INSPIRE HEALTH

Inspire Health Journal
Vol. 1, No. 1, 2026
https://doi.org/10.65718/inspireSS.2026.2005

InspireQuill.org
Inspire Academic Press Inc. (Canada)

Research Paper OPEN ACCESS

Optimized Deep Learning Framework for Diabetic Retinopathy Detection and Classification Using Fundus Imaging

Pooja Verma [1], Ghazanfar Latif [1, *], Jaspreet Kaur [1], Mohsin Butt [2]

[1] Department of Computing Science, Thompson Rivers University, Canada
[2] College of General Studies, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia
[*] Corresponding Author: glatif@tru.ca

Abstract

Diabetic Retinopathy (DR) is one of the primary causes of blindness all around the world, early and accurate detection to avoid loss of human vision is important. This research presents an optimized hybrid deep learning approach for DR detection from fundus images. Deep learning models such as Convolutional Neural Networks (CNNs) are proficient in capturing local features of an image while on the other hand, Vision Transformers (ViTs) excel at modeling global context, both have certain limitations in a standalone context. This research offers a novel hybrid feature fusion framework to overcome these individual limitations. We extracted deep feature vectors simultaneously from a pre-trained AlexNet (CNN) and a Swin Transformer model, then we combined them into a complete picture. The proposed strategy was rigorously tested on the public APTOS 2019 dataset, where it surpassed all baseline models significantly. The Random Forest classifier, when trained on the combined features, achieved a state-of-the-art accuracy of 98.2% and a huge refinement of 70% compared to the best individual deep learning model. Our study demonstrates that by combining CNN and ViT we can create a strong and reliable tool for detecting DR. Our approach presents an effective way to build automated systems/tools that could help the healthcare providers in finding out about the disease early on and saving the eyesight of their patients.

Keywords: Diabetic Retinopathy, Deep Learning, Convolutional Neural Network (CNN), Vision Transformer (ViT), Feature Fusion, Fundus Images.

1. Introduction

Diabetes Mellitus (DM) is long term a health condition that leads to high levels of sugar in the blood. The World Health Organization has issued an alert that the number of individuals with the condition of diabetes has been growing significantly worldwide at an alarming rate, especially in the countries that don't have sufficient resources [1]. Diabetic Retinopathy is one such serious complication of diabetes. In this condition tiny blood vessels in retina are damaged, the part of the light that is responsible for receiving light that is focused by the lens and then convert them into neural signals as shown in Figure 1. If this condition is not diagnosed and treated early, it can lead to permanent blindness, and it also plays a major role in loss of vision among the working-age adults [2].

Academic Editor:
Jaafar Alghazo

Received: 25/06/2025
Revised: 04/08/2025
Accepted: 11/09/2025
Published: 05/01/2026

Citation
Verma, P., Latif, G., Kaur, J., Butt, M. (2026). Diabetic Retinopathy Detection and Classification from Fundus Images using Optimized Deep Learning Models. *Inspire Health Journal, 1*(1), 47-62.

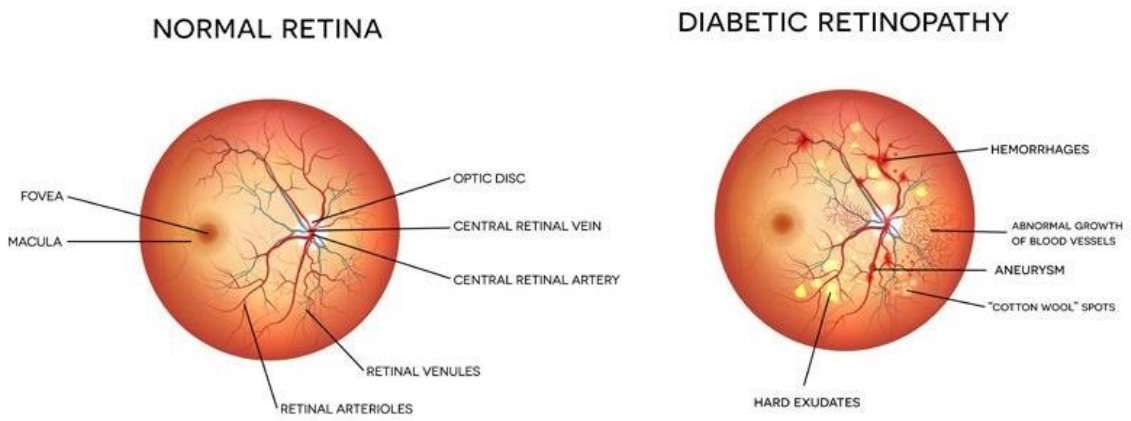

Figure 1. Lesions Indicating Signs of DR in the Retina [3].

One of the big issues with the DR is that there are no symptoms initially. By the time people notice that something is wrong with their vision, severe damage has been done already [4]. However, if the DR was detected in its early stages, adequate treatment would have prevented vision loss by a strong 95% [5]. This is why regular checkups hold utmost importance. The current standard to examine is color fundus photography done by an ophthalmologist [6]. Unfortunately, manually reading every picture is labor-intensive, time-consuming and expensive, as the growing need of resources and the number of diabetic patients is increasing worldwide [7]. This can lead to long waits for appointments and backlogs leading to delays in treatment and making the patient more prone to vision loss.

This is where Computer-Aided Diagnosis (CAD) systems have emerged to expand DR screening services. Modern CAD tools are dependent on artificial intelligence to automatically analyse and label the retinal lesions, presenting a second opinion. Deep learning, especially Convolutional Neural Network (CNN) is a major cause of this improvement thanks to its ability to learn significant visual patterns from the raw images [8-10]. The CNNs are great at seizing local texture but they are not as effective when it comes to modeling long-range relations among the images. In order to overcome this limitation of the CNNs, ViTs have been applied to medical imaging very recently. ViTs make use of a self-attention mechanism, which was earlier made to understand the global content [11]. Early research portrays that ViTs alone have the capability to surpass CNNs in tasks like DR grading, but their downside is that they usually need a very large amount of image training and extensive computing resources [12-13].

Since both CNNs and ViTs have different strengths when it comes to focusing on separate visual cues, we believe that merging them can result in more complete and robust description of an eye scan. In this paper, we have introduced a simple yet strong hybrid feature fusion approach that combines the strengths of both approaches. First, a pre-trained AlexNet (a classic CNN) and a pre-trained Swin Transformer (a hierarchical ViT) process each fundus photo in parallel, generating two complementary feature vectors. Second, we concatenate or “fuse” these vectors to form one comprehensive signature of the retina. Finally, we feed the fused features into lightweight, conventional classifiers: Support Vector Machine (SVM), Multi-Layer Perceptron (MLP), and Random Forest (RF). Extensive experiments on public DR datasets show that the fusion, when paired with an RF classifier, achieves an overall accuracy of 98.2 %, a notable boost over any single model trained end-to-end on the same data.

