

$$\theta = A^* R \tag{2}$$

Flipping: Data augmentation with flip horizontal and flip vertical can be mathematically represented as a transformation matrix applied to each point in the image.

For flip horizontal, the transformation matrix value,

Fh = [[-1, 0, 0], [0, 1, 0], [0, 0, 1]]

$$A = A * Fh \tag{3}$$

For flip vertical, the transformation matrix value,

Fv = [[1, 0, 0], [0, -1, 0], [0, 0, 1]]

$$A = A * Fv \tag{4}$$

However, the testing data will also be used in conjunction with the first augmentation strategy. Fig. 4 illustrates the distribution of images in the test and train sets both before and after applying the augmentation techniques.

C. Model Selection

In order to detect cataracts in fundus eye images obtained from ocular datasets, various model architectures were employed, namely VGG19, NASNet, ResNet50, and MobileNetV2. These architectures were selected to enhance the accuracy of cataract detection and improve the overall performance of the models. A detailed analysis of the method architecture is presented below for a better understanding.

1) Visual geometry group architecture: The Visual Geometry Group at the University of Oxford developed a CNN model called VGG19, known for its depth and use of small convolutional filters and max pooling layers [39]. In VGG19, feature maps are down-sampled by small 3x3 convolutional filters that focus on the most important features. To enhance the overall effectiveness of the model and address overfitting, we implemented several techniques, such as adding a dense layer with 512 neurons and a ReLU activation function, a dropout layer with a rate of 0.5, and a dense layer with 49 neurons and a sigmoid activation function. The model also includes a global average pooling layer, which reduces the spatial dimensions of the input by taking the average of the values in each channel, allowing the model to focus on the most important features and improve performance. Finally, we added a 1-unit dense layer with a sigmoid activation function [shown in Fig. 5(a)] to make the final prediction.

2) Neural architecture search: Neural Architecture Search (NAS) is a type of convolutional neural network developed by the Google Brain team [40]. The goal is to design an architecture with minimal human intervention and limited resources. In our study, we used a pre-trained NASNet model with ImageNet weights and added some additional layers. We introduced a fully connected layer architecture of 512 neurons with ReLU activation, followed by dropout with a rate of 0.5. Next, we added another fully connected layer of 49 neurons with sigmoid activation. We down-sampled the output using a 2D global max pooling layer and then extracted all-combinational features using a dense layer to make the final prediction [Fig. 5(b)].

3) Residual network architecture: The author in [41] is a convolutional neural network (CNN) developed by Microsoft Research, which has gained widespread popularity for image classification tasks. It is a deep neural network comprising 50 layers and is designed with a residual architecture. This model has produced outstanding results on various image classification tasks. It is referred to as a "residual" network due to its residual functions related to the input rather than learning the desired functions directly. This property enables the network to learn complex, highly non-linear functions more efficiently, resulting in much higher accuracy than traditional feedforward networks. In this study, we utilized the pre-trained ResNet50 model on ImageNet and added additional layers, such as dropout with a rate of 0.5, a dense layer with 512 neurons, Global average pooling layers, and a single-node dense layer, as shown in Fig. 5(c).

4) MobileNetV2: [42] is an advanced neural network architecture designed to efficiently classify images on mobile and embedded devices. Developed by Google and introduced in their 2018 publication "MobileNetV2: Inverted Residuals and Linear Bottlenecks," this architecture represents an improved version of its predecessor, MobileNetV1. MobileNetV2 utilizes a combination of linear bottlenecks, inverted residual blocks, and short connection paths to reduce the number of computations required by the network. This results in a faster and more efficient model while maintaining a similar level of accuracy. Additionally, MobileNetV2 is highly versatile and can perform well on a variety of devices. It can easily be integrated with existing neural network architectures to build larger, more complex models. To further improve accuracy, we added some additional layers to the pretrained MobileNetV2 and fine-tuned it for cataract detection [Fig. 5(d)].

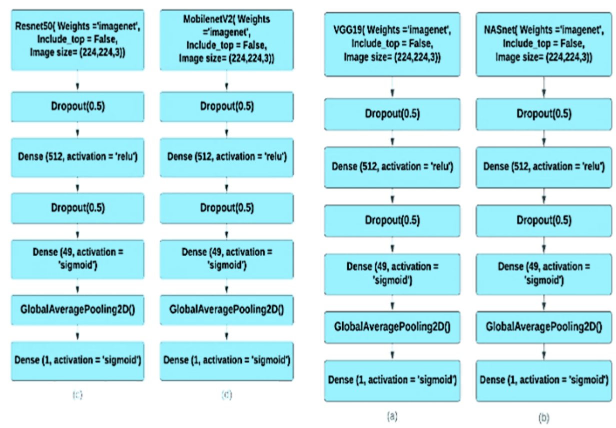


Fig. 5. Layer visualization of four different DCNN models.

