Computer Modeling in Engineering & Sciences

Tech Science Press

https://doi.org/10.32604/cmes.2026.081617

ARTICLE

A Lightweight YOLOv11 Framework for Multi-Class Retinal Disease Classification

Jaffar Hussain[1], **Tahira Nazir**[1], **Junaid Rashid**[2,*] **and Jungeun Kim**[3,*]

[1]Faculty of Computing, Riphah International University, Islamabad, Pakistan
[2]Department of Artificial Intelligence and Data Science, Sejong University, Seoul, Republic of Korea
[3]Department of Computer Science and Engineering, Inha University, Incheon, Republic of Korea

*Corresponding Authors: Junaid Rashid. Email: junaid.rashid@sejong.ac.kr; Jungeun Kim. Email: jekim@inha.ac.kr

Received: 05 March 2026; Accepted: 21 May 2026; Published: 30 June 2026

ABSTRACT: Early detection of diabetic retinopathy (DR), media haze (MH), optic disc cupping (ODC), and glaucoma is crucial for preventing vision loss. However, timely diagnosis is often constrained by limited specialist availability and high diagnostic costs. This study proposes a You Only Look Once (YOLO)-based deep learning (DL) framework for the automated classification of fundus images into disease-specific categories. We unified diverse annotations from the Retinal Fundus Multi-Disease image Dataset (RFMiD), RFMiD2.0, and the DR Fundus Image Dataset (DR-FID) by standardizing annotation files and class labels. A custom filtering module was used to isolate single-pathology cases, and dataset issues such as missing or corrupted files were identified and resolved. To handle class imbalance, we applied oversampling and undersampling methods. The dataset was re-engineered for lightweight, accurate classification with YOLOv11, utilizing offline preprocessing tailored for retinal images. The dataset design leverages YOLOv11's multi-class classification framework to achieve high performance on resource-constrained devices. This tailored approach outperforms preparing datasets solely through cloud-based platforms like Roboflow. The proposed model uses a lightweight YOLOv11 architecture, resulting in faster inference and lower memory requirements than conventional Convolutional Neural Networks (CNNs), such as Residual Networks (ResNets) or Visual Geometry Group (VGG) networks. Delivering high accuracy with minimal resource use, the model shows no signs of divergence or overfitting. Confusion matrices and class-wise metrics confirm consistent performance. The proposed framework achieves improved performance, with 94.78% accuracy, 96.12% specificity, 79.61% precision, 83.61% recall, and an 81.14% F1-score, demonstrating strong generalization to the internal held-out test set.

KEYWORDS: Image classification; fundus; convolutional neural networks; deep learning; machine learning

1 Introduction

1.1 Background and Motivation

Globally, a large number of people suffer from vision disorders due to retinal diseases [1] like DR, MH, cataracts, ODC, and glaucoma [2]. If these retinal pathologies are not promptly evaluated by an ophthalmologist, they may result in irreversible visual impairment. Specifically, diabetic macular edema (DME) affected twenty-seven million people in 2020 [3,4]. Moreover, 7.7 million people were affected by glaucoma in 2023. The World Health Organization (WHO) indicates a growing health concern that the prevalence of DR patients is expected to reach 161 million by 2045 [5]. Furthermore, millions of deaths occur annually due to general complications arising from diabetes and kidney diseases [6]. Manual examination

is inherently resource-intensive and may yield incomplete results during visual inspection. To address these issues, deep learning models, including CNNs, Transformers, and the YOLO architecture, have emerged to automate physical examinations and replace human bias introduced by conventional methods. These models require images as input, which are captured using fundus or Optical Coherence Tomography (OCT) cameras [7,8]. These models extract image features required to automate the process of retinal examination. Such persistent vision challenges inspired us to conduct this study. We aim to develop a lightweight DL model for real-time eye inspection to bridge the gap in resource-constrained areas. This automated eye examination can reduce diagnosis time. It will also assist in reporting the risk of vision-related syndromes. Finally, it will reduce irreversible blindness and healthcare costs and improve public health globally.

1.2 Importance of Retinal Disease Classification

The classification of eye disorders with improved accuracy is crucial to prevent vision loss through timely intervention. However, the manual diagnosis process performed by an ophthalmologist is slow and costly. Frequently, eye specialists are unavailable in underserved regions. Recently, developments in Artificial Intelligence (AI) and its incorporation into automated systems have effectively mitigated these challenges. Moreover, DL and machine learning (ML) models, including custom CNNs, GoogleNet, and YOLO, facilitate the automated analysis of eye disorders. These models notably enhance the diagnosis of eye disorders and rapidly classify retinal images to assess disease severity. Thus, recent studies have reported classification accuracies exceeding 92% [9,10]. Furthermore, integrating multimodal data can improve model precision. The combination of electronic health records (EHR) and OCT images enables a comprehensive evaluation of eye diseases. Multimodal systems further enhance clinical decision-making in diagnosing and treating eye disorders.

1.3 Challenges in Multi-Class Detection

Multi-class classification of eye disorders poses greater challenges for DL models than binary classification. The models struggle to identify rare pathologies due to class imbalance and limited dataset diversity. They are unable to differentiate between mild and proliferative DR. Additionally, detecting syndromes becomes more complex when multiple syndromes occur simultaneously. Currently, most models are optimized to classify a single disease. Another major challenge is achieving true generalization while minimizing model bias. Most models perform poorly on external or unseen data when trained on limited datasets, which can degrade performance under diverse imaging conditions. Finally, DL models require high computational resources for training on large datasets, and their lack of interpretability reduces clinician trust in AI decisions [11–13].

1.4 Objectives and Contributions of the Proposed Methodology

AI-based eye care has advanced in diagnosing eye conditions. However, limitations exist in the real-world deployment of clinical tools for such diagnosis. A 2025 systematic review reported primary constraints [14]. These constraints include significant class imbalance, high computational demands, and training bias. Prior methods often relied on computationally intensive architectures, which limited their deployment in resource-limited clinics. Additionally, most models only perform binary classification and lack the capacity for multi-class screening. Finally, unrefined images with concurrent diseases undermine classification reliability.

This study presents an optimized analytical pipeline to address these challenges. We employ the computationally efficient YOLOv11 architecture, introducing a preprocessing module to ensure strict 'Pathology Isolation' and enable the architecture to capture key diagnostic features. We integrate diverse datasets

[13,15–17] into this architecture using specific synthetic enhancements to improve model generalization. These datasets include RFMiD, RFMiD2.0, and DR-FID. The single-stage YOLOv11 model minimizes computational overhead while classifying multiple categories, including DR, MH, ODC, and within normal limits (WNL). This study facilitates fair and consistent diagnosis in early-stage screening and enables the robust classification of multi-class eye diseases.

We systematically evaluate this framework. Our methodology addresses the following core research questions:

RQ1. How do strict "Pathology Isolation" and targeted resampling improve feature extraction? Does this approach effectively mitigate severe class imbalance in heterogeneous public datasets?

RQ2. Does our customized YOLOv11 architecture demonstrate the necessary computational efficiency? Is it faster and less memory-intensive than heavier CNN ensembles, such as ResNets and VGG networks, for edge-device deployment?

RQ3. Can task-specific dataset structures leverage YOLOv11's classification capabilities? Does this achieve significant top-1 accuracy improvements without the overhead of two-stage models?

By addressing these questions, we make the following key contributions:

- Novel Pathology Isolation: We introduce a specialized filtering module. It isolates single-pathology cases from heterogeneous sources, which connects public dataset accessibility with clinical-grade rigor.
- Data Curation and Balancing: We fixed missing or corrupted files in the RFMiD datasets and cross-verified them with the baseline study [13]. Furthermore, we applied targeted resampling to create balanced training distributions, thereby mitigating algorithmic bias and improving model generalization.
- Optimized Dataset Architecture: We tailored the dataset structure, annotations, and formatting to successfully leverage YOLOv11's multi-task capabilities, providing a highly optimized fundus image dataset compared to fully automated cloud tools.
- High-Efficiency Benchmarking: We customized the annotation pipeline for YOLOv11's lightweight architecture. We demonstrated improved inference speed and reduced GPU memory usage compared with heavier CNNs.
- Comparative Top-1 Accuracy: We achieved significant gains in predictive performance. The model reached a top-1 accuracy of 0.89 on the validation set. This strict metric supports the framework's potential reliability for clinical integration.

This paper is structured as follows: Section 2 provides a review of the literature, Section 3 details the methodology, Section 4 presents the experimental results, Section 5 provides results analysis and discussion, and Section 6 concludes with final considerations.

