International Journal of Computational and Experimental Science and ENgineering (IJCESEN)

Vol. 11-No.2 (2025) pp. 3341-3357
http://www.ijcesen.com

ISSN: 2149-9144

Research Article

CNN and SVM Hybrid Model: A Robust Solution for Diabetic Retinopathy Classification

Karthika Gidijala[1*], Vijaya Kumar Sagenela[2]

[1 *] Department of Computer Science and Engineering GITAM (Deem to be University), Hyderabad.
* **Corresponding Author Email:** kgidijal@gitam.edu – **ORCID:** 0000-0002-7470-1294

[2] Department of Computer Science and Engineering GITAM (Deem to be University), Hyderabad.
Email: vsagenel@gitam.edu – **ORCID:** 0000-0002-2867-3093

Article Info:

DOI: 10.22399/ijcesen.1971
Received : 05 February 2025
Accepted : 22 April 2025

Keywords :

Diabetic Retinopathy
Hybrid CNNs
Deep learning
EfficientNetV2S
EfficientNetB0

Abstract:

Diabetic Retinopathy (DR) is a common problem of diabetes mellitus, which causes lesions on the retina that affect vision. If it is not detected early, it can lead to blindness. Unfortunately, DR is not reversible, and treatment only sustains vision. Early detection and treatment of DR can significantly reduce the risk of vision loss. Unlike computer-aided diagnosis systems, ophthalmologists' manual diagnosis process of DR retina fundus images is time-, effort-, and cost-consuming and prone to misdiagnosis. Recently, deep learning has become one of the most common techniques that have achieved better performance in many areas, especially in medical image analysis and classification. Convolutional neural network models are more widely used as a deep learning method in medical image analysis, and they are highly effective. In this context, this work proposes and investigates hybrid CNNs using support vector machines and compares them with state-of-the-art CNN architectures. To select which models to use we tested 10 state-of-art CNN architectures: EfficientNetV2S, EfficientNetB0, ResNet50, DenseNet121, MobileNetV2, InceptionV3, Xception, VGG16, VGG19 and NASNetMobile. We formed the 9,815 DR dataset with images from the Indian Diabetic Retinopathy Image Dataset (IDRiD), Kaggle's Diabetic Retinopathy dataset, and images from American Eye Hospital Hyderabad. The results showed that the hybrid CNNs using support vector machines tend to present the best results. The experimentation outcome showed that the proposed approach classifies all the classes of Diabetic Retinopathy and performs better compared to other methods with an accuracy of 90.02%.

1. Introduction

Diabetic Retinopathy, one of the complications caused by diabetes, is globally considered a public health concern, with a notably high incidence in India. The World Health Organization (WHO) predicts that diabetes will be 'roughly 783 million people impacted by diabetes' by 2045[1]. WHO's observation is verified by the fact that DR is rapidly increasing in prevalence. If it is not properly diagnosed and treated effectively, people who suffer from DR will lose their vision permanently. The DR is at staggering levels throughout the world and in high amounts in India. About 22.4 % of cases of DR among the urban population in Andhra Pradesh were found in the state of India by itself.
The same pattern holds true for the area in the south Indian state of Telangana. Factors such as DR of diabetes and inadequate glycaemic management have been identified as the cause for the speedy growth rate of the DR in Telangana [2].
The DR is projected to be the leading cause of blindness in the world, and hence, an effective method to screen retinas is an imperative solution. Customized screening methods are used by resource-rich nations, which incorporate the individual's profile of risk using factors such as duration of diabetes and prior diagnosis of DR. It offers a customized way to test, catch early, detect accurately, and intervene quickly. In cases of lack of robust Eye Screening Systems or systems to screen DR, there is delayed diagnosis, and chances of vision impairment increase. The problem that this study aims to solve is that societies with an

elevated number of Diabetic Retinopathy patients lack vast machinery to perform individualized screening implemented subsequently by Artificial Intelligence (AI) based retinal screening.

The AI has the possibility of creating a cheaper and easier alternative solution for diabetic retinopathy screening [32]-[34]. The sensitivity and specificity that AI provides make it a viable system that can withstand the screening workload. Studies suggest that AI algorithms may have achieved high levels of sensitivity (80-90%) and specificity (86-100%) for the identification of the DR. The ability to automate and improve the efficacy of the retinal screening processes for DR makes AI-based systems a more affordable solution with a higher degree of accuracy. The application of AI in the screening methods may prove useful in removing the economic barrier on the way to ensuring early detection and thereby reducing the strain on healthcare systems. This technology offers a helpful solution to the global challenge of the DR and its associated costs in healthcare.

Convolutional neural networks (CNNs) are made to distinguish patterns among the input data and link it to the class labels that fall out. Tasks such as image recognition and classification are the major examples of the uses of CNNs [4, 5, 6]. The field of medical image processing has documented numerous CNN based architectures explored [6,7]. Furthermore, previous work [8–10] performed classification tasks using CNNs while integrating them with traditional machine learning techniques like Support Vector Machines (SVM), k nearest Neighbour (k-NN), and Decision Trees (DT). In this paper, a hybrid model for diabetic retinopathy image classification as healthy, mild, moderate, severe, and proliferative diabetic retinopathy (PDR) is presented.

The structure of this paper is organized as follows: Section 2 provides an overview of the stages of diabetic retinopathy, Section 3 reviews related studies, Section 4 outlines the proposed methodology, Section 5 discusses results and findings, and Section 6 concludes the study.

2. Stages of Diabetic Retinopathy

The development of the DR among diabetic patients occurs in multiple progressive stages. Distinct retinal changes will be observed in each stage. At every stage, the severity of damage to the retinal blood vessels increases, resulting in minor alterations to abnormal blood vessel growth that impairs vision. Understanding the progression of DR is critical for timely diagnosis, intervention, and prevention of vision loss. The stages of DR are categorized into Mild, Moderate, and Severe Non-Proliferative Diabetic Retinopathy (NPDR), culminating in Proliferative Diabetic Retinopathy (PDR), each described in detail below.

Stage 1: Mild NPDR

The first stage of diabetic retinopathy, known as mild non-proliferative diabetic retinopathy (NPDR), is characterized by localized changes in the retinal blood vessels. The outward signs of these changes are tiny protuberances or microaneurysms in the blood vessels. Microaneurysms can cause little blood and fluid leaks into the retina and show up as red focal spots on retinal imaging. The macula, the central part of the retina responsible for direct and clear vision, may somewhat enlarge as a result of fluid infiltration during this stage. Despite these early changes, vision is usually unaffected, and people with moderate NPDR frequently don't exhibit any symptoms at all. Timely identification is crucial, even if therapy is typically not required at this stage. Physicians who detect moderate non-proliferative diabetic retinopathy (NPDR) may advise lifestyle modifications and more precise control of diabetes to avoid exacerbation of the disease. Effective management of blood glucose, blood pressure, and cholesterol levels is crucial for decelerating the advancement of diabetic retinopathy.

Stage 2: Moderate NPDR

The second stage of the DR, known as Moderate Non-Proliferative Diabetic Retinopathy (NPDR), is characterized by the enlargement of blood veins that limit the blood flow to the retina. In this stage, significant damage is caused to blood veins, which constricts the flow of blood to the retina, thereby restricting access to necessary nutrients. In contrast to mild NPDR, which is characterized by the existence of a small number of microaneurysms, moderate NPDR entails the presence of several microaneurysms and a progressive increase in blood vessel leakage. During this phase, the excessive buildup of fluid and blood in the macula might result in more prominent symptoms, such as impaired vision. An individual with intermediate NPDR is more susceptible to illness advancement. According to a retrospective study conducted in 2020, there is a 17.6% probability that an individual with mild non-proliferative diabetic retinopathy (NPDR) may advance to severe NPDR or PDR within five years after being diagnosed.

