Electronic Letters on Computer Vision and Image Analysis 25(2):179-211, 2025

A Lightweight Deep Neural Network for Multi-label Eye Diseases Classification

Nzanzu Katasohire Romeo*, Megha Chhabra+ and Saumya Chaturvedi‡

* *Department of Computer Science & Engineering, School of Engineering & Technology, Sharda University, UP, India*
Email: nzanzukatasohire@gmail.com
+ *Department of Computer Applications, School of Engineering & Technology, SGT University, Gurugram (Haryana)-122505, India*
Email: megha.chhbr@gmail.com
‡ *Professor, CSE (AI & ML), Greater Noida Institute of Technology, Greater Noida, India*
Email: saumyanmishra5@gmail.com

Received 12 June 2025; accepted 19 November 2025

Abstract

Cataracts, Diabetic Retinopathy, and Glaucoma are the leading causes of vision loss worldwide. Traditionally, these diseases are diagnosed manually by inspecting retinal images, a process that is time-consuming and resource intensive. To solve this problem, researchers around the world are working on computer-aided diagnosis systems to assist healthcare professionals. In this study, the NASNetMobile architecture is used to classify four eye conditions: cataract, diabetic retinopathy, glaucoma, and normal. The proposed model was trained on the EDC dataset, containing 4 classes and 4217 images. The architecture integrates global average pooling, Swish-activated dense layers with dropout, and a softmax classifier, optimized using Adam. To ensure a robust evaluation, 5-fold cross-validation was performed, which yielded a mean precision of $98.26\% \pm 0.62$. Comparative ablation with VGG16 and SqueezeNet demonstrates that NASNetMobile achieves the best balance between accuracy and computational complexity. Interpretability was validated through Grad-CAM++ heatmaps, which highlighted clinically relevant regions, while a FLOPs analysis confirmed the framework's efficiency for deployment. These results established NASNetMobile as a practical and accurate backbone for multi-label eye disease classification. Future work will extend robustness analysis to image artifacts and external datasets and explore attention- and transformer-based enhancements for improved clinical reliability.

Key Words: eye diseases, deep learning, transfer learning, multi-label classification.

1 Introduction

An ocular disease is any disorder or condition that hinders the normal functioning of the eye and negatively affects one's visual acuity [1]. Ocular diseases such as diabetic retinopathy, glaucoma, and cataracts are significant public health concerns throughout the world. In fact, they can cause visual impairment or blindness.

Correspondence to: megha.chhbr@gmail.com

Recommended for acceptance by Angel D. Sappa
DOI: 10.5565/rev/elcvia.2233
ELCVIA ISSN:15108-5097
Published by Computer Vision Center / Universitat Autonoma de Barcelona, Barcelona, Spain

Diabetic retinopathy, for example, is the leading cause of blindness among adults [2]. According to the World Report on Vision released by the World Health Organization, at least 1 billion people suffer from preventable or yet-to-be-addressed vision impairment. Among these 1 billion people, the main diseases that cause vision impairment are cataracts, glaucoma, diabetic retinopathy, age-related macular degeneration, refractive errors, and trachoma [2]. Early detection and treatment of these diseases is crucial in preventing their devastating effects and ensuring a better quality of life for patients. In fact, early intervention can slow or even stop the progression of diseases such as diabetic retinopathy and glaucoma, which can lead to irreversible blindness if not properly treated [3].

Visual impairment has been found to have a high impact on the economy and social development [4]. In 2020, the cost of vision impairment globally was estimated to be between US$160 billion and US$216.32 billion [5]. In addition, approximately 90% of the people affected by visual impairment live in low and middle-income countries with a very low patient-to-doctor ratio [6].

Eye diseases are traditionally detected through the manual examination of medical images [7] by ophthalmologists. The diagnosis process can be not only time-consuming but also prone to subjectivity and human error [8]. In the ophthalmic context, medical imaging encompasses a variety of tools and techniques that provide visual representations of the internal structures of the eye [9]. It is useful for non-invasive diagnosis of eye diseases. [10] Such techniques include fundus photography and optical coherence tomography. A fundus image is a 2D projection of the interior surface of the eye, also known as the fundus. It captures a range of important biomarkers such as the optic cup (OC), optic disk (OD), blood vessels, macula, fovea, and different types of lesions including soft and hard exudates, hemorrhages, and micro-aneurysms. Fundus images can be used to diagnose a variety of diseases, including glaucoma, diabetic retinopathy, cataracts, and age-related macular degeneration [11]. On the other hand, an optical coherence tomography (OCT) scan is a non-invasive imaging test that uses light waves to take cross-section pictures of the retina. It allows ophthalmologists to examine the distinctive layers of the retina and is based on the principles of interferometry and confocal microscopy. OCT scans are efficient in the diagnosis of age-related macular degeneration and diabetic eye diseases [12].

The equipment used for medical imaging generates large amounts of data in hospitals and eye clinics. This data can be leveraged to create Artificial Intelligence powered solutions that will allow ophthalmologists to review patient retinal images promptly [13]. Additionally, Artificial Intelligence integration can allow doctors to remotely diagnose and monitor their patients' conditions, reducing the patients' need for frequent medical visits [14] and solving the low doctor-to-patient ratio problem faced by many countries [15]. Artificial Intelligence can be used to detect and assess the severity of eye diseases. This can help to categorize patients into risk groups based on the severity of their condition [16] and assist healthcare professionals in allocating the right resources to provide appropriate care for patients at a higher risk [17]. During the past few years, deep learning techniques have been used to classify eye diseases using retinal fundus images and optical coherence tomography (OCT) scans [18]. Compared to traditional diagnosis methods, deep learning can achieve better performance by automatically extracting the relevant features from the images. To this date, most studies on Deep Learning applications on retinal images can be divided into classification and segmentation tasks [19–23]. These studies mostly classify between the absence and presence or focus on the grading of a single disease. Few of them have focused on the classification of multiple diseases [24].

This paper presents a novel approach for multi-label eye disease classification. The aim is to create and train a lightweight deep-learning model capable of detecting the following eye diseases in fundus images: glaucoma, diabetic retinopathy, and cataracts. The proposed model is based on NASNetMobile, a lightweight mobile-friendly convolutional neural network (CNN) architecture based on the Neural Architecture Search (NAS) Network. Below are the point-wise contributions of this study:

- Multi-labelled classification of Diabetic Retinopathy, Glaucoma, Cataract, and Healthy Eye using NASNetMobile feature extractor.
- Efficient lightweight model with reduced resource utilization
- Comparison of the proposed model against transfer-learning-based models.

The rest of the paper is organized as follows: After the introduction, section 2 presents some related recent works on the topic of this study. In section 3, the materials and methods are discussed. Section 4 is reserved for the results and their analysis while section 5 and section 6 respectively discuss and conclude the work.

2 Related Work

Recent developments in artificial intelligence and deep learning have brought significant advancements to the field of ophthalmology [25]. This section reviews some studies focusing on the applications of deep learning for eye disease recognition, highlighting the methodologies and the achieved results. A summary of the review can be seen in Table 1.

[26] implemented a deep-learning model capable of detecting Diabetic Retinopathy in low-resolution retinal images. This study highlighted the efficacy of convolutional neural networks even on low-quality images with their model performing with an accuracy of up to 80.93%. [27] used orbital computed tomography (CT) scans to train a thyroid eye disease detection model. The model, based on the VGG16 feature extractor, achieved 89.5% accuracy when trained on data from 141 patients. In contrast, oculoplastic surgeons could only achieve a 70.0% classification accuracy on the same dataset. An 18-layer deep neural network architecture is proposed in [28]. Based on the SqueezeNet feature extractor, the model was used to detect Glaucoma from retinal images and achieved a high accuracy of 97.7%. Age-related macular degeneration was tackled in [29] where transfer learning was used to classify between healthy and unhealthy eyes. The proposed model works in two phases. In the first stage, the image is classified as healthy or unhealthy. In case the image is unhealthy, the severity of the disease is assessed at the second stage. The study experimented with seven pre-trained models for each stage, namely VGG19, Xception, DenseNet201, EfficientNetB7, InceptionV3, NASNetLarge, and ResNet152V2. The DenseNet201 feature extractor outperformed the other models in terms of accuracy. On the same note, [30] introduced a new framework for converting color fundus images into synthetic fundus autofluorescence (FAF) images. This concept involves using the natural fluorescence of the back of the eye to provide insight into the health of the retinal pigment epithelium. The synthetic images were generated using Generative Adversarial Networks (GANs) trained on pairs of color fundus images and their corresponding FAF images. The model's efficiency was assessed in a grading task for Age-Related Macular Degeneration on the LabelMe dataset.

Researchers in [31] pointed up the deep-rooted limitation of traditional diagnosis procedures due to limited medical resources. Their study brought to life the OIA-ODIR dataset and hinted at the positive impact of feature fusion for multi-disease classification by bench-marking nine prominent convolutional neural network (CNN) architectures. Continuing on the same track, the study [32] proposed a hybrid deep-learning network to detect and grade the severity of diabetic retinopathy. They were able to achieve up to 96% accuracy by using the ResNet50 as their feature extractor on the IDRiD dataset. Multi-label disease classification was tackled by this [33] study which proposed a model for timely detection of Diabetic retinopathy, Diabetic Macular Edema, and Glaucoma. They built a custom CNN model that was able to achieve 81.33% of accuracy. [34] also addressed multiple eye diseases, including Macular Degeneration, Myopia, Glaucoma, and Retinitis Pigmentosa. Their study used the MobileNetV2 model as a feature extractor and included gradient-weighted Class Activation Mapping for visualization. The EyeDeep-Net was introduced in [35]. It is a multi-label eye disease classification model trained on the RFMiD dataset. In addition to a healthy eye, the model was trained to identify the following diseases: Diabetic Retinopathy, Media Haze, and Optic Disc Cupping. The results of the experiments showed validation accuracy of up to 82.13%. The work in [36] took a different approach to multiple eye disease classification by using the flower pollination optimization algorithm to build an optimized deep convolutional neural network. A summary of the reviewed related work can be found in Table 1.

Deep learning has showcased significant potential in the field of retinal image analysis. By leveraging CNNs, models can be trained to detect and classify a variety of eye conditions. These models are capable of learning complex patterns in retinal images that may sometimes be imperceptible to human clinicians.

2.1 Convolutional Neural Networks (CNNs)

CNNs are a type of deep learning network that has greatly contributed to the ever-growing adoption of deep learning in computer vision applications. They have one or many convolutional layers which consist of filters that automatically extract features from images. Basic features of CNN include [37]:

- *Kernels*: Also known as filters, kernels are small matrices that are convolved with input data to extract features such as edges, shapes, textures, or more complex patterns. Kernels contain learnable parameters and their shape can vary depending on the network's architecture and on the features to be extracted.
- *Convolution operation*: In mathematics, a convolution is an operation that combines two functions to produce a third function. The resulting third function expresses the combined effect of the two previous functions, highlighting areas where their features overlap. Let and g(x) be two functions. The convolution of and is defined as:

$$(g * f)(t) = \int_{-\infty}^{\infty} g(t - x) * f(x)\, dx \tag{1}$$

  In deep learning, the convolution operation involves the application of kernels to the input data to produce feature maps. The mathematical expressions of convolution in the context of deep learning are shown in Equation 2 and Equation 3.

  For a 1D input signal x and a filter w, the convolution operation at position t is:

$$y(t) = (x * w)(t) = \sum_{i=0}^{k-1} x(t + i) \cdot w(i) \tag{2}$$

  Where:

  - $y(t)$ is the output at position t,
  - $x(t)$ is the input signal at position t,
  - $w(i)$ is the filter (kernel) value at position i,
  - k is the size of the filter.

  In the case of 2D input data (such as images), the convolution operation can be expressed as:

$$y(i, j) = (x * w)(i, j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(i + m, j + n) \cdot w(m, n) \tag{3}$$

  Where:

  - $y(i, j)$ is the output at position (i, j),
  - $x(i + m, j + n)$ is the input value at position $(i + m, j + n)$,
  - $w(m, n)$ is the filter value at position (m, n),
  - $M \times N$ is the size of the filter.

- *Activation function*: Activation functions introduce non-linearity to the deep learning model. They scale the weighted sum in the neuron down to a specific range and replicate the firing of a neuron. Some common activation functions include the ReLU (Rectified Linear Unit), Tanh (hyperbolic tangent), SoftMax, and Sigmoid activation functions.
- *Pooling operation*: This operation is used to reduce the spatial dimensions of the feature map generated by the convolutional layer. Besides decreasing the amount of data to process, the pooling layer captures the most important information from the feature map.

Today, CNNs are ubiquitous with applications ranging from facial recognition in security systems to precision agriculture. They are also extensively used for medical image analysis. In fact, Lo et al.'s work on lung nodule detection in chest radiographs [38] can be regarded as one of the earliest successful attempts to use Convolutional Neural Networks for medical image analysis.

Table 1: Comparative summary of related work

Ref	Contribution	Results	Gap analysis
[26]	Eye disease classification using low resolution images	80.93% Acc	Detection of only one disease (Diabetic Retinopathy)
[27]	VGG16 model for Thyroid eye disease detection	89.5% Acc	Detection of only one disease (Thyroid eye disease)
[28]	SqueezeNet model for detection of glaucoma from retinal images	97.7% Acc	Detection of only one disease (Glaucoma)
[29]	9 Pre-trained deep learning model for the detection and grading of Age-related Macular Degeneration	96% Acc	Detection and grading of only one disease (Age-related Macular Degeneration)
[30]	Generation of synthetic Autofluorescence images	88.5% Acc	Detection of only one disease (Age-related Macular Degeneration)
[31]	9 CNN models for multiple diseases classification using the ODIR dataset	90% Acc	Imbalanced dataset
[32]	Detection and Grading of Diabetic Retinopathy using ResNet50	96% Acc	Detection of only one disease (Diabetic Retinopathy)
[33]	CNN models for multiple diseases classification using multiple datasets (Messidor, Messidor2, DRISTHTI-GS)	81.33% Acc	Small datasets
[34]	MobileNetV2 for multiple diseases classification	96.2% Acc	Limited Dataset
[35]	Deep Neural network for multiple diseases classification	82.13% Acc	Limited and Imbalanced dataset
[36]	Optimized Deep Neural Network with whale flower pollination optimization algorithm	Up to 95.27% Acc	Imbalanced dataset

2.2 Research gaps

Several works have been published in the field of retinal image analysis using deep learning. However, from the review of the recent studies, the following limitations can be pointed out:

- Limited research on multi-label eye disease classification:Most of the existing models often focus on the detection of a single disease, overlooking the possibility of co-existing ophthalmic disorders in patients. By proposing a model capable of detecting multiple diseases simultaneously, this study addresses this gap.