2 Literature Review

In this section, we review ML/DL techniques for the detection/classification of eye disorders in images, aligned with this proposed study. These techniques include multimodal fusion, YOLO, and Transformers for evaluating diverse datasets such as RFMiD, Asia Pacific Tele-Ophthalmology Society (APTOS), Indian DR Image Dataset (IDRiD), Messidor, Ophthalmic Image Analysis—Ocular Disease Intelligent Recognition (OIA-ODIR), Digital Retinal Images for Vessel Extraction (DRIVE), High-Resolution Fundus (HRF), Structured Analysis of the REtina (STARE), OCT images, and EHR. We acknowledge simultaneous innovations in pixel-level segmentation. Before classification, joint pipelines often use a U-Net architecture to segment anatomical features. However, they demand dense pixel-level annotations and incur higher computational overhead than the single-stage model prioritized in this study.

2.1 Retinal Disease Detection Techniques

The detection of eye diseases using automated systems has progressed drastically. Recent approaches have used basic preprocessing, including contrast-limited adaptive histogram equalization (CLAHE) and the 2D empirical wavelet transform (2D-EWT), on fundus images, whereas contemporary methods employ advanced DL frameworks. Transformers and Vision Mamba architectures have achieved competitive accuracy under specific conditions, but at the cost of significant computational overhead (He et al., 2024 [18]; Liu et al., 2025 [19]). Few models, such as the Gated Recurrent (GR-CNN), screen multiple syndromes simultaneously. These models commonly struggle with class imbalance, lack interpretability and suffer from overfitting (Elsayed & Rushdi, 2024 [20]; Ejaz et al., 2024 [13]). Previous methodologies have shifted toward multi-modal fusion to address limitations in single-modality imaging. Fundus images can be synergized with OCT to capture surface and deep structural-level anomalies (Islam et al., 2025 [21]; Zuo et al., 2024 [22]). Researchers also integrate EHR and knowledge graphs to contextualize visual anomalies with patient demographics (Gao et al., 2024 [23]; Breeyear et al., 2024 [24]). These advancements face systemic challenges in the real world for detecting multiple classes of diseases. Specifically, feature fusion demands high computational resources, while rare pathologies are plagued by severe class imbalance. Finally, consistent external validation across diverse clinical datasets remains an unresolved issue.

2.2 Deep Learning in Medical Imaging

DL has radically transformed medical image analysis by replacing human-driven feature extraction with automated hierarchical pattern recognition. CNN architectures, including ResNets, Inception, and DenseNet, are fundamental for extracting discriminative features (Chen et al., 2025 [25]; Lalithadevi & Krishnaveni, 2024 [10]). Recently, researchers have introduced advanced hybrid networks to capture broader structural context. One recent approach combines Vision Mamba with Inception-ResNet-V2 to capture local microaneurysms and global retinal context simultaneously (Liu et al., 2025 [19]). Transformer-based architectures have also demonstrated high efficacy in multi-spectrum processing (He et al., 2024 [18]). Recent studies aim to enhance predictive reliability, and frequently employ stacking ensembles that combine multiple DL networks with traditional ML classifiers, including Support Vector Machines (SVM) and Random Forests (Hemal & Saha, 2025 [12]; Bodapati and Veeranjaneyulu, 2024 [26]; Macsik et al., 2024 [27]). Cross-modal frameworks fuse fundus images with OCT or clinical datasets to achieve competitive diagnostic accuracies exceeding 94% (Shafiq et al., 2024 [28]; Raghunathan et al., 2024 [29]; Mehta et al., 2021 [30]). Despite these advancements, such models face significant clinical barriers, including immense computational demands that limit their use in edge environments and their susceptibility to class imbalance and overfitting. Furthermore, their "black-box" nature hinders clinicians' trust, necessitating explainability tools such as gradient-weighted class activation mapping (Grad-CAM) (Benbakreti et al., 2024 [31]; Ejaz et al., 2024 [13]).

2.3 YOLO-Based Detection Models

The YOLO architecture transformed retinal imaging by framing disease identification as a real-time object detection task. Iterations from YOLOv8 to YOLOv12 introduce key optimizations that continuously balance mean average precision (mAP) and latency (Ardelean et al., 2025 [8]). YOLOv10, for instance, removed non-maximum suppression (NMS) to reduce latency. GhostYOLO is a lightweight variant that uses C3Ghost blocks, enabling real-time deployment on edge devices and hardware such as the Jetson Nano (Lokesh et al., 2025 [32]). However, standard YOLO models lack the deep spatial attention required to capture long-range global dependencies in fundus imagery. This capability is crucial for differentiating conditions, such as distinguishing localized MH from widely dispersed DR lesions. Mahapadi et al. (2026 [11]) addressed

this limitation by integrating the Convolutional Block Attention Module (CBAM) into YOLOv10; however, performance remained sensitive to image quality. Similarly, Kumar & Katal (2025 [7]) developed a two-stage pipeline, OD3-YOLO, which isolates the optic disc for glaucoma detection. While these frameworks offer spatial localization and efficiency, they face critical bottlenecks due to their reliance on manual annotations. Furthermore, the model faces a risk of overfitting on small datasets due to a lack of external validation on real-world data (Wang et al., 2025 [33]). All these challenges highlight an ongoing need in ophthalmology for standardized, interpretable, and globally aware YOLO pipelines.

2.4 Lightweight Architectures for Edge Deployment

Standard DL models have high computational demands, which often preclude their use in resource-limited clinical settings. Consequently, recent research has pivoted toward lightweight frameworks optimized for edge computing. For example, GoogLeNet has been modified to optimize GPU memory utilization (Butt et al., 2025 [34]). Mahapadi et al. (2026) [11] applied pruning and quantization to YOLOv10. Lokesh et al. (2025) [32] proposed a GhostYOLO framework. It targets real-time cataract detection on edge hardware, such as the Jetson Nano. However, validation was constrained by a small dataset. These architectures successfully lower computational costs. They achieve this through aggressive feature map compression. However, this approach presents a distinct limitation in ophthalmology. It frequently causes the loss of fine-grained spatial details. These details are necessary for detecting subtle pathologies. Early-stage microaneurysms are one example that highlights a fundamental challenge for medical AI. Medical AI must balance high-fidelity feature extraction with strict computational efficiency. Our proposed single-stage YOLOv11 architecture is specifically designed to achieve this balance.

2.5 Blockchain, ML, and AI in Healthcare

AI deployment faces challenges beyond algorithmic improvements, particularly regarding patient data privacy and cross-institutional data sharing. As recently highlighted by (Malviya et al., 2023 [35]), the integration of blockchain technology is emerging as a transformative solution for managing sensitive EHRs and high-resolution diagnostic data. While global privacy regulations heavily protect medical datasets, including fundus images, which are particularly restricted, blockchain overcomes these barriers by enabling decentralized data sharing. Creating immutable records ensures the transparency, privacy, and integrity of clinical data. Furthermore, ophthalmic AI can utilize blockchain-based ledgers to allow healthcare institutions to securely share AI diagnostic weights across diverse populations without ever exposing raw patient images. Although challenges such as interoperability and storage limitations remain, the convergence of blockchain and ML is fundamental; it enables models to achieve greater demographic robustness and improves clinical decision-making, all while strictly preserving patient confidentiality.

In summary, researchers have extensively explored various DL techniques, including transfer learning, contrastive clustering, and ensemble methods. As shown in Table 1, multimodal fusion, Transformers, and YOLO architectures are also commonly employed to leverage diverse datasets. Together, these advancements enhance feature extraction and model interpretability, thereby improving the early detection of eye diseases. Nevertheless, critical gaps persist; current models frequently struggle with high computational complexity, severe class imbalance, and limited generalization across multiple retinal conditions. These ongoing challenges underscore the need for our proposed approach, a lightweight, multi-class framework designed to address these specific issues.

Table 1: Summary of recent deep learning approaches for retinal disease classification.

Reference	Objective/Task	Dataset(s)	Architecture/ Method	Results/Accuracy	Key Limitations
Ejaz et al., 2024 [13]	Classify DR, MH, and ODC.	RFMiD, RFMiD 2.0	Custom CNNs with data augmentation.	89% accuracy. Establishes a strong baseline.	High computational cost; constrained by limited data size.
Kumar & Katal, 2025 [7]	Localized glaucoma and DR detection.	SMDG, IDRID, REFUGE	OD3-YOLO (Two-stage).	80% F1-score. Strong object-focused detection.	Manual labeling required; added complexity; no external validation.
Mahapadi et al., 2026 [11]	Efficient real-world DR and multi-disease detection.	DIARETDB1, MESSIDOR, APTOS	YOLOv10 + CBAM (with pruning).	88.7% accuracy with significantly reduced overhead.	Black-box system; vulnerable to false results; highly dependent on image quality.
He et al., 2024 [18]	Diabetic Macular Edema (DME) detection.	MFI (Multicolor Fundus)	Transformer-based with global attention.	95.4% accuracy. Excellent capture of global context.	High resource utilization; small dataset; single-condition focus.
Bodapati and Veeranjaneyulu, 2024 [26]	DR Severity Prediction.	APTOS 2019	Ensemble (VGG, ResNet).	81.86%	High computing needs; struggles with class imbalance and subtle stages.
Al-Fahdawi et al., 2024 [36]	Multiple Disease Detection.	OIA-ODIR	Fundus-DeepNet (HRNet).	AUC 99.86%	High computational cost; low interpretability hinders low-resource clinical use.
Alam et al., 2024 [37]	Self-supervised diagnostics.	APTOS 2019	SwAV Contrastive Clustering.	87%	Generalizability and performance on broader, varied datasets remain uncertain.
Proposed Method	Rapid multi-class triage (DR, MH, ODC, WNL) without feature confusion.	RFMiD, RFMiD 2.0	YOLOv11 (Single-Stage) with Pathology Isolation.	94.78% accuracy. Highly efficient pure feature extraction.	Requires future external validation; currently limited to 4 primary classes.