Stage 3: Severe NPDR

In the third stage of DR, known as NPDR, the retinal blood vessels undergo more substantial damage. Retina loses access to nutrients as the blood flow is severely hampered. Unlike in the mild NPDR, characterized by a small number of microaneurysms, moderate NPDR experiences the presence of several microaneurysms and a progressive increase in blood vessel leakage. During this phase, the excessive buildup of fluid and blood in the macula might result in more prominent symptoms, such as impaired vision. An individual with intermediate NPDR is more susceptible to illness advancement. According to a retrospective study conducted in 2020, there is a 17.6% probability that an individual NPDR may advance to severe NPDR or Proliferative Diabetic Retinopathy (PDR) within five years after being diagnosed. As the illness progresses, severe NPDR leads to blockages in more significant segments of blood vessels in the retina. This results in a substantial reduction in blood circulation in this region. A physician may detect severe NPDR caused by intraregional microvascular abnormalities (IRMA). Angiogenesis refers to the atypical branching and dilatation of pre-existing blood vessels. The blood vessels in question are remarkably narrow and delicate, which can manifest prominent symptoms, including impaired eyesight, dark spots, and areas of limited vision. A clinician can classify severe non-proliferative diabetic retinopathy (NPDR) using the 4-2-1 criteria, reflecting Hemorrhages are observed in all four retinal quadrants or Venous beading is seen in two or more quadrants or Intense retinal migratory avascular necrosis presents in at least one quadrant

Stage 4: PDR

The last stage and the most advanced form of diabetic retinopathy is called PDR. Vascular problems at this point have led to the retina's oxygen deprivation and the growth of new, fragile blood vessels. The retina and the vitreous, the gelatinous substance that fills the eye's posterior cavity, are where these blood vessels form. Visual impairment may result from these newly developed blood vessels, often leaking blood into the virtue. Additionally, these recently created blood vessels may produce scar tissue, which could result in macula problems or retinal detachment downstream. This can also damage the optic nerve and increase intraocular pressure. People may experience serious consequences at this stage and require medical assistance to stabilize their visual acuity. Failure to treat PDR can result in severe vision impairment and complete loss of eyesight. Diabetic macula edema refers to the swelling of the macula, the central region of the retina responsible for providing color vision and fine detail perception. Diabetic macula edema is a possible consequence of retinopathy, characterized by fluid buildup due to leaky blood vessels. This fluid induces macula swelling, leading to decreased visual acuity. Glaucoma is a medical disorder characterized by increased fluid pressure within the eye, which can harm the optic nerve and result in the progressive loss of eyesight. Diabetes can harm the ocular blood vessels, leading to elevated intraocular pressure. Consequently, individuals with diabetes have a higher susceptibility than the general population to the development of glaucoma. A cataract is a spherical abnormality in the lens layer of the eye. In older people, cataracts often develop due to the degradation of proteins in the eye. Figure 1 shows the different classes of diabetic retinopathy.

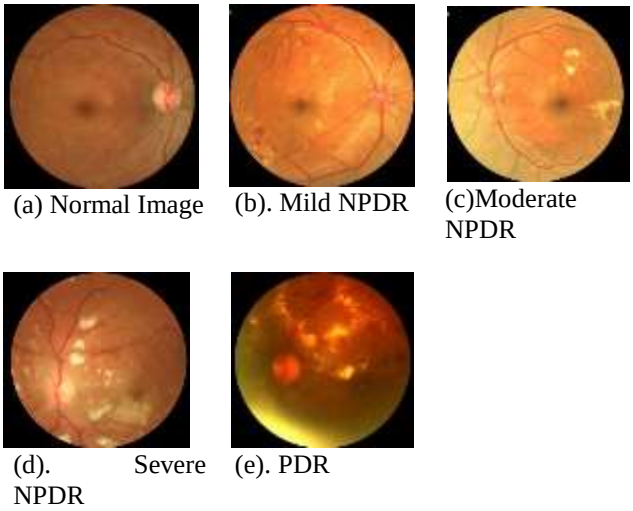

Figure 1. *Different Classes of Diabetic Retinopathy.*

Distinct differences can be noticed when a normal fundus image and a DR image are compared, as shown in Figure 2. Veins, arteries, the optic disc, macula, and fovea appear clear, healthy, and uniformly distributed with no visible damage in the normal funds. But pathological features such as aneurysms (small dark spots), haemorrhages (bleeding areas), cotton wool spots (white fluffy patches indicating nerve fibre damage), hard exudates (yellow lipid deposits), and abnormal blood vessel growth associated with proliferative DR. are observed in the DR-affected fundus, these abnormalities highlight the damage caused by DR that can lead to vision impairment if not diagnosed and treated promptly.

Detecting the onset of DR is crucial to preserve the patient's visual acuity. It has been established by numerous studies in this field that early detection

can prevent 90% of diabetic patients from developing diabetic retinopathy [11]. The diagnosis of DR can be performed either by a human ophthalmologist or by an automated methodology, with inherent advantages and limitations. The expertise of the ophthalmologist is the only advantage of the human intervention model. However, the skill and expertise of the ophthalmologist face limitations when the trace of the onset of DR is subtle. AI, with all its advancements, offers an alternative solution in early disease identification. The accuracy of an AI-driven automated method is higher and offers greater advantages than the manual DR detection approach. The AI-driven method significantly lowers the likelihood of human error because it reduces the load on the ophthalmologist. Furthermore, compared to manual testing, an automated system can detect lesions and anomalies far more easily and effectively. Therefore, it is essential to automate the identification of diabetic retinopathy. Both machine learning and deep learning approaches can be used to create DR-automated systems. Gathering retinal images showing signs of DR is the initial stage ML techniques. These photos then go through a number of picture pre-processing steps.

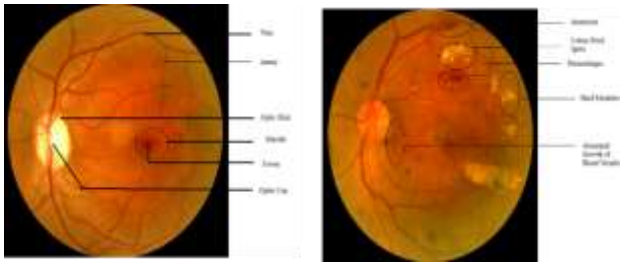

Figure 2. *Difference between the normal fundus image and diabetic retinopathy image.*

3. Related work

A modified ResNet-50 network with pre-processing methods was used by Lin CL and Wu KC to improve the detection of diabetic retinopathy (DR) in retinal fundus images. The approach addressed overfitting and loss fluctuation by incorporating structural changes to ResNet-50, such as regularization and adaptive learning rates, as well as a standard operating procedure (SOP) for image pre-processing. The model achieved a training accuracy of 83.95% and a test accuracy of 74.32%, outperforming well-known CNNs like Xception and VggNet-16. [12] The study's main flaw is its narrow focus on getting the best detection accuracy for DR. The strategy does not thoroughly examine sophisticated techniques to improve accuracy further or address problems like imbalanced datasets, which are common in DR detection, even though it places emphasis on preprocessing and visualization techniques to improve model calibration. Pratt et al. introduced a CNN-based method for detecting diabetic retinopathy from digital fundus images, highlighting the challenges of accurately assessing its severity due to its complexity and time-intensive nature. Their approach demonstrated a sensitivity of 95% and an accuracy of 75% on validation datasets, showcasing its potential for effective DR diagnosis [13].

A method developed by combining CNN with an IDX-DR device was used to detect and classify DR in fundus images by Abramoff et al. [14]. They employed data augmentation and integrated various CNNs with a Random Forest classifier to identify DR lesions and normal retinal anatomy on the Messidor-2 dataset with 1748 images. Images on the dataset were categorized into no DR, referring to DR, or vision-threatening DR. An AUC of 0.980, sensitivity of 96.8%, and specificity of 87.0% were achieved through this approach. However, classifying mild DR images as no DR and not considering all five DR stages are highlighted as limitations of the work.

The evaluated the performance of three pre-trained CNN architectures—VGG16 [15], [16], AlexNet [17], and Inception Net V3 [18]—for detecting the five stages of diabetic retinopathy (DR) using the Kaggle [19] dataset is presented. During pre-processing, images were resized to 224×224 pixels for VGG16, 227×227 pixels for AlexNet, and 299×299 pixels for InceptionNet V3. The dataset contained only 166 images, leading to an average accuracy of 50.03% for VGG16, 37.43% for AlexNet, and 63.23% for InceptionNet V3. The study was limited by the small dataset size, which restricted feature learning, inadequate pre-processing, and the use of a single dataset for evaluation. A CNN-TL DenseNet121 model for diabetic retinopathy detection, trained on 3050 images is presented in [20]. The model was fine-tuned and tested against various architectures, including Inception V1, Inception V2, Inception V3, Xception, VGG16, ResNet-50, DenseNet, and AlexNet. Their approach achieved a validation accuracy of 84.10%, demonstrating its effectiveness in detecting DR.

A weakly-supervised framework for diabetic retinopathy detection using DRNet and CNN is presented in [33]. The methodology included image augmentation, resizing, Gaussian distribution, and Euclidean distance techniques. Tested on the IDRiD dataset, the model achieved an accuracy of 84.50%, demonstrating its effectiveness in retinal image analysis for DR detection. The authors in

[34] have proposed a diabetic retinopathy detection approach combining improved Grey Wolf optimization for feature selection with CNN classification. Techniques such as augmentation, image transformation, and contrast enhancement were applied. Tested on datasets Messidor-2, e-Ophtha, and EyePACS, the method achieved specificities of 87%, 94%, and 98%, respectively.