- Lack of lightweight models: Many of the state-of-the-art architectures are not suitable for deployment in resource-constrained environments due to their complex and resource-intensive architectures. This gap is bridged by leveraging NASNetMobile, an efficient and lightweight architecture.

- Although some models achieve acceptable levels of accuracy, they often fail to improve other critical metrics such as precision, recall, and Area Under the Curve(AUC). This work addresses this gap by achieving superior performance across the cited metrics, thereby offering a more holistic evaluation.

2.3 Trends

Beyong CNN-based feature extractors, recent studies have explored alternative and hybrid approaches for improving robustness and interpretability in image recogntion. Handcrafted descriptors such as fractional-order Legendre moments [39], jacobi moments [40], and optimized quaternion Meixner moments [41] have demonstrated strong invariance to scale, rotation, and noise across diverse applications, including 3D image recognition and automatic lip reading. More recent advances, such as multichannel orthogonal radial triangular pulse moments [42], extend these descriptors to biometric recognition, highlighting their adaptability to different domains. Parallel trends in interpretable AI include attention-based NN modules (e.g., CBM, SENet), explainable multimodel pipelines integrating clinical and imaging data, and even quantum-enhanced classifiers [43], all of which emphasize transparency and efficiency. These developments collectively suggest that hybrid pipelines combining deep feature extractors with mathematically grounded descriptors or attention mechanisms may further improve both robustness and interpretability in retinal disease classification, and thus represent promising directions for future research.

3 Materials and Methods

This study aims to build a lightweight deep-learning model for detecting and classifying eye diseases in fundus images. This section provides a detailed layout of the implemented workflow. The proposed work is based on the use of transfer learning to build a convolutional neural network (CNN). The overall proposed workflow is shown in Figure 1.

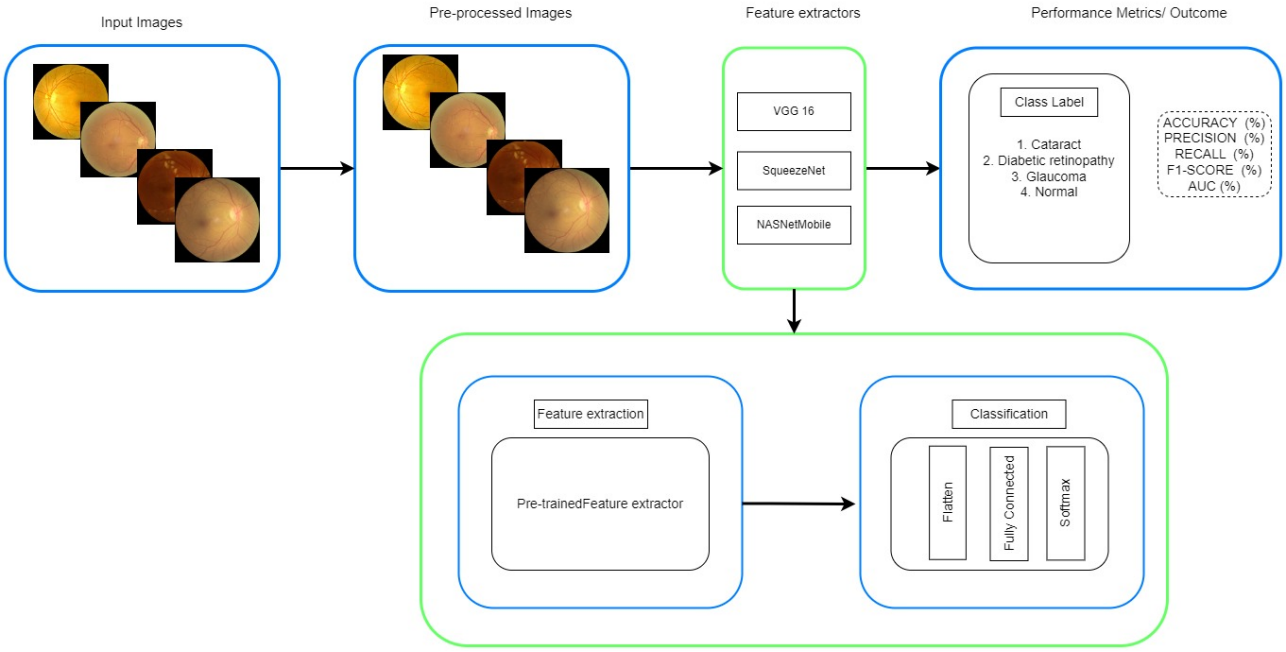


Figure 1: Implementation flow of proposed work

3.1 Roadmap

- Data description: The EDC dataset contains four classes of fundus images: Diabetic retinopathy, glaucoma, cataract, and normal, with a total of 4217. The images have a fixed resolution of 512 x 512.
- Data preprocessing and augmentation: at this stage, the images are resized to a smaller resolution of 224 x 224. Basic color space and geometric transformations are also applied during data augmentation
- Transfer learning and feature extraction: The NASNetMobile architecture is chosen as the feature extractor for this study.
- Model description: details about the proposed model's architecture are presented.
- Performance metrics: the model is evaluated in terms of accuracy, precision, recall, area under the curve, and f1-score.

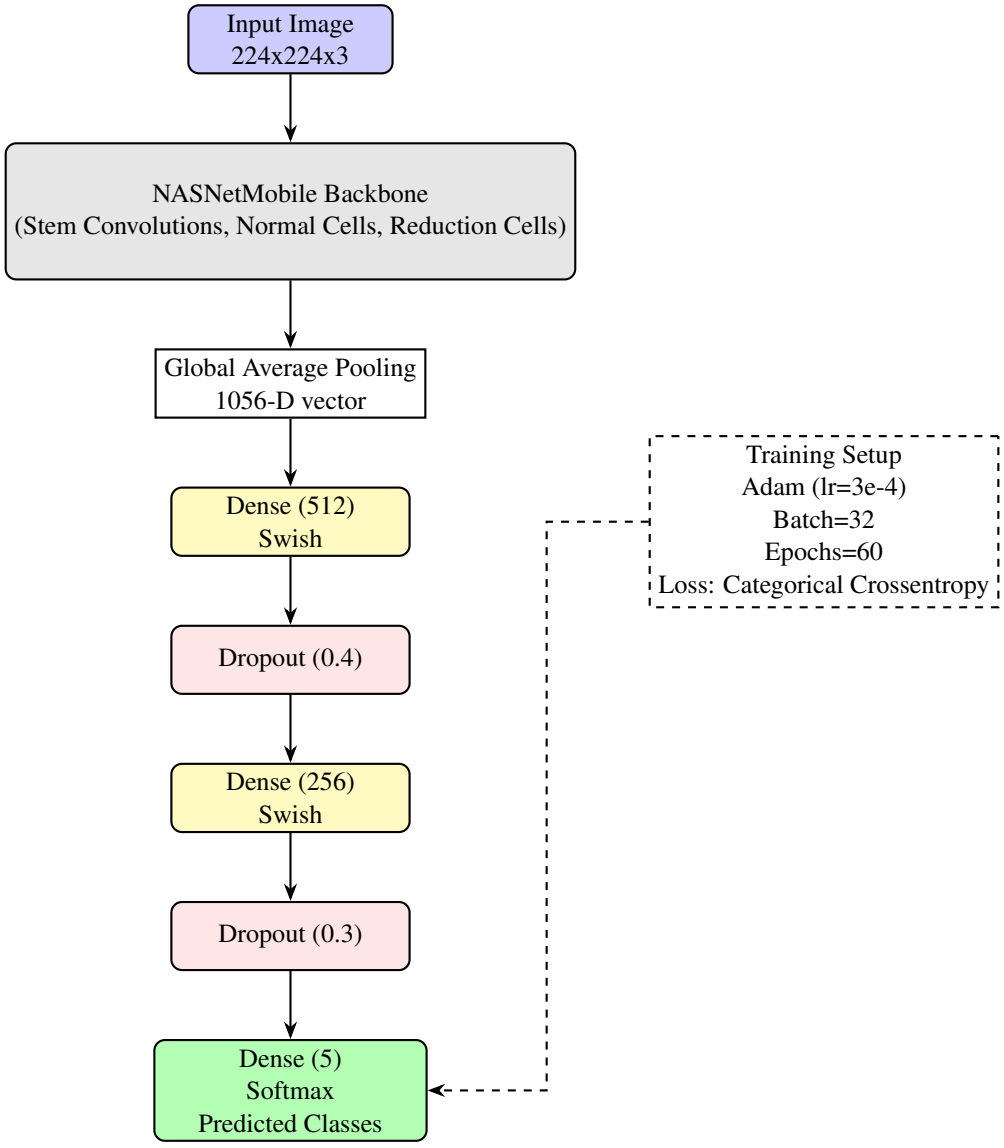


Figure 2: Proposed NASNetMobile-based lightweight deep neural network for multi-label eye disease classification.

3.2 Data Description

The framework for the development of any deep learning model begins with the collection of data [44].

Over the years, several datasets that contain retinal images have been made publicly available online. The model presented in this study was trained using the Eye Disease Classification (EDC) dataset [45]. It is publicly available on Kaggle and contains 4217 retinal images collected from various sources and features four classes based on the kind of eye diseases: diabetic retinopathy, glaucoma, cataract, and normal. All the images have a resolution of 512 X 512. In the dataset, the distribution of images per class is relatively uniform. The image count per class does not have a significant difference, hence class balancing is not applied. The data set was divided into training, testing, and validation sets, with the training set accounting for 80% of the data. The remaining 20% were equally distributed between the validation and the testing set. A description of the EDC dataset can be found in Table 2

Table 2: Dataset Description

Classes	**Image Resolution**	**Image Count**
cataract	512 X 512	1038
diabetic retinopathy	512 X 512	1098
glaucoma	512 X 512	1007
normal	512 X 512	1074
Total	**4217**	

Table 3: Distribution of image counts across different classes and dataset splits.

Class	**Split**	**Image Count**
cataract	Training	830
	Validation	103
	Testing	105
diabetic retinopathy	Training	878
	Validation	109
	Testing	111
glaucoma	Training	805
	Validation	100
	Testing	102
normal	Training	859
	Validation	107
	Testing	108

3.3 Data Preprocessing and Data Augmentation

The first stages of the preprocessing phase were image normalization and image resizing. In image analysis and deep learning, image normalization is a step that adjusts the pixel values of an image to fit in a certain range [46]. This results in improved performance and stability during the training process. In other words, image normalization is a standardization technique that improves the efficiency of the learning process [47]. In this study, the Min-Max Scaling approach was followed, transforming the pixel values to a range of [0,1]. The formulas for Min-Max scaling are shown in Equation 4 and Equation 5.

For scaling to the range [0, 1]:

$$X' = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{4}$$

For scaling to a custom range [a, b]:

$$X' = a + \frac{(X - X_{\text{min}}) \times (b - a)}{X_{\text{max}} - X_{\text{min}}} \tag{5}$$

Where:

- X is the original value.
- X' is the scaled value.
- X_{min} is the minimum value in the original dataset.
- X_{max} is the maximum value in the original dataset.

Following up, all the images were resized to a standard size of 224 X 224. This specific size was chosen to fit the required input size of the feature extractors that were later used in the study. Furthermore, resizing the input can have a great impact on training time and memory usage [48].

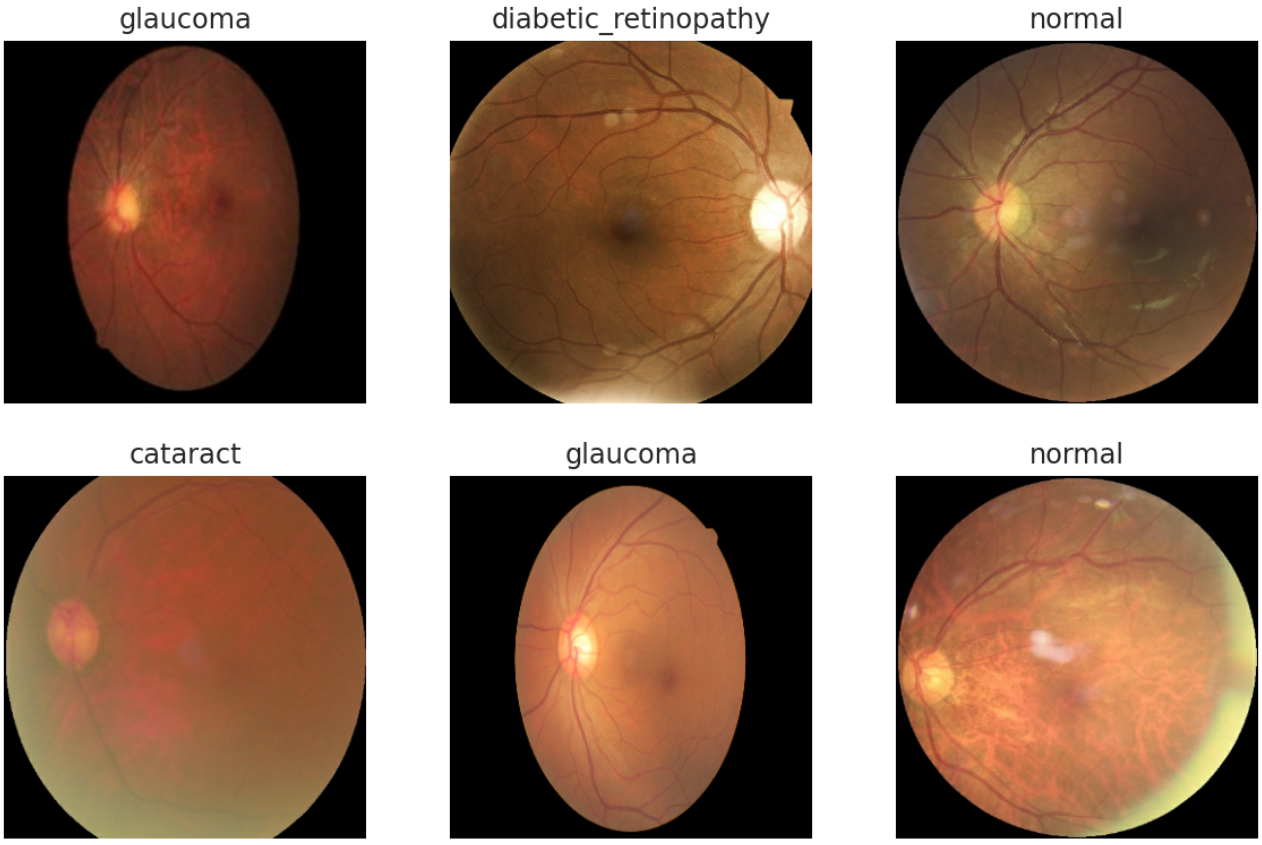


Figure 3: Dataset sample

Subsequently, the input images went through the data augmentation phase. Data augmentation is a technique used to increase the diversity of the training data without collecting new data [49]. The aim is to enhance the generalization capabilities of the model over unseen data [50]. Data augmentation can help prevent overfitting and is beneficial when dealing with a very small dataset [51]. Many transformation techniques can be applied to augment the dataset. In this study, the focus was made on simple transformations, avoiding complex transformations that could misrepresent critical features from fundus images and lead to improper classification. Typically, brightness, hue, and saturation changes were used along with horizontal flipping and rotation. The data augmentation strategy is summarized in Table 4.