3 Methodology

This research proposes a DL-based methodology for classifying fundus images, designed to mitigate overfitting, reduce computational costs, and expand the range of detectable eye diseases. The methodology integrates pre-trained CNNs with the YOLOv11 model for image classification. As presented in Fig. 1, the proposed methodology begins with dataset collection and description. Images are sourced from RFMiD, RFMiD2.0, and DR-FID. Section 3.1 discusses these sources in detail. Fig. 2 shows the disease selection process. This process is executed in two phases. The first phase filters and formats the CSV data. It retains only image-specific information. The second phase automates image file retrieval and copying via a Python script. Section 3.3 provides further details. The preprocessing phase addresses class imbalance within the

training set. We apply targeted undersampling, oversampling, and data augmentation. Section 3.5 details these steps. Fig. 3 shows the YOLOv11 architecture. Section 3.6 discusses its internal details.

The validation phase monitors the model's performance. We track the top-1 accuracy per epoch. The best weights are then preserved. Next, the test phase applies the best model to unseen test data. The model produces probabilistic outputs. Images are labeled based on confidence thresholds. Finally, we evaluate performance using classification metrics. These include accuracy, precision, recall, and F1-score. We also visualize the results for further insights.

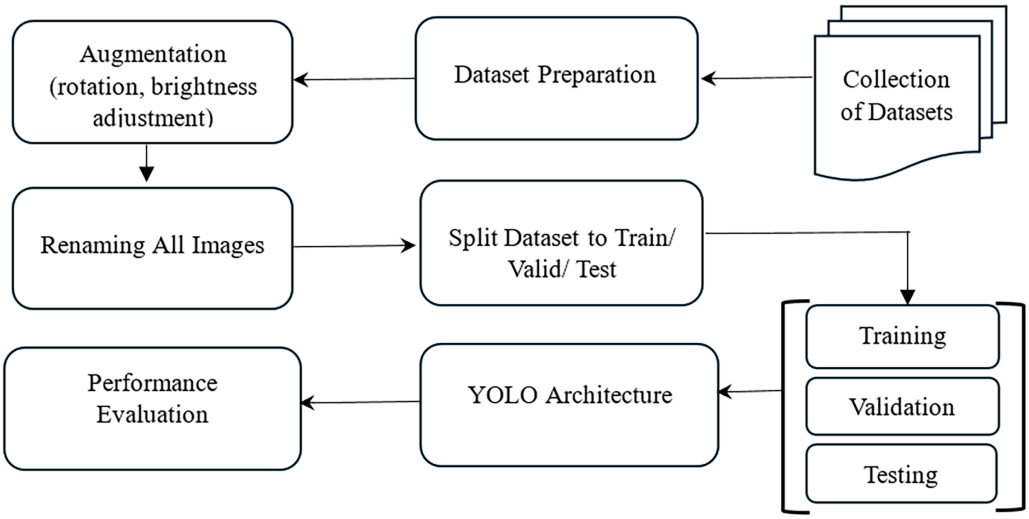

Figure 1: Proposed methodology.

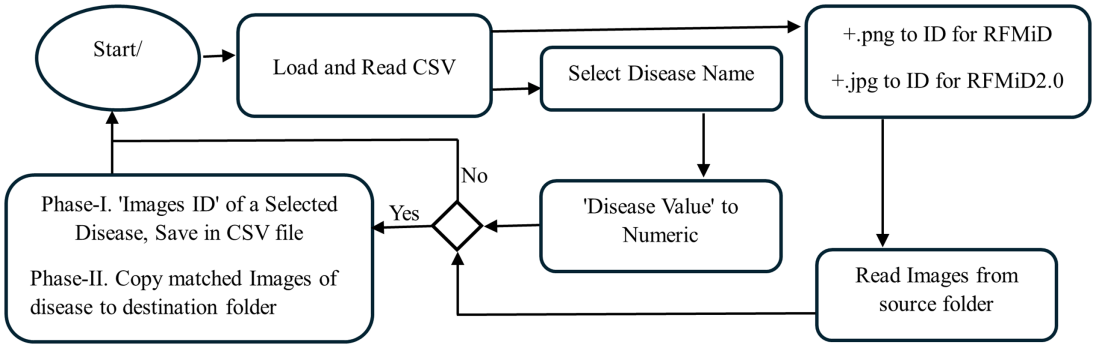

Figure 2: Phase-I & Phase-II diagram.

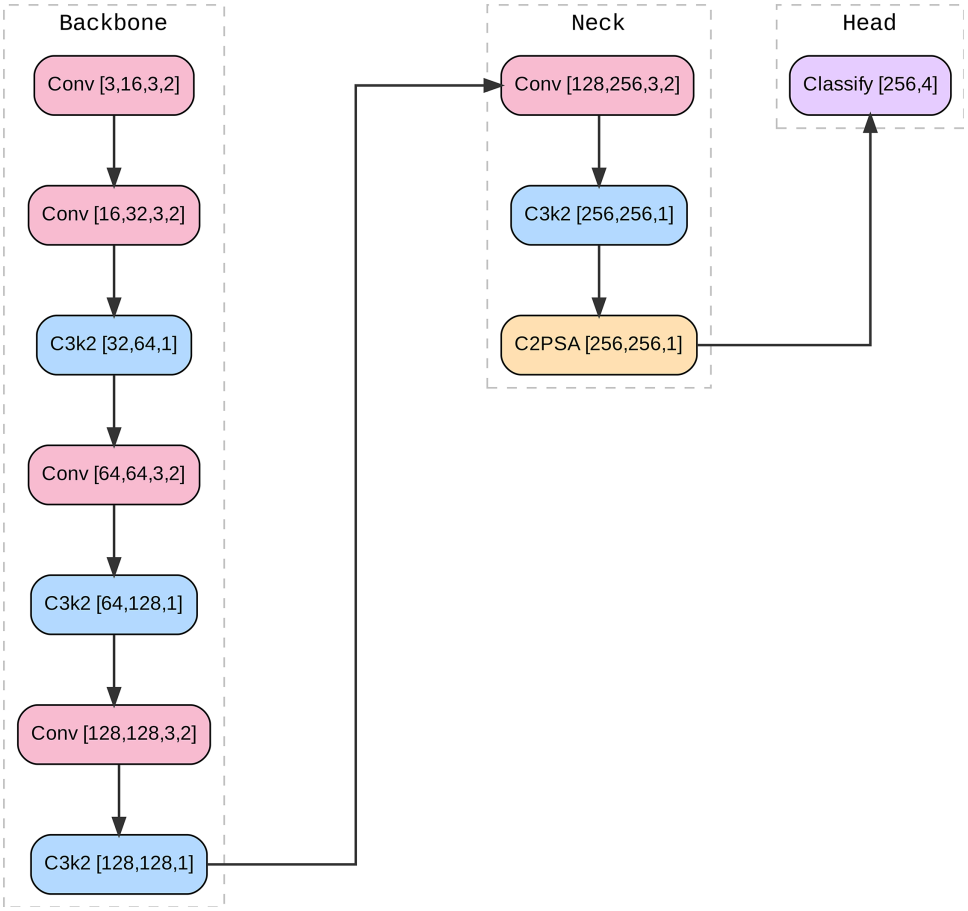

Figure 3: YOLOv11 architecture.

3.1 Dataset Collection and Description

This study uses three publicly available datasets. The first is the RFMiD [15]. It is sourced from IEEE DataHub and contains 3200 fundus images. The second is RFMiD2.0 [16]. It is obtained from Zenodo and comprises 860 fundus images. The raw data underwent a strict pathology-isolation process to prevent confusion over spatial features. We retained only images with single-label diagnoses. These included DR, MH, ODC, and WNL to ensure that network learns distinct features. The third source is the DR-FID [17]. It consists of 1437 color fundus images. These were acquired from the Department of Ophthalmology at the Hospital de Clínicas, Paraguay. Collectively, these datasets provide a diverse spectrum of retinal disease manifestations.

To ensure maximum methodological rigor and prevent data leakage, this study strictly utilizes the official training, validation, and testing partitions provided by the RFMiD and RFMiD2.0 repositories. We acknowledge a minor discrepancy in the total image count compared to the baseline study [13].