The remainder of this paper is organized as follows. Section 2 reviews related work on DR detection and hybrid deep learning methods. Section 3 explains the proposed feature-fusion pipeline in detail. Section 4 presents

experimental settings and the resultant quantitative results. Section 5 discusses the limitations and future extensions, and Section 6 concludes the study.

2. Literature Review

The research on detecting and grading diabetic retinopathy (DR) has grown a lot in recent years, shifting from old-school image processing to smart deep learning tools that can spot tiny changes in eye scans. Back in the day, experts manually tweak images to pull out key details like blood vessel shapes or spots on the retina, then feed those into basic machine learning models. These old ways worked okay but needed tons of handholding and didn't handle messy real-world photos well, plus they got bogged down on big piles of data. Now, with faster computers and huge collections of eye images, folks are leaning hard into convolutional neural networks (CNNs) and even fancier things like transformers. These let the models learn straight from raw pictures, spotting DR signs like leaks or swelling without as much fuss.

A bunch of studies have switched to transfer learning, where you start with a model trained on everyday photos and tweak it for eye scans, this saves time and boosts results on smaller datasets. For example, one team took a pre-trained SqueezeNet setup, slimmed down its features, and paired it with a hybrid weighted bidirectional LSTM to grade DR levels [14]. They tested it on the APTOS 2019 and MESSIDOR datasets, hitting higher accuracy than plain baselines, but the whole chain of steps makes it a bit clunky to run or copy in a busy clinic. Another group went for a team-up of EfficientNet backbones, voting together like a panel of experts, on a Kaggle set of 35,000 images [15]. They got a solid 86.5% accuracy on average, though they skimped on details about which DR stages trip it up most or how they fixed uneven data spread.

Lately, attention mechanisms and transformers are stealing the show because they help models zoom in on important spots like lesions while ignoring blurry edges. Take this masked autoencoder Vision Transformer that learns from unlabeled eye patches first, then fine-tunes for full grading [16]. On APTOS 2019, it nailed a 93.4% AUC for spotting serious DR, but it guzzles GPU power and doesn't explain much about what it's "seeing" in the images. A hybrid setup mixes ResNet for local details with a tiny ViT decoder that pays attention to the big picture, tested on EyePACS and Messidor-2 [17]. Sensitivity peaked at 95.7% with 0.96 AUC, yet they only checked a small slice of data, so who knows how it'd hold up in a real hospital mix.

Not everyone's going full transformer, some stick to lighter CNN tweaks for quicker checks. One approach feeds DenseNet maps into a bi-LSTM to track DR changes over time, pulling 97% F1 on APTOS 2019 [18]. But heavy data tricks might fake those high scores, and it skips head-to-heads with newer transformer rivals. Then there's a two-stream CNN that fuses info through gated attention for grading on EyePACS and Messidor-2, landing 94.3% accuracy [19]. It's promising, but the code's not out yet, and no real-time patient tests mean we can't say if it'd shine in the wild. Preprocessing fans add steps like contrast boosts or super-resolution GANs before a custom CNN, claiming 97.8% on APTOS [20]. Sounds great, but sticking to one dataset leaves generalization in question, and all that extra work could slow things down.

For tougher spots like low-quality shots from handheld cameras, some mix object detectors like YOLO with lightweight classifiers [21]. On EyePACS, it hit 98.4% accuracy, though chaining models risks error pile-up and longer waits. VGG-16 with hand-picked features fused in gets 94.8% on the same set [22], but blending old and new tricks adds hassle without huge payoffs. Dual paths for segmenting and grading lesions together, using MSGDA-Net, manage 87.2% grading on DDR and APTOS [23], but weak segmentation means it misses stuff, and scores tank on other datasets.

Interpretable models are a hot push too, like IDANet with dual attention and heatmaps for docs to trust [24]. AUC of 0.961 on EyePACS and Messidor, but 90 million parameters make it a beast for phones or remote spots. Multi-scale attention pyramids grab tiny and big lesions in one go, 92% F1 on DDR and APTOS [25], though tiny test sets scream overfitting. Priors from lesion maps guide a CNN-ViT fusion for segmentation and grading [26], upping mAP to 67% and 88% accuracy on DDR, but needing those maps upfront is a pain.

Cascades and ensembles keep popping up for reliability. A three-stage CNN chain, Triple-DRNet, sorts five DR classes on APTOS and EyePACS at 93.1% [27], but stages drag on speed. Five-CNN stack with a meta-layer hits 95.6% on APTOS and Messidor-2 [28], yet the bloat hurts deployment. Wavelets before CNN get 94.8% on APTOS [29], a small edge though. Older SVM on CNN features lags at 92% on EyePACS [30], feeling dated next to end-to-end setups. Beyond plain grading, some eye progression over time, like a DenseNet-LSTM for predicting shifts [18]. Or ensembles of Inception, ResNet, Xception stacked for 0.95 AUC on EyePACS [31], slow but steady. Gaussian denoising and crops into ResNet50 yield 91.7% [32], camera-specific though.

Transfer learning shines in tight data spots, like fine-tuned EfficientNet for early DR on APTOS and EyePACS over 90% [33]. Or ResNet50 with focal loss, 0.965 AUC on Indian sets [34], but binary only. Cross-modality GANs adapt wide-field to standard, kappa 0.87 on UWF [35]. VGG-16 for remote real-time, 97.6% on private handheld data [36]. Dual Inception-DenseNet branches, 98.5% binary on Kaggle [37]. 20-CNN vote, 96.4% on APTOS [38]. Datasets and benchmarks fuel this boom. New handheld mBRSET with 5k Brazilian shots benchmarks ConvNeXt at >0.93 AUC [39]. DeepDR Plus predicts 5-year risk, 0.79 C-index on huge cohorts [40]. Prospective trials like DR-DLS in Indigenous groups, 94% sensitivity [41]. LuxIA cloud tool, 90-99% sens/spec across sets [42]. Head-to-head of seven AIs, EyeArt leads to 0.936 AUC [43].

In conclusion, DR tools have leveled up from fiddly features to plug-and-play DL that catches early signs fast. Custom CNNs rule big data, transfers handle scraps, and attention/transformers add smarts for tricky lesions. But snags like single-dataset tests, heavy compute power, and missing real-clinic runs hold back trust. Ensembles and priors help, yet lighter, explainable models for mobiles could tip the scale. Future trajectories may prioritize lightweight, interpretable architectures for mobile deployment, extending DL's salience across pathologies such as oncology and pulmonology.

3. Methodology

This section provides a comprehensive description of the dataset, the experimental setup, and the multi-phase methodology employed in this study. Figure 2 shows the proposed architecture of the hybrid feature extraction and classification of fundus images. Our approach is divided into two primary phases: first, the implementation and evaluation of several baseline end-to-end deep learning models, and second, the development and testing of our proposed hybrid feature fusion framework.