Table 4: Summary of data augmentation techniques and their parameters.

Augmentation Technique	Parameters	Values
Color Space Transformation	Brightness	$\pm 20\%$
	Saturation	$\pm 20\%$
	Hue	$\pm 20\%$
Geometric Transformation	Rotation	$\pm 10^{\circ}$
	Flipping	Horizontal

3.4 Transfer learning and feature extraction

Transfer learning is a technique used in machine learning whereby a model trained to solve a particular task is reused as a starting point to solve a different task [52]. It takes the knowledge gained from solving one problem and transfers it to a new related problem [53]. This approach is particularly useful when the task at hand features a limited dataset. Transfer learning offers advantages such as:

- *Reduced training time*: By using a pre-trained model, transfer learning requires less training time since it uses the weights and biases from the pre-trained model as a starting point [54]

- *Reduced data requirements*: Transfer learning methods can exceed baseline models using only a fraction of labeled dataset as compared to models trained from scratch [55]. This can help address the problem of data scarcity, especially in the medical field.

- *Better performance*: Jo Asanuma et al. demonstrated in their study on Topology Optimization of Electric Motor [56] that transfer learning reduced the computational costs by 13% to 15% as compared to models trained from scratch.

When it comes to Deep Learning, feature extraction is the process of automatically identifying and extracting relevant patterns from raw data [57]. It is a critical task, especially when dealing with high dimensional and complex data [58]. Pre-trained deep-learning models can also be used as feature extractors. This is particularly relevant in the field of image processing, where CNNs are used to obtain feature maps from new unseen images [59]. These feature maps capture special and hierarchical patterns.

The NASNetMobile model is used as a feature extractor for this study. It was selected due to its lightweight design and the level of accuracy it offered. The NASNet architecture was developed by researchers from Google Brain using a reinforcement learning approach [60]. This approach involved using a parent Recurrent Neural Network (RNN) called 'The Controller' that samples child networks and learns their architecture directly from the dataset of interest. Figure 4 shows the operational blocks used by the controller to build the child network. These blocks enable the Controller (the parent RNN) to construct the NASNet architecture by performing architectural adjustments on the child network. It is subsequently trained on two sets of images containing images with different sizes to generate two models: NASNetLarge, which has 88,949,818 parameters and NASNetMobile with 5,326,716 parameters [61].

3.5 Model description

The NASNetMobile was used with the imagenet weights. The lightweight nature of the model ensures its ability to deliver accurate results even with reduced computational resources. To customize the model for multiple eye diseases classification, a flatten layer was added on top of the pre-trained model followed by two fully connected (FC) layers.

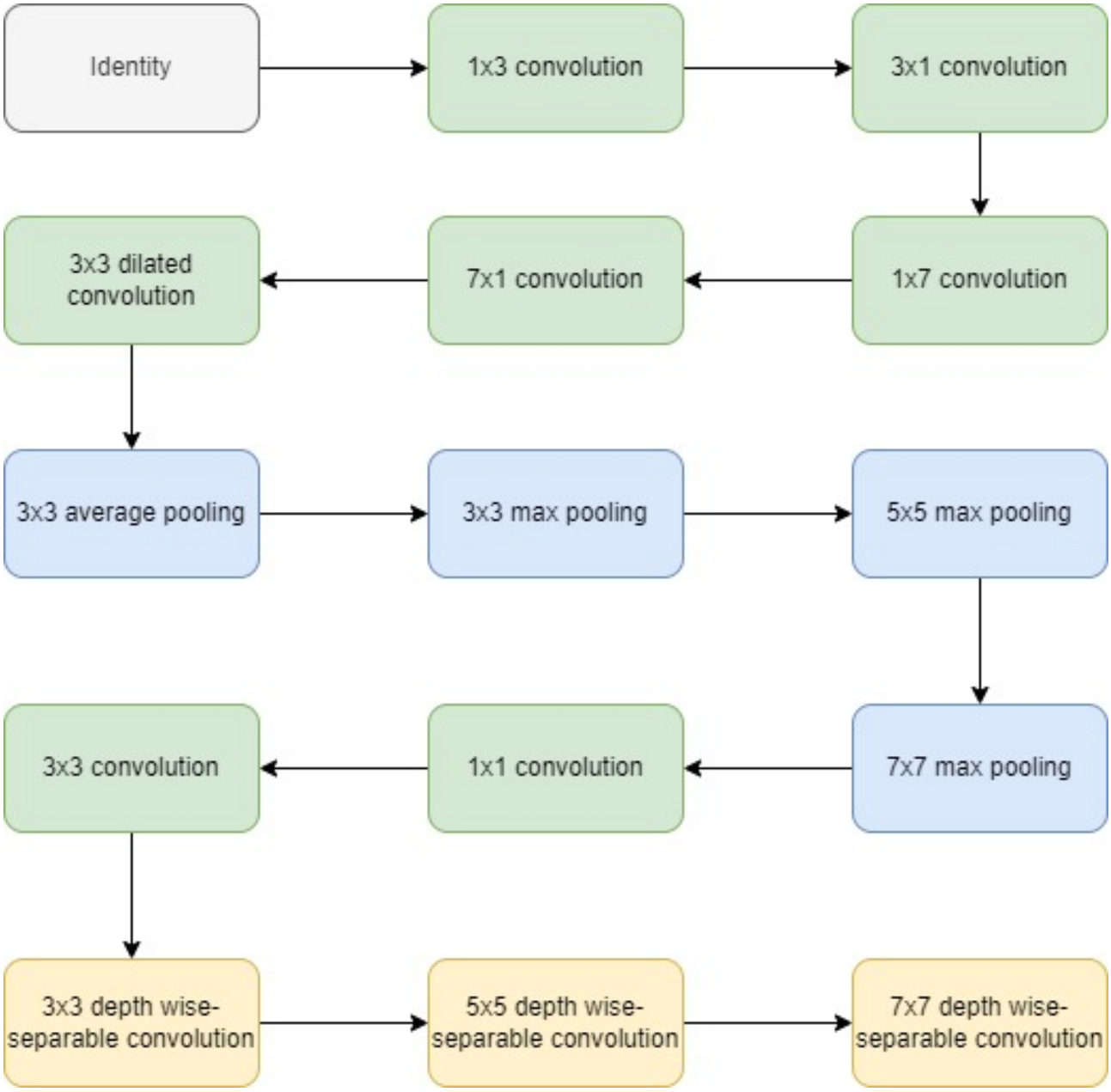


Figure 4: NASNet Controller's operational blocks

3.5.1 Purpose to use NASNetMobile

1. Lightweight Design: NASNetMobile is optimized for efficiency. It comprises about 5.3 million parameters, while its larger counterpart, NASNetLarge, has nearly 88.9 million [52]. This reduced number of parameters makes the NASNetMobile model suitable for devices with constrained computational resources such as mobile and embedded systems.

2. Cell-Based Structure: The network is composed of two types of cells:

   - Normal cells: learn features from input images while preserving their spatial dimension.
   - o Reduction cells: focus on deeper features by reducing spatial dimensions.

3. High Accuracy: The NASNetMobile architecture maintains a high level of accuracy in image recognition and classification tasks despite its compact size. It offers a good balance between efficiency and performance, ensuring reliable results without the necessity of extensive computational power.

4. Scalability: The NASNet architecture was inherently designed to be scalable, allowing the depth and widths of the network to be adjusted to fit different applications requirements

The models comprise the following key components:

- Input Layer
- Feature Extractor
- Flatten Layer
- Fully Connected (FC) Layers
- Output Layer with SoftMax Activation

3.5.2 Implementation of pre-trained feature extractors

The implementation of the chosen feature extractors in this study involved several key steps:

- Initialization with Pre-trained Weights: The chosen pre-trained models were imported from the Keras applications module with pre-trained weights on the ImageNet dataset. This initialization provides a robust starting point for the models by capitalizing upon the feature extraction capabilities they have learned over a sizeable and diverse set of images.
- Removal of Top Layers: The top layer, that is, the fully connected layer of the pre-trained architecture was removed to fine-tune the pre-trained models for the task at hand. This allows the proper tailoring of the output according to the classes of interest, namely Diabetic Retinopathy, Glaucoma, Cataract, and Normal.
- Freezing the Base Model: Freezing the pre-trained weights of the pre-trained models ensures that they are not updated during the training phase of the task at hand. Doing so would help the networks maintain the generalized feature extraction abilities learned on ImageNet and hence be applicable in most image classification tasks.
- Integration with Custom Layers: On top of the frozen convolutional base, custom layers are added that are tailored for the classification task. These dense layers are further discussed in the Model Description section below.
- Fine-Tuning: Fine-tuning was done after training the custom top layers by unfreezing a few of last layers of pre-trained base models. This allows the models to further adjust the pre-trained weights so that it can better capture the specific features relevant for eye disease classification, resulting in better performance of the model on the target task.

3.5.3 NASNetMobile as Feature Extractor

This section presents the detailed architecture of the model based on the NASNetMobile feature extractor. Table 5 presents a summary of the model's configuration, outlining each layer, its type, output shape, and the number of parameters involved.

- Input Layer

1. Shape: the images are resized to a shape of 224 x 224 x 3. This corresponds to the height, width and color channels of the fundus images. This particular shape was implemented to comply with the input shape requirements of the NASNetMobile architecture.
2. Purpose: this layer serves as the entry point for data.

- Feature Extractor (NASNetMobile)
  1. Architecture: The NASNetMobile mode has a depth of 88 layers, made of normal cell and reduction cells.
  2. Output Shape: The feature extractor outputs a tensor of shape $7 \times 7 \times 1056$, representing the high-level features extracted from the input images.
  3. Parameters Count: The feature extractor accounts for 4,269,716 non-trainable parameters. These parameters are frozen during the initial training phase to use the weights learned from the ImageNet dataset.
- Flatten Layer
  1. Purpose: This layer is used to transform the high dimensional data from the feature extractor into a one-dimensional vector. The resulting vector is used as a base to the subsequent fully connected layers
  2. Output Shape: The layer has an output shape of (None, 51744).
  3. Parameters Count: No additional parameters are introduced by this layer.
- Fully Connected (FC) Layers
  1. First FC Layer:
     (a) Configuration: A dense layer with 64 neurons and the ReLU (Rectified Linear Unit) activation function.
     (b) Purpose: This layer is added to introduce non-linearity to the model. The choice of 64 neurons offers a balance between complexity and computational efficiency.
     (c) Parameters Count: This layer accounts for 3,311,680 parameters.
  2. Second FC Layer (Output Layer):
     (a) Configuration: A dense layer with 4 neurons corresponding to the four classes: Cataract, Glaucoma, Diabetic Retinopathy, and Normal. This layer employs the SoftMax activation function.
     (b) Purpose: The SoftMax layer converts the raw logits into a probability distribution over the four classes, facilitating the final classification decision.
     (c) Parameters Count: This last layer accounts for 260 parameters.
- SoftMax Activation Function The SoftMax activation function is applied in the output layer, so the model predicts with probabilities.

Figure 5 offers an illustration of the architecture of the proposed model.

3.5.4 Training process

1. Epochs: In order to allow sufficient iterations for the network to learn patterns in retinal images, the model was trained for 60 epochs. Early stopping was also implemented to watch the validation loss and stop training if there was no improvement for 5 consecutive epochs. This mechanism helps prevent overfitting.

Table 5: Summary of the model's configuration.

Layer	Type	Output Shape	Parameters Count
0	Input	(224, 224, 3)	-
1	Feature Extractor (NASNetMobile)	(7, 7, 1056)	4,269,716
2	Flatten	(None, 51744)	-
3	Fully Connected	(None, 64)	3,311,680
4	Fully Connected	(None, 4)	260
Total parameters: 7,581,656 **Trainable parameters: 3,311,940** **Non-trainable parameters: 4,269,716**			

2. Batch Size: A batch size of 32 was selected in an effort to offer a balance between memory usage and training efficiency.

3. Loss Function: The Categorical Cross Entropy loss function was utilized. This function is commonly used when a classification task involves more than two classes. It works by measuring the difference between the probability distributions of the actual labels and the predicted distribution. This loss function is defined as:

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}}, \quad L = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$$

Where:
C is the number of classes
y_i is the true label class i
$\hat{y}_i$ is the predicted probability for class i
Log is the natural logarithm (base e)

4. Optimizer: The Adam (Adaptive Moment Estimation) optimizer was used to trained the model. This optimizer is renowned for its efficiency and ability to handle sparse gradient. The Adam optimizer was chosen because of its adaptive learning capabilities allowing for faster convergence.