5) Ensemble learning with combined ResNet50+NASNet and ResNet50+MobileNetV2: Ensemble learning is a popular technique in machine learning that combines multiple models to improve accuracy and reduce overfitting. Fig. 6 showcases the layer visualization of the ResNet50+NASNet hybrid deep convolutional neural network (DCNN). In this case, two powerful models, ResNet50+NASNet and ResNet50+MobileNetV2, are combined to create a hybrid model. By combining these models, the resulting hybrid model benefits from the strengths of each individual model, resulting in higher accuracy and robustness. Additionally, the non-trainable parameters in both models contribute to the overall performance by improving the generalization ability of the model. It is worth noting that the trainable parameters in both ResNet50+MobileNetV2 and ResNet50+NASNet are significantly lower than the total parameters, which is due to the use of transfer learning. It utilizes the strengths of each individual model and benefits from the non-trainable parameters to achieve state-of-the-art performance.

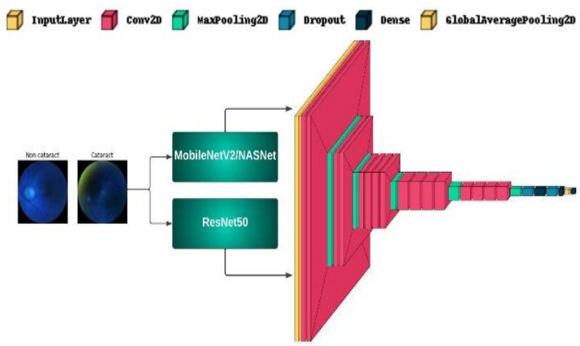


Fig. 6. Layer visualization ResNet50+NASNet hybrid DCNN.

D. Evaluation Metrics

This subsection elucidates the confusion metrics employed to gauge the efficiency of the models. The matrix provides insight into the number of true positives, true negatives, false positives, and false negatives, allowing for a more in-depth analysis of the model's strengths and weaknesses. Confusion matrix is widely used in various fields, such as medical diagnosis, fraud detection, and image classification, where the accuracy of the predictions is critical. By analyzing the confusion matrix, data scientists can identify the areas where the model is performing well and areas where it needs improvement, enabling them to fine-tune the model to achieve better results.

Accuracy: This is one of the evaluation metrics, represented by the ratio of correct prediction and the total predictions made.

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

Sensitivity: This metric is used to evaluate model performance by measuring its ability to detect positive prediction. It also called true positive rate (TPR) or recall.

$$\text{Sensitivity/Recall} = TP/(TP + FN) \quad (6)$$

Specificity: This metric calculate the percentage of true negatives which are actually negative. It can also be referred to as the True Negative Rate (TNR).

$$\text{Specificity} = TN/(TN + FP) \quad (7)$$

Precision: Precision is another evaluation element that calculates a model's performance by finding the ratio of Positives predictions and total number of Positives predictions. It measures the accuracy of positive predictions made by the model, particularly in the case of minority class predictions.

$$\text{Precision} = TP/(TP + FP) \quad (8)$$

F1 score: F1 score is an alternative evaluation metric that combines precision and recall scores to determine a model's accuracy. It is reliable when the classes in the dataset are balanced and have a similar number of data points.

$$\text{F1 score} = (2 \times \text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall}) \quad (9)$$

E. Experimental Setting

During the experimental phase, the binary cross-entropy loss function and Adam optimizer were employed to train the models. The batch size was set to 49, and the models underwent training for a duration of 15 epochs. The loss between the true and predicted labels was calculated throughout the training process to assess the model's performance. The execution environment for this experiment was Google Colab, which provided 25 GB of RAM but no GPU acceleration. The compiling parameters used in this experiment are summarized in Table III.

TABLE III. COMPILING EXPERIMENTAL SETUP PARAMETERS

Parameters	Value
Initial learning rate	1*e*-3
Optimizer	Adam
Loss	Binary cross-entropy
Epochs	15
Batch size	49
Executable Environment	Colab

IV. RESULT AND DISCUSSION

In this section, a comprehensive analysis and interpretation of the results obtained from the different models performed will be presented. The analysis will provide insights into the effectiveness of the models in the given task, and highlight the strengths and weaknesses of each approach. Additionally, the results will be evaluated using appropriate statistical methods to determine the level of significance and confidence in the findings.