3. Proposed Method

This work can be broken down into six phases: 1) merging three datasets, 2) preparing the fundus images, 3) applying three kinds of augmentation methods, 4) developing a foundational model, 5) conducting model optimization, and 6) analyzing performance and results. The workflow of this research is illustrated in Figure 3. This section provides a detailed explanation of each of the steps.

1. Dataset creation

Many datasets used for DR detection suffer from data imbalance, where images are unevenly distributed across classes. For example, Ghosh et al. [10] utilized the Kaggle dataset to classify DR into stages. The dataset includes 35,126 training images and 53,576 test images, but the training set is heavily imbalanced: 73.5% of images are labeled as class 0, while only 6.9%, 15.1%, 2.5%, and 2.0% belong to classes 1, 2, 3, and 4, respectively.

Information in Table 1 and 2 conclude that the images in both two datasets are not appropriately balanced. Therefore, to overcome the imbalanced distribution in the DR images we formed the new dataset by combining the images from the Indian Diabetic Retinopathy Image Dataset (IDRiD), Kaggle's Diabetic Retinopathy dataset, and in house images collected from American Eye Hospital Hyderabad and shown in the table 3.

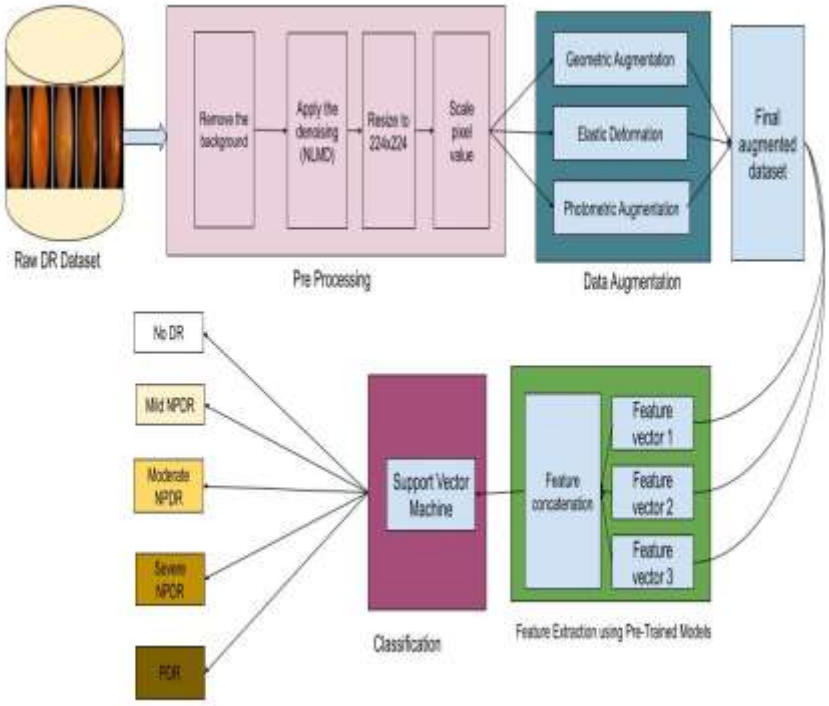

Figure 3. *Block diagram of the proposed method.*

Table 1. *Distribution of DR images in Kaggle dataset.*

Class Label	No. of Images	Percentage
0-No DR	25,810	73.50%
1-Mild NPDR	2443	6.90%
2-Moderate NPDR	5292	15.10%
3-Severe NPDR	873	2.50%
4-Proliferative DR	708	2.00%

The in-house dataset initially exhibited class imbalance, with the majority class (No DR) containing 2805 images, while severe and proliferative DR had only 383 and 585 images, respectively. To address this imbalance, the Synthetic Minority Oversampling Technique (SMOTE) [22] was applied. SMOTE generates synthetic samples for minority classes by interpolating between existing instances, avoiding simple duplication of images. By creating synthetic samples along the feature space, SMOTE not only increases the number of minority class images but

also enhances diversity, leading to a more robust training dataset. After applying SMOTE, the dataset became balanced, as shown in the table below, which significantly improved model performance. This balanced dataset ensures a fair representation of all classes, leading to improved model generalizability and reliability.

Table 2. *Distribution of DR images in IDRiD dataset.*

Class Label	No. of Images	Percentage
0-No DR	129	28.35%
1-Mild NPDR	22	4.84%
2-Moderate NPDR	156	34.29%
3-Severe NPDR	84	18.46%
4-Proliferative DR	64	14.07%

Table 3: *Inhouse dataset by combine the data Indian Diabetic Retinopathy Image Dataset (IDRiD), Kaggle's Diabetic Retinopathy dataset, and images from American Eye Hospital Hyderabad.*

Class Label	No. of Images
0-No DR	2805
1-Mild NPDR	740
2-Moderate NPDR	1899
3-Severe NPDR	383
4-Proliferative DR	585

Table 4. *In-house dataset after augmentation.*

Class Label	No. of Images	Final Images After SMOTE
0-No DR	2805	2805
1-Mild NPDR	740	1899
2-Moderate NPDR	1899	1899
3-Severe NPDR	383	1899
4-Proliferative DR	585	1899

2. Pre-Processing

Preparing images for neural network input is a crucial step to achieving optimal accuracy. This involves tasks such as removing artefacts, eliminating noise, and enhancing features that may be unclear but significant. The precise and timely detection of diabetic retinopathy (DR) depends on high-quality retinal images. However, publicly available retinal fundus datasets often vary in resolution and compression formats, frequently containing background noise. DR classification becomes difficult without effective pre-processing, as neural networks generally perform better with clean, enhanced, and well-processed images. In this study, we first remove the background and denoise the images using the Non-Local Means denoising method. Subsequently, resizing and pixel scaling are performed, as detailed in the following section.

A. Background Removal:

Background removal is an essential step in image pre-processing, and it can be implemented using a binary mask. This process involves applying the mask M(x,y) to the image I(x,y) to retain only the foreground pixels while eliminating the background. Mathematically, it is represented as (1).

$$I_{bg}(x,y) = I(x,y) \cdot M \tag{1}$$

In (1), I(x,y) is the pixel value of the input image,M(x,y) is the binary mask (1 for foreground and 0 for background).

B. Denoising Using NLMD (Non-Local Means Denoising):

Denoising retinal fundus images are essential, especially to enhance image quality while preserving critical features such as lesions and exudates necessary for accurate classification [23]. The Non-Local Means Denoising (NLMD) technique [24] is employed to effectively reduce noise without compromising these vital features. Non-Local Means Denoising (NLMD) reduces noise by computing the weighted average of similar pixels within the neighbourhood Ω of a pixel (x,y). The denoised pixel value is calculated using:

$$I_{denoised}(x,y) = \frac{\sum_{p\in\Omega} w\,(p,q)\cdot I(p)}{\sum_{p\in\Omega} w\,(p,q)} \tag{2}$$

In (2), $w(p,q)$ is the weight measuring the similarity between pixel p and q, given by:

$$w(p,q) = \exp\left(-\frac{\| I(p) - I(q) \|^2}{h^2}\right) \tag{3}$$

In (3), Ω is the neighbourhood of pixel (x,y) and h is the filtering parameter.

C. Resizing:

Once the image pre-processing is completed, it is important to standardize the image size before inputting them into the neural network to minimize the model's computational complexity. For this purpose, the processed images are resized to dimensions of 224×224. After these steps, the resulting images are free from artifacts and noise. As shown in Fig. 10(A), the original image includes irrelevant black background regions, classified as artifacts, and the features of the fundus images appear unclear. The preprocessing effectively enhances image quality, making critical features more distinguishable for analysis. Resizing can be represented mathematically using interpolation (4).

$$I_{resized}(x', y') = I\left(\frac{x \cdot W_{orig}}{W_{new}}, \frac{y \cdot H_{orig}}{H_{new}}\right) \quad (4)$$

In (4), W_{orig}, H_{orig} are the original width and height and are the resized dimensions.