5. Classifier: The SoftMax activation function [62] is applied in the output layer, so the model predicts with probabilities. Normalizing the output logits guarantees that the sum of the predicted probabilities over all classes equals one, which means that the classification result is clear and interpretable. The formula of the SoftMax activation function is shown in Equation below:

Where:
zi is the ith element of the input vector z
e is the base of the natural logarithm
$\sum_{j=1}^{n} e^{Z_j}$ is the sum of the exponentials of all the elements in the vector.

The hyperparameter setup can be found in Table 6.

Table 6: Summary of training parameters and their values.

Parameter	Value
Optimizer	Adam
Epochs	10,20,30,40,50,60
Batch size	32
Loss	Categorical cross entropy
Classifier	SoftMax

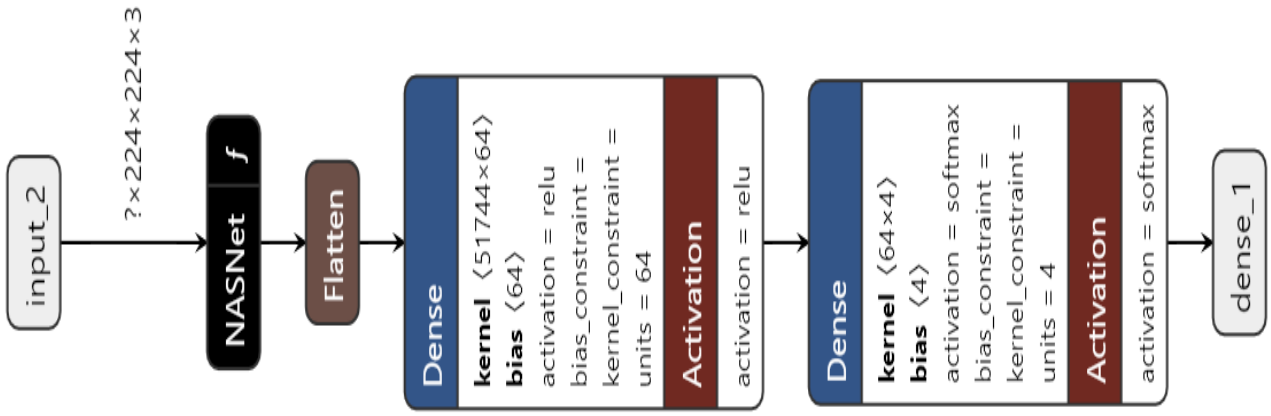


Figure 5: Proposed model architecture

3.6 Evaluation metrics

In addition to accuracy, four other metrics are used to measure the effectiveness of the proposed model: precision, recall, F1 score, and Area Under the Curve (AUC) [63]. The following lines provide a mathematical representation of these metrics:

- **Precision**, also called Positive Predictive Value, is the measure of the rate of true predictive predictions over all positive predictions made by a model.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{6}$$

- **Recall**, also called Sensitivity, measures the rate of true positive predictions over all actual positives in the data sample.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{7}$$

- **F1-score**, also called F-Measure (FM), is the harmonic mean between recall and precision values.

$$\text{F1-score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{8}$$

- **AUC**is the area under the Receiver Operating Characteristic (ROC) curve, which plots the Recall (true positive rate) against the false positive rate (FPR) at various threshold settings.

$$\text{AUC} = \int_{x=0}^{1} \text{Recall}\left(\text{FPR}^{-1}(x)\right) dx \tag{9}$$

$$\text{FPR} = \frac{\text{False Positives}}{\text{False Positives} + \text{True Negatives}} \tag{10}$$

Additionally, a confusion matrix was also used during the evaluation process. It is a table that provides the summary of the model's classification performance. It does so by comparing the actual target classes with the model's predicted values.

4 Experiments and Results

The proposed workflow as shown in Figure 1 paved the way for the experimentation. This section offers a detailed description of the steps followed during the experimentation phase.

4.1 Roadmap

- Experimental setup: details about computational setup are given
- Choice of feature extractor: the process of selecting the feature extractor is described
- Impact of the number of epochs: tuning of the number of epochs is performed.
- Impact of regularizer: tuning of regularizer is performed

4.2 Experimental setup

Google Colab was used to conduct the experiments in this study. It is a cloud-based platform provided by Google that hosts an infrastructure for training deep learning models. This platform offers several advantages, including a Jupyter Notebook environment, support for collaboration, and high-performance hardware. The computations were performed using the TPU v4 (Tensor Processing Unit) hardware accelerator. The summary of the experimental setup is as follows:

- **Platform**: Google Colab
- **Hardware Accelerator**: TPU v4
- **Software Framework**: Tensorflow 2.17.0
- **Dataset**: EDC dataset

4.3 Choice of feature extractor

To determine the most suitable feature extractor for the proposed model, an extensive evaluation of different pre-trained architectures was conducted. Three different architectures were tested: SqueezeNet, NASNetMobile, and VGG16. All the models were trained for 60 epochs on the EDC dataset. Table 7 shows a comparative summary of these feature extractors and Figure 6 shows a bar plot of the performance of each feature extractor. The results showed that the NASNetMobile architecture provided better performance. It was subsequently chosen as the main feature extractor for the proposed model.

Table 7: Comparative summary of feature extractors

Evaluation Metric	SqueezeNet	VGG16	NASNetMobile
Accuracy	83.26%	95.27%	**98.95%**
Precision	86.24%	90.69%	**98.99%**
Recall	84.76%	89.25%	**98.98%**
F1 Score	84.55%	88.71%	**98.98%**

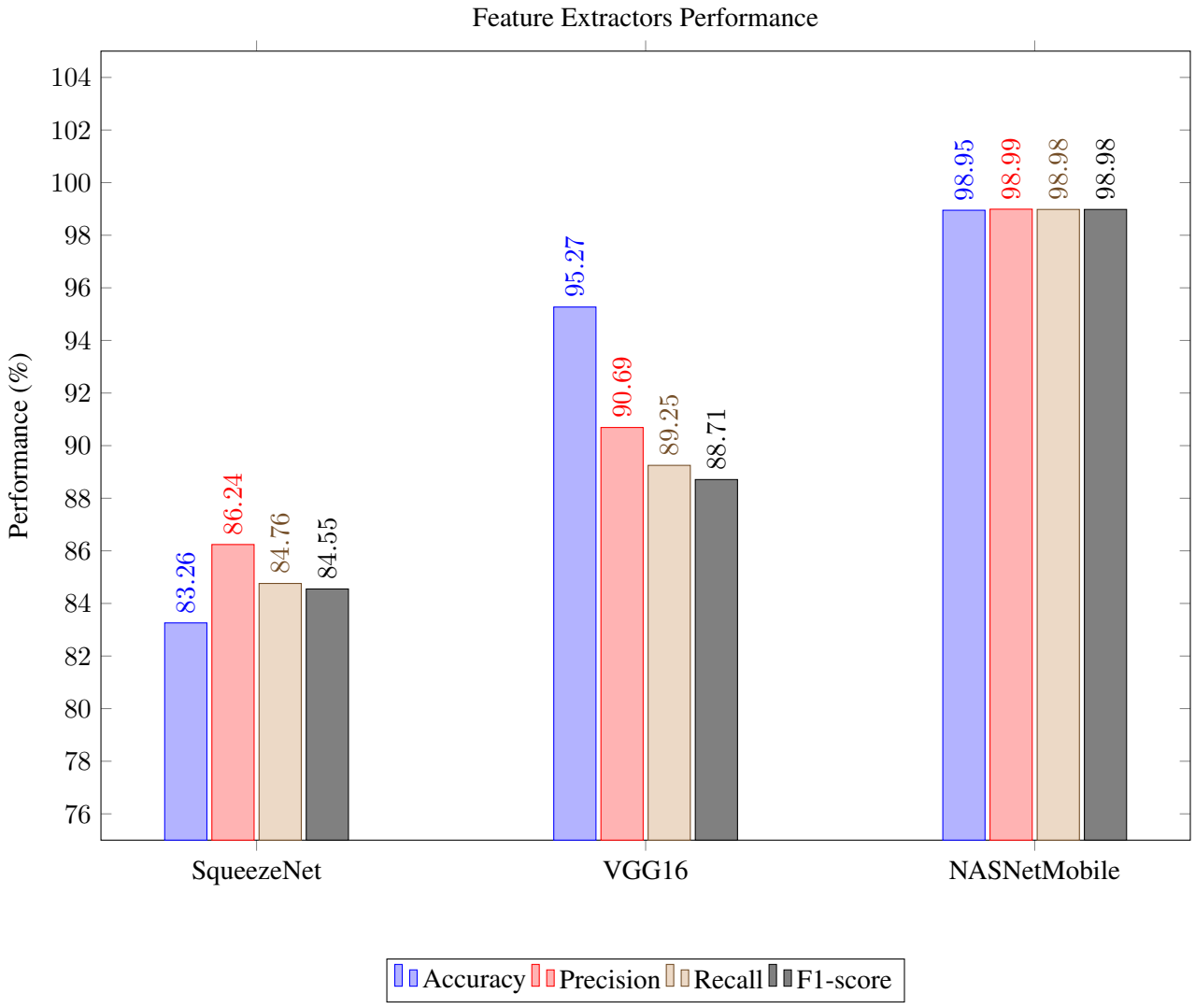


Figure 6: Bar plot of feature extractors performance

4.4 Impact of number of epochs

The subsequent steps involved experimenting with the impact of the number of epochs on the performance of the model. Although the accuracy of the model increased with the number of epochs, the gap between the training and the accuracy of the validation also increased, suggesting overfitting. Table 8 offers a summary of these observations.

4.5 Impact of Regularizer

Additional experimentation was done to determine the influence of a kernel regularizer on the model's performance. An L2 regularizer was added with a regularization factor of 0.015. Table 9 shows the parameters used for this experiment.

Although the regularizer improved the loss function of the model, it significantly reduced its accuracy. In fact, the training accuracy went from 98.95% without a regularizer down to 93.68% with an L2 regularizer. These observations are illustrated in Figure 8. The regularizer also reduced the gap between training and validation accuracy from 8.42% to 5.76%. When evaluated on the testing data split, the model without regularization outperformed the regularized model in terms of Precision, Recall, F1-score, and AUC as demonstrated in Table

Table 8: Influence of number of epochs on model performance

Epochs	Accuracy	Precision	Recall	F1-score	AUC
10	92.75%	92.43%	92.36%	92.36%	97.88%
20	93.02%	92.55%	92.84%	92.91%	98.06%
30	94.61%	94.63%	94.38%	93.44%	98.08%
40	97.93%	96.72%	96.81%	96.82%	98.40%
50	98.59%	98.16%	98.10%	98.10%	98.66%
60	98.95%	98.99%	98.98%	98.98%	98.93%

Table 9: Hyper-parameters Tuning during training.

Parameters	NASNetMobile	
	Without Regularizer	With Regularizer
Regularizer	-	L2
Regularization factor	-	0.015
Learning rate	0.001	0.001
Early stopping	No	Yes
Epochs	60	43

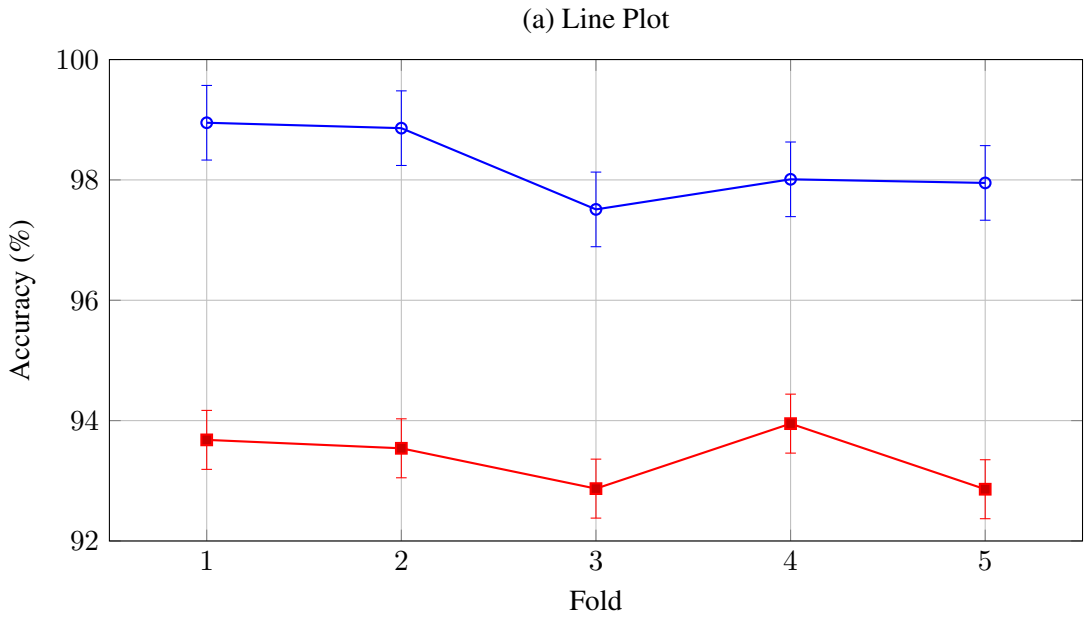
10 and Table 11. Figure 9 gives an illustration of the confusion matrices of all the models. Additionally, Figure 10 shows their respective ROC curves.