The Tables IV and V represents the performance of Six different deep learning algorithms with and without data augmentation, namely VGG19, ResNet50, NASNet, MobileNetV2, and two combinations of ResNet50 with either NASNet or MobileNetV2, in terms of accuracy, specificity, sensitivity, F1 score, and precision. It can be observed from Table IV that MobileNetV2 achieved the highest accuracy score of 99.00%, followed closely by Combine ResNet50+MobileNetV2 with an accuracy of 98.75%. NASNet also performed well, achieving an accuracy of 98.50%. On the other hand, ResNet50 had the lowest accuracy among the algorithms, achieving only 72.50%. This indicates that ResNet50 may not be the most appropriate algorithm for the given task. However, it is important to note that accuracy alone may not be sufficient to evaluate the performance of these algorithms. Other metrics such as sensitivity, specificity, F1 score, and precision should also be considered for a more comprehensive understanding of their performance. Additionally, the choice of algorithm should be based on the specific requirements and constraints of the task at hand, such as computational cost, training time, and resource availability. Overall, based on the analysis of the provided table, it can be concluded that MobileNetV2 achieved the best performance in terms of accuracy, specificity, sensitivity, F1 score, and precision without data augmentation.

TABLE IV. PERFORMANCE OF THE DL ALGORITHMS WITHOUT DATA AUGMENTATION

Model	Accuracy	Specificity	Sensitivity	F1 score	Precision
VGG19	96.25%	95.48%	97.01%	96.30%	95.59%
ResNet50	72.50%	72.86%	72.14%	72.50%	72.86%
NASNet	98.50%	99.50%	97.51%	98.49%	99.49%
MobileNetV2	99.00%	99.50%	98.51%	99.00%	99.50%
Combine ResNet50+NASNet	97.75%	98.49%	97.01%	97.74%	98.48%
Combine ResNet50+ MobileNetV2	98.75%	99.50%	98.01%	98.75%	99.49%

TABLE V. PERFORMANCE OF THE DL ALGORITHMS WITH DATA AUGMENTATION

Model	Accuracy	Specificity	Sensitivity	F1 score	Precision
VGG19	97.41%	98.01%	96.81%	97.39%	97.98%
ResNet50	88.65%	93.44%	83.83%	88.05%	92.72%
NASNet	98.51%	100.00%	97.01%	98.48%	100.00%
MobileNetV2	98.61%	99.80%	97.41%	98.59%	99.80%
Combine ResNet50+NASNet	97.81%	100.00%	95.61%	97.76%	100.00%
Combine ResNet50+ MobileNetV2	99.00%	99.60%	98.40%	99.00%	99.60%

From the accuracy metric, it can be seen in Table V that MobileNetV2 performed with an accuracy score of 98.61%, followed by NASNet with an accuracy of 98.51% and Combine ResNet50+MobileNetV2 performed best with an accuracy of 99.00% with data augmentations. ResNet50 performed poorly with an accuracy of only 88.65%. The specificity metric shows how well the models can identify true negatives. It can be observed that all models except ResNet50 achieved high specificity scores, with NASNet and the combined models achieving a perfect score of 100%. The sensitivity metric shows how well the models can identify true positives. MobileNetV2 had sensitivity score of 97.41%, followed by Combine ResNet50+MobileNetV2 had the highest with a score of 98.40%. ResNet50 had the lowest sensitivity score of 83.83%. The F1 score, which is the harmonic mean of precision and recall, provides a balance between the two metrics. It can be seen that all models except ResNet50 achieved high F1 scores, the combined ResNet50+ MobileNetV2 model achieving a perfect score 99.00%. Finally, the precision metric shows how well the models can avoid false positives. All models except ResNet50 achieved high precision scores, with NASNet and the combined ResNet50+NASNet model achieving a perfect score of 100%.

The provided Fig. 7(a-f) displays plots that depict the relationship between the training and validation accuracy of each deep learning model without data augmentation. The graphs illustrate how the accuracy of each model changes as the number of epochs increase during the training process. It can be observed that for some models, such as MobileNetV2 and NASNet, the validation accuracy increases in a consistent and linear manner with the training accuracy. However, for other models such as ResNet50, the validation accuracy starts to plateau while the training accuracy continues to increase. These plots provide valuable insights into how each model performs during training and can be useful for further model optimization.

Fig. 8(a-f) displays plots that exhibit the relationship between the training and validation loss of each deep learning model without data augmentation. The plots demonstrate varying behaviors of the models during training, where some models exhibit a consistent and linear decrease in both training and validation loss, while others experience a plateau in validation loss and a continued decrease in training loss. These plots provide valuable information on the convergence properties of each model and can be used for optimizing the models' performance. Furthermore, analyzing the loss function during the training phase is a common technique used in evaluating the effectiveness of deep learning models, and these plots can provide insights into the models' learning dynamics.