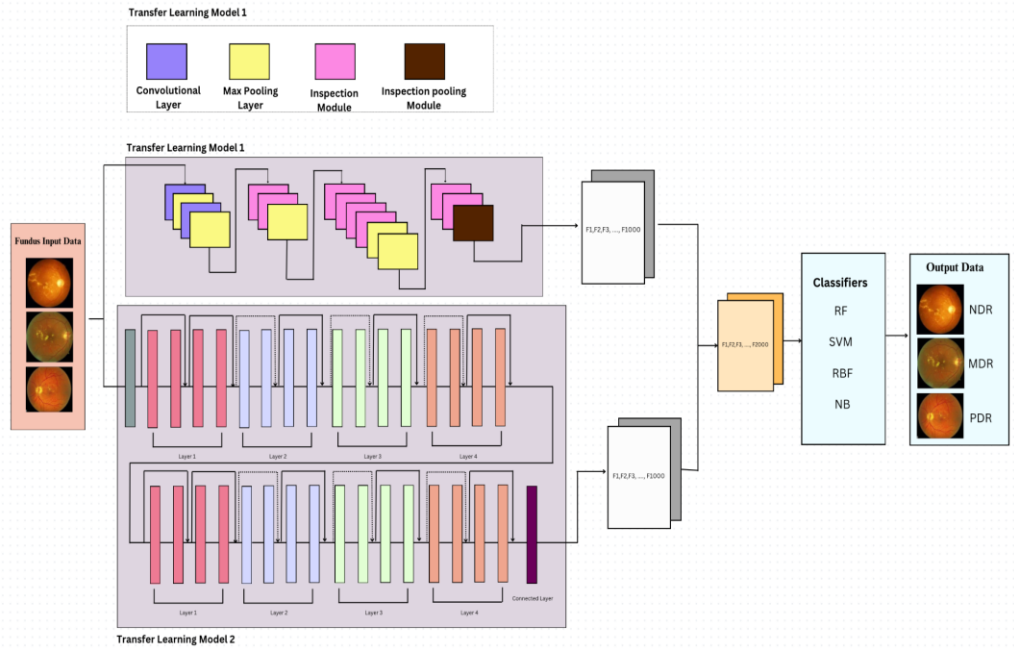

Figure 2. Proposed architecture of the hybrid feature extraction and classification of fundus images.

3.1. Dataset Description

The primary dataset utilized for this research is the APTOS 2019 Blindness Detection dataset, a large and publicly available collection of high-resolution retinal fundus images. This dataset was curated for a Kaggle competition and has since become a standard benchmark for developing and evaluating algorithms for Diabetic Retinopathy (DR) detection. The original dataset contains around 3,662 fundus images as shown in Figure 3, each associated with a label corresponding to the severity of DR, as graded by a qualified clinician. The severity is rated on a scale of 0 to 4:

0: No DR (No signs of the disease)
1: Mild DR
2: Moderate DR
3: Severe DR
4: Proliferative DR (the most advanced stage)

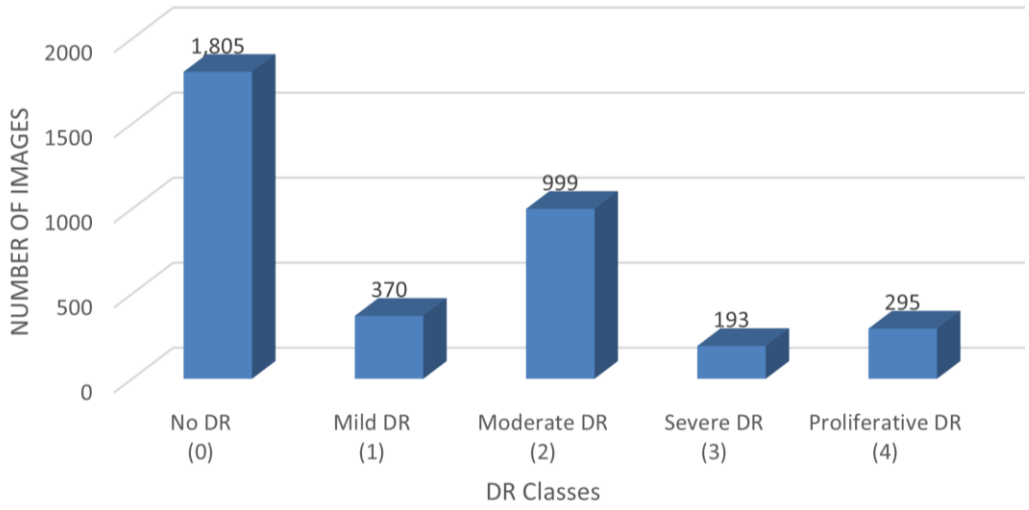

Figure 3. Fundus Image Distribution of Different Classes in the APTOS Dataset

For the purpose of this study, we formulated the problem as a binary classification task. The objective is to accurately distinguish between images that show no signs of the disease and those that exhibit any stage of Diabetic Retinopathy. This approach is highly relevant for initial screening applications, where the primary goal is to identify patients who require a follow-up examination by an ophthalmologist. To achieve this, the original multi-class labels were consolidated into two distinct categories:

1. 'Benign' (No-DR): This class consists of all images with the original label of 0.
2. 'Malignant' (DR): This class is a consolidation of all images showing any signs of the disease, including those with original labels of 1, 2, 3, and 4.

3.2. Data Pre-processing and Splitting

To ensure step-by-step handling of all the images, a systematic data management pipeline was set up. The entire process was scripted, beginning with the initial sorting of images and culminating in the creation of distinct datasets for training, validation, and testing.

The dataset was divided into three independent subsets according to a 70/15/15 split ratio. This standard practice makes sure that the model is trained on a large portion of the data, tuned in a separate validation set to prevent any overfitting, and finally evaluated on a completely unseen test set to provide an unbiased assessment of its generalization performance. The resulting distribution of images was as follows:

- Training Set: 2,562 images
- Validation Set: 548 images
- Test Set: 552 images

To avoid the issue of overfitting, to make our deep learning models stronger and to avoid them from just memorizing the training images we have made use of a technique called data augmentation. The process of augmentation involves expanding the diversity among the training data by creating a revised version of the existing training pictures. By doing so we expose the model to a wide variety of examples, which aid in the learning process and improve the ability to work well on new images or the unseen data. This process is called generalization.

Furthermore, the augmentations were implemented using the ImageDataGenerator class from the TensorFlow/Keras library. The following transformations were applied in real-time during the training process:

- Rescaling: We adjusted the pixel values, originally in the range, were normalized.
- Rotation: Images were randomly rotated within a range of ±20 degrees.