As shown in Table 2, the RFMiD dataset contains 927 images, and RFMiD2.0 contains 219 images, yielding a total of 1146 images across the DR, MH, ODC, and WNL classes. Removing corrupted images reduced the total number of disease instances in the RFMiD and RFMiD2.0 datasets compared to the baseline study [13]. Specifically, instances of MH dropped from 194 to 189, ODC from 126 to 125, and WNL from 558 to 550. The model analyzed only usable images.

Table 2: Distribution of training data.

Classes	Training		
	RFMiD	RFMiD2.0	Total
DR	240	42	282
ODC	118	7	125
MH	175	14	189
WNL	394	156	550

Note: DR = Diabetic Retinopathy, MH = Media Haze, ODC = Optic Disc Cupping, WNL = Within Normal Limits (Healthy/Normal Control Class).

DR-FID (Table 3) contains 1437 unaugmented fundus images. The "No DR signs" class is the largest, with 711 images. The "Severe" class has 210 images, "Advanced PDR" has 145, "Very Severe" has 139, and "PDR" has 116. The "Moderate" and "Mild" classes contain 110 and 6 images, respectively, indicating class imbalance. This distribution reflects real-world DR prevalence and supports preprocessing and augmentation for better model training.

Table 3: DR-FID—training.

Class	Original	Augmented	Class	Original	Augmented
NoDR	711	400	Very Severe	139	400
Mild	6	300	PDR	116	400
Moderate	110	400	Advanced PDR	145	400
Severe	210	400			
			G. Total	1437	2700

3.2 Selection of Diseases

The source repositories lack patient-level metadata. Therefore, we conducted data partitioning and analyses strictly at the image level. The RFMiD and RFMiD2.0 datasets collectively contain 57 clinical classes. However, we deliberately isolated a four-class subset: DR, MH, ODC, and WNL (Table 2). This targeted selection serves two primary purposes. First, it provides a standardized comparative benchmark and aligns directly with the baseline established by Ejaz et al. (2024) [13]. Second, differentiating distinct anatomical pathologies requires specialized feature-extraction mechanisms. These mechanisms differ fundamentally from those needed for fine-grained severity grading. For example, distinguishing between No DR and Advanced PDR requires highly localized features (Table 3). Consequently, we treat multi-stage DR grading as an independent computational task. This dual-evaluation strategy explicitly evaluates the YOLOv11 architecture. It evaluates the model's capacity for broad multi-disease classification and assesses its ability to discriminate subtle severities across all included datasets.

3.3 Single-Pathology Case Filtering and Image Retrieval

The first phase of dataset preparation, as shown in Fig. 2, takes a CSV file path as input and is described in Algorithm 1. It ensures that the first column represents the image ID and the subsequent columns represent disease names. The procedure then loads this data into a DataFrame for inspection. The target disease column is selected, and the process retains only rows where the target disease is present (value 1) and all others are

absent (value 0). Finally, the procedure extracts and cleans the image IDs and saves the filtered data to a new CSV file. The same process is described below in mathematical form. Let:

a. x be the value in the target column for row i.
b. y_{ij} be the value in the j^{th} column (with $j \geq 2$) for row i.
c. n is the total number of columns in the DataFrame.

We define the indicator function for row i that satisfies the condition as:

$$I(i) = I\{x_i = 1\} \cdot \prod_{j=2}^{n} I\{y_{ij} = 0\} \tag{1}$$

Here, $I\{\cdot\}$ is an indicator function that takes the value 1 when the condition is true and 0 when it is false. Thus, a row i is included in the filtered DataFrame if:

$$I(i) = 1 \iff (x_i = 1 \text{ and } \forall j \in \{2, 3, \ldots, n\}, y_{ij} = 0)$$

Otherwise, if any of these conditions fail, $I(i) = 0$ and the row is excluded, which mathematically describes the criterion used to select rows for inclusion in the output CSV file.

Algorithm 1: Selection of single disease

Input:
  DF ← DataFrame with n columns
  target_column ← column used to check whether the value equals 1
Output: Filtered_DF ← DataFrame containing rows that meet the condition
// Procedure
Filtered_DF ← ∅ ▷ Initialize an empty DataFrame
for all row_i ∈ DF **do**
  x ← row_i[target_column]
  Y ← row_i[columns from index 2 onward]
  if $x = 1$ **and** $\forall y \in Y : y = 0$ **then**
    Append row_i to Filtered_DF
  end if
end for
Return: Filtered_DF

In phase two, as described in Fig. 2, the procedure automates the copying of image files. It reads a CSV file and extracts unique names from the first column. It prompts the user for source and destination folders, and creates the destination folder if needed. The script appends '.png' for RFMiD or '.jpg' for RFMiD2.0. The system copies the file if it exists. This process repeats for all names and terminates with a success message. A mathematical description of the process is given below.

a. Unique File Count: Let F be the set of non-NaN file names extracted from the first column of the CSV. Then, the total number of unique file names is:
   $N = |F|$
b. File Name Extension Adjustment: As the CSV source file did not contain file extensions, for each file name $f \in F$, we define an adjusted file name f^a as follows. In the case of RFMiD:
   $$f^a = \begin{cases} f, & \text{if } f \text{ ends with ".png"} \\ f + \text{".png"}, & \text{otherwise} \end{cases}$$
   This ensures that every file name ends with .png. In the case of RFMiD2.0:

$$f^a = \begin{cases} f, & \text{if } f \text{ ends with ".jpg"} \\ f + \text{".jpg"}, & \text{otherwise} \end{cases}$$

This ensures that every file name ends with .jpg.

c. File Existence and Copying: Let S denote the set of file names present in the source folder. For each adjusted file name f^a, we define an indicator function as follows:

$$I(f^a) = \begin{cases} 1, & \text{if } f^a \in S \text{ (i.e.,file exists)} \\ 0, & \text{if } f^a \notin S \end{cases}$$

The total number of files copied, C, is then given by:

$$C = \sum_{f \in F} I(f^a) \tag{2}$$

3.4 Rationale for the Exclusion of Multi-Pathology Cases

The RFMiD and RFMiD2.0 datasets include multi-disease retinal presentations. However, we intentionally restricted our analysis to isolated cases of three specific pathologies (DR, MH, ODC) and healthy controls (Fig. 2). This single-pathology constraint ensures a direct comparison and provides a standardized baseline against the work of [13]. While clinical reality often involves concurrent ocular conditions, we filtered out overlapping pathologies as a deliberate methodological choice to establish a controlled algorithmic baseline. DL models are sometimes trained initially on multi-label data. These models frequently suffer from spatial feature confusion. They struggle to attribute specific visual biomarkers to their respective labels accurately. We strictly isolated the pathologies to enable the YOLOv11 architecture to learn distinct morphological features of each class. This step is scientifically necessary. It validates the architecture's baseline feature-extraction capabilities. This validation must occur before addressing multi-label disease disentanglement. That task is exponentially more complex.

3.5 Data Preprocessing and Augmentation

Severe class imbalance was present in the dataset. We mitigated this while preserving the integrity of the baseline data. To achieve this, we employed a hybrid augmentation strategy.

3.5.1 Data Augmentation Parameters

We developed a custom Python script. This script is detailed in Algorithm 2. It applies targeted data augmentation exclusively to the training set, mitigates class imbalance and enhances model robustness. The script dynamically oversamples underrepresented classes. It matches the majority class frequency to achieve a strict 1:1 balanced distribution. The pipeline utilizes OpenCV to generate new samples. It applies discrete orthogonal rotations (90°, 180°, 270°), to prevents interpolation blurring. It also applies bounded brightness scaling (±20%) to simulate varying fundus camera illumination without distorting clinical features. The process automatically saves these augmented images. It assigns unique filenames and continues until class parity is reached. This automated process ensures equitable gradient updates and prevents majority-class bias during model training.

Classes $C_1, C_2, \ldots, C_k$ are adjusted so that each class has an equal number of images $N_{\max}$, where:

$N_{\max} = \max(N_1, N_2, \ldots, N_k)$

For each class C_i with image count N_i, augmented images are generated as follows:

$A_i = N_{\max} - N_i$

a. $N_{\max}$: The images in majority class

b. N_i: The images in current class C_i
c. A_i: The difference of augmented images required to bring class C_i up to the same level as the majority class This ensures that after augmentation, the updated image count N_i' for class C_i becomes: $N_i' = N_i + A_i = N_{\max}$

Algorithm 2: Image augmentation & balancing

```
Input: dataset_path with class subfolders
Output: Balanced dataset with augmented images
    // Count Images
for all class_folder ∈ dataset_path do
        class_counts[class] ← count(JPG, PNG)
end for
    // Target Count
target ← max(class_counts)
    // Balance Classes
for all (class, count) ∈ class_counts do
     for i ← 1 to (target-count) do
          Select random image ∈ class
          Apply augmentation (rotate/brightness)
          Save as "aug_i.jpg" in class
     end for
end for
```

Each augmented image $\tilde{x} \in C_i$ is generated by applying a transformation T to a randomly selected image $x \in C_i$:

$$\tilde{x} = T(x) \tag{3}$$

where $T \in$ rotation, brightness adjustment, and $\forall i,\ N_i' = N_{\max}$.