Pixel Scaling:

Pixel scaling is a crucial pre-processing step in image-based tasks, normalizing pixel values to a consistent range, typically [0,1]. This transformation improves neural network performance by preventing the dominance of large intensity values and ensuring equal weight for all features. It accelerates model convergence by stabilizing gradients during backpropagation and reduces computational errors caused by floating-point precision issues. Pixel scaling also ensures compatibility with pre-trained models, which are often trained on normalized data. By enhancing numerical stability and training efficiency, pixel scaling plays a vital role in achieving accurate and reliable results in image processing and machine learning tasks. Normalization of pixel values to [0,1] range is given by (5).

$$I_{scaled}(x, y) = \frac{I(x,y) - \min(I)}{\max(I) - \min(I)} \quad (5)$$

3. Data Augmentation

Data augmentation [25] played a pivotal role in this study to address the challenges of dataset imbalance and enhance data diversity. While numerous techniques exist to prevent overfitting, augmentation is a core strategy as it allows the generation of additional training samples from limited data. Augmentation-based oversampling techniques effectively expand the dataset and mitigate the risk of overfitting by introducing variations that closely resemble real-world data [26]. Ensuring efficient data generation is critical, especially in the medical domain, where maintaining the quality of images and preserving critical features is essential [27, 30].

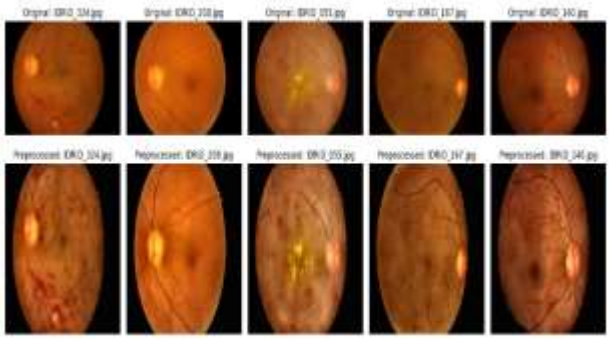

Figure 4. *Prepressed images.*

Improper augmentation can compromise the integrity of medical data, which is why care was taken to generate realistic samples that accurately represent all potential variations. This study applied three augmentation methods: geometric transformations, elastic deformations and photometric adjustments. These techniques created varied and representative augmented samples while maintaining the same number of images across all methods. By employing these approaches, the study ensured that the dataset was not only balanced but also enriched with high-quality and diverse samples, crucial for improving the robustness and accuracy of the model.

A. Geometric Transformations (Rotation):

Rotation is a widely used technique in image augmentation [28], allowing images to be rotated at any desired angle based on specific requirements. This method ensures that the essential information in the image remains unchanged, regardless of its orientation. Rotation helps create diverse perspectives of the same image, improving the model's robustness and reducing bias toward orientations. Equation (6) [29] gives the mathematical representation for the rotation technique, which defines how pixel positions are transformed during rotation. This approach is especially beneficial in tasks requiring invariance to object orientation, such as medical image analysis and classification. For rotation by an angle θ is given by (6).

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} \quad (6)$$

B. Elastic Deformation:

When forces are applied to a continuous body, they create a stress field that causes deformation. If the original shape is restored after the removal of the stress field, the deformation is classified as elastic [19]. This concept is utilized to generate deformed retinal fundus images, where the image appears stretched but retains all its critical information. Elastic deformation is particularly useful for augmenting medical images, as it preserves essential features required for accurate analysis, as shown in the equation (7). Figure 11 demonstrates a deformed version of the pre-processed retinal fundus image, showing that, despite the deformation, all significant features remain intact and clearly visible.

Elastic deformation applies displacement fields Δ_x (x,y) and Δ_y (x,y) to pixel coordinates:

$$I_{deformed}(x,y) = I\left(x + \Delta_x(x,y), y + \Delta_y(x,y)\right) \quad (7)$$

C. Photometric Augmentation:

Photometric augmentation involves modifying the RGB channels of an image by transforming each pixel's values (x,y,z) into new values (x′,y′) based on predefined heuristics. This technique alters the image's color and lighting while maintaining its geometric structure [31], [32]. Common methods include color jittering, grayscale conversion, filtering, light perturbation, noise addition, vignetting, contrast adjustment, and random erasing [33]. While increasing dataset diversity is essential, it must be done carefully to preserve critical pixel information and avoid overfitting. In this study, several photometric methods were explored, including histogram equalization (HE), saturation, Gaussian noise, hue adjustment, brightness, contrast, color, and sharpness modifications. Among these, adjusting brightness and contrast yielded the best results and were selected as the photometric augmentation techniques used in this research and shown in the equation (8).

Adjust brightness β and contrast α:

$$I_{augmented}(x,y) = \alpha \cdot I(x,y) + \beta \quad (8)$$

To improve diversity as well as robustness of the model, the dataset was thoroughly augmented. A 20° rotation was performed as part of Geometric Transformations in order to simulate changes in orientation. Realistic distortions were introduced to the images by Elastic Deformation with α=40 and σ=6 for the morphological changes that are expected in the images. Through photometric Augmentation we adjusted brightness and contrast within ±0.2 to match different bright and dark lighting environments common in real world. An additional variation was added by applying a Zoom Transformation of a zoom factor of 1.3, focusing on various parts of the images and simulating different observation scales. Collectively, these augmentations produced enriched datasets for (more) feature learning and model generalization, as shown in figure 5.

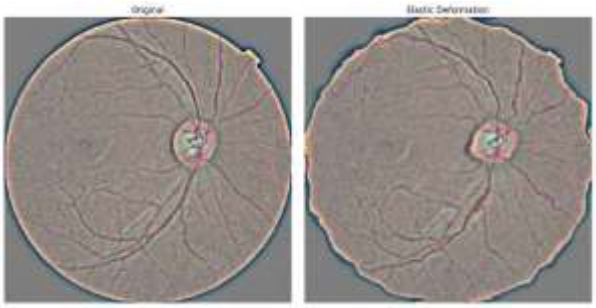

Figure 5. *Elastic Deformation.*

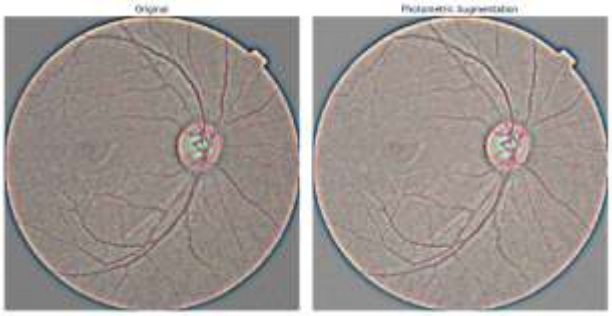

Figure 6. *Photometric Augmentation.*

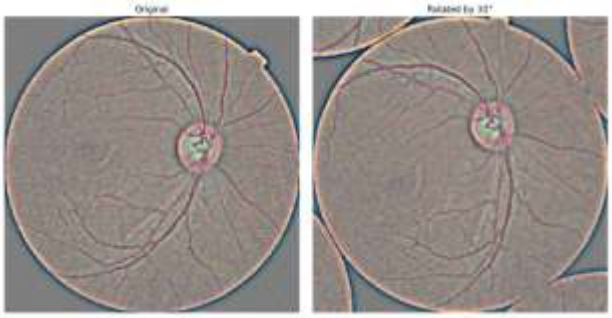

Figure 7. *Geometric Transformations (Rotation).*

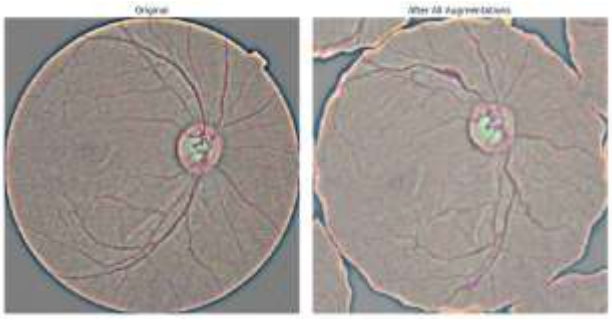

Figure 8. *Final Augmented image.*

4. Feature Extraction Using Pre-Trained Models

A. CNN Feature Extraction:

The pre-trained models used for feature extraction include EfficientNetV2S, EfficientNetB0, ResNet50, DenseNet121, MobileNetV2, InceptionV3, Xception, VGG16, VGG19, and

NASNetMobile. All models were optimized using the Adam optimizer. The models were trained with varying numbers of epochs and a consistent batch size of 32. Specifically, EfficientNetV2S and Xception were trained for 7 epochs, EfficientNetB0, DenseNet121, InceptionV3, VGG16, and VGG19 for 10 epochs, ResNet50 for 5 epochs, MobileNetV2 for 4 epochs, and NASNetMobile for 6 epochs. These configurations were chosen to balance computational efficiency and model performance for effective feature extraction and are shown in equation (9).

$$F_l = \sigma(W_l * I + b_l) \quad (9)$$

In (9), * is the convolution operator, W_l and b_l are the weights and biases of layer l. σ is the activation function (e.g., ReLU). F_l is the feature map.