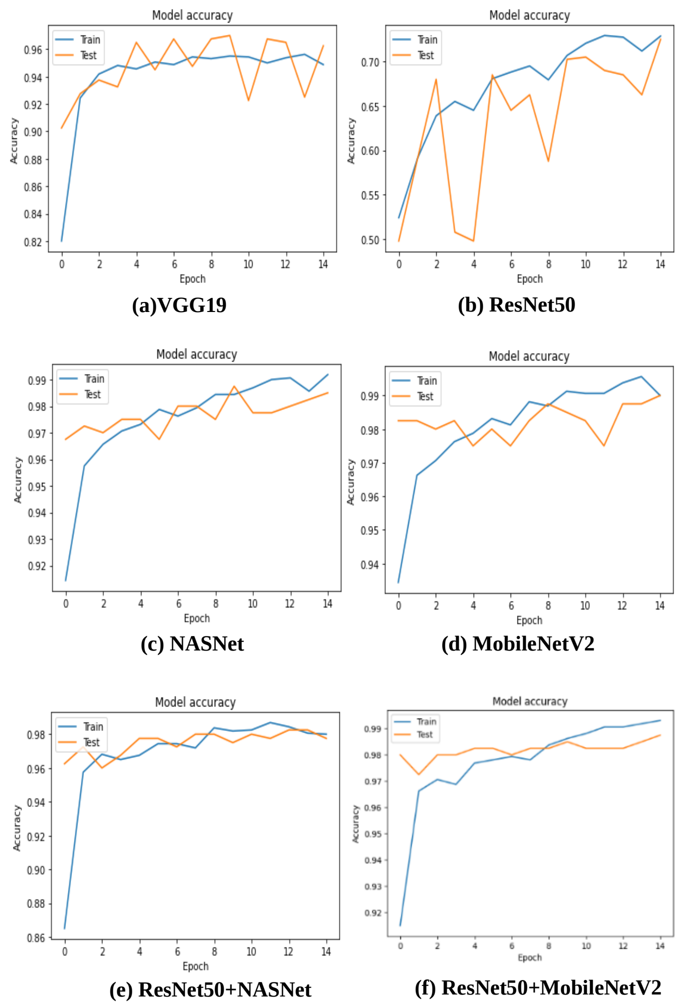
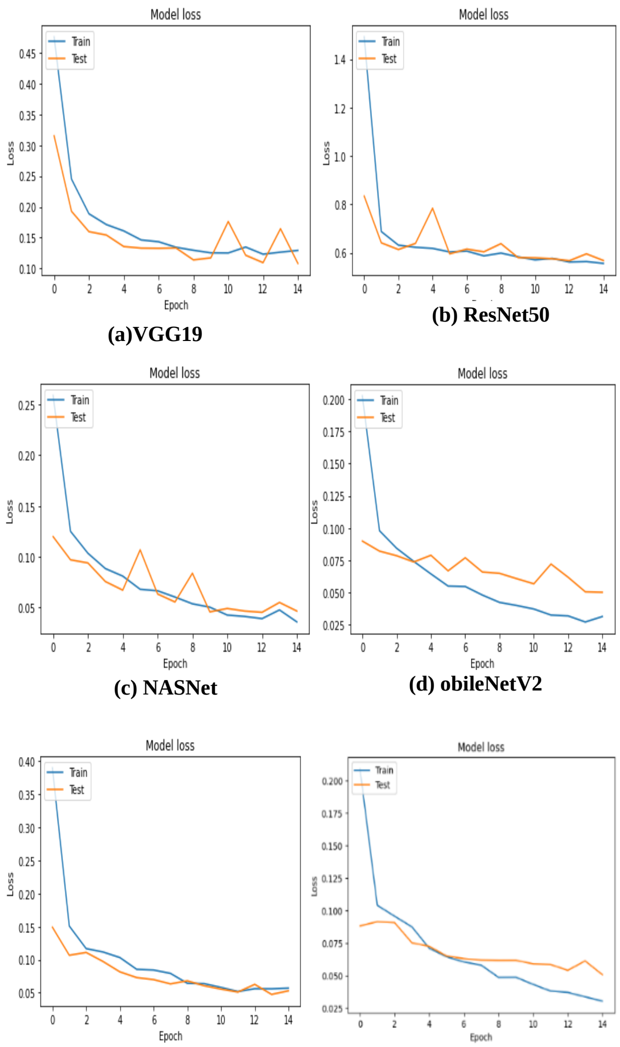


Fig. 7. (a-f) Plot showing the relationship between each deep learning model’s training & validation accuracy without data augmentation.