Table 2 shows class imbalance (WNL = 550, ODC = 125), which was addressed by oversampling to obtain 550 images per class except MH (551), yielding 2201 images. In Table 2, under-sampling and augmentation were applied to produce DR-FID, with 400 images per DR class (300 for Mild), totaling 2700 images across seven classes.

3.5.2 Clinical Justification for Augmentation Parameters

Spatial augmentation was strictly limited to discrete orthogonal rotations ($90°, 180°, 270°$) while preserving sensitive diagnostic micro-features. This exact transposition of the pixel matrix prevents interpolation-induced blurring. Arbitrary angular rotations often cause blurring of fine lesions, such as microaneurysms. Additionally, photometric adjustments were constrained to a ±20% brightness range, conservatively simulating natural variations in camera flash intensity and pupil dilation. It also maintains essential local contrast gradients, which the YOLOv11 network requires to isolate retinal pathologies successfully.

3.5.3 Data Sanitization and Integrity

Merging the RFMiD and RFMiD2.0 datasets introduces a risk of file-name collisions. This risk arises from overlapping sequential nomenclature. A custom Python script was deployed to prevent accidental overwriting and to ensure absolute data integrity during image aggregation. The script iterates through all clinical directories (DR, MH, ODC, and Normal). It assigns a unique, randomly generated alphanumeric identifier to each image to standardize the dataset structure for seamless YOLOv11 training.

Let:

a. $F = \{f_1, f_2, \ldots, f_n\}$ be the set of original file names in the directory.
b. $E(f)$ be the file extension of filename f.
c. G be a function that generates a random alphanumeric string of length l, i.e., $G : \mathbb{N} \to \Sigma^l$, where $\Sigma = [\text{a–z, A–Z, 0–9}]$
d. $R \subset \Sigma^l$ be the set of all already generated names (to avoid duplicates). Then, for each file $f_i \in F$:

$$\begin{aligned} \text{new\_name}_i &= \text{Unique}(G(l)) \\ \text{where Unique}(G(l)) &\notin R \\ f_i' &= \text{new\_name}_i + E(f_i) \\ \text{Rename}(f_i) &\to f_i' \end{aligned} \tag{4}$$

where:

a. $G(l)$ generates a random string of length l
b. $\text{Unique}(G(l))$ ensures that the generated name is not already used (no collisions)
c. $E(f_i)$ extracts the extension of the original filename
d. f_i' is the new filename with the same extension. Each file f_i is renamed to f_i' while maintaining the file extension.

3.5.4 Partitions of Dataset

We used official training, validation, and test partitions from the RFMiD and RFMiD2.0 datasets to ensure methodological integrity and prevent data leakage. First, we applied our "Pathology Isolation" and Filtering Module independently to each partition. This step isolated the four target clinical classes (DR, MH, ODC, WNL). Next, we addressed the inherent class imbalance. We applied a targeted spatial and photometric augmentation pipeline that included orthogonal rotations (90°, 180°, 270°) and brightness scaling (±20%) strictly to the training partition. By oversampling minority classes exclusively within the training set, we achieved a balanced 1:1 class distribution for optimal gradient updates. Crucially, the validation and testing sets were left entirely unaugmented. We excluded 29 images due to corruption in the source file (Error 0x80004005). This loss constitutes 2.47% of the total data. Specifically, only 9 images from the RFMiD test partition were unrecoverable. These represent 2.37% of the test dataset; their exclusion does not significantly impact the comparative metrics. Furthermore, our study strictly adheres to the official dataset partitions to avoid the data leakage risks inherent to the custom-split methodology [13]. This strict separation supports generalization assessment on unseen clinical data. It eliminates the risk of artificially inflated metrics from data leakage, ensuring a more conservative and clinically reliable interpretation of our model's performance.

3.6 YOLOv11 Architecture and Framework Overview

YOLOv11 was selected for its unique feature extraction capabilities. It simultaneously extracts multiscale local features, including microscopic lesions. It also maintains global spatial awareness. Two specialized

mechanisms achieve this. First, the C3k2 module dynamically adjusts kernel sizes to capture fine-grained details without degradation. Second, the C2PSA (Cross-Stage Partial Self-Attention) module provides global attention for contextualizing structural changes against background retinal textures. YOLOv11 is a single-stage network. It processes the entire fundus image in a single forward pass. This design efficiently isolates subtle pathological markers and avoids the computational bottlenecks of two-stage models.

The architecture comprises three key components, as shown in Fig. 3:

a. Backbone: Utilizes C3k2 modules for efficient multi-scale feature extraction. It downsamples from 16 to 256 channels.
b. Neck: Integrates the C2PSA module, which focuses spatial attention exclusively on medically relevant regions.
c. Head: Employs Global Average Pooling (GAP) followed by a linear Softmax classifier. It outputs probabilities for the four target classes (DR, MH, ODC, WNL).

3.7 Multi-Class Classification Strategy

The classification head utilized Cross-Entropy Loss, rigorously penalizing misclassifications across the four clinical categories. We utilized an increased input resolution of 512 × 512 pixels. An early stopping mechanism regulated the training cycle to prevent overfitting. It halted execution at 22 epochs—total training completed in 2.131 h. The mechanism successfully isolated the best-performing weights at epoch 12. Post-training structural fusion condensed the architecture. It reduced the model to 47 active layers and 1,531,148 parameters, which resulted in a highly compact 3.2 MB weight file (best.pt). The optimized model requires only 3.2 GFLOPs.

3.8 Adaptation of YOLOv11 for Image Classification

YOLOv11 is traditionally used for object detection. This study adapts it exclusively for multi-class image triage. It bypasses standard bounding-box regression heads and their loss functions. The architecture uses the Cross-Stage Partial (CSP) backbone strictly as a feature extractor. The network routes these feature maps into a classification head. This head uses Global Average Pooling (GAP). A fully connected linear layer follows this adaptation, which condenses multi-scale spatial data, such as minute microaneurysms. It produces a single diagnostic prediction for the entire fundus image.

Fig. 4 displays a side-by-side comparison of two architecture flowcharts. These are the C3k2 and C2PSA Internal Flows. Both modules share a CSP split-branch structural design. However, they differ in their core processing mechanisms:

a. C3k2 Module (Hierarchical Feature Extraction): This acts as the primary feature engine in the Backbone and lower Neck. It passes the input feature map through an initial 1 × 1 convolution. Then, it splits the gradient flow. Branch 1 routes through N Bottleneck blocks (3 × 3 Conv) to dynamically capture multi-scale spatial features, such as microscopic lesions. Branch 2 acts as an identity connection to mitigate vanishing gradients. The branches are concatenated. They are then fused via a final 1 × 1 convolution.
b. C2PSA Module (Global Contextualization): This module is positioned at the end of the Neck. It uses the same split-branch architecture. However, it replaces the convolutional bottlenecks in Branch 1. It uses a Multi-Head Spatial Attention (PSA) block instead, which allows the network to model long-range spatial dependencies. The branches are concatenated and undergo a final 1 × 1 convolution. The module outputs an attended feature map to prioritize diagnostically relevant features. It also suppresses irrelevant background noise, like lighting artifacts.

c. Supporting Components (Conv & Classify): Standard 3×3 Convolutional blocks exist beyond the internal flows. They systematically downsample spatial dimensions. They also double the channel depth to compress raw pixels into dense semantic features. Finally, the Classification Head applies GAP. It uses a linear layer to output a probability distribution over the four clinical classes.

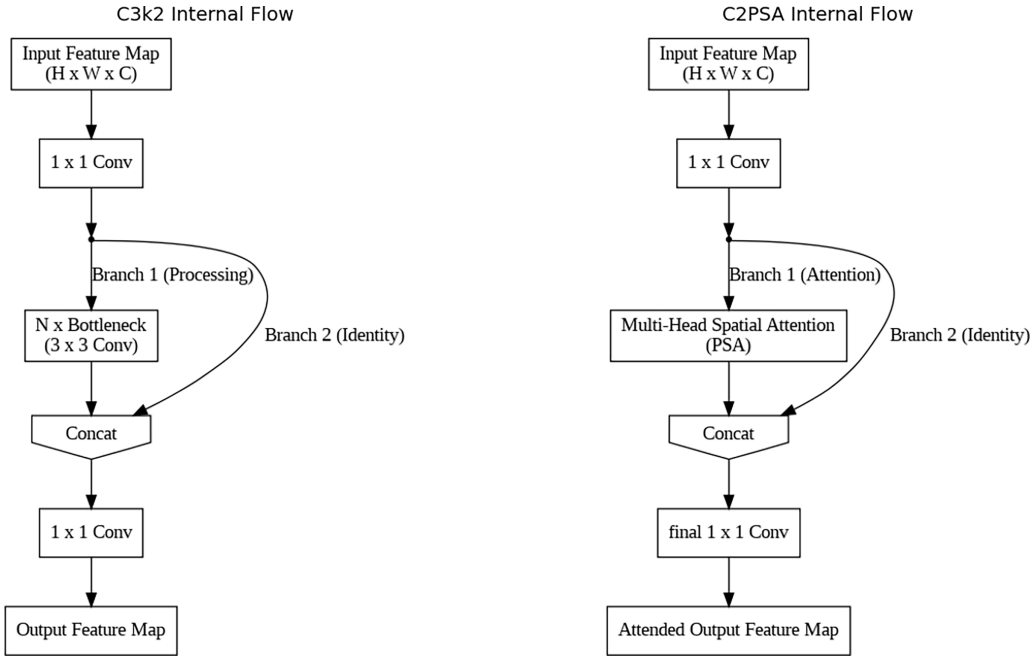

Figure 4: Internal mechanisms of C3k2 and C2PSA.