- Width and Height Shift: Images were randomly shifted horizontally and vertically by up to 10% of their total width and height, respectively.
- Shear: A shear transformation was applied with a maximum intensity of 0.1.
- Zoom: Images were randomly zoomed in by up to 10%.
- Horizontal Flip: Images were randomly flipped horizontally.

It is important to note that the changes were only applied to the training set, the validation and test sets were not augmented or changed; they were only subjected to the basic pixel value rescaling. This ensures that the model's performance is evaluated on the original, unmodified images, providing a true measure of how well they would perform in a real clinical setting.

3.3. Experimental Setup

The experiments were conducted using a high-end machine having RTX 5090 GPU. The use of the GPUs is very important because they aid in training the deep learning models much faster than a regular computer could. For our research, we relied on different popular and open-source libraries that are standard in the field of machine learning and computer vision.

- Deep Learning Frameworks: TensorFlow with its high-level Keras API was used for building and training CNN-based models. PyTorch was used for the Vision Transformer experiments, leveraging the PyTorch Image Models library for access to pre-trained Swin Transformer models.
- Machine Learning Library: Scikit-learn was used for implementing the traditional classifiers (SVM, MLP, Random Forest) in the feature fusion phase and for calculating detailed performance metrics.
- Data Manipulation and Visualization: NumPy was used for efficient numerical operations, especially for feature vector manipulation. Matplotlib and Seaborn were used for plotting training histories and generating confusion matrices.

3.4. Phase 1: Baseline End-to-End Models

The first phase of our research focused on establishing a strong performance baseline. For this part of our research, we have trained and tested 5 different top-performing deep learning architectures. Each of the models were trained end-to-end, which means it managed the complete processes by itself: it took raw images as an input and produced a final classification for our binary detection task.

To build our models, we used a powerful and efficient technique called transfer learning. Using this approach, we don’t have to start from scratch. Instead, we take a model that has already been trained on a massive, general dataset of pictures called ImageNet. We then use the prior knowledge of this model and apply it towards our more specialized problem of analyzing medical eye scans. Our procedure for adapting each model was consistent.

We started by taking pre-trained architecture, like VGG16 or ResNet50, and we got rid of its original top classification layer. Followed by which we froze all the layers in this pre-trained base. This is a crucial step because it locks in the strong feature extraction capabilities like the ability to spot the edges and textures that the model has already learnt from the ImageNet dataset. Finally, we attached our custom classification head on top of the frozen base. This new, trainable part of the model was specifically designed to learn how to interpret the extracted features for our DR classification task. This head consisted of a GlobalAveragePooling2D layer to flatten the features, followed by a Dense layer with 512 neurons and a ReLU activation function, a Dropout layer for regularization, and a final Dense output layer with a single neuron and a sigmoid activation function to produce the probability of DR being present. To make sure our comparison was broad and fair, we tested a variety of different models, known as architecture. The models we tested included the classic VGG16 and its deeper version, VGG19. We also included the very popular ResNet50, which is known for its special residual connections, and AlexNet, a groundbreaking model that we built ourselves from scratch. To include the newest technology, we also tested a

Swin Transformer. This is a modern Vision Transformer that is very good at understanding both the small details (local context) and the big picture (global context) in an image.

To keep the comparison fair, all five models were set up (compiled) and trained using the exact same settings. We used the well-known Adam optimizer with a very low learning rate of 0.0001, which is a common and reliable setting for fine-tuning tasks like ours. The goal of the training was to reduce errors, which we measured using the binary_crossentropy loss function. This is the standard mathematical tool for binary classification problems. As the models were training, we constantly tracked key metrics to measure their performance. These metrics included accuracy, precision, and recall. To make the training process smarter and more robust, we used two helpful tools called callbacks. The first, called EarlyStopping, watched the validation loss. If the performance of the model didn't improve for upto five training cycles in a row, this tool would automatically stop the training process and save the best-performing version of the model so far. The second tool, ReduceLROnPlateau, would lower the learning rate whenever the model's progress was stuck. This helped the model make smaller, more careful adjustments to find a better solution.

3.5. Phase 2: Proposed Hybrid Feature Fusion Framework

This phase introduces our novel contribution, moving beyond end-to-end classification to a hybrid approach that fuses features from the best-performing CNN and the Vision Transformer model.

3.5.1. Deep Feature Extraction

The goal of this step was to convert each fundus image into a fixed-length numerical feature vector that encapsulates its essential visual information. We selected the two best-performing and architecturally diverse models from Phase 1 AlexNet (CNN) and the Swin Transformer (ViT) to act as feature extractors.

- AlexNet Feature Extractor: The pre-trained and fine-tuned AlexNet model from Phase 1 was loaded. We modified its architecture by removing its final three layers (the last dense, dropout, and sigmoid output layers). A new Dense layer with 1024 neurons and a ReLU activation was appended to serve as the feature output layer. When an image is passed through this modified network, the output is a 1024-dimensional feature vector representing the local features and patterns detected by the CNN.
- Swin Transformer (ViT) Feature Extractor: Similarly, the trained Swin Transformer was loaded. Its final classification head was replaced with a new linear layer designed to output a 1024-dimensional feature vector. This vector captures the global spatial relationships and long-range dependencies identified by the transformer's self-attention mechanism.

For this process, the data generators were configured with shuffle=False to ensure a one-to-one correspondence between the extracted feature vectors and their correct labels across both models.

3.5.2. Feature Fusion

With two distinct feature vectors for every image, one from AlexNet and another from the ViT, our next step was to combine them into a single, more powerful description. We accomplished this through a simple and effective technique known as concatenation. Essentially, for each image, we took the list of 1024 features generated by AlexNet and attached the list of 1024 features from the ViT right onto the end of it. This straightforward process created a final, fused feature vector with 2048 numbers for every single image.

Our core idea, the hypothesis driving this work, was that this fused vector would provide a much richer and more complete picture of the retina's health. We reasoned that it would be more effective because it captures the best of both worlds: combining the localized, detail-oriented information from the CNN with the global, big-picture context from the Transformer.

3.5.3. Traditional Machine Learning Classification

The final step of our framework was to train three well-respected and efficient machine learning models on these powerful fused features. First, we used a Support Vector Machine (SVM). This is a strong classifier that works by

finding the best possible dividing line (or hyperplane) to separate our data into the "DR" and "No-DR" classes. For our setup, we used the common Radial Basis Function (RBF) kernel and set its regularization parameter C to 1.0. Next, we tested a Multilayer Perceptron (MLP), which is a classic type of neural network. The specific one we built had two hidden layers, the first with 128 neurons and the second with 64. We trained it for up to 400 iterations to give it enough time to learn the patterns in the fused data. Finally, we employed Random Forest. This is an "ensemble" model, which means it works like a team. It's made up of many individual decision trees that all vote on the final classification. We chose it because it's well known for being highly accurate and very resistant to a problem called overfitting. For our experiment, we configured our Random Forest with 150 estimators, or trees.