B. Feature Concatenation:

Concatenating feature vectors from multiple pre-trained models enhances performance by combining diverse representations. Each model, such as EfficientNet, ResNet, and VGG, is designed with unique architectures that extract different aspects of the input data. By aggregating their feature vectors, a richer and more comprehensive representation is created. This approach leverages the complementary strengths of different models, capturing both low-level details (e.g., textures, edges) and high-level semantic features (e.g., shapes, patterns), which are crucial for complex tasks like medical imaging. Moreover, concatenation reduces reliance on a single model's limitations, improving robustness and generalizability across datasets. It preserves complementary information that individual models may miss, thus increasing the discriminative power of the combined feature space. This method is particularly beneficial for challenging tasks requiring fine-grained classification or detailed analysis, as the unified feature vector provides a robust foundation for downstream tasks, boosting accuracy and reducing the risk of overfitting. As shown in equation (10) Concatenate n feature vectors F_1, F_2,…,F_n into a single vector (10).

$$F_{concat} = [F_1; F_2; \ldots; F_n] \quad (10)$$

5. Classification Using SVM

The strengths of multiple pre-trained networks can be harnessed by combining their extracted features for DR classification. This method involves concatenating the feature vectors from various pre-trained models to create a comprehensive representation. The combined feature set is then fed into an SVM classifier, which processes these features to predict the final output. By leveraging the complementary capabilities of different models, the approach ensures that critical and diverse features are utilized, enhancing classification accuracy. The SVM classifier, trained on the concatenated features, predicts the DR classes, including No DR, Mild, Moderate, Severe, and PDR (Proliferative Diabetic Retinopathy). This methodology ensures that the input fundus image is classified accurately into one of the categories, providing a robust framework for automated DR detection and analysis. This combination of feature extraction and SVM classification demonstrates a powerful pipeline for improving the diagnostic accuracy of DR classification systems.

A. SVM Objective Function (Primal Form):

The Support Vector Machine (SVM) classifier used in this framework optimizes the following objective function (11):

$$\min_{w,b,\xi} \frac{1}{2} \| w \|^2 + C \quad (11)$$

Subject to

$$\left(w^T F_{concat}^{(i)} + b\right) \geq 1 - \xi_i, \quad \xi_i \geq 0 \quad (12)$$

Where w is the weight vector,b is the bias,ξ_i is the slack variable for misclassification,C is the regularization parameter, and y_i∈{-1,1} are the labels.

Decision Boundary:

For classification, the Support Vector Machine (SVM) predicts the class label of an input using the decision function (13):

$$f(x) = \text{sgn}\left(w^T F_{concat} + b\right) \quad (13)$$

Where f(x): The predicted class label for the input, sgn is the sign function, which outputs +1 for positive values and −1 for negative values, w is the weight vector, representing the orientation of the hyperplane, F_concat is the concatenated feature vector from multiple pre-trained models and b is the bias term, determining the offset of the hyperplane. This function evaluates the position of the input feature vector relative to the decision boundary defined by the hyperplane. If f(x)>0, the input is classified into one class (e.g., DR), and if f(x)<0, it belongs to the other class (e.g., No DR). This approach ensures accurate and robust classification.

B. Kernel Function:

In non-linear SVMs, a kernel function K*(F_i,,F_j)* maps input features into a higher-dimensional space, enabling the SVM to separate classes that are not linearly separable. The kernel function is defined as (14).

$$K(F_i, F_j) = \phi(F_i)^T \phi(F_j) \quad (14)$$

Common kernels include Linear: $K(F_i, F_j) = F_i^T F_j$, Polynomial: $K(F_i, F_j) = (F_i^T F_j + c)^d$ and RBF (Gaussian): $K(F_i, F_j) = \exp\left(-\frac{\|F_i - F_j\|^2}{2\sigma^2}\right)$.

4. Results and Discussions:

4. 1. Implementation

Google Colab with a Python 3 runtime and a v5e-1 TPU hardware accelerator was utilized to simulate the categorization models. Three sections comprised the results section. While the results for DR classification using a classifier with a pre-trained network are shown in Section A the results for different pre-trained networks are shown in Section B The use of hybrid networks with ML classifiers at the outer layer for DR classification is covered in Section C.

A. The performance of various pre-trained models for diabetic retinopathy classification

As indicated in table 5, the outcomes were assessed with consistent hyperparameters, such as the Adam optimizer, a batch size of 32, and a learning rate of 0.0001. With the best accuracy (0.892), precision (0.83), recall (0.841), and F1-Score (0.829) among the models, Xception was the clear winner and a solid option for this task. With F1-Scores of 0.827and 0.823, respectively, DenseNet121 and InceptionV3 also showed strong performance, demonstrating their capacity to extract and classify features.

But, ResNet50 and MobileNetV2 exhibited significantly lower accuracy and F1-Scores, indicating limited effectiveness for this dataset. The results highlight the advantages of using high-performing architectures like DenseNet121 and Xception. These methods exhibited remarkable robustness in feature representation and model accuracy. These findings suggest Xception as the most suitable model, while DenseNet121and EfficientNetV2S as promising alternatives for further optimization.

Table 5. *Comparative Performance of Pre-Trained Models on in-house Diabetic Retinopathy Dataset.*

Model	Accuracy	Precision	Sensitivity (Recall)	F1 Score
EfficientNetV2S	0.879	0.829	0.778	0.797
EfficientNetB0	0.827	0.763	0.742	0.751
ResNet50	0.336	0.248	0.326	0.224
DenseNet121	0.888	0.817	0.830	0.827
MobileNetV2	0.441	0.396	0.207	0.138
InceptionV3	0.887	0.816	0.833	0.823
Xception	0.892	0.83	0.841	0.829
VGG16	0.804	0.739	0.581	0.599
VGG19	0.800	0.716	0.581	0.592
NASNetMobile	0.606	0.314	0.365	0.330

B. Results for classification of DR images using pre-trained networks and ML classifiers

Table 6 to table 15 evaluates the performance of ten pre-trained deep learning models paired with machine learning classifiers across multiple metrics: Accuracy, Precision, Sensitivity (Recall), and F1-Score. The pre-trained models include EfficientNetV2S, EfficientNetB0, ResNet50, DenseNet121, MobileNetV2, InceptionV3, Xception, VGG16, VGG19, and NASNetMobile.

With an accuracy of 0.871102, precision of 0.836623, and F1-Score of 0.782577, Xception with XGBoost outperformed the other combinations, demonstrating its capacity to capture intricate information and provide reliable categorization. With an accuracy of 0.856549 and an F1-Score of 0.767784, DenseNet121 with XGBoost likewise demonstrated remarkable performance, demonstrating its potent feature extraction capabilities. In the majority of networks, XGBoost continuously beat other classifiers, demonstrating its proficiency in handling intricate feature spaces. Additionally, Random Forest and Bagging produced competitive outcomes, especially when combined with InceptionV3,

Xception, and DenseNet121. However, classifiers like SVM struggled with networks such as EfficientNetB0 and ResNet50, likely due to their limited capacity for handling non-linear relationships. AdaBoost showed lower performance compared to other classifiers, often underperforming in recall and F1-Score. Among the networks, DenseNet121, Xception, and InceptionV3 consistently outperformed others, highlighting their ability to extract meaningful features crucial for classification tasks. In contrast, ResNet50 and EfficientNetB0 displayed mixed results, suggesting a need for optimization or alternative classifiers. Overall, DenseNet121 and Xception paired with XGBoost emerged as the most effective combinations for DR classification, delivering high accuracy and robust results. These findings emphasize the importance of combining strong feature extractors with advanced classifiers to improve DR detection systems.

Table 6. *Performance Metrics of Machine Learning Classifiers Using EfficientNetV2S Pre-Trained Model.*

Class-ifier	Acc-uracy	Precision	Sensitivity	F1-Score
SVM	0.709979	0.785058	0.454572	0.436227
Random Forest	0.713098	0.57742	0.473579	0.482953
Logistic Regression	0.705821	0.631727	0.451739	0.446781
Decision Tree	0.60395	0.421492	0.421752	0.421445
KNN	0.694387	0.520598	0.471361	0.469663
XGBoost	0.703742	0.530823	0.470632	0.478552
Bagging	0.712058	0.571028	0.480628	0.489006
AdaBoost	0.665281	0.436327	0.403798	0.378398

Table 7. *Performance Metrics of Machine Learning Classifiers Using EfficientNetB0 Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity	F1-Score
SVM	0.43763	0.087526	0.2	0.121764
Random Forest	0.756757	0.669424	0.529622	0.548218
Logistic Regression	0.439709	0.159163	0.210698	0.162772
Decision Tree	0.655925	0.495532	0.5002	0.496999
KNN	0.695426	0.523623	0.473308	0.476314
XGBoost	0.766112	0.663054	0.562092	0.585483
Bagging	0.758836	0.644396	0.55048	0.566896
AdaBoost	0.679834	0.526074	0.398987	0.379849

Table 8 *Performance Metrics of Machine Learning Classifiers Using ResNet50 Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity (Recall)	F1-Score
SVM	0.673597	0.281771	0.368366	0.310791
Random Forest	0.758836	0.696026	0.532697	0.552617
Logistic Regression	0.682952	0.480082	0.40321	0.377047
Decision Tree	0.653846	0.494624	0.497411	0.495069
KNN	0.722453	0.585796	0.527992	0.526723
XGBoost	0.787942	0.723698	0.601598	0.630141
Bagging	0.759875	0.697848	0.55941	0.581448
AdaBoost	0.692308	0.531234	0.436166	0.421877

Table 9. Performance Metrics of Machine Learning Classifiers Using DenseNet121 Pre-Trained Model.