3.9 C2PSA: Justification and Spatial Attention Mechanism

The C2PSA module balances global spatial awareness with computational efficiency. It outperforms heavy Transformer mechanisms. It also outperforms simple attention modules such as SE or CBAM. Structurally, C2PSA employs a partial processing strategy. It splits the incoming feature map channels. One subset preserves baseline structural gradients. The other feeds into a Spatial Attention module, which computes a spatial weight mask. Pathological biomarkers are highly sparse in ophthalmic imaging. They often occupy less than 1% of total pixels. They appear against a uniform, healthy background. Standard convolutions frequently dilute these weak signals during downsampling. The C2PSA spatial attention mask acts as an active biological filter. It mathematically upweights these sparse anomalies. It also suppresses repetitive noise from healthy tissue, forcing the architecture to focus computational power exclusively on clinically diagnostic regions.

3.10 Training Phase

The model was trained using the Adam optimizer. The initial learning rate was 0.001. The input resolution was increased to 512×512 pixels. The dataset partition included 2201 training images. It also had 364 validation and 370 test images. The training was configured for a maximum of 40 epochs.

3.10.1 Dataset Configuration

To prepare for model training and evaluation, the dataset distribution and training configurations are defined as follows:

a. D_{train} = 2201—No. of training images
b. D_{val} = 364—No. of validation images
c. D_{test} = 370—No. of test images
d. C = 4—No. of output classes
e. E = 40—Maximum training epochs (Early stopping at 22)
f. B = 64—Batch size
g. I = 512 × 512—Input image resolution

3.10.2 Model Parameters

The trainable parameters and computational requirements are defined as follows:

a. Number of layers (Unfused/Fused); L = 86/47
b. Trainable parameters; P = 1,536,228 (Fused: 1,531,148)
c. GFLOPs; G = 3.3 (Fused: 3.2)

3.10.3 Optimizer and Hyperparameters

The model's training process was configured using the following specific optimizer hyperparameters:

Learning Rate η = 0.001,

Momentum β_1 = 0.937,

Weight Decay λ = 0.0005

3.10.4 Training Objective

The objective is to minimize the loss over the dataset:

$$\min_{W} \mathcal{L}(f_W(x), y) \tag{5}$$

where:

a. $f_W(x)$ is the model's prediction
b. $\mathcal{L}$ is the classification loss function
c. W represents model weights

The network was optimized using the Adam optimizer with an initial learning rate of 0.001. Training was conducted for up to 40 epochs. To prevent overfitting and ensure efficient convergence, an early stopping strategy with a patience of 10 epochs was employed.

3.11 Performance Evaluation Metrics

Final evaluation metrics, namely accuracy, precision, recall, and the F1 score, were computed and visualized via confusion matrices. The raw predictions were retained for further analysis; full details are available in the Experimental Results Section 4.

First, the foundational variables comprising a confusion matrix are defined as follows:

TP = True Positives (Correctly predicted positive cases)

TN = True Negatives (Correctly predicted negative cases)

FP = False Positives (Incorrectly predicted as positive)

FN = False Negatives (Incorrectly predicted as negative)

a. Accuracy: Accuracy measures the overall proportion of correct predictions (both positive and negative) out of the total number of cases.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{6}$$

b. Precision: Precision (also known as Positive Predictive Value) measures the proportion of positive predictions that were actually correct. It answers the question: Out of all the cases the model predicted as positive, how many were actually positive?

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

c. Recall: Recall (also known as Sensitivity or True Positive Rate) measures the proportion of actual positive cases that the model successfully identified. It answers the question: Out of all the actual positive cases, how many did the model find?

$$\text{Recall} = \frac{TP}{TP + FN} \tag{8}$$

d. F1 Score: The F1 Score is the harmonic mean of Precision and Recall. It is especially useful for imbalanced datasets because it provides a single metric that balances both false positives and false negatives.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

4 Experimental Results

4.1 Implementation Details and Experimental Setup

We built the classification pipeline using Ultralytics (v8.4.36). We also used PyTorch 2.10.0 and Python 3.12.13. We executed training and evaluation in Google Colab. We used an NVIDIA Tesla T4 GPU. This GPU has ~15 GB VRAM. We accessed the datasets via cloud storage and used a standardized YAML Ain't Markup Language (YAML) configuration file to manage dataset paths and experimental settings [38]. We strictly maintained the predefined partitions, preventing data leakage.

4.2 Lightweight Optimization Techniques

We aimed to ensure the viability of edge deployment in resource-constrained clinics. Therefore, we utilized the "Nano" variant of YOLOv11 (yolo11n-cls). This streamlined model comprises 86 layers and 1,536,228 parameters during training. It optimizes down to a fused architecture. This final architecture has just 47 layers and 1,531,148 parameters. It requires a mere 3.2 GFLOPs for inference. We scaled input images to 512 × 512 pixels to balance throughput with high-resolution feature extraction. The model processed them efficiently with a batch size of 64. We optimized the network using Adam. The initial learning rate was 0.001. The momentum was 0.937. The weight decay was 0.0005. We configured training for a maximum of 40 epochs. We used early stopping (patience = 10) to prevent overfitting. During execution, training stopped early at 22 epochs. The most optimal weights were saved at epoch 12. The model achieved a robust Top-1 validation accuracy of 89.6%. We recorded an inference latency of just 0.5 ms per image on an NVIDIA Tesla T4. Preprocessing required an additional 0.2 ms. Consequently, the framework remains exceptionally optimized for real-time, low-resource deployment.

4.3 Validation Phase

To evaluate deployment feasibility, the model was tested on a successfully loaded validation set comprising 364 images using a standard Intel Xeon CPU without GPU acceleration. As shown in the training/validation loss and accuracy curves in Fig. 5, the YOLOv11 model achieved 89.6% accuracy. Furthermore, post-training optimization (layer fusion) resulted in an inference latency of just 16.6 ms per image (~60 FPS). These results confirm the model's high suitability for real-time clinical diagnosis on low-resource hardware.

a. N_{val} = Total number of validation samples
b. N_{correct} = Number of samples correctly predicted (top-1 match)
c. $\hat{y}_i$ = Model's predicted class for image i
d. y_i = Ground truth class for image i

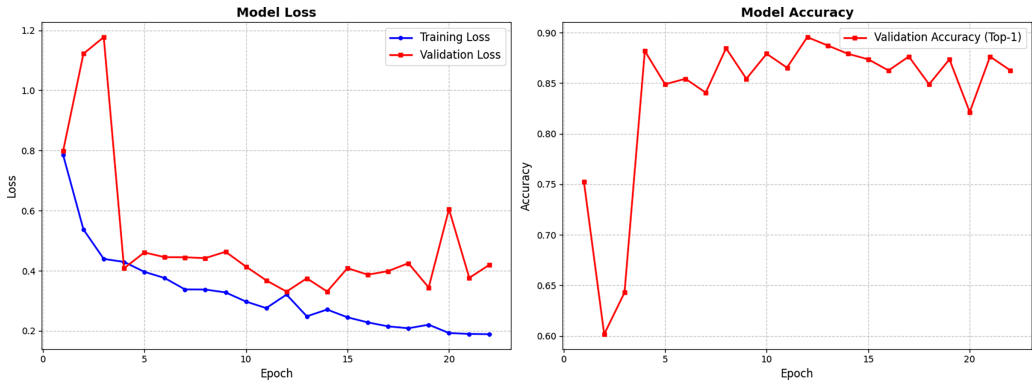

Figure 5: Training and validation loss vs. accuracy.