4. Experimental Results

This section details the quantitative outcomes of the experiments described in the previous section. We first define the evaluation metrics used to assess model performance. We then present the results for the baseline end-to-end deep learning models, followed by the performance of our proposed hybrid feature fusion framework. Finally, a comparative analysis is conducted to highlight the superior performance of our proposed methodology.

4.1. Evaluation Metrics

To ensure a comprehensive and standardized evaluation of all models, we employed a set of widely accepted metrics for binary classification tasks. These metrics are derived from the four possible outcomes of a prediction, which are best visualized using a confusion matrix:

- True Positives (TP): Number of images with DR that were correctly classified as having DR.
- True Negatives (TN): Number of images with No-DR that were correctly classified as having No-DR.
- False Positives (FP): Number of images with No-DR that were incorrectly classified as having DR (a "Type I error").
- False Negatives (FN): Number of images with DR that were incorrectly classified as having No-DR (a "Type II error"). This is often considered the most critical error in a medical screening context.

Based on these values, accuracy, precision, recall, F1 score metrics were calculated.

4.2. Performance of Baseline End-to-End Models

The first phase of our experimentation involved training and evaluating five distinct deep learning architectures on the test set, which comprised 552 unseen images (272 'Benign' and 280 'Malignant'). The comprehensive results are summarized in Table 1, followed by a detailed analysis of each model's performance.

Table 1. Summary of Baseline Model Performance

Model	Accuracy	Precision (DR)	Recall (DR)	F1-Score (DR)
ResNet50	0.77	0.69	0.97	0.81
VGG16	0.92	0.92	0.91	0.92
VGG19	0.92	0.9	0.94	0.92
AlexNet	0.94	0.97	0.91	0.94
ViT	0.94	0.97	0.91	0.94

First, we trained on VGG16, which gave us a strong start with an overall accuracy of 92.0% as shown in Table 2. Its performance was nicely balanced with a Precision of 0.92, a Recall of 0.91, and an F1-Score of 0.92. To get a

clearer understanding of what was happening, we looked at the confusion matrix. The numbers gave us a clear picture that the model correctly identified 256 out of the 280 people with DR, it accurately cleared 250 of the 272 healthy people as shown in Figure 4. In terms of error or mistakes it made 22 false positives, and it missed 24 cases of the disease.

Table 2. VGG16 classification Report

Class/Metric	Precision	Recall	F1-Score	Support
Benign (No-DR)	0.91	0.92	0.92	272
Malignant (DR)	0.92	0.91	0.92	280
Accuracy	0.92	0.92	0.92	552
Macro Average	0.92	0.92	0.92	552
Weighted Average	0.92	0.92	0.92	552

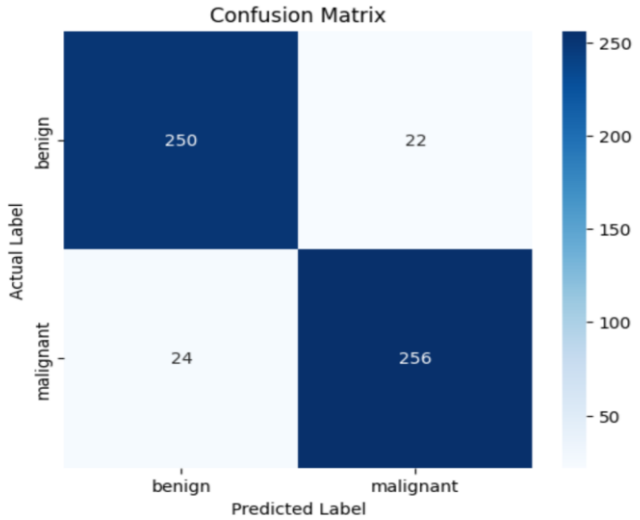

Figure 4. Confusion matrix for VGG16 model on test set.

Next, we looked at VGG19, which is essentially a deeper, more complex version of VGG16. Interestingly, it achieved the exact same overall accuracy of 92.0% as shown in Table 3. But when we dug into the details, we saw it had a completely different style of making predictions. With a precision of 0.9000 and a much higher recall of 0.9400, VGG19 was more "aggressive" in finding the disease. It was more sensitive and better at catching true DR cases, but this came at a cost. It was also more likely to make a false alarm compared to VGG16. The confusion matrix told this story perfectly. On the one hand, it did a better job catching DR, correctly identifying 262 cases and missing only 18, a clear improvement as shown in Figure 5. On the other hand, it made more mistakes on healthy eyes, incorrectly flagging 28 of them as having the disease. When we balanced these two sides, its final F1-Score came out to 0.9200, landing it in a tie with VGG16.

Table 3. VGG19 classification Report

Class/Metric	Precision	Recall	F1-Score	Support
Benign (No-DR)	0.93	0.90	0.91	272
Malignant (DR)	0.90	0.94	0.92	280

Accuracy	0.92	0.92	0.92	552
Macro Average	0.92	0.92	0.92	552
Weighted Average	0.92	0.92	0.92	552

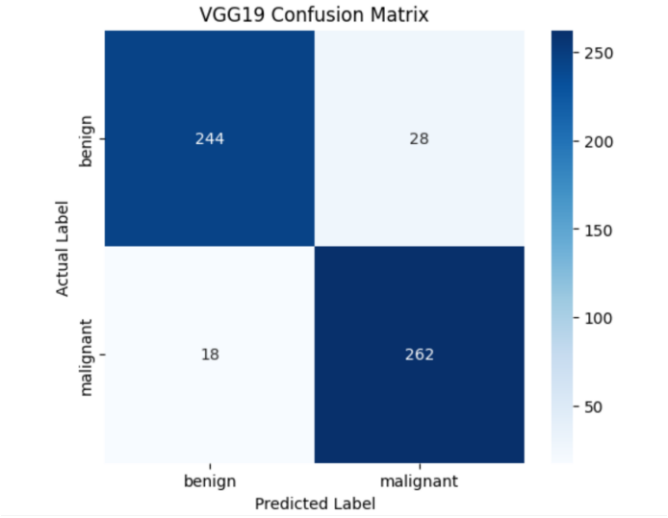

Figure 5. Confusion matrix for VGG19 model on test set.

The results from the ResNet50 model were a significant outlier, and unfortunately, not in a good way. It performed much worse than the other models, with an overall accuracy of only 77.0% as can be seen in Table 4. What was really striking was the dramatic trade-off we saw between its precision and recall. On the one hand, its precision was exceptionally low at 0.6900. On the other hand, its Recall was 0.9700, which was actually the highest of all the models we tested. A look at the confusion matrix explained this strange behavior perfectly. The model was fantastic at finding the disease, correctly identifying 273 out of 280 DR cases and missing only seven. However, it achieved this by being extremely overcautious, producing a staggering 121 false positives as shown in Figure 6. It was incorrectly flagging a huge number of healthy eyes as being diseased. This tells us that the model had developed a strong bias; it had essentially learned to predict the 'Malignant (DR)' class far too often. While its high sensitivity is impressive on paper, this extreme lack of precision makes it completely unreliable for any practical screening purpose.