(a)VGG19 **(b) ResNet50**

(c) NASNet **(d) obileNetV2**

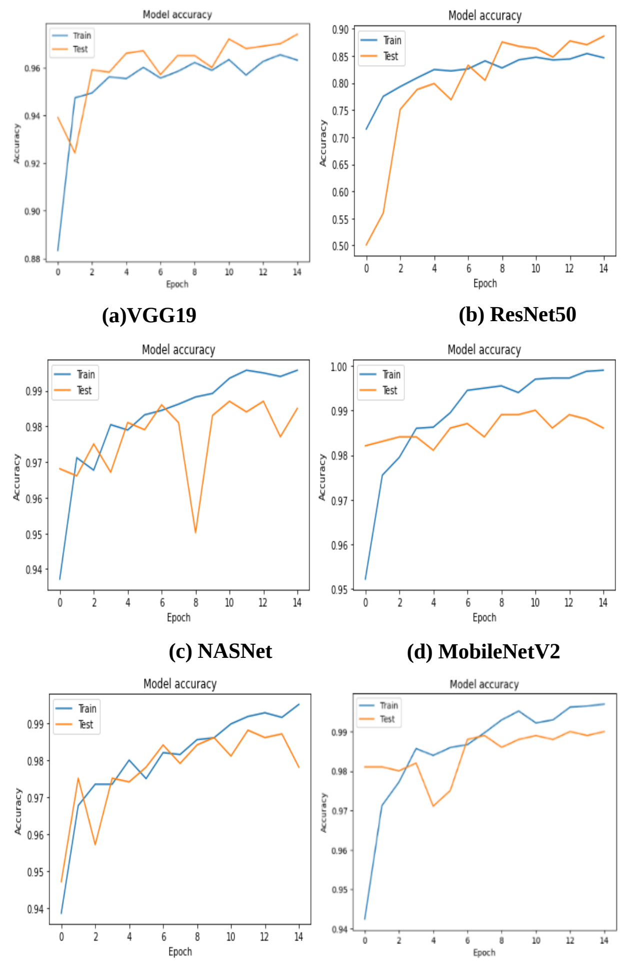
(e) ResNet50+NASNet **(f) ResNet50+MobileNetV2**

Fig. 8. (a-f) Plot showing the relationship between each deep learning model's training & validation loss without data augmentation.

Fig. 9 and 10 present plots that illustrate the relationship between the training and validation accuracy (Fig. 9) and training and validation loss (Fig. 10) of each deep learning model with data augmentation. Comparing the results of the two Fig. with those without data augmentation (Fig. 7 and 8) provides insights into the impact of data augmentation on model performance. The graphs in Fig. 9 show an overall improvement in the training and validation accuracy of all models with data augmentation, where the gap between the two accuracies is smaller compared to the previous figure. Moreover, the models' accuracy tends to increase faster during the initial epochs of training. In Fig. 10, the behaviors of the models' training and validation loss is similar to that without data augmentation, although the loss values are slightly higher. This observation suggests that data augmentation can help to improve model accuracy without increasing overfitting.

Overall, the comparative analysis of Fig. 7 and 8 with Fig. 9 and 10 suggests that data augmentation can help to enhance the performance of deep learning models in image classification tasks. By introducing variations in the training data, data augmentation can help models to generalize better to new data and improve their accuracy. Furthermore, analyzing the changes in the loss function and accuracy during the training process with and without data augmentation can provide useful insights into the learning dynamics of deep learning models and can guide model selection and optimization.

The comparison of accuracy between deep learning architectures with and without data augmentation highlights a significant improvement in performance when augmentation techniques are applied. The combined ResNet50+MobileNetV2 model achieved the highest accuracy of 99.00% with augmentation, while without augmentation; the accuracy was slightly lower at 98.75%. Notably, ResNet50 demonstrated a substantial increase in accuracy with augmentation, scoring 88.65% compared to 72.50% without augmentation. Fig. 11 visually represents this comparative performance, emphasizing the positive impact of data augmentation techniques on the accuracy and effectiveness of deep learning architectures. These findings affirm the benefits of employing augmentation techniques for improved model performance in image classification tasks.



(a)VGG19 **(b) ResNet50**

(c) NASNet **(d) MobileNetV2**

(e) ResNet50+NASNet **(f) ResNet50+MobileNetV2**

Fig. 9. (a-f) Plot showing the relationship between each deep learning model's training & validation accuracy with data augmentation.