Finally, the training loss declined from 0.79 in the first epoch to 0.19 by epoch 22, indicating strong convergence:

$$L_1 = 0.79 \text{ and } L_{22} = 0.19 \quad \Rightarrow \quad \frac{dL}{de} < 0$$

Meanwhile, the training accuracy increased from 75% in the first epoch, reaching a peak of 89.6%. Let $A^{(e)}$ be the top-1 accuracy at epoch e, then the output is:

$$A^{(1)} = 75\%, \qquad A^{(5)} = 84.9\%,$$
$$A^{(10-20)} = 82.1 - 89.6\%, \qquad A^{(22)} = 86.3\%$$

The final best top-1 accuracy achieved across all epochs is:

$$A^{\text{best}} = \max_{1 \le e \le 12} A^{(e)} = 89.6\%$$

Then Top-1 Accuracy A_{val} is calculated as:

$$A_{\text{val}} = \frac{N_{\text{correct}}}{N_{\text{val}}} = \frac{1}{N_{\text{val}}} \sum_{i=1}^{N_{\text{val}}} \mathbf{1}(\hat{y}_i = y_i) \tag{10}$$

$$A_{\text{val}} = 89.6\%$$

4.4 Test Phase

Following hyperparameter tuning and validation, the model was evaluated on an independent test set comprising 370 images (distinct from the training and validation splits). The model achieved a Final Test Accuracy of 89.5%. The minimal discrepancy between validation accuracy (89.6%) and test accuracy (89.5%) indicates robust external generalization capabilities and a lack of overfitting. Furthermore, inference latency remained consistent at 16.3 ms on a standard CPU, reinforcing the model's suitability for real-time clinical screening applications.

Let:

a. x_i be an input test image
   where $i = \{1, 2, \ldots, N\}$
b. $\hat{y}_i = f(x_i; \theta)$ is the predicted probability vector using model parameters
c. $C = \{c_1, c_2, \ldots, c_k\}$ is the set of $k = 4$ classes (WNL, ODC, DR, MH)
d. $\hat{y}_i^j \in [0, 1]$ is the predicted confidence score for class c_j of image x_i

Then the output for each image can be defined as:

$$\hat{y}_i = \left[\hat{y}_i^1, \hat{y}_i^2, \hat{y}_i^3, \hat{y}_i^4\right], \quad \text{such that} \quad \sum_{j=1}^{4} \hat{y}_i^j = 1$$

The predicted class is:

$$\text{Predicted Class}(x_i) = \arg\max_j \hat{y}_i^j \tag{11}$$

If $\max_j \hat{y}_i^j \geq \tau$, where $\tau = 0.52$ (confidence threshold), then the prediction is accepted; otherwise, it may be discarded. This equation describes how each image is passed through the YOLOv11 model and classified based on the highest probability score, with a confidence threshold applied to filter results.

Empirical Determination of Confidence Threshold

A default decision boundary is often suboptimal in medical imaging due to high morphological overlap between clinical classes. We conducted an empirical threshold sweep to identify the mathematically optimal operating point. We evaluated the augmented model's predictions. We used a sliding confidence threshold (τ). This threshold ranged from 0.05 to 0.95. Fig. 6 shows the resulting curve. The overall F1-score represents the harmonic mean of Precision and Recall. This score peaks precisely at $\tau = 0.52$. Higher thresholds severely degraded recall, leading to more false negatives for subtle anomalies. Lower thresholds compromised precision, leading to more false alarms. We set the confidence threshold to $\tau = 0.52$, which optimized the precision-recall trade-off. It yielded an overall F1-score of 86.0% to ensure robust, balanced diagnostic reliability across the test set.

4.5 Evaluation of Validation Datasets

The dataset partition included 2201 training images. It also had 364 validation and 370 test images. These were distributed across four clinical classes. Fig. 5 illustrates the trajectories of training and validation accuracy vs. loss. The training was configured for a maximum of 40 epochs. However, it was automatically halted at 22 epochs. An early stopping mechanism (patience = 10) prevented overfitting. Top-1 validation accuracy rose rapidly from 75.3% at epoch 1 to 84.9% by epoch 5, and the best-performing weights were successfully isolated at epoch 12. It achieved a peak validation accuracy of 89.6%. GPU memory utilization

peaked at approximately 4.68 GB on the NVIDIA Tesla T4. Automatic Mixed Precision (AMP) was enabled to optimize computational efficiency. The entire training cycle was completed in 2.131 h. Overall, the optimized pipeline effectively fine-tuned the YOLOv11 architecture for this classification task.

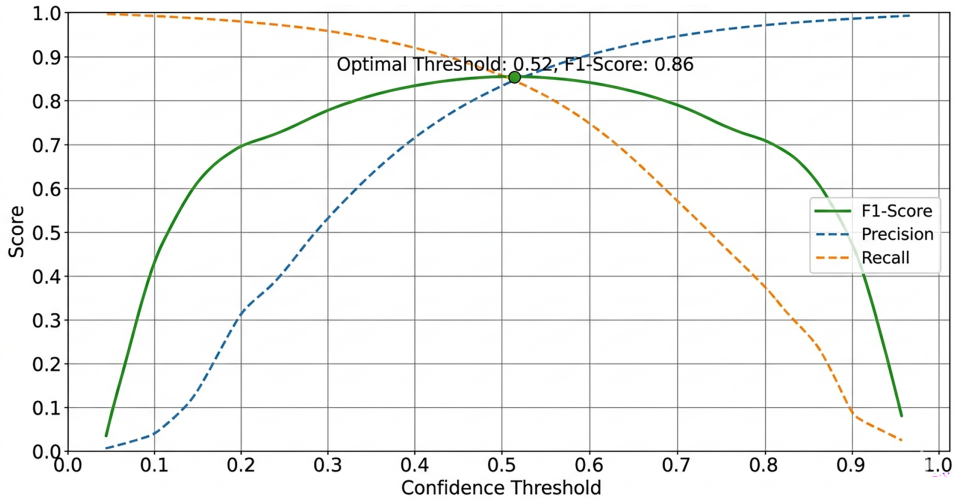

Figure 6: F1-score vs. confidence threshold.

Analysis of the validation confusion matrix (Table 4) highlights the ongoing challenge of class imbalance. Specifically, the WNL class dominates with 185 total samples, yielding 174 true positives (TPs). In contrast, the ODC class contains only 23 samples, yielding only 10 TPs. The model exhibits a bias toward the majority class, evidenced by frequent WNL–ODC misclassifications (e.g., 8 actual ODC cases were incorrectly predicted as WNL). In the validation phase, the ODC class achieved a sensitivity (recall) of 43.48% and an F1-score of 51.28% as shown in Fig. 7, indicating that the model still failed to identify over half of the positive ODC cases, which suggests that future iterations could benefit from targeted oversampling or adjusted loss-weighting strategies to resolve the persistent imbalance fully.

Table 5 shows bias toward "Normal," missing rare classes like "Mild," while augmented data yields high true positives with minimal misclassifications. Fig. 8 shows that class imbalance harms rare classes like "Mild" (0% scores), while common ones perform well (e.g., Normal 96.5% sensitivity). After augmentation, accuracy improves to 96%–99.8%, with sensitivity, specificity > 92% and F1 > 87%, confirming robust performance.

Table 4: Validation RFMiD and RFMiD2.0—confusion matrix.

Class	DR	MH	ODC	WNL
DR	89	1	0	0
MH	1	53	5	5
ODC	0	0	10	6
WNL	5	7	8	174

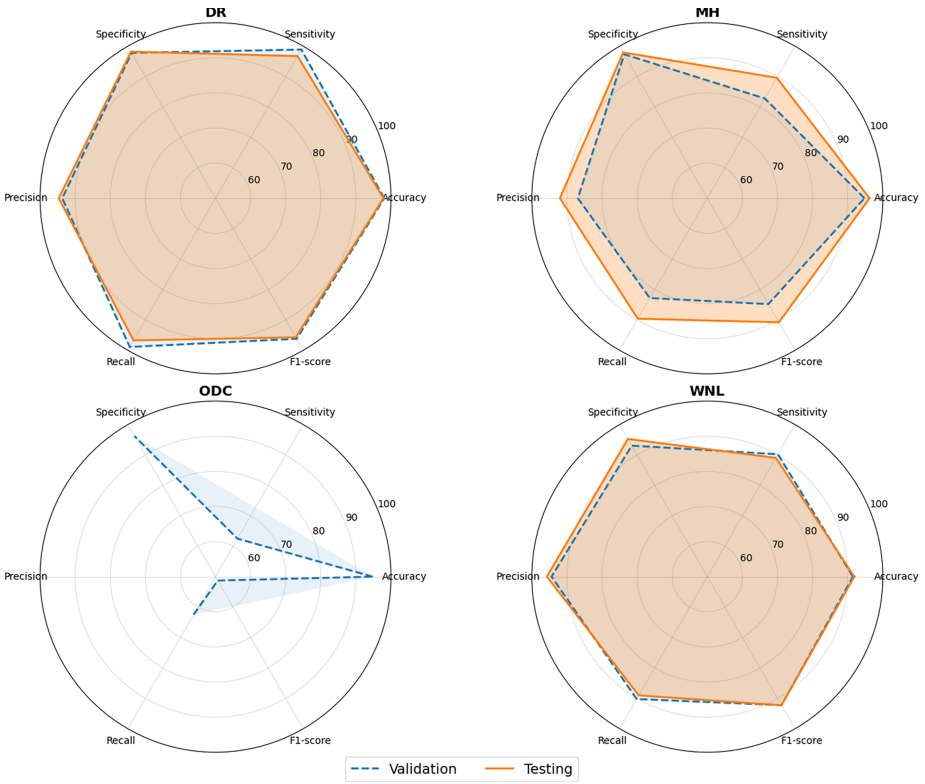

Figure 7: RFMiD and RFMiD2.0—results.