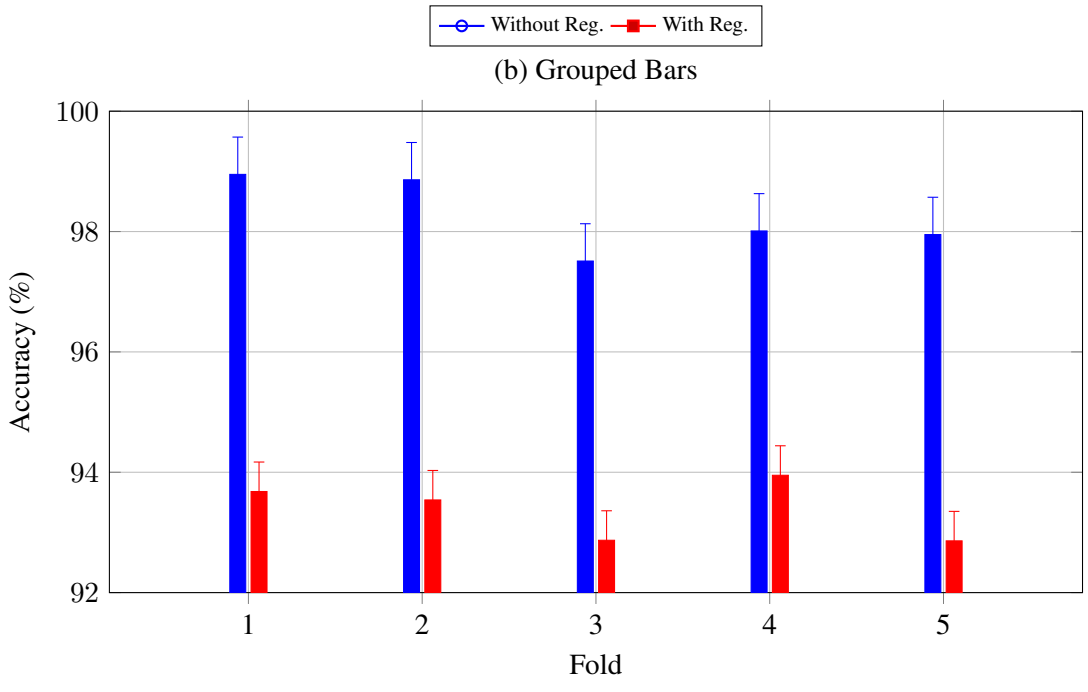
Table 10: Evaluation metrics of NASNetMobile without regularizer

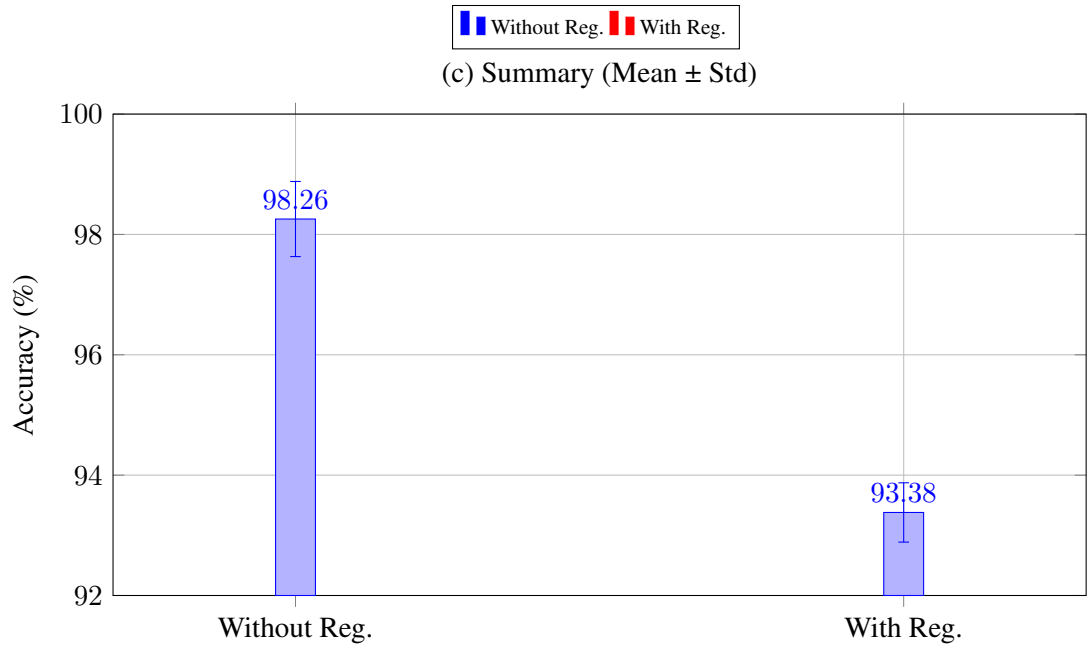
Class	Evaluation metrics				
	Accuracy	Precision	Recall	F1-score	AUC
cataract	98.43%	99.75%	98.43%	99.09%	99.49%
diabetic retinopathy	99.90%	98.98%	99.92%	99.49%	99.99%
glaucoma	99.01%	98.51%	99.00%	98.76%	97.89%
normal	98.49%	98.71%	98.48%	98.60%	98.37%
Average	**98.95%**	**98.99%**	**98.98%**	**98.98%**	**98.93%**

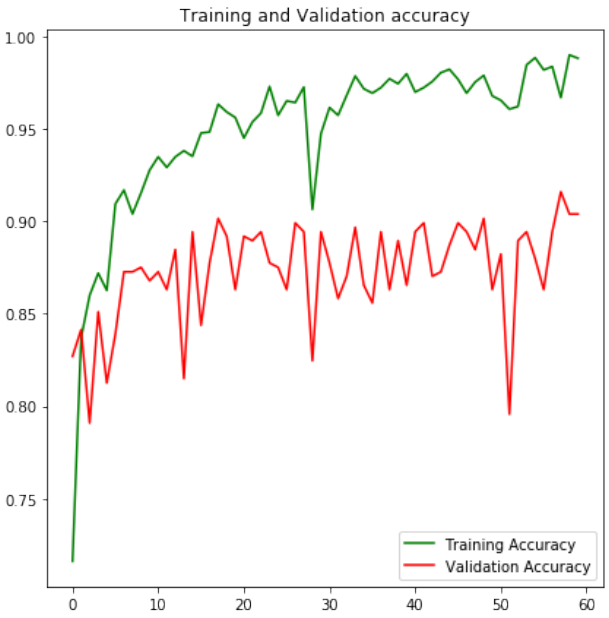
Figure 7 compares the fold-wise performance of NASNetMobile with and without regularization using three complementary visualizations. The line plot illustrates that accuracy remains consistently high across all five folds, with values 97.5% in the unregularized case and above 92.8% with regularization. The grouped bar chart highlights the direct fold-by-fold comparison, confirming that regularization lowers average accuracy but reduces fluctuations across folds. Finally, the summary bar chart reports the consolidated mean accuracies of $98.26\% \pm 0.62$ (without regularization) and $93.38\% \pm 0.49$ (with regularization). Together, these results demonstrate that the model performs reliably across multiple data splits, with only minor variance, thereby strengthening confidence in the reported classification performance.

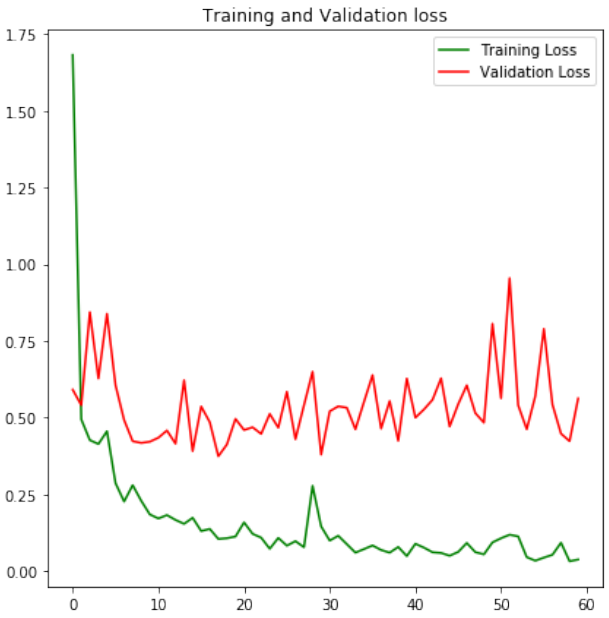
To assess the generalization of the proposed model, we conducted 5-fold cross-validation. Each data set was divided into five equal folds and the model was trained 5five times,each time using a different fold for valiation and the remaining folds for training. The average results are reported in Table 12. The results are trained for 60 epochs. Without regularization, the model achieved an average accuracy of 98.26%, indicating highly stable performance across all folds. The fold-wise accuracies remain consistently high, which confirms that the high reported accuracy is not an artifact of a favorable data split.


(a) Line Plot
100
98
96
94
92
Accuracy (%)
1
2
3
4
5
Fold


Without Reg.
With Reg.
(b) Grouped Bars
100
98
96
94
92
Accuracy (%)
1
2
3
4
5
Fold

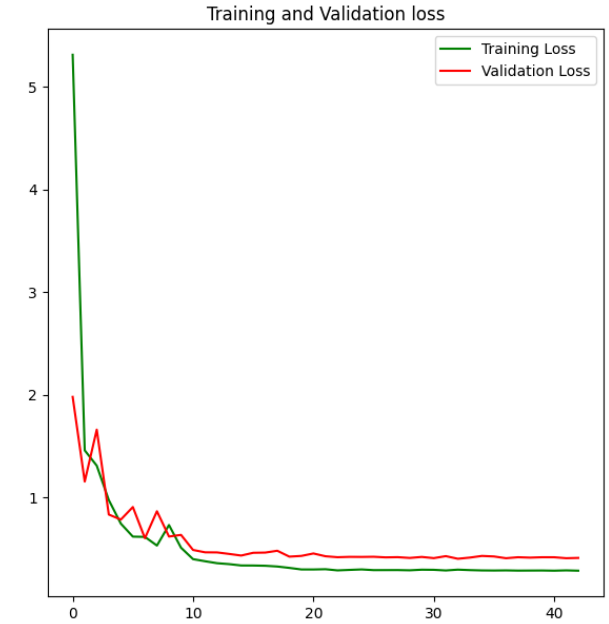

Without Reg.
With Reg.
(c) Summary (Mean ± Std)
100
98
96
94
92
Accuracy (%)
98.26
93.38
Without Reg.
With Reg.





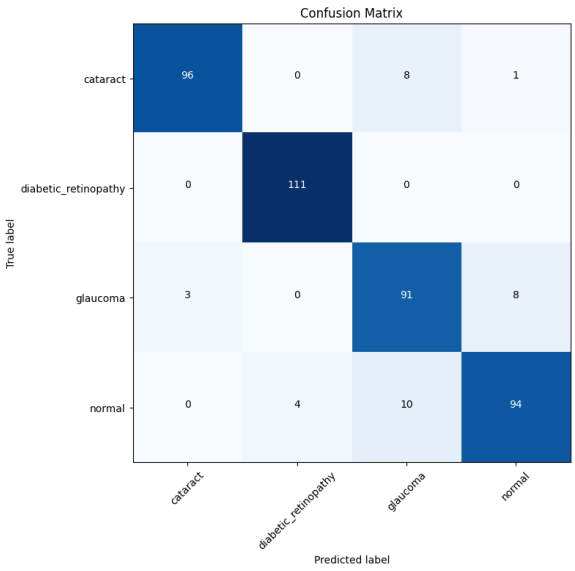
(a) Accuracy and Loss without regularizer



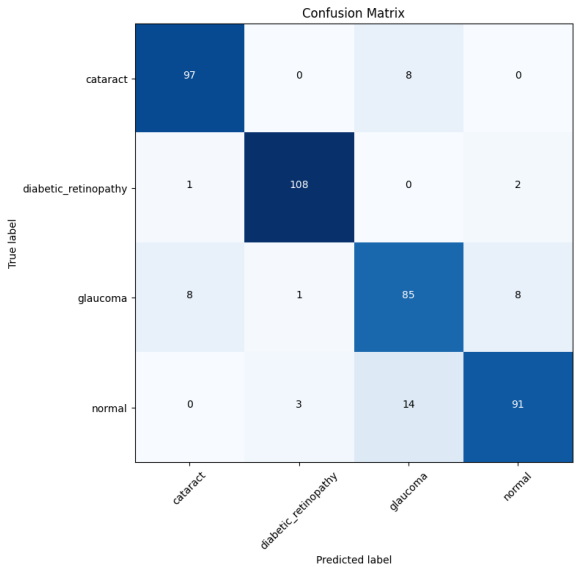


(b) Accuracy and Loss with regularizer

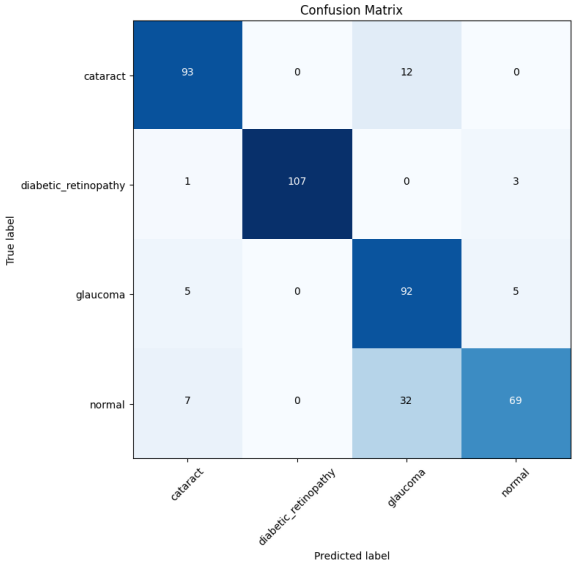
Figure 8: Model Performance



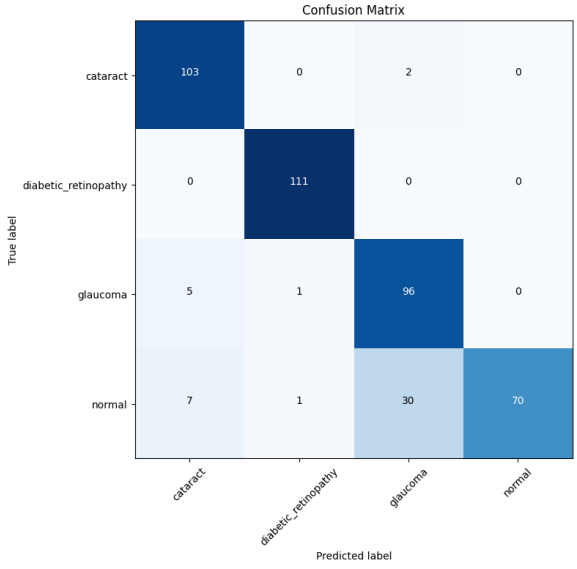
(a) NASNestMobile without regularizer



(b) NASNestMobile with regularizer



(c) SqueezeNet



(d) VGG 16

Figure 9: Confusion matrices of all the models

Table 11: Evaluation metrics of NASNetMobile with regularizer

Class	Evaluation metrics				
	Accuracy	**Precision**	**Recall**	**F1-score**	**AUC**
cataract	95.54%	94.97%	95.54%	95.25%	99.29%
diabetic retinopathy	99.43%	99.43%	99.43%	99.43%	99.87%
glaucoma	85.34%	91.60%	85.34%	88.36%	97.24%
normal	94.41%	89.21%	94.41%	91.74%	98.49%
Average	**93.68%**	**93.80%**	**93.68%**	**93.69%**	**98.72%**