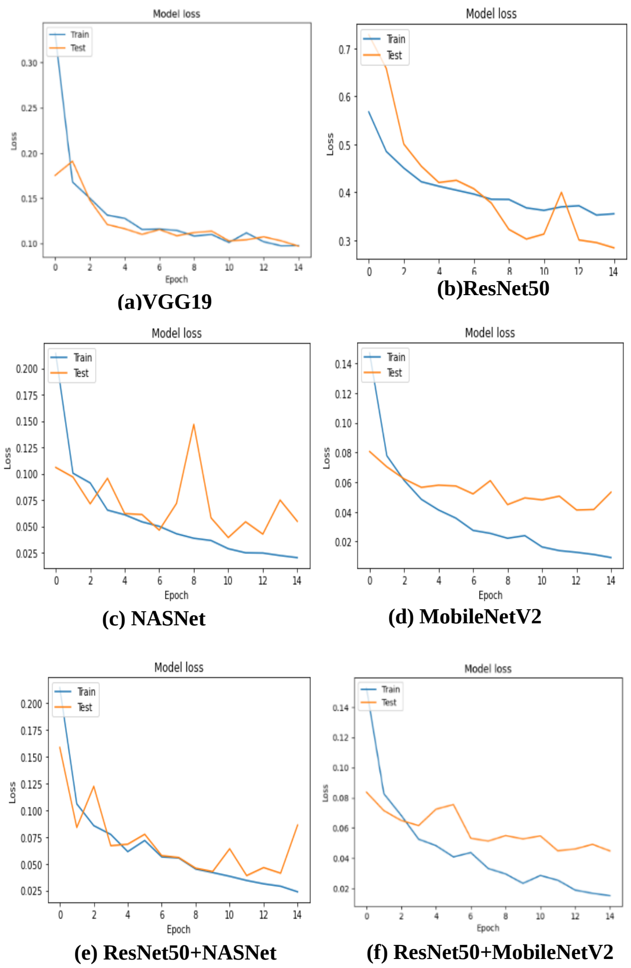


Fig. 10. (a-f) Plot showing the relationship between each deep learning models training & validation loss with data augmentation.

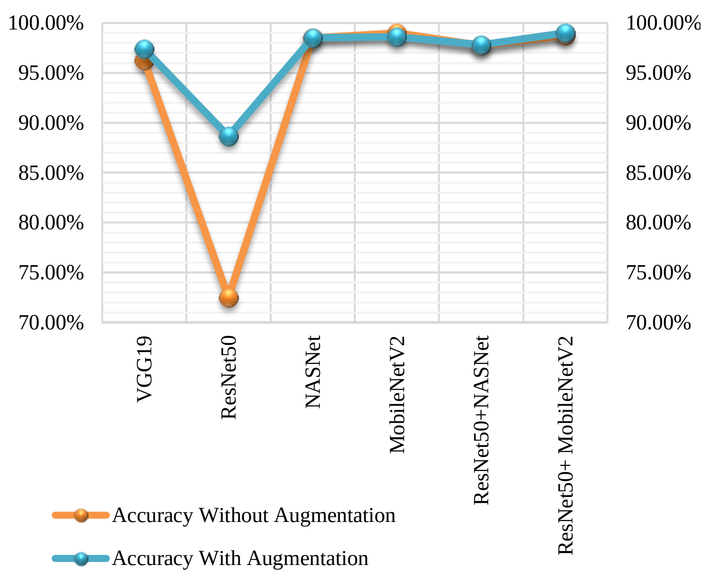


Fig. 11. Comparative performance of DL architectures with and without data augmentation.

V. Study Limitations and Scope for Future Research

Despite our efforts to conduct a comprehensive study, it is important to acknowledge certain limitations that should be taken into consideration when interpreting the results. First, the sample size in our study was relatively small, which may limit the generalizability of the findings. Additionally, the study was conducted within a specific geographic region, and the results may not be applicable to other populations or settings. Moreover, the data collection process relied on self-report measures, which may introduce response biases or inaccuracies. Lastly, the study design was cross-sectional, which limits our ability to establish causal relationships between variables.

Our study has opened up several avenues for future research in this field. Firstly, future studies could consider employing a larger and more diverse sample to enhance the external validity of the findings. Longitudinal studies could be conducted to explore the causal relationships between the variables of interest. Additionally, conducting similar research in different geographic regions or cultural contexts could provide valuable insights into the generalizability of the results. Furthermore, incorporating objective measures or alternative data collection methods would enhance the reliability and validity of the findings. Lastly, exploring the effectiveness of intervention programs or strategies targeting the identified variables could be an important area for future research.

VI. Conclusion

Various deep convolutional neural network architectures were compared with transfer learning for the accurate classification of fundus images for cataract diagnosis. This study demonstrates the efficacy of deep convolutional neural network (DCNN) architectures for automatic cataract detection in fundus images. Specifically, MobileNetV2 and the combined ResNet50+MobileNetV2 models exhibit superior performance, achieving an impressive accuracy of 99.00%. The utilization of diverse datasets, data augmentation, and hybrid architecture models, such as ResNet50-NASNet and ResNet50+MobileNetV2, contributes to accurate cataract diagnosis. The findings highlight the potential of deep learning and image processing techniques in early detection and treatment of medical conditions, particularly cataracts. Furthermore, the proposed system shows promise in reducing the reliance on ophthalmologists, decreasing the cost of eye check-ups, and improving access to eye care for a wider population. This research underscores the successful application of innovative hybrid architectures and emphasizes the importance of leveraging pretrained DCNN models for accurate and efficient cataract diagnosis.

References

[1] Xiong, Li, Huiqi Li, and Liang Xu. "An approach to evaluate blurriness in retinal images with vitreous opacity for cataract diagnosis." Journal of healthcare engineering 2017 (2017).

[2] Rawat, Waseem, and Zenghui Wang. "Deep convolutional neural networks for image classification: A comprehensive review." Neural computation 29, no. 9 (2017): 2352-2449.