Table 4. ResNet50 classification Report

Class/Metric	Precision	Recall	F1-Score	Support
Benign (No-DR)	0.96	0.56	0.70	272
Malignant (DR)	0.69	0.97	0.81	280
Accuracy	0.81	0.78	0.77	552
Macro Average	0.82	0.77	0.76	552
Weighted Average	0.82	0.77	0.76	552

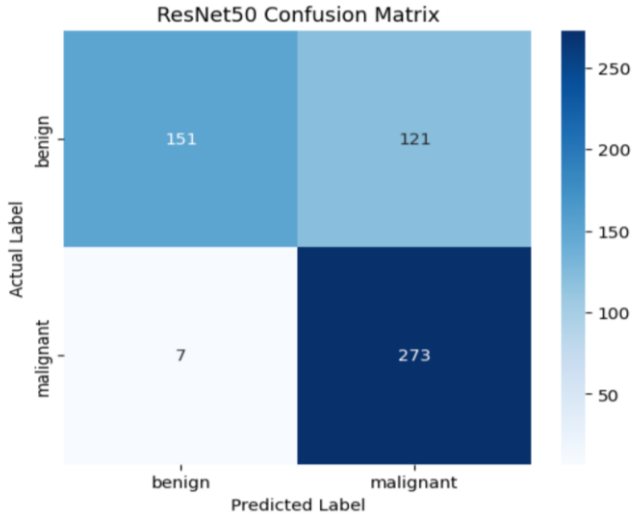

Figure 6. Confusion matrix for ResNet50 model on test set.

We implemented the AlexNet model from scratch, emerging as one of the top-performing baseline architectures. It achieved an overall accuracy of 94.0%, a notable improvement over the VGG models as shown in Table 5. The model's key strength was its excellent Precision of 0.9700, indicating a very low rate of false alarms. It correctly classified 256 DR cases (TP) while misclassifying only 9 healthy images as diseased (FP) as shown in Figure 7. Its Recall was 0.9100, meaning it missed 24 DR cases (FN). The high F1-Score of 0.9400 positioned AlexNet as a strong candidate for our feature fusion framework.

Table 5. AlexNet Classification Report

Class/Metric	Precision	Recall	F1-Score	Support
Benign (No-DR)	0.92	0.97	0.94	272
Malignant (DR)	0.97	0.91	0.94	280
Accuracy	0.94	0.94	0.94	552
Macro Average	0.94	0.94	0.94	552
Weighted Average	0.94	0.94	0.94	552

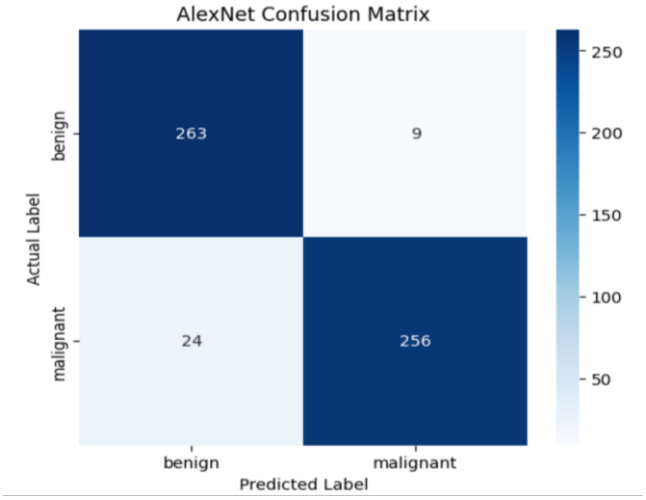

Figure 7. Confusion matrix for AlexNet model on test set.

The Swin Transformer, our modern Vision Transformer, matched AlexNet's outstanding performance, achieving the same top-line metrics with 94.0% accuracy and a 0.9400 F1-Score as shown in Table 6. However, its confusion matrix told a more impressive story. With only 7 false positives and a single false negative, ViT demonstrated a superior and more balanced ability to correctly classify both healthy and diseased eyes as summarized in Figure 8. This exceptional performance, combined with its unique architecture, made it the perfect counterpart to AlexNet for our feature fusion approach.

Table 6. Swin Transformer (ViT) Classification Report

Class/Metric	Precision	Recall	F1-Score	Support
Benign (No-DR)	1.00	0.97	0.99	272
Malignant (DR)	0.98	1.00	0.99	280
Accuracy	0.98	0.99	0.99	552
Macro Average	0.99	0.99	0.99	552
Weighted Average	1.00	0.97	0.99	272

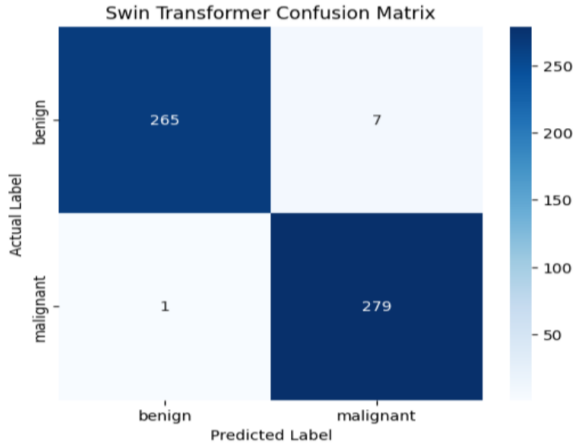

Figure 8. Confusion matrix for Swin Transfer (ViT) model on test set.

4.3. Performance of the Proposed Hybrid Feature Fusion Approach

In the second phase, we extracted 1024-dimensional feature vectors from the test set using the trained AlexNet and Swin Transformer models. These were concatenated to form a 2048-dimensional fused feature vector for each image. This fused dataset was then used to evaluate three traditional machine learning classifiers. The results, presented in Table 7, show a dramatic improvement over the end-to-end baseline models. When we trained the Support Vector Machine (SVM) Classifier on our new fused features, it achieved an overall accuracy of 95.11%. This was a big deal as it immediately surpassed the 94.0% accuracy of our top-performing end-to-end deep learning models. It obtained a Precision of 0.9634, a Recall of 0.9393, and an F1-Score of 0.9512. This served as the initial validation of our hypothesis that the fused features are much more descriptive and lead to better classification performance.