Table 12: 5-Fold Cross Validation of the performance of NasnetMobile (Accuracy in %) with and without regularization

E=60	**NasnetMobile 5-Fold Cross-Validation (Accuracy in %)**	
Folds	**Without Regularization**	**With Regularization**
1	98.95	93.68
2	98.86	93.54
3	97.51	92.87
4	98.01	93.95
5	97.95	92.86
Average	**98.256**	**93.38**
Std Dev	0.62	0.49

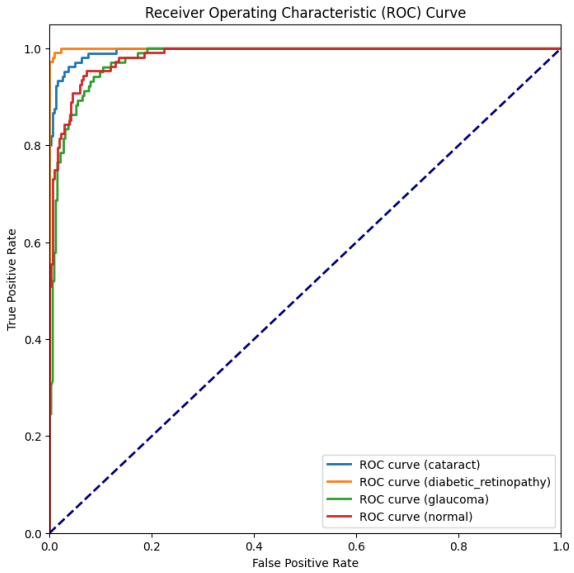
When regularization was introduced, the average accuracy dropped to 93.38% with a slightly lower stadeviation.ation. Although the mean accuracy decreased, the reduction in variance demonstrates that regularization improved the model stability and reduced overfitting tendencies. This trade-off is expected, as regularization penalizes complex weight patterns, leading to mode conservative but generalizable learning. Future work will validate these findings with external datasets to further establish generalization.

4.6 Class Activation Heat Maps (CAM)

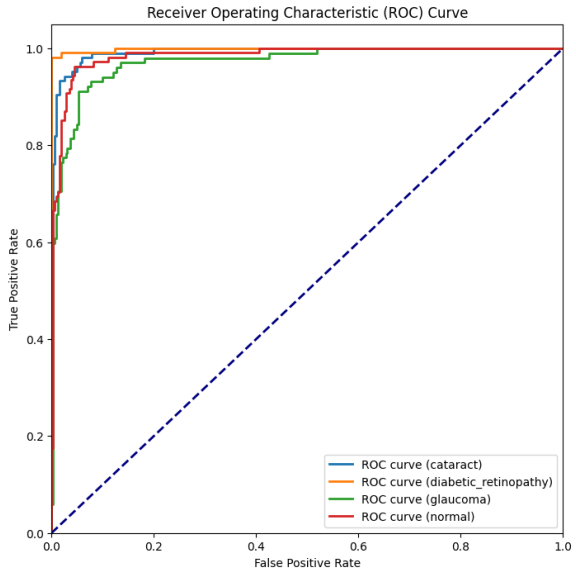
In order to understand the decision making process of the model, we employ Class Activation Heat Maps. These heat maps indicate which parts of the input image influenced the classification outcome. In diabetic retinopathy, for example, the model examines micro aneurysms and hemorrhages, but in the detection of glaucoma, it examines the head region of the optic nerve. A comparison can be made between the CAM output and expert medical comments to ensure that the model accurately detects prominent disease features. In misclassification instances, these heat maps can be further examined to determine whether incorrect predictions are the result of unclear patterns or model biases. Below is the presentation of the Activation Maps from each of the 4 classes:

1. Glaucoma : As described above, glaucoma is a type of disease that is characterized by an elevated intra-ocular pressure. This condition may progress to permanent vision loss if not treated over time. Traditionally, this disease is detected in fundus images by looking at the head of the optic nerve and measuring the cup-to-disc ratio.
   The class activation map shown in Figure 11a shows how the model was effective in detecting glaucoma in an input fundus image by analyzing the head of the optic nerve.

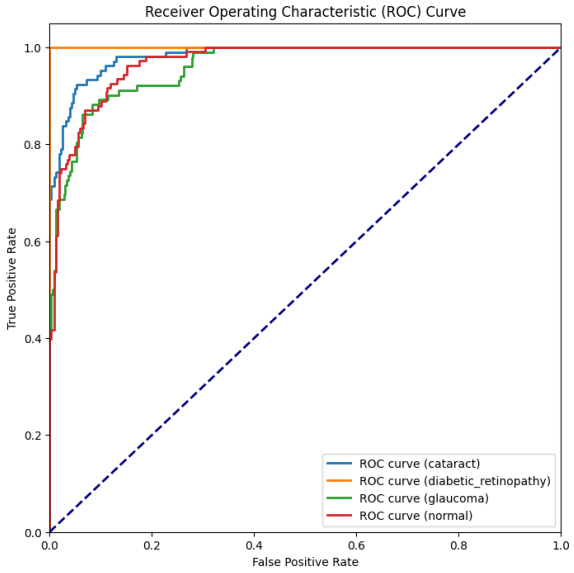
2. Cataract : Cataract is characterized by the clouding of the lens of the eye. This can cause blurry vision and loss of vision if not treaded om time. On a fundus image, Cataract is typically detected by analyzing the image to detect a blurry or hazy appearance, in particular around the optic disc. Figure 11b shows



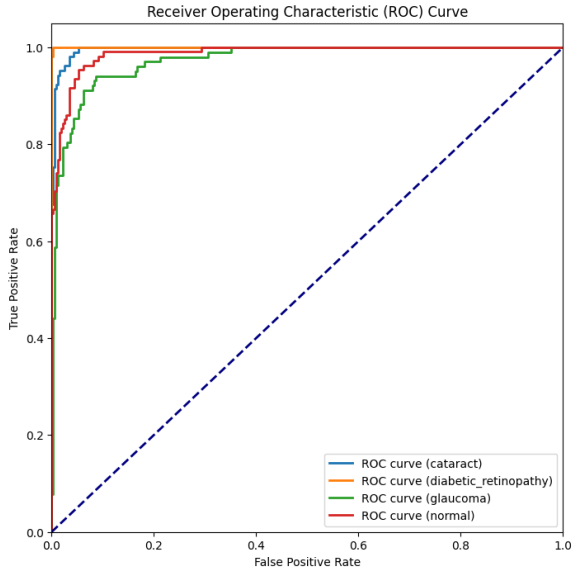
(a) NASNestMobile w/o regularizer



(b) NASNestMobile with regularizer



(c) SqueezeNet



(d) VGG 16

Figure 10: ROC curves of all the models

Table 13: VGG16-based model profiling results

Parameter	Value
Total Parameters	11.80 Million
Inference Latency	220.97 ms
Estimated Memory Usage	45.58 MB
Floating Point Operations per Second	30.72 GFLOPs

Table 14: NASNetMobile-based model profiling results

Parameter	Value
Total Parameters	7.58 Million
Inference Latency	468.05 ms
Estimated Memory Usage	29.05 MB
Floating Point Operations per Second	1.14 GFLOPs

how the model looks at the optic disc and the entire image in general, effectively detecting cataract in a fundus image.

3. Diabetic Retinopathy: Diabetic retinopathy is an eye condition that is directly linked to high blood sugar levels. The excess sugar in the blood damage the small blood vessels in the retina, causing them to swell and lead fluid in the retina. Sometimes, this condition can cause abnormal growth of new blood vessels which can lead to vision loss. In fundus images, diabetic retinopathy is detected by examining the region around blood vessels in search of features like micro aneurysms, hemorrhages and hard exudates. Figure 11c shows how the model looks at that region, effectively detecting diabetic retinopathy in a fundus image.

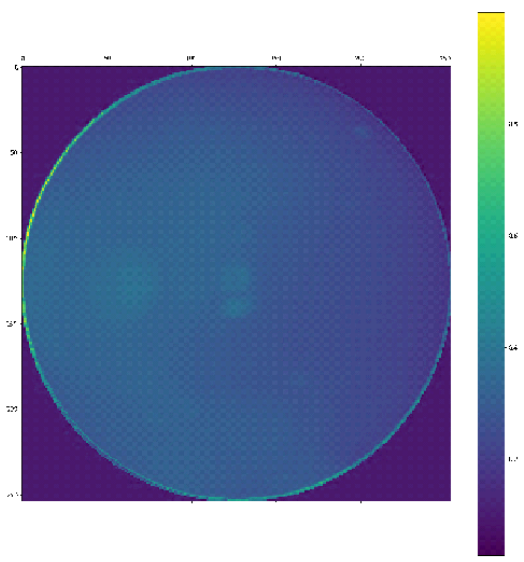
4. Normal : Finally, the last class of the "Normal" class containing images with no diseases. Since the image does not contain any pathology, the model looks at it as a whole. This is illustrated in Figure 11d .

4.7 Computational Profiling

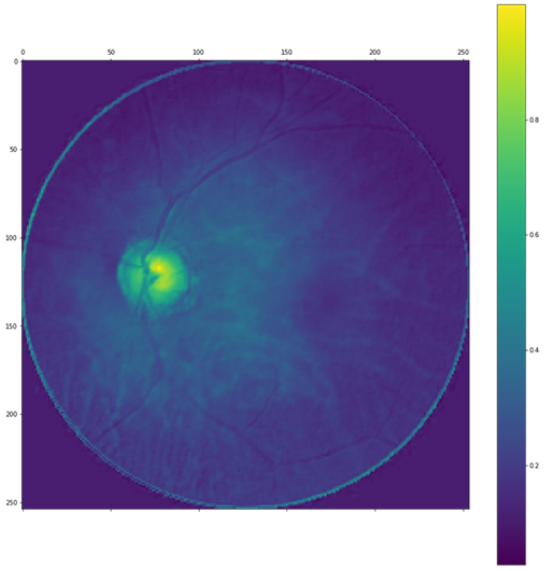
In deep learning, model profiling refers to the process of assessing a model's efficiency by analyzing its resource utilization and execution time for different operations.
In this section, profiling is conducted to analyze the computational efficiency of all transfer learning-based models. Typically, memory consumption, inference latency, total parameter count, and floating point operations per second (FLOPs) are the parameters involved in this experiment. The results of the profiling are shown in Table 13, Table14 and Table 15.

Table 15: SqueezeNet-based model profiling results

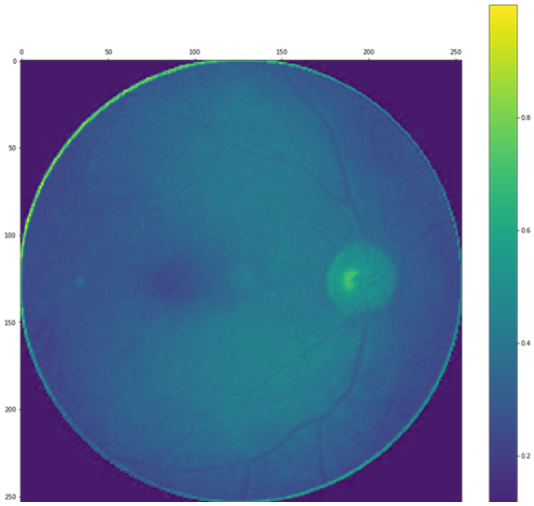
Parameter	Value
Total Parameters	17.93 Million
Inference Latency	49 ms
Estimated Memory Usage	68.96 MB
Floating Point Operations per Second	0.66 GFLOPs



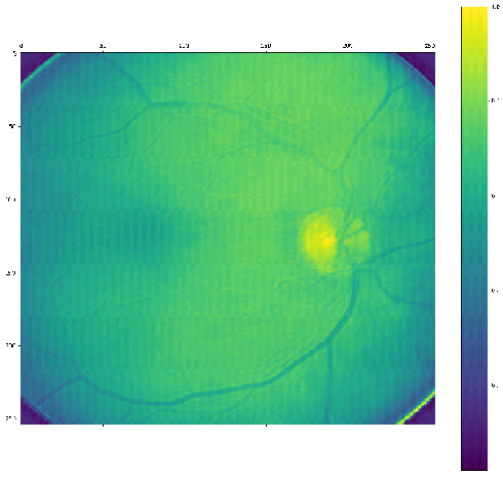
(a) Class A: Normal



(b) Class B: Glaucoma



(c) Class C: Cataract



(d) Class D: Diabetic Retinopathy

Figure 11: Grad-CAM++ visualizations for different disease classes. Each heatmap highlights the discriminative regions used by the NASNetMobile model to classify retinal images.

By looking at the models profiling results, we can observe that SqueezeNet is the fastest model with an inference time of 49.00 ms, which is ideal for real-time applications, but consumes the most memory (68.96 MB). NASNetMobile is the lightest, consuming only 29.05 MB of memory, but the inference time is the longest (468.05 ms), and hence it is not ideal for applications requiring fast response. VGG16 provides an average trade-off between efficiency and speed with a medium inference time (220.97 ms) and computational cost (30.72 GFLOPs). SqueezeNet is ideal for real-time applications, and NASNetMobile is ideal for low-memory applications. VGG16 is ideal for applications requiring good performance and efficiency.