Classifier	Accuracy	Precision	Sensitivity	F1-Score
SVM	0.817048	0.72719	0.698485	0.708112
Random Forest	0.808732	0.778227	0.621774	0.653073
Logistic Regression	0.81185	0.714624	0.68127	0.695396
Decision Tree	0.751559	0.623244	0.607948	0.614441

KNN	0.796258	0.69981	0.623724	0.639344
XGBoost	0.856549	0.82122	0.741098	0.767784
Bagging	0.807692	0.755679	0.631019	0.658686
AdaBoost	0.704782	0.497216	0.447731	0.440925

Table 10. *Performance Metrics of Machine Learning Classifiers Using MobileNetV2 Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity (Recall)	F1-Score
SVM	0.819127	0.718805	0.706868	0.709055
Random Forest	0.777547	0.722683	0.55464	0.579449
Logistic Regression	0.801455	0.693751	0.670411	0.679307
Decision Tree	0.686071	0.536254	0.549865	0.539605
KNN	0.787942	0.701236	0.61131	0.625482
XGBoost	0.831601	0.795583	0.664326	0.700846
Bagging	0.786902	0.717548	0.586956	0.61047
AdaBoost	0.697505	0.494012	0.466798	0.460719

Table 11*. Performance Metrics of Machine Learning Classifiers Using InceptionV3 Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity (Recall)	F1-Score
SVM	0.846154	0.770963	0.762911	0.7639
Random Forest	0.776507	0.782681	0.550447	0.586265
Logistic Regression	0.837838	0.764498	0.73423	0.747865
Decision Tree	0.637214	0.489683	0.493829	0.491095
KNN	0.739085	0.630377	0.543386	0.560559
XGBoost	0.83368	0.814906	0.67963	0.722775
Bagging	0.796258	0.790088	0.598157	0.64084
AdaBoost	0.641372	0.468453	0.411696	0.407499

Table 12. *Performance Metrics of Machine Learning Classifiers Using Xception Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity (Recall)	F1-Score
1	0.861746	0.774492	0.764011	0.766436
2	0.828482	0.816556	0.654163	0.694806
3	0.834719	0.730863	0.689512	0.7053
4	0.726611	0.589541	0.592148	0.590648
5	0.760915	0.612562	0.543719	0.553851
6	0.871102	0.836623	0.750708	0.782577
7	0.81185	0.75736	0.636355	0.665453
8	0.677755	0.504011	0.448132	0.442363

Table 13 *Performance Metrics of Machine Learning Classifiers Using VGG16 Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity (Recall)	F1-Score
SVM	0.744283	0.764121	0.498021	0.489816
Random Forest	0.818087	0.792944	0.64931	0.681999
Logistic Regression	0.746362	0.634977	0.509345	0.520961
Decision Tree	0.727651	0.586645	0.588036	0.586667
KNN	0.766112	0.661251	0.59862	0.609162
XGBoost	0.870062	0.822872	0.764068	0.785995
Bagging	0.81289	0.764053	0.640804	0.66912
AdaBoost	0.691268	0.552352	0.433573	0.429775

Table 14. *Performance Metrics of Machine Learning Classifiers Using VGG19 Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity (Recall)	F1-Score
SVM	0.740125	0.762281	0.495038	0.490333
Random Forest	0.800416	0.757117	0.606718	0.635681
Logistic Regression	0.74948	0.729165	0.520903	0.540834
Decision Tree	0.697505	0.548115	0.54779	0.547829
KNN	0.756757	0.678029	0.575961	0.593623
XGBoost	0.83368	0.781503	0.682753	0.713401
Bagging	0.799376	0.745551	0.62241	0.649872
AdaBoost	0.679834	0.515137	0.436628	0.439241

Table 15. *Performance Metrics of Machine Learning Classifiers Using NASNetMobile Pre-Trained Model.*

Classifier	Accuracy	Precision	Sensitivity (Recall)	F1-Score
SVM	0.820166	0.735867	0.713954	0.718274
Random Forest	0.797297	0.767603	0.591133	0.620573
Logistic Regression	0.804574	0.703658	0.666949	0.679842
Decision Tree	0.686071	0.513725	0.512233	0.512686
KNN	0.776507	0.670065	0.591569	0.603399
XGBoost	0.829522	0.771686	0.667106	0.69493
Bagging	0.805613	0.760873	0.61359	0.643334
AdaBoost	0.68711	0.500903	0.459207	0.452805

The images of Figure 9 to Figure 13 display comparative bar charts showcasing the accuracy of ten pre-trained models (EfficientNetV2S, EfficientNetB0, ResNet50, DenseNet121, MobileNetV2, InceptionV3, Xception, VGG16, VGG19, and NASNetMobile) paired with various machine learning classifiers. Classifiers include SVM, Random Forest, Logistic Regression, Decision Tree, KNN, XGBoost, Bagging, and AdaBoost. Each chart highlights the accuracy performance of classifiers for a specific pre-trained model. Ensemble methods such as XGBoost and Bagging generally show higher accuracy across models, while Decision Tree often underperforms. DenseNet121 and Xception with XGBoost are among the top-performing combinations, emphasizing the synergy between pre-trained models and advanced ensemble classifiers.

C. Results for DR classification using hybrid networks and SVM classifier

Ten different CNN models are employed in the feature extraction process. The extracted features were concatenated to form a comprehensive feature set. Multiple Machine Learning classifiers were used in evaluating and subsequently determining the best-performing model. The results are shown in table 16. Logistic Regression and XG Boost are achieved with an accuracy of 88% and a macro average of 81%. But the Voting Classifier and SVM outperformed other models, achieving an accuracy of 90.02%, with macro averages of 83% and 82%, respectively. SVM has emerged as the best method because of its superior performance; the performance can be attributed to its ability to handle high-dimensional data effectively and find an optimal hyperplane for classification.

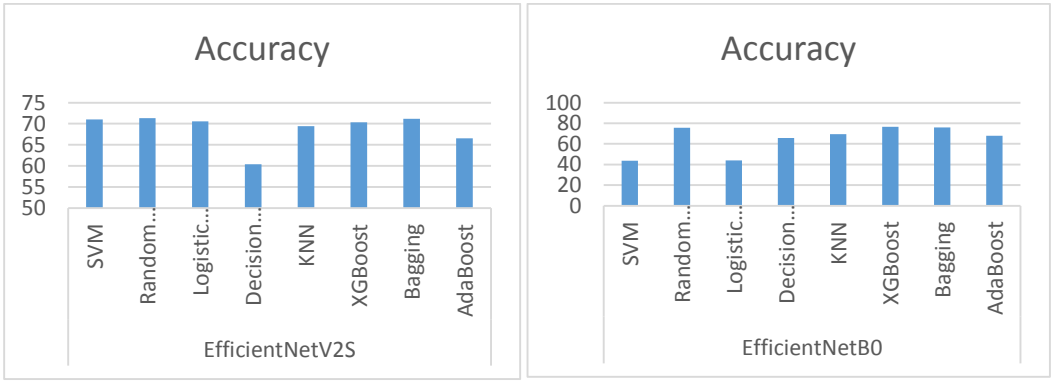

Figure 9. *Accuracy Comparison of Classification Models on EfficientNetV2S, EfficientNetB0*