[3] Li, Tao, Wang Bo, Chunyu Hu, Hong Kang, Hanruo Liu, Kai Wang, and Huazhu Fu. "Applications of deep learning in fundus images: A review." Medical Image Analysis 69 (2021): 101971

[4] Lai, Chi-Ju, Ping-Feng Pai, Marvin Marvin, Hsiao-Han Hung, Si-Han Wang, and Din-Nan Chen. "The Use of Convolutional Neural Networks and Digital Camera Images in Cataract Detection." Electronics 11, no. 6 (2022): 887.

[5] Shin, Hoo-Chang, Holger R. Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura, and Ronald M. Summers. "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning." IEEE transactions on medical imaging 35, no. 5 (2016): 1285-1298.

[6] Hasan, Md Kamrul, Tanjum Tanha, Md Ruhul Amin, Omar Faruk, Mohammad Monirujjaman Khan, Sultan Aljahdali, and Mehedi Masud. "Cataract Disease Detection by Using Transfer Learning-Based Intelligent Methods." Computational and Mathematical Methods in Medicine 2021 (2021).

[7] Weni, Indra, Pradita Eko Prasetyo Utomo, Benedika Ferdian Hutabarat, and Muksin Alfalah. "Detection of Cataract Based on Image Features Using Convolutional Neural Networks." Indonesian Journal of Computing and Cybernetics Systems) 15, no. 1 (2021): 75-86.

[8] Hossain, Md Rajib, Sadia Afroze, Nazmul Siddique, and Mohammed Moshiul Hoque. "Automatic detection of eye cataract using deep convolution neural networks (DCNNs)." In 2020 IEEE region 10 symposium (TENSYMP), pp. 1333-1338. IEEE, 2020.

[9] Zhou, Yue, Guoqi Li, and Huiqi Li. "Automatic cataract classification using deep neural network with discrete state transition." IEEE transactions on medical imaging 39, no. 2 (2019): 436-446.

[10] Zhang, Xiaofei, Jiancheng Lv, Heng Zheng, and Yongsheng Sang. "Attention-based multi-model ensemble for automatic cataract detection in B-scan eye ultrasound images." In 2020 international joint conference on neural networks (IJCNN), pp. 1-10. IEEE, 2 020.

[11] Bhandary, Nihal, and Anish Adnani. "Cataract Eye Detection using Machine Learning Models." (2020)

[12] Triyadi, Ahmad Bondan, Alhadi Bustamam, and Prasnurzaki Anki. "Deep Learning in Image Classification using VGG-19 and Residual Networks for Cataract Detection." In 2022 2nd International Conference on Information Technology and Education (ICIT&E), pp. 293-297. IEEE, 2022.

[13] Imran, Azhar, Jianqiang Li, Yan Pei, Faheem Akhtar, Ji-Jiang Yang, and Yanping Dang. "Automated identification of cataract severity using retinal fundus images." Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization 8, no. 6 (2020): 691-698.

[14] Gour, Neha, and Pritee Khanna. "Multi-class multi-label ophthalmological disease detection using transfer learning based convolutional neural network." Biomedical Signal Processing and Control 66 (2021): 102329.

[15] N. Prof. V. Lakshmi, N. L. Vinay, N. L. Monisha, N. Mounesh, and N. N. M. S, "Ocular Disease Recognition and Detection using VGG Algorithm," International Journal of Advanced Research in Science, Communication and Technology, Jul. 2022.

[16] Dipu, Nadim Mahmud, Sifatul Alam Shohan, and K.Salam. "Ocular Disease Detection Using Advanced Neural Network Based Classification Algorithms." Asian J. Converg. Technol 7 (2021): 91-99.

[17] Khan, Md Sajjad Mahmud, Mahiuddin Ahmed, Raseduz Zaman Rasel, and Mohammad Monirujjaman Khan. "Cataract detection usings convolutional neural network with VGG-19 model." In 2021 IEEE World AI IoT Congress (AIIoT), pp. 0209-0212. IEEE, 2021.

[18] Zhang, Linglin, Jianqiang Li, He Han, Bo Liu, Jijiang Yang, and Qing Wang. "Automatic cataract detection and grading using deep convolutional neural network." In 2017 IEEE 14th international conference on networking, sensing and control (ICNSC), pp. 60-65. IEEE, 2017.

[19] Ran, Jing, Kai Niu, Zhiqiang He, Hongyan Zhang, and Hongxin Song. "Cataract detection and grading based on combination of deep convolutional neural network and random forests." In 2018 international conference on network infrastructure and digital content (IC-NIDC), pp. 155-159. IEEE, 2018.

[20] Zhang, Xiao-Qing, Yan Hu, Zun-Jie Xiao, Jian-Sheng Fang, Risa Higashita, and Jiang Liu. "Machine Learning for Cataract Classification/Grading on Ophthalmic Imaging Modalities: A Survey." Machine Intelligence Research 19, no. 3 (2022): 184-208.

[21] Pratap, Turimerla, and Priyanka Kokil. "Deep neural network based robust computer-aided cataract diagnosis system using fundus retinal images." Biomedical Signal Processing and Control 70 (2021): 102985.