4.8 Ablation study

Table 16: Ablation study of the proposed framework.

Model Variant	Accuracy (%)	Params (M)	FLOPs (M)
VGG16 backbone + dense head	94.5	138	15,000
SqueezeNet backbone + dense head	90.8	1.2	350
NASNetMobile (proposed backbone)	**98.3**	5.3	564
NASNetMobile w/o dropout	98.9	5.3	564
NASNetMobile + dropout (0.4/0.3)	98.3	5.3	564
NASNetMobile + regularization	93.4	5.3	564

To assess the contribution of different design components, we compared the proposed NASNetMobile pipeline against variants using VGG16 and SqueezeNet backbones, as well as NASNetMobile configurations without dropout or with explicit regularization (Table 16). NASNetMobile consistently outperformed the alternatives, achieving the best trade-off between accuracy and efficiency. The dropout slightly reduced the peak accuracy but improved stability, while additional regularization further reduced variance at the expense of mean accuracy. These results justify the final architecture design. The table reports classification accuracy, parameter count, and FLOPs for different backbone architectures (VGG16, SqueezeNet, NASNetMobile) and for variants of the NASNetMobile pipelines with/without dropout and regularization. The results show that NASNetMobile achieves the best trade-off between accuracy and efficiency, while dropout improves stability and regularization reduces variance at the cost of lower mean accuracy.

5 Discussion

This study proposes a deep learning mode for multilabel eye disease classification using fundus images. Several architectures were tested and the NASNetMobile was selected as the feature extractor. It was later tuned to improve its performance on the EDC dataset. This section provides a comparison between the proposed model and the other studies on multi-label eye disease classification mentioned earlier.The architecture was further tunes with global average pooling , Swish-activated dense layers and dropout, and trained with the Adam optimizer. To ensure generalization, 5-fold cross validation was conducted on the EDC dataset, confirming stable performance.

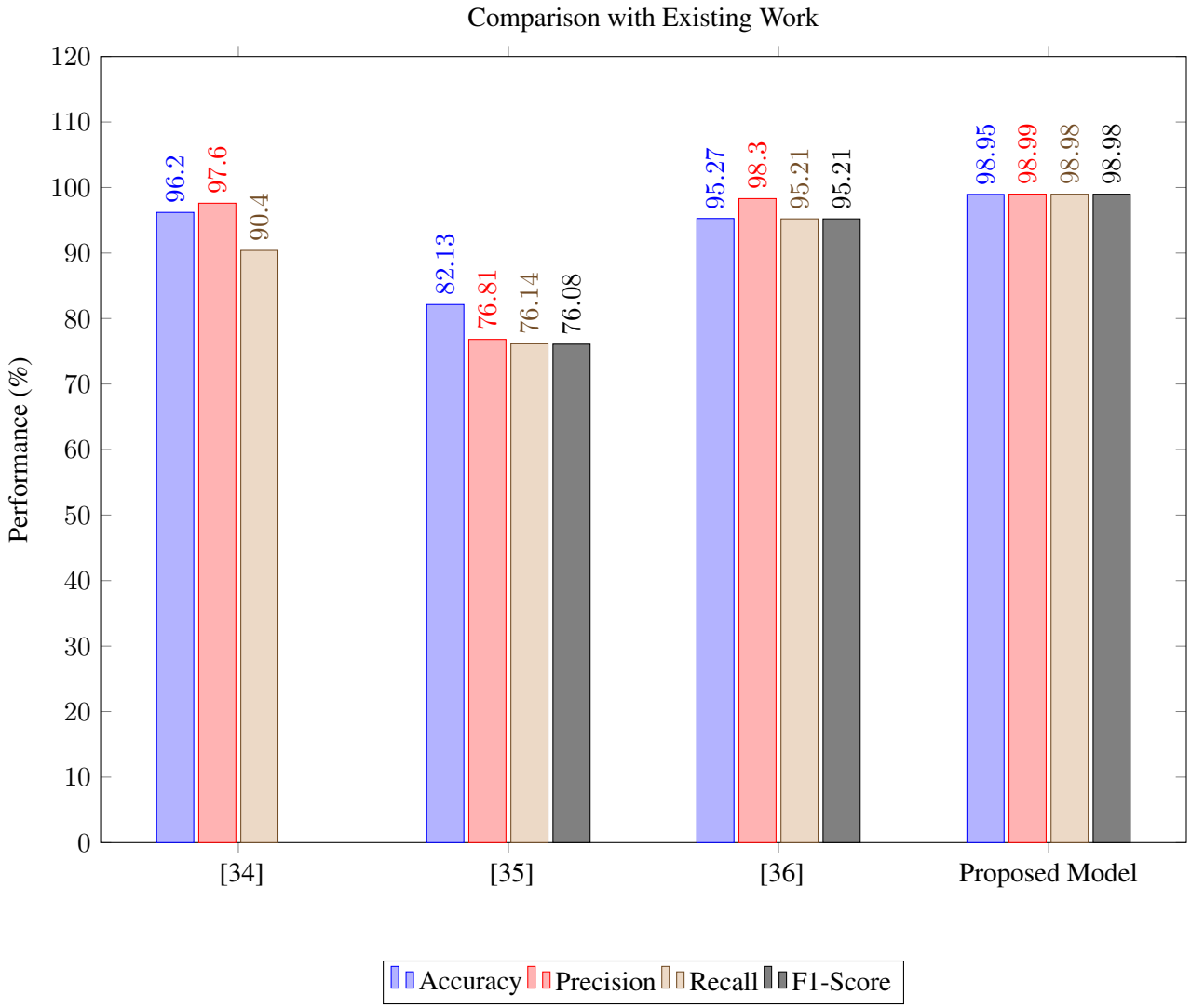


Figure 12: Comparison between state-of-the-art models and the proposed model

Bambang Triwijoyo et al. [26] trained a convolutional neural network on low-resolution images. In their experimentations on the STARE dataset, they obtained the highest accuracy of 80.93% on images with a resolution as low as 31 x 35 pixels. The researchers in [31] benchmarked 9 CNNs on the ODIR data set. Among all the models, the VGG-16 offered the best benchmark performance with an F1 score of up to 87.30%. Sarki et al. [33] worked on diabetic eye diseases and built a custom CNN that achieved 81.33% of accuracy. The MobileNetV2 architecture was used in [34] to address the classification of multilabel eye disease. Their proposed model displayed good results, with an average accuracy of 96.2%. The EyeDeep-Net model [35] on the other hand was trained on the RFMiD dataset. This 17-layer neural network yielded an accuracy of up to 82.13%. Finally, the optimized neural network proposed in [36] achieved an accuracy of 95.27% with a high precision of 98.30%. Interpretability ws validated through Grad-CAM++ heatmaps,which confirmed that the model's predictions align with disease-relevant regions of the retina.Additionally, a FLOPs and parameter analysis demonstrated lower computational complexity than VGG16 with better accuracy than squeezeNet, supporting deployment feasibility.

The proposed model offers better performance in terms of accuracy, precision, and recall as compared to the state-of-the-art techniques. In fact, with an accuracy of 98.95%, it showed a 2.75% accuracy advantage over the most accurate model proposed in [34]. It also performed with Precision, Recall, F1-score, and AUC at 98.99%, 98.98%, 98.98%, and 98.93% respectively. A visual comparison proposed model against state-of-the-art models is shown in Figure 12.

6 Conclusion

This study addressed the problem of multi-label eye disease classification. Leveraging the capabilities of pre-trained architectures, a deep learning model was built using the NASNetMobile feature extractor. The proposed model demonstrated high performance in classifying multiple ocular diseases from retinal fundus images. It was trained on the EDC dataset, which contains 4 classes of retinal fundus images: Cataract, Diabetic Retinopathy, Glaucoma, and Normal. The model was evaluated using five performance metrics: Accuracy, Precision, Recall, F1-score, and Area Under the Curve (AUC). It performed with an accuracy of 98.95%

The model's high accuracy and recall highlight its potential for integration into clinical workflows. In fact, the integration of multi-label classification techniques would tackle the complexity of real-world clinical scenarios where patients often exhibit multiple co-existing eye disorders. The study also underscores the effectiveness of transfer learning in medical imaging applications, where high-quality annotated data can be scarce. Further work may focus on the following points:

- Expanding the retinal fundus image dataset
- Optimizing the model's architecture to further improve accuracy
- Reducing computational requirements
- Integrating the model into a clinical framework

In conclusion, the ability to identify and classify multiple eye diseases simultaneously is a significant leap forward in ophthalmic diagnostics.

References

[1] Sajid, Muhammad Zaheer, Qureshi, Imran, Abbas, Qaisar, Albathan, Mubarak, Shaheed, Kashif, Youssef, Ayman et al.. "Mobile-HR: An Ophthalmologic-Based Classification System for Diagnosis of Hypertensive Retinopathy Using Optimized MobileNet Architecture." *Diagnostics*. vol. 13, no. 8, 2023. doi: doi:10.3390/diagnostics13081439.

[2] World Health Organization. "World Report on Vision." 2019. Available: https://www.who.int/publications/i/item/world-report-on-vision. Accessed: 2024-07-18.

[3] Bourne, Rupert, Stevens, G.A., White, Richard, Smith, J.L., Flaxman, S.R., Barr, Holly et al.. "Causes of vision loss worldwide, 1990-2010: a systematic analysis." *Lancet Glob Health*. vol. 1, pp. 1990-2010, 2013. doi:/10.1016/s2214-109x(13)70113-x.

[4] Mojon-Azzi, Stefania, Sousa-Poza, Alfonso, Mojon, Daniel. "Impact of Low Vision on Employment." *Ophthalmologica. Journal international d'ophtalmologie. International journal of ophthalmology. Zeitschrift für Augenheilkunde*. vol. 224, pp. 381-8, 2010. doi: doi:10.1159/000316688.

[5] Eckert, Kristen A., Lansingh, Van C., Carter, Marissa J., Frick, Kevin D.. "Update of a Simple Model to Calculate the Annual Global Productivity Loss Due to Blindness and Moderate and Severe Vision Impairment." *Ophthalmic Epidemiology*. vol. 30, no. 2, pp. 142–150, 2023. doi: doi:10.1080/09286586.2022.2072899.

[6] Burton, Matthew J, Jacqueline Ramke, Marques, Ana Patricia, Bourne, Rupert R A, Nathan Congdon, Iain Jones et al.. "The Lancet Global Health Commission on Global Eye Health: vision beyond 2020." *The Lancet. Global health*. vol. 9, no. 4, pp. e489–e551, 2021. doi: doi:10.1016/S2214-109X(20)30488-5.

[7] Kriti, Jitendra Virmani, Ravinder Agarwal. "Assessment of despeckle filtering algorithms for segmentation of breast tumours from ultrasound images." *Biocybernetics and Biomedical Engineering*. vol. 39, no. 1, pp. 100-121, 2019. doi: doi:10.1016/j.bbe.2018.10.002.

[8] Haraburda, Patrycja, Dabała, Łukasz. "Eye Diseases Classification Using Deep Learning." *Image Analysis and Processing – ICIAP 2022*. pp. 160–172, 2022. doi:10.1007/978-3-031-06427-2_14.

[9] An, Guangzhou, Omodaka, Kazuko, Hashimoto, Kazuki, Tsuda, Satoru, Shiga, Yukihiro, Takada, Naoko et al.. "Glaucoma Diagnosis with Machine Learning Based on Optical Coherence Tomography and Color Fundus Images." *Journal of Healthcare Engineering*. vol. 2019, pp. 1-9, 2019. doi: doi:10.1155/2019/4061313.

[10] Daniel S.W. Ting, Lily Peng, Avinash V. Varadarajan, Pearse A. Keane, Philippe M. Burlina, Michael F. Chiang et al.. "Deep learning in ophthalmology: The technical and clinical considerations." *Progress in Retinal and Eye Research*. vol. 72, pp. 100759, 2019. doi: doi:10.1016/j.preteyeres.2019.04.003.

[11] Tao Li, Wang Bo, Chunyu Hu, Hong Kang, Hanruo Liu, Kai Wang et al.. "Applications of deep learning in fundus images: A review." *Medical Image Analysis*. vol. 69, pp. 101971, 2021. doi: doi:10.1016/j.media.2021.101971.

[12] Maryam Badar, Muhammad Haris, Anam Fatima. "Application of deep learning for retinal image analysis: A review." *Computer Science Review*. vol. 35, pp. 100203, 2020. doi: doi:10.1016/j.cosrev.2019.100203.

[13] González-Gonzalo, Cristina, Sánchez-Gutiérrez, Verónica, Hernández-Martínez, Paula, Contreras, Inés, Lechanteur, Yara, Domanian, Artin et al.. "Evaluation of a deep learning system for the joint automated detection of diabetic retinopathy and age-related macular degeneration." *Acta Ophthalmologica*. vol. 98, 2019. doi: doi:10.1111/aos.14306.

[14] Siegfried Karl Wagner, Dun Jack Fu, Livia Faes, Xiaoxuan Liu, Josef Huemer, Hagar Khalid et al.. "Insights into Systemic Disease through Retinal Imaging-Based Oculomics." *Translational Vision Science & Technology*. vol. 9, 2020. Available: https://api.semanticscholar.org/CorpusID:212871561.

[15] Arcadu, Filippo, Benmansour, Fethallah, Maunz, Andreas, Willis, Jeff, Haskova, Zdenka, Prunotto, Marco. "Deep learning algorithm predicts diabetic retinopathy progression in individual patients." *npj Digital Medicine*. vol. 2, 2019. doi: doi:10.1038/s41746-019-0172-3.

[16] Yan, Qi, Weeks, Daniel E., Xin, Hongyi, Huang, Heng, Swaroop, Anand, Chew, Emily Y. et al.. "Deep-learning-based Prediction of Late Age-Related Macular Degeneration Progression." *medRxiv*. 2019. doi: doi:10.1101/19006171.

[17] Wahab Sait, Abdul Rahaman. "Artificial Intelligence-Driven Eye Disease Classification Model." *Applied Sciences*. vol. 13, no. 20, 2023. doi: doi:10.3390/app132011437.