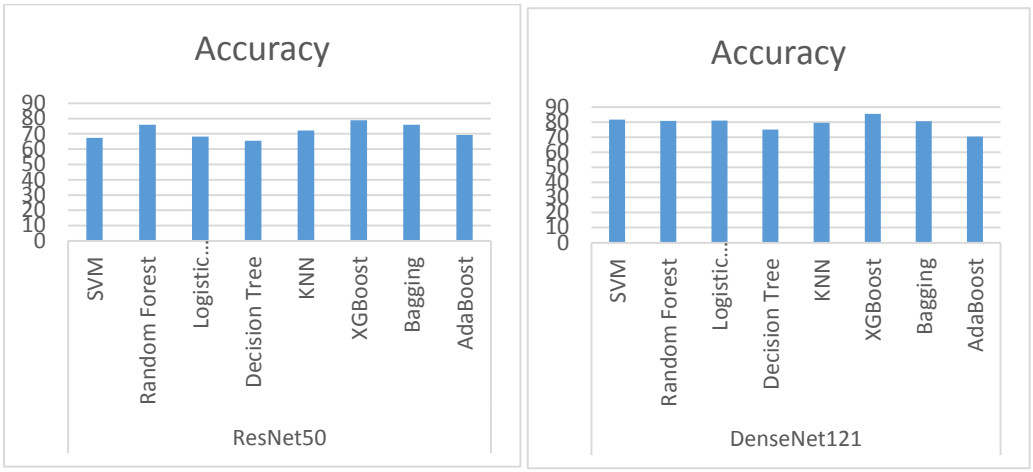

Figure 10. *Accuracy Comparison of Classification Models on ResNet50, DenseNet121*

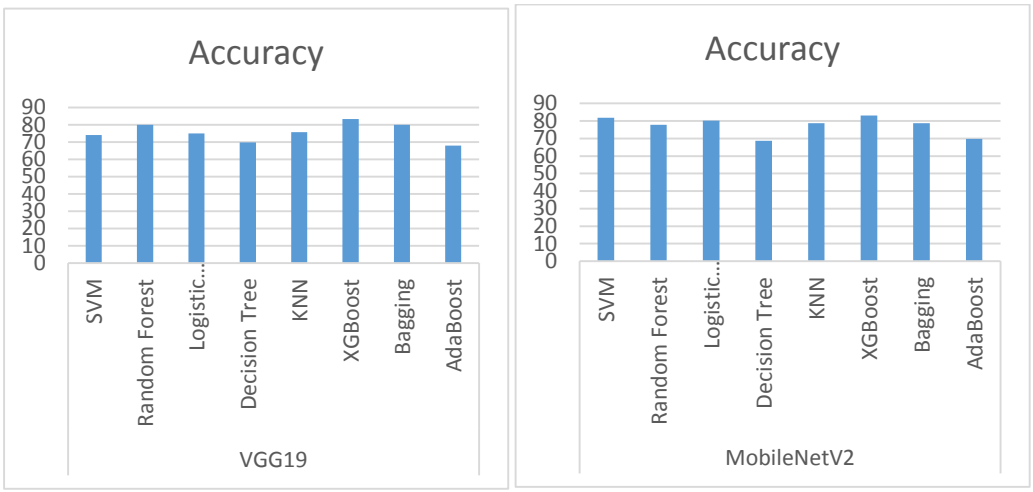

Figure 11. *Accuracy Comparison of Classification Models on VGG19, MobileNetV2.*

SVM demonstrated its robustness in working with complex, concatenated feature sets compared to Decision Tree and KNN, which exhibited lower accuracies (67% and 68%, respectively). These results emphasize SVM's reliability and effectiveness for accurate classification, making it a preferred choice in the ensemble of evaluated classifiers.

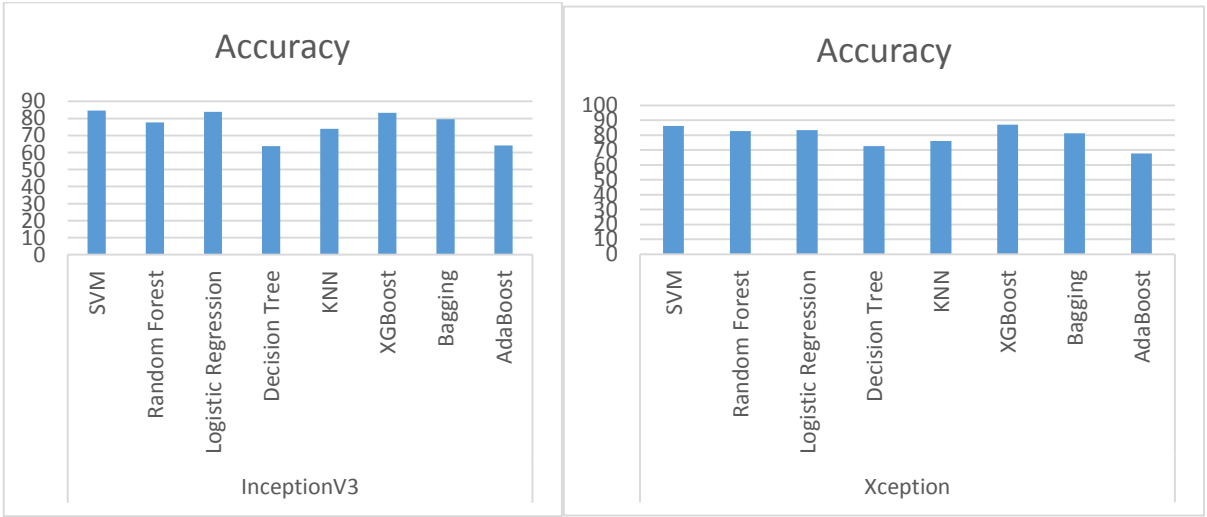

Figure 12. *Accuracy Comparison of Classification Models on InceptionV3, Xception.*

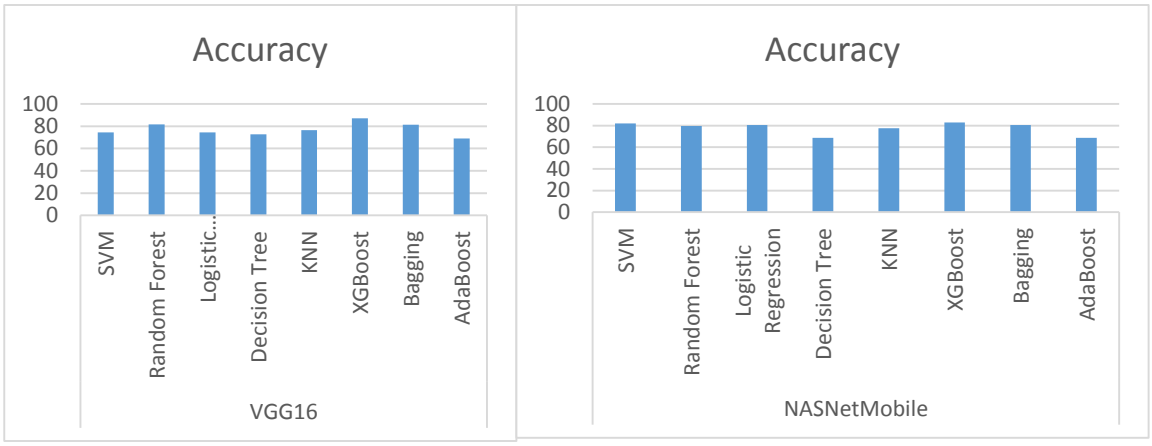

Figure 13. *Accuracy Comparison of Classification Models on VGG16, NASNetMobile.*

The table 17 compares the accuracy of various methods with the proposed hybrid method. Among the existing approaches, the highest accuracy is achieved by Method 34 with 87%, closely followed by Methods 20 and 33 with 84% and 84.5% accuracy, respectively. Method 15 records the lowest accuracy at 63.4%. The proposed hybrid method outperforms all others, achieving an impressive accuracy of 90.02%, indicating significant improvement and validating its robustness in addressing the problem. This highlights the effectiveness of the proposed hybrid approach in surpassing the limitations of traditional methods.