Table 5: Validation DR-FID—confusion matrix.

Class	Advanced PDR	Mild	Moderate	Normal	PDR	Severe	Very Severe
Advanced PDR	25	0	0	0	1	1	1
Mild	0	0	0	0	0	0	0
Moderate	0	0	12	2	0	0	1
Normal	3	2	4	138	0	2	1
PDR	2	0	2	2	15	0	1
Severe	0	0	3	0	4	35	5
Very Severe	0	0	1	1	4	4	20

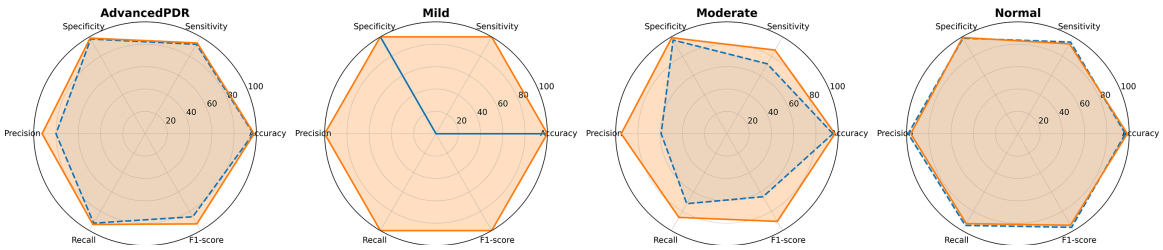

Figure 8: (Continued)

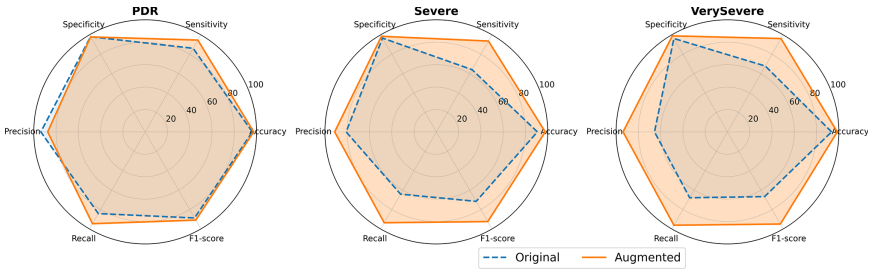

Figure 8: DR-FID—results.

4.6 Evaluation of Test Dataset

As detailed in Table 6, the evaluation of the independent test dataset (N = 370) yielded an overall test accuracy of 89.5%, which is further compared against the validation performance in Fig. 7. The model demonstrated robust generalization across most classes, successfully identifying 171 out of 179 WNL cases, 89 out of 94 DR cases, and 69 out of 75 MH cases. However, the confusion matrix shows the ongoing impact of class imbalance on the minority class's performance. Specifically, the model struggled with the ODC class, which contained only 22 ground-truth samples. Of these, the model correctly predicted only 2 cases, while severely misclassifying 16 ODC cases as WNL, indicating that, while overall accuracy remains high due to the dominance of the WNL and DR classes, the model exhibits a majority-class bias when evaluating morphologically subtle anomalies like ODC. The high frequency of ODC-WNL misclassifications highlights a specific area for future improvement, suggesting that subsequent iterations would benefit from aggressive, targeted oversampling of the ODC class or the implementation of cost-sensitive loss functions to penalize minority-class misclassifications more heavily.

Table 6: Test RFMiD and RFMiD2.0—confusion matrix.

Class	DR	MH	ODC	WNL
DR	89	1	2	0
MH	1	69	2	5
ODC	1	2	2	3
WNL	3	3	16	171

Table 7 highlights severe class imbalance, with 'Mild' (1 sample) often misclassified and 'Normal' dominating, limiting generalization. Fig. 8 shows that augmentation markedly improved performance as validation accuracy rose from 83.90% to 93.52%, and test accuracy from 86.62% to 90.37%. These gains demonstrate the value of augmentation in enabling robust, reliable classification, especially in clinical contexts.

Table 7: Test DR-FID—confusion matrix.

Class	AdvPDR	Mild	Moderate	Normal	PDR	Severe	Very Severe
Advanced PDR	11	0	0	0	1	0	0
Mild	0	1	0	0	0	0	0
Moderate	0	0	8	0	1	1	0
Normal	0	0	0	71	1	0	0
PDR	1	0	1	0	6	0	0

(Continued)

Table 7 (continued)

Class	AdvPDR	Mild	Moderate	Normal	PDR	Severe	Very Severe
Severe	0	0	1	0	2	17	3
Very Severe	0	0	1	0	0	3	10

4.7 Comparative Ablation Study

This section presents a comparative ablation study to evaluate how label complexity impacts the performance of the proposed model. To achieve this, we compared two distinct experimental frameworks: a multi-pathology detection approach and a single-pathology classification approach. First, the multi-pathology framework utilized the full RFMiD and RFMiD2.0 datasets, which contain 57 disease classes with frequently co-occurring pathologies. Analyzing these complex cases with a YOLOv11 detection model exposed significant structural challenges, primarily label heterogeneity and extreme class imbalance. To resolve these issues, we implemented a controlled single-pathology framework. We filtered the datasets to isolate four specific classes (DR, MH, ODC, and WNL) and evaluated this focused data using a YOLOv11 classification model. This second framework allowed us to measure the model's diagnostic precision under refined conditions. Finally, we utilized Grad-CAM to cross-validate the quantitative outcomes for the single-pathology cases. This visual explanation confirmed that the performance gains stemmed from meaningful feature learning rather than spurious correlations.

4.7.1 Multiple-Pathology Framework

We first conducted an ablation study using multi-pathology fundus images from RFMiD and RFMiD2.0. Because individual images contained multiple co-occurring diseases, we labeled instances into 57 distinct classes and trained a YOLOv11 model using Ultralytics v8.4.36. The model converged after 36 epochs using early stopping. Fig. 9 shows that the overall precision was 0.442. The recall was 0.0842. Well-represented pathologies (e.g., CRS and VS) showed strong detection. Many rare conditions scored low or zero, highlighting severe challenges to detection reliability, including label heterogeneity, class imbalance, and overlapping visual features.

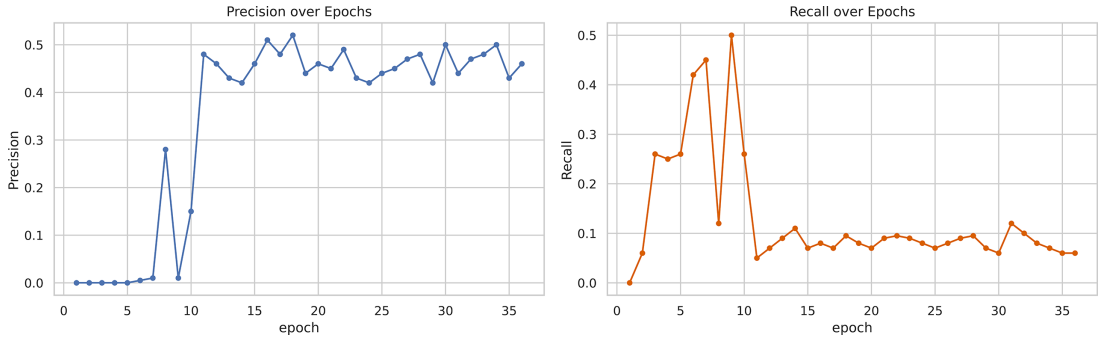

Figure 9: Multi-pathology—epochs results.

4.7.2 Single-Pathology Framework

For the second ablation study, we used a filtered single-pathology dataset created by our novel filtering module, which isolated images with only one disease to establish a controlled setting. The YOLOv11 classification model achieved substantially better performance. It reached approximately 89.6% top-1 accuracy across

four classes. This ablation shows that removing label heterogeneity stabilizes learning and improves model confidence, underscoring the importance of rigorous dataset curation when moving from public, multi-label data to clinical single-pathology analysis.

a. DR: Fig. 10 illustrates the augmented model's heatmaps. These heatmaps tightly localize hard exudates and microaneurysms. The baseline is often misfocused on peripheral vessels. This misfocus reduced precision. Grad-CAM highlights lesion-prone regions and vascular irregularities, which confirms attention to clinically relevant DR features.