[18] Juneja, Mamta, Singh, Shaswat, Agarwal, Naman, Bali, Shivank, Gupta, Shubham, Thakur, Niharika et al.. "Automated detection of Glaucoma using deep learning convolution network (G-net)." *Multimedia Tools and Applications*. vol. 79, pp. 15531–15553, 2020. doi: doi:10.1007/s11042-019-7460-4.

[19] Abbadullah .H Saleh, Oğuzhan Menemencioğlu. "Study the effect of eye diseases on the performance of iris segmentation and recognition using transfer deep learning methods." *Engineering Science and Technology, an International Journal*. vol. 47, pp. 101552, 2023. doi: doi:10.1016/j.jestch.2023.101552.

[20] Alkhadrawi, Adham M., Lin, Lisa Y., Langarica, Saul A., Kim, Kyungsu, Ha, Sierra K., Lee, Nahyoung G. et al.. "Deep-Learning Based Automated Segmentation and Quantitative Volumetric Analysis of Orbital Muscle and Fat for Diagnosis of Thyroid Eye Disease." *Investigative Ophthalmology & Visual Science*. vol. 65, no. 5, pp. 6-6, 2024. doi: doi:10.1167/iovs.65.5.6.

[21] Lai WEI, Menghan HU. "A review of medical ocular image segmentation." *Virtual Reality & Intelligent Hardware*. vol. 6, no. 3, pp. 181-202, 2024. doi: doi:10.1016/j.vrih.2024.04.001.

[22] Vadduri, Maneesha, Kuppusamy, P.. "Enhancing Ocular Healthcare: Deep Learning-Based Multi-Class Diabetic Eye Disease Segmentation and Classification." *IEEE Access*. vol. 11, pp. 137881-137898, 2023. doi: doi:10.1109/ACCESS.2023.3339574.

[23] Yadav, Niranjan, Dass, Rajeshwar, Virmani, Jitendra. "Despeckling filters applied to thyroid ultrasound images: a comparative analysis." *Multimed. Tools Appl.*. vol. 81, no. 6, pp. 8905–8937, 2022. doi: doi:10.1007/s11042-022-11965-6.

[24] Abbas, Qaisar, Albathan, Mubarak, Altameem, Abdullah, Almakki, Riyad Saleh, Hussain, Ayyaz. "Deep-Ocular: Improved Transfer Learning Architecture Using Self-Attention and Dense Layers for Recognition of Ocular Diseases." *Diagnostics*. vol. 13, no. 20, 2023. doi: doi:10.3390/diagnostics13203165.

[25] Tayal, Akash, Gupta, Jivansha, Solanki, Arun, Bisht, Khyati, Nayyar, Anand, Masud, Mehedi. "DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases." *Multimedia systems*. vol. 28, no. 4, pp. 1417–1438, 2022. doi: doi:10.1007/s00530-021-00791-9.

[26] Triwijoyo, Bambang, Sabarguna, Boy, Budiharto, Widodo, Abdurachman, Edi. "Deep learning approach for classification of eye diseases based on color fundus images." *Computer-Assisted Diagnosis*. pp. 25-57, 2020. doi: doi:10.1016/B978-0-12-817440-1.00002-4.

[27] Lin, Lisa, Zhou, Paul, Shi, Min, Lu, Jonathan, Jeon, Soomin, Kim, Doyun et al.. "A Deep Learning Model for Screening Computed Tomography Imaging for Thyroid Eye Disease and Compressive Optic Neuropathy." *Ophthalmology Science*. vol. 4, pp. 100412, 2023. doi: doi:10.1016/j.xops.2023.100412.

[28] Rohit Thanki. "A deep neural network and machine learning approach for retinal fundus image classification." *Healthcare Analytics*. vol. 3, pp. 100140, 2023. doi: doi:10.1016/j.health.2023.100140.

[29] Ngoc Thien Le, Thanh Le Truong, Pear Ferreira Pongsachareonnont, Disorn Suwajanakorn, Apivat Mavichak, Rath Itthipanichpong et al.. "Deep Learning Approach for Age-related Macular Degeneration Detection Using Retinal Images: Efficacy Evaluation of Different Deep Learning Models." *Egyptian Informatics Journal*. vol. 24, no. 4, pp. 100402, 2023. doi: doi:10.1016/j.eij.2023.100402.

[30] Fan Song, Weiyi Zhang, Yingfeng Zheng, Danli Shi, Mingguang He. "A deep learning model for generating fundus autofluorescence images from color fundus photography." *Advances in Ophthalmology Practice and Research*. vol. 3, no. 4, pp. 192-198, 2023. doi: doi:10.1016/j.aopr.2023.11.001.

[31] Li, Ning, Li, Tao, Hu, Chunyu, Wang, Kai, Kang, Hong. "A Benchmark of Ocular Disease Intelligent Recognition: One Shot for Multi-disease Detection." *Benchmarking, Measuring, and Optimizing*. pp. 177–193, 2021. doi: doi:10.1007/978-3-030-71058-3_11.

[32] V. Purna Chandra Reddy, Kiran Kumar Gurrala. "Joint DR-DME classification using deep learning-CNN based modified grey-wolf optimizer with variable weights." *Biomedical Signal Processing and Control*. vol. 73, pp. 103439, 2022. doi: doi:10.1016/j.bspc.2021.103439.

[33] Sarki, Rubina, Ahmed, Khandakar, Zhang, Yanchun. "Early Detection of Diabetic Eye Disease through Deep Learning using Fundus Images." *EAI Endorsed Transactions on Pervasive Health and Technology*. vol. 6, pp. 164588, 2018. doi: doi:10.4108/eai.13-7-2018.164588.

[34] Guo, Chen, Yu, Minzhong, Li, Jing. "Prediction of Different Eye Diseases Based on Fundus Photography via Deep Transfer Learning - jcm-10-05481-2." *Journal of Clinical Medicine*. vol. 10, 2021. doi: doi:10.3390/jcm10235481.

[35] Sengar, Neha, Joshi, Rakesh Chandra, Dutta, Malay Kishore, Burget, Radim. "EyeDeep-Net: a multi-class diagnosis of retinal diseases using deep neural network." *Neural Computing and Applications*. vol. 35, 2023. doi: doi:10.1007/s00521-023-08249-x.

[36] P. Glaret subin, P. Muthukannan. "Optimized convolution neural network based multiple eye disease detection." *Computers in Biology and Medicine*. vol. 146, pp. 105648, 2022. doi: doi:10.1016/j.compbiomed.2022.105648.

[37] Albawi, Saad, Mohammed, Tareq Abed, Al-Zawi, Saad. "Understanding of a convolutional neural network." *2017 International Conference on Engineering and Technology (ICET)*. pp. 1-6, 2017. doi: doi:10.1109/ICEngTechnol.2017.8308186.

[38] Lo, Shih-Chung, Lin, Jyh-Shyan, Freedman, Matthew, Mun, Seong. "Computer-Assisted Diagnosis of Lung Nodule Detection using Artificial Convolution Neural Network." *Proceedings of SPIE - The International Society for Optical Engineering*. vol. 1898, pp. 859-869, 1993. doi: doi:10.1117/12.154572.

[39] Zhou, Y., Sun, X., others. "3D image recognition using new set of fractional-order Legendre moments and deep neural networks." *Signal Processing: Image Communication*. vol. 98, pp. 116410, 2021. doi: doi:10.1016/j.image.2021.116410.

[40] Wang, J., Xu, H., others. "Novel fractional-order Jacobi moments and invariant moments for pattern recognition applications." *Neural Computing and Applications*. vol. 33, no. 20, pp. 13539–13565, 2021. doi: doi:10.1007/s00521-021-05828-1.

[41] El Ogri, O., El-Mekkaoui, J., Benslimane, M., and Hjouji, A. 'Automatic lip-reading classification using deep learning approaches and optimized quaternion Meixner moments by GWO algorithm." *Knowledge-Based Systems*. vol. 304, Article 112430, 2024. doi:10.1016/j.knosys.2024.112430.

[42] Sahmoudi, Y., El Ogri, O., Hjouji, A., and El-Mekkaoui, J. "New Biometric Recognition System By Multichannel Orthogonal Radial Triangular Pulse Moments." In: *Proceedings of the 2024 3rd International Conference on Embedded Systems and Artificial Intelligence (ESAI)*, pp. 1–11, 2024. IEEE. doi: doi:10.1109/ESAI62891.2024.10913855.

[43] Singh, A., Zhao, Y., others. "Quantum-inspired classifiers for medical image recognition." *IEEE Transactions on Emerging Topics in Computing*. 2023. doi: doi:10.1109/TETC.2023.000XXX.

[44] Huang Yi, Sun Shiyu, Duan Xiusheng, Chen Zhigang. "A study on Deep Neural Networks framework." *2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC)*. pp. 1519-1522, 2016. doi: doi:10.1109/IMCEC.2016.7867471.

[45] Guna Venkat Doddi. "eye diseases classification." 2022. Available: `https://www.kaggle.com/gunavenkatdoddi/eye-diseases-classification`. Accessed: 2024-07-18.

[46] Krig, Scott. "Image Pre-Processing." *Computer Vision Metrics: Textbook Edition*. pp. 35–74, 2016. doi: doi:10.1007/978-3-319-33762-3_2.

[47] Koo, Kyung-Mo, Cha, Eui-Young. "Image recognition performance enhancements using image normalization." *Human-centric Computing and Information Sciences*. vol. 7, pp. 1–11, 2017. doi: doi:10.1186/s13673-017-0114-5.

[48] Saponara, Sergio, Elhanashi, Abdussalam. "Impact of Image Resizing on Deep Learning Detectors for Training Time and Model Performance." *Applications in Electronics Pervading Industry, Environment and Society*. pp. 10–17, 2022. doi: doi:10.1007/978-3-030-95498-7_2.

[49] Alhassan Mumuni, Fuseini Mumuni. "Data augmentation: A comprehensive survey of modern approaches." *Array*. vol. 16, pp. 100258, 2022. doi: doi:10.1016/j.array.2022.100258.

[50] Perez, Luis, Wang, Jason. "The Effectiveness of Data Augmentation in Image Classification using Deep Learning." *Convolutional Neural Networks Vis. Recognit*. vol. 11, pp. 1-8, 2017. doi: doi:10.48550/arXiv.1712.04621.

[51] Shorten, Connor, Khoshgoftaar, Taghi. "A survey on Image Data Augmentation for Deep Learning." *Journal of Big Data*. vol. 6, 2019. doi: doi:10.1186/s40537-019-0197-0.

[52] Ribani, Ricardo, Marengoni, Mauricio. "A Survey of Transfer Learning for Convolutional Neural Networks." *2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T)*. pp. 47-57, 2019. doi: doi:10.1109/SIBGRAPI-T.2019.00010.

[53] Hosna, Asmaul, Merry, Ethel, Gyalmo, Jigmey, Alom, Zulfikar, Aung, Zeyar, Azim, Mohammad. "Transfer learning: a friendly introduction." *Journal of Big Data*. vol. 9, 2022. doi: doi:10.1186/s40537-022-00652-w.

[54] Zehui Zhao, Laith Alzubaidi, Jinglan Zhang, Ye Duan, Yuantong Gu. "A comparison review of transfer learning and self-supervised learning: Definitions, applications, advantages and limitations." *Expert Systems with Applications*. vol. 242, pp. 122807, 2024. doi: doi:10.1016/j.eswa.2023.122807.

[55] Krishnan, Rayan, Rajpurkar, Pranav, Topol, Eric. "Self-supervised learning in medicine and healthcare." *Nature Biomedical Engineering*. vol. 6, pp. 1-7, 2022. doi: doi:10.1038/s41551-022-00914-1.

[56] Asanuma, Jo, Doi, Shuhei, Igarashi, Hajime. "Transfer Learning Through Deep Learning: Application to Topology Optimization of Electric Motor." *IEEE Transactions on Magnetics*. vol. PP, pp. 1-1, 2020. doi: doi:10.1109/TMAG.2019.2956849.

[57] Dara, Suresh, Tumma, Priyanka. "Feature Extraction By Using Deep Learning: A Survey." *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*. pp. 1795-1801, 2018. doi: doi:10.1109/ICECA.2018.8474912.

[58] Amraee, Somaieh, Chinipardaz, Maryam, Charoosaei, Mohammadali. "Analytical study of two feature extraction methods in comparison with deep learning methods for classification of small metal objects." *Visual Computing for Industry Biomedicine and Art*. vol. 5, pp. 13, 2022. doi: doi:10.1186/s42492-022-00111-6.

[59] Abbas, Qaisar, Albathan, Mubarak, Altameem, Abdullah, Almakki, Riyad Saleh, Hussain, Ayyaz. "DeepOcular: Improved Transfer Learning Architecture Using Self-Attention and Dense Layers for Recognition of Ocular Diseases." *Diagnostics*. vol. 13, no. 20, 2023. doi: doi:10.3390/diagnostics13203165.

[60] B. Zoph, V. Vasudevan, J. Shlens, Q. V. Le. "Learning Transferable Architectures for Scalable Image Recognition." *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 8697-8710, 2018. Available: https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00907.

[61] Radhika, K., Devika, K., Aswathi, T., Sreevidya, P., Sowmya, V., Soman, K. P.. "Performance Analysis of NASNet on Unconstrained Ear Recognition." *Nature Inspired Computing for Data Science*. pp. 57–82, 2020. doi: doi:10.1007/978-3-030-33820-6_3.

[62] S, Raghuram, Bharadwaj, Anirudh S, S K, Deepika, Khadabadi, Mridula S, Jayaprakash, Aditya. "Digital Implementation of the Softmax Activation Function and the Inverse Softmax Function." *2022 4th International Conference on Circuits, Control, Communication and Computing (I4C)*. pp. 64-67, 2022. doi: doi:10.1109/I4C57141.2022.10057747.

[63] Hossin, Mohammad, M.N, Sulaiman. "A Review on Evaluation Metrics for Data Classification Evaluations." *International Journal of Data Mining & Knowledge Management Process*. vol. 5, pp. 01-11, 2015. doi: doi:10.5121/ijdkp.2015.5201.