Table 16. *Performance Comparison of Machine Learning Classifiers Using Concatenated CNN Features.*

Method	Accuracy	Macro avg
Logistic Regression	88%	81
Decision Tree	67%	54
Random Forest	86%	77
Gradient Boosting	82%	71
KNN	68%	63
XGBoost	88%	81
Voting Classifier	90%	83
Stacking Classifier	86%	76
SVM	90.02%	82

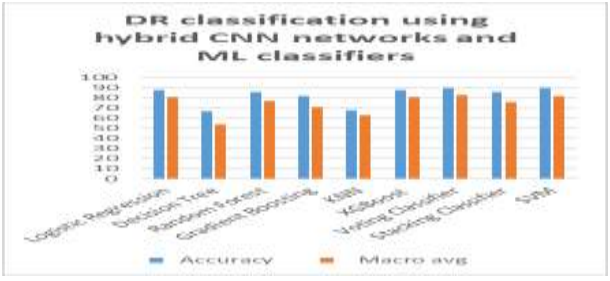

Figure 14. *Performance Comparison of Hybrid CNN and ML Classifiers for DR Classification.*

Table 17 *Accuracy Comparison of Existing Methods and Proposed Hybrid Method.*

Method	Accuracy
[12]	74.32%
[13]	75%
[20]	84%
[33]	84.5%
[34]	87%
[15]	63.4%
[14]	78%
Proposed hybrid method	90.02%

4. Conclusion

With the aim of providing new in-house data, this study analyzes DR photos from IDRid, Kaggle, and the American Eye Hospital in Hyderabad. Following it, a hybrid idea for the categorization of the DR images using support vector machines and pre-trained CNN algorithms is presented. Several pre-trained networks are concatenated to create a hybrid model. The SVM classifier, which had the maximum accuracy of 90.02%, is fed the features that were derived from the hybrid model. Future research can test the suggested hybrid networks using various classifiers, optimizers, batch sizes, and epochs. Before being fed into the networks, pre-processing steps can be added to the input pipeline.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could

have appeared to influence the work reported in this paper

- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] World Health Organization. (2024). *Diabetes prevalence to affect 783 million individuals by 2045*. https://www.who.int/diabetes

[2] Brar, A. S., Sahoo, J., Behera, U. C., Jonas, J. B., Sivaprasad, S., & Das, T. (2022). Prevalence of diabetic retinopathy in urban and rural India: A systematic review and meta-analysis. *Indian Journal of Ophthalmology*, *70*(6), 1945–1955. https://doi.org/10.4103/ijo.IJO_2206_21

[3] Gulshan, V., Peng, L., Coram, M., et al. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*, *316*(22), 2402–2410. https://doi.org/10.1001/jama.2016.17216

[4] Venugopal, V., Joseph, J., Vipin Das, M., & Kumar Nath, M. (2022). An EfficientNet-based modified sigmoid transform for enhancing dermatological macro-images of melanoma and nevi skin lesions. *Computer Methods and Programs in Biomedicine*, *222*, 106935. https://doi.org/10.1016/j.cmpb.2022.106935

[5] Brinker, T., Hekler, A., Utikal, J., Grabe, N., Schadendorf, D., Klode, J., Berking, C., Steeb, T., Enk, A., & Kalle, V. (2018). Skin cancer classification using convolutional neural networks: Systematic review. *Journal of Medical Internet Research*, *20*(10). https://doi.org/10.2196/11936

[6] Lameski, J., Jovanov, A., Zdravevski, E., Lameski, P., & Gievska, S. (2019). Skin lesion segmentation with deep learning. *Proceedings of the International Conference*, 1–5.

[7] Harangi, B. (2018). Skin lesion classification with ensembles of deep convolutional neural networks. *Journal of Biomedical Informatics*, *86*, 25–32. https://doi.org/10.1016/j.jbi.2018.08.006

[8] Zheng, L., Wang, Z., Liang, J., Luo, S., & Tian, S. (2021). Effective compression and classification of ECG arrhythmia by singular value decomposition. *Biomedical Engineering Advances*, *2*, 100013. https://doi.org/10.1016/j.bea.2021.100013

[9] Roy, T. S., Roy, J. K., & Mandal, N. (2022). Classifier identification using deep learning and machine learning algorithms for the detection of valvular heart diseases. *Biomedical Engineering Advances*, *3*, 100035. https://doi.org/10.1016/j.bea.2022.100035

[10] Vasilakos, C., Kavroudakis, D., & Georganta, A. (2020). Machine learning classification ensemble of multitemporal Sentinel-2 images: The case of a mixed Mediterranean ecosystem. *Remote Sensing*, *12*(12), 2005. https://doi.org/10.3390/rs12122005

[11] Tarasewicz, D., Karter, A. J., Pimentel, N., Moffet, H. H., Thai, K. K., Schlessinger, D., Sofrygin, O., & Melles, R. B. (2023). Development and validation of a diabetic retinopathy risk stratification algorithm. *Diabetes Care*, *46*(5), 1068–1075. https://doi.org/10.2337/dc22-2037

[12] Lin, C. L., & Wu, K. C. (2023). Development of revised ResNet-50 for diabetic retinopathy detection. *BMC Bioinformatics*, *24*(1), 157. https://doi.org/10.1186/s12859-023-05293-1

[13] Pratt, H., Coenen, F., Broadbent, D. M., Harding, S. P., & Zheng, Y. (2016). Convolutional neural networks for diabetic retinopathy. *Procedia Computer Science*, *90*, 200–205. https://doi.org/10.1016/j.procs.2016.07.014

[14] Abramoff, M. D., et al. (2016). Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning. *Investigative Ophthalmology & Visual Science*, *57*(13), 5200–5206.

[15] Wang, X., Lu, Y., Wang, Y., & Chen, W. B. (2018). Diabetic retinopathy stage classification using convolutional neural networks. In *International Conference on Information Reuse and Integration for Data Science* (pp. 465–471).

[16] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*.

[17] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

[18] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2818–2826). https://doi.org/10.1109/CVPR.2016.308

[19] Kaggle. (n.d.). *Diabetic retinopathy detection*. https://www.kaggle.com/c/diabetic-retinopathy-detection

[20] Samanta, A., Saha, A., Satapathy, S. C., Fernandes, S. L., & Zhang, Y. (2020). Automated detection of diabetic retinopathy using convolutional neural networks on a small dataset. *Pattern Recognition Letters*, *135*, 293–298. https://doi.org/10.1016/j.patrec.2020.04.026

[21] Ghosh, R., Ghosh, K., & Maitra, S. (2017). Automatic detection and classification of diabetic retinopathy stages using CNN. *arXiv preprint*, arXiv:1706.09640.

[22] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of*

Artificial Intelligence Research, 16, 321–357. https://doi.org/10.1613/jair.953

[23] Sathiya, G., & Gayathri, P. (2014). Automated detection of diabetic retinopathy using GLCM. *International Journal of Applied Engineering Research, 9*(22), 7019–7027.

[24] Buades, A., Coll, B., & Morel, J.-M. (2005). A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (Vol. 2, pp. 60–65). https://doi.org/10.1109/CVPR.2005.38

[25] Wong, S. C., Gatt, A., Stamatescu, V., & McDonnell, M. D. (2016). Understanding data augmentation for classification: When to warp? In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)* (pp. 1–6). https://doi.org/10.1109/DICTA.2016.7797091

[26] Nie, Y., Zamzam, A. S., & Brandt, A. (2021). Resampling and data augmentation for short-term PV output prediction based on an imbalanced sky images dataset using convolutional neural networks. *Solar Energy, 224*, 341–354. https://doi.org/10.1016/j.solener.2021.06.023

[27] Hussain, Z., Gimenez, F., Yi, D., & Rubin, D. (2017). Differential data augmentation techniques for medical imaging classification tasks. In *AMIA Annual Symposium Proceedings* (p. 979).

[28] Khosla, C., & Saini, B. S. (2020). Enhancing performance of deep learning models with different data augmentation techniques: A survey. In *International Conference on Intelligent Engineering and Management (ICIEM)* (pp. 79–85). https://doi.org/10.1109/ICIEM48762.2020.9160170

[29] Khalifa, N. E., Loey, M., & Mirjalili, S. (2021). A comprehensive survey of recent trends in deep learning for digital image augmentation. *Artificial Intelligence Review, 55*, 1–27. https://doi.org/10.1007/s10462-021-09987-w

[30] Castro, E., Cardoso, J. S., & Pereira, J. C. (2017). Elastic deformations for data augmentation in breast cancer mass detection. In *IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)* (pp. 479–482). https://doi.org/10.1109/BHI.2017.7897308

[31] Aazad, S. K., Saini, T., Ajad, A., Chaudhary, K., & Elsayed, E. E. (2024). Deciphering blood cells – Method for blood cell analysis using microscopic images. *Journal of Modern Technology, 1*(1), 9–18.

[32] Gayathri, L., Muralidhara, B. L., & Rajesh, B. (2025). Comparative evaluation of feature selection techniques and machine learning algorithms for Alzheimer's disease staging. *International Journal of Computational and Experimental Science and Engineering, 11*(2). https://doi.org/10.22399/ijcesen.1077

[33] Kalnoor, G., Dasari, K. S., Suma, S., Waddenkery, N., & Pragathi, B. (2025). Enhanced brain tumor detection from MRI scans using frequency domain features and hybrid machine learning models. *Journal of Modern Technology, 1*(2), 141–149.

[34] A, V., & Avanija, J. (2025). AI-driven heart disease prediction using machine learning and deep learning techniques. *International Journal of Computational and Experimental Science and Engineering, 11*(2). https://doi.org/10.22399/ijcesen.1669