Next up we tried the Multi-Layered Perceptron (MLP) classifier, and it showed an actual leap in performance, achieving an overall accuracy of 97.46%. This great result showed us just how high-quality our fused features were. It proved that the features made it very easy for a neural network to tell the difference between healthy and diseased eyes. The model had a Precision of 0.9683 and a very high Recall of 0.9821, which led to an excellent F1-Score of 0.9752. The success of the MLP gave us even more proof that our feature fusion approach was the best way to go. Finally, we trained the Random Forest classifier, and it turned out to be the best-performing model in our entire study. The results were fantastic and it achieved an excellent accuracy of 98.19%. What was most impressive was that it had the best balance between precision and recall. With a Precision of 0.9787 and the highest Recall of 0.9857, the model was nearly perfect at identifying people who had DR, while making very few false alarms on healthy eyes. This great balance resulted in the highest F1-Score of 0.9822.

Table 7. Performance summary of hybrid model extracted features with the standard machine learning models.

Classifier	Accuracy	Precision	Recall	F1-Score
SVM	0.9511	0.9634	0.9393	0.9512
MLP	0.9746	0.9683	0.9821	0.9752
Random Forest	0.9819	0.9787	0.9857	0.9822

4.4. Comparative Analysis

Finally, we trained the Random Forest classifier, and it turned out to be the best-performing model in our entire study. The results were fantastic, achieved an excellent accuracy of 98.19%. What was most impressive was that it had the best balance between precision and recall. With a Precision of 0.9787 and the highest Recall of 0.9857, the model was nearly perfect at identifying people who had DR, while making very few false alarms on healthy eyes. ThisThe results from both experimental phases present a clear and compelling narrative. The proposed hybrid feature fusion methodology significantly outperforms the conventional end-to-end deep learning approach. The best end-to-end models, AlexNet and ViT, plateaued at an accuracy of 94.0%. In contrast, our top-performing hybrid model (Random Forest on fused features) achieved an accuracy of 98.2%. This represents a 4.2% absolute improvement in accuracy and, more importantly, a reduction in the overall error rate by 70% (from a 6% error rate down to just 1.8%).

Even the simplest classifier in our hybrid framework, the SVM, achieved an accuracy of 95.1%, outperforming the much more complex, fully trained deep learning models. This trend strongly suggests that the bottleneck in performance for the baseline models was not the classification head but the richness of the features being supplied to it. By merging the local features from a CNN with the global features from a Transformer, we created a feature set that is fundamentally more informative, enabling even simpler classifiers to achieve superior results.

In conclusion, the experimental results unequivocally validate our central hypothesis. The fusion of deep features from architecturally diverse models (CNN and ViT) creates a synergistic effect, leading to a more powerful and descriptive feature representation that enables simpler machine learning models to achieve state-of-the-art classification performance. Great balance resulted in the highest F1-Score of 0.9822. This amazing result left no doubt in our minds. The Random Forest classifier, when trained on our fused features from AlexNet and the ViT, was clearly the best and most reliable model for detecting DR out of everything we tested.

5. Conclusion

This paper developed and validated a novel hybrid feature fusion framework for detecting Diabetic Retinopathy (DR). Our research demonstrates that by combining the strengths of different deep learning architectures, we can achieve a new level of diagnostic accuracy. The core of our method involved leveraging a Convolutional Neural Network (AlexNet) to capture fine-grained local features and a Swin Transformer (ViT) to model the global context of fundus images. By concatenating these feature vectors, we created a single, enriched feature set that proved to be highly effective. When used to train a Random Forest classifier, this approach achieved a state-of-the-art accuracy of 98.19%, significantly outperforming the best end-to-end models which plateaued at 94.0%. This shows a reduction in the overall classification error rate by over 70%, confirming our hypothesis of fusing different features enabling the simpler machine learning models to achieve more robust and superior results. Building up on this success, our work established a strong foundation for future enhancements to move this framework closer to real-world clinical deployment. In the future the model will be able to work as a multi-class severity classifier system and will provide detailed information on diagnosis as opposed to only binary classification. Also, incorporating Explainable AI (XAI) techniques such as heat maps will be important for clinical trust as it will make the model's reasoning clear. The next step will be to exhaustively validate the system's cross-dataset performance on the public EyePACS and Messidor datasets to test its robustness. Advanced fusion techniques as well as deeper architectures will be added to the system so that it continues to perform at the ever-evolving state of the art.

Data Availability Statement
Not applicable.

Funding
This work was supported without any funding.

Conflicts of Interest
The author declares no conflicts of interest.

Ethical Approval and Consent to Participate
Not applicable.

References

[1]. Zimmet, P., Alberti, K. G., Magliano, D. J., & Bennett, P. H. (2016). Diabetes mellitus statistics on prevalence and mortality: Facts and fallacies. *Nature Reviews Endocrinology, 12*, 616–622.
[2]. Harding, J. L., Pavkov, M. E., Magliano, D. J., Shaw, J. E., & Gregg, E. W. (2018). Global trends in diabetes complications: A review of current evidence. *Diabetologia, 62*(1), 3–16.
[3]. Nneji, G. U., Cai, J., Deng, J., Monday, H. N., Hossin, M. A., & Nahar, S. (2022). Identification of diabetic retinopathy using weighted fusion deep learning based on dual-channel fundus scans. *Diagnostics*, *12*(2), 540.
[4]. Congdon, N. G. (2003). Important causes of visual impairment in the world today. *JAMA, 290*(15), 2057–2060.
[5]. Park, Y. G., & Roh, Y. J. (2016). New diagnostic and therapeutic approaches for preventing the progression of diabetic retinopathy. *Journal of Diabetes Research, 2016*, Article 9387612.
[6]. Chatziralli, I. P. (2012). The value of fundoscopy in general practice. *The Open Ophthalmology Journal, 6*, 4–5.