[22] Paul Jacob, Aaron, Aryan Bansal, and Ruchika Malhotra. "A Novel Approach for Early Recognition of Cataract using VGG-16 and Custom User-based Region of Interest." In 2022 4th Asia Pacific Information Technology Conference, pp. 15-18. 2022.

[23] Dong, Ke, Chengjie Zhou, Yihan Ruan, and Yuzhi Li. "Mobilenetv2 model for image classification." In 2020 2nd International Conference on Information Technology and Computer Application (ITCA), pp. 476-480. IEEE, 2020.

[24] Zoph, Barret, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. "Learning transferable architectures for scalable image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8697-8710. 2018.

[25] Dondeti, Venkatesulu, Jyostna Devi Bodapati, Shaik Nagur Shareef, and Naralasetti Veeranjaneyulu. "Deep Convolution Features in Non-linear Embedding Space for Fundus Image Classification." Rev. d'Intelligence Artif. 34, no. 3 (2020): 307-313.

[26] Manaswi, Navin Kumar. "Understanding and working with Keras." In Deep Learning with Applications Using Python, pp. 31-43. Apress, Berkeley, CA, 2018.

[27] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 2010.

[28] K. Hu, L. Zhao, S. Feng, S. Zhang, Q. Zhou, X. Gao, and Y. Guo, “Colorectal polyp region extraction using saliency detection network with neutrosophic enhancement,” Computers in Biology and Medicine, vol. 147, p. 105760, 2022.

[29] Y. Dai, J. Wu, Y. Fan, J. Wang, J. Niu, F. Gu, and S. Shen, “MSEVA: A musculoskeletal rehabilitation evaluation system based on EMG Signals,” ACM Transactions on Sensor Networks, vol. 19, no. 1, pp. 1–23, 2022.

[30] A. Qi, D. Zhao, F. Yu, A. A. Heidari, Z. Wu, Z. Cai, F. Alenezi, R. F. Mansour, H. Chen, and M. Chen, “Directional mutation and crossover boosted ant colony optimization with application to covid-19 X-ray image segmentation,” Computers in Biology and Medicine, vol. 148, p. 105810, 2022.

[31] J. Zhou, X. Zhang, and Z. Jiang, “Recognition of imbalanced epileptic EEG signals by a graph-based Extreme Learning Machine,” Wireless Communications and Mobile Computing, vol. 2021, pp. 1–12, 2021.

[32] I. Ahmadianfar, A. A. Heidari, A. H. Gandomi, X. Chu, and H. Chen, “Run beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method,” Expert Systems with Applications, vol. 181, p. 115079, 2021.

[33] C. Llopis-Albert, F. Rubio, and S. Zeng, “Multiobjective Optimization Framework for designing a vehicle suspension system. A comparison of optimization algorithms,” Advances in Engineering Software, vol. 176, p. 103375, 2023.

[34] P. Singh and M. K. Muchahari, “Solving multi-objective optimization problem of convolutional neural network using Fast Forward Quantum Optimization Algorithm: Application in Digital Image Classification,” Advances in Engineering Software, vol. 176, p. 103370, 2023.

[35] Z. Liu, P. Jiang, J. Wang, and L. Zhang, “Ensemble forecasting system for short-term wind speed forecasting based on optimal sub-model selection and multi-objective version of Mayfly Optimization algorithm,” Expert Systems with Applications, vol. 177, p. 114974, 2021.

[36] Sang-To, T., Hoang-Le, M., Khatir, S., Mirjalili, S., Wahab, M. A., & Cuong-Le, T. (2021). Forecasting of excavation problems for high-rise building in Vietnam using planet optimization algorithm. Scientific Reports, 11(1), 23809.

[37] S. Xian, K. Chen, and Y. Cheng, “Improved seagull optimization algorithm of partition and xgboost of prediction for Fuzzy Time Series forecasting of COVID-19 daily confirmed,” Advances in Engineering Software, vol. 173, p. 103212, 2022.

[38] Manaswi, Navin Kumar. "Understanding and working with Keras." In Deep Learning with Applications Using Python, pp. 31-43. Apress, Berkeley, CA, 2018.

[39] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[40] Zoph, B., & Le, Q. V. (2017). Neural architecture search with reinforcement learning. arXiv preprint arXiv:1611.01578.

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

[42] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4510-4520).

[43] Larxel, “Ocular disease recognition,” Kaggle, 24-Sep-2020. [Online]. Available: https://www.kaggle.com/datasets/andrewmvd/ocular-disease-recognition-odir5k. [Accessed: 31-Jan-2023].

[44] G. V. Doddi, “Eye_diseases_classification,” Kaggle, 28-Aug-2022. [Online]. Available: https://www.kaggle.com/datasets/gunavenkatdoddi/eye-diseases-classification. [Accessed: 31-Jan-2023].