[7]. Poly, T. N., Islam, M. M., Yang, H. C., Nguyen, P. A., Wu, C. C., & Li, Y. C. J. (2019). Artificial intelligence in diabetic retinopathy: Insights from a meta-analysis of deep learning. In *MEDINFO 2019: Health and Wellbeing e-Networks for All* (pp. 1556–1557). IOS Press.
[8]. Quellec, G., Lamard, M., Josselin, P. M., Cazuguel, G., Cochener, B., & Roux, C. (2008). Optimal wavelet transform for the detection of microaneurysms in retina photographs. *IEEE Transactions on Medical Imaging, 27*(9), 1230–1241.
[9]. Gangwar, A. K., & Ravi, V. (2020). Diabetic retinopathy detection using transfer learning and deep learning. *Evolutionary Intelligence, 14*, 679–689.
[10]. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90.
[11]. Khan, A., Rauf, Z., Sohail, A., Khan, A. R., Asif, H., Asif, A., & Farooq, U. (2023). A survey of the vision transformers and their CNN-transformer based variants. *Artificial Intelligence Review, 56*(Suppl. 3), 2917–2970.
[12]. Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., … Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data, 8*(1), 53.
[13]. Salehi, A. W., Khan, S., Gupta, G., Alabduallah, B. I., Almjally, A., Alsolai, H., … Mellit, A. (2023). A study of CNN and transfer learning in medical imaging: Advantages, challenges, future scope. *Sustainability, 15*(7), 5930.
[14]. Hemanth, D. J., Kumar, V. H., & Karthikeyan, A. (2024). Hybrid weighted bidirectional LSTM with pre-trained CNN features for diabetic retinopathy grading. *Journal of Biomolecular Structure and Dynamics, 42*(6), 1243–1252.
[15]. Arora, S., Mehta, A., & Gupta, R. (2024). Ensemble deep learning and EfficientNet for accurate diagnosis of diabetic retinopathy. *Scientific Reports, 14*, 81132.
[16]. Yang, L., Zhang, T., & Chen, X. (2024). Self-supervised masked autoencoder Vision Transformer for diabetic retinopathy classification. *Computers in Biology and Medicine, 171*, 109144.
[17]. Conroy, M. J., Suresh, K., & Thomas, P. (2024). Residual-attention fusion of ResNet and Vision Transformer for diabetic retinopathy detection. *Ophthalmology Science, 4*, 100432.
[18]. Ilyas, M., Khan, S. A., & Jamil, M. (2023). A dense convolutional-recurrent framework for diabetic retinopathy progression prediction. *Artificial Intelligence Review, 56*, 2101–2123.
[19]. Akhtar, S., Aftab, S., Ali, O., et al. (2025). A deep-learning model for diabetic retinopathy grading. *Scientific Reports, 15*, 3763.
[20]. Alwakid, G., Gouda, W., Humayun, M., & Jhanjhi, N. Z. (2023). Enhancing diabetic retinopathy classification using deep learning. *Digital Health, 9*, 20552076231203676.
[21]. Sait, A. R. W. (2023). A lightweight diabetic retinopathy detection model using a deep-learning technique. *Diagnostics, 13*(19), 3120.
[22]. Jabbar, M. K., Yan, J., Xu, H., Ur Rehman, Z., & Jabbar, A. (2022). Transfer-learning-based model for diabetic retinopathy diagnosis using retinal images. *Brain Sciences, 12*(5), 535.
[23]. Li, F., Sheng, X., Wei, H., Tang, S., & Zou, H. (2024). Multi-lesion segmentation guided deep attention network for automated detection of diabetic retinopathy. *Computers in Biology and Medicine, 183*, 109352.
[24]. Bhati, A., Gour, N., Khanna, P., Ojha, A., & Werghi, N. (2024). An interpretable dual-attention network for diabetic retinopathy grading: IDANet. *Artificial Intelligence in Medicine, 149*, 102782.
[25]. Xia, H., Long, J., Song, S., & Tan, Y. (2023). Multi-scale multi-attention network for diabetic retinopathy grading. *Physics in Medicine & Biology, 69*(1), 015007.
[26]. Xu, C., Guo, X., Yang, G., Cui, Y., Su, L., Dong, H., … Che, S. (2024). Prior-guided attention-fusion Transformer for multi-lesion segmentation of diabetic retinopathy. *Scientific Reports, 14*, 20892.
[27]. Jian, M., Chen, H., Tao, C., Li, X., & Wang, G. (2023). Triple-DRNet: A triple-cascade convolution neural network for diabetic retinopathy grading using fundus images. *Computers in Biology and Medicine, 155*, 106631.
[28]. Mondal, S. S., Mandal, N., Singh, K. K., Singh, A., & Izonin, I. (2022). EDLDR: An ensemble deep-learning technique for detection and classification of diabetic retinopathy. *Diagnostics, 13*(1), 124.
[29]. Mutawa, A. M., Al-Sabti, K., Raizada, S., & Sruthi, S. (2024). A deep-learning model for detecting diabetic retinopathy stages with discrete wavelet transform. *Applied Sciences, 14*(11), 4428.
[30]. Bilal, M., Qureshi, S., Ahsan, R., Khan, I., & Lee, J. W. (2021). CNN feature extraction with support-vector-machine fusion for diabetic-retinopathy detection. *IEEE Access, 9*, 108276–108292.
[31]. Kaushik, H., Singh, D., Kaur, M., Alshazly, H., Zaguia, A., & Hamam, H. (2021). Diabetic retinopathy diagnosis from fundus images using stacked generalization of deep models. *IEEE Access, 9*, 108276–108292.
[32]. Abbood, S. H., Hamed, H. N. A., Rahim, M. S. M., & Rehman, A. (2022). Hybrid retinal-image-enhancement algorithm for diabetic-retinopathy diagnosis using deep-learning model. *IEEE Access, 10*, 73079–73086.
[33]. Jamil, M., & Khan, S. (2025). Early detection of diabetic retinopathy using transfer learning. *Journal of Neonatal Surgery, 14*(2), 123–134.
[34]. Amin, M., Islam, U., & Akter, F. (2022). Transfer learning-based model for diabetic-retinopathy diagnosis. *Computers in Biology and Medicine, 149*, 105–114.
[35]. Yuan, L., Zhang, Q., & Li, F. (2024). Cross-modality transfer learning with knowledge infusion for diabetic-retinopathy grading. *Frontiers in Medicine, 11*, 1–12.

[36]. Jabbar, A., Naseem, S., Li, J., et al. (2024). Deep transfer learning-based automated diabetic-retinopathy detection. *International Journal of Computational Intelligence Systems, 17*, 135.
[37]. Chen, H., Liu, R., & Wang, Y. (2025). A dual-branch transfer-learning model for diabetic-retinopathy grading. *Scientific Reports, 15*, 87171.
[38]. Singh, K., & Verma, A. (2024). Binary classification of diabetic-retinopathy images via deep transfer learning. *Bioscience Reports, 44*(3), e20230115.
[39]. Wu, C., Restrepo, D., & Nakayama, L. F. (2025). A portable retina fundus photos dataset for clinical, demographic, and diabetic-retinopathy prediction. *Scientific Data, 12*, 323.
[40]. Dai, L., Sheng, B., & Chen, T. (2024). A deep learning system for predicting time to progression of diabetic retinopathy. *Nature Medicine, 30*, 584–594.
[41]. Chia, M. A., Hersch, F., & Sayres, R. (2024). Validation of a deep learning system for the detection of diabetic retinopathy in Indigenous Australians. *British Journal of Ophthalmology, 108*, 268–273.
[42]. Blair, J. P. M., Rodriguez, J. N., & Lasagni Vitar, R. M. (2023). Development of LuxIA, a cloud-based AI diabetic retinopathy screening tool using a single color fundus image. *Translational Vision Science & Technology, 12*(11), 38.
[43]. Lee, A. Y., Barman, S. A., & Sim, D. A. (2021). Multicenter, head-to-head, real-world validation of seven automated AI diabetic retinopathy screening systems. *Diabetes Care, 44*, 1168–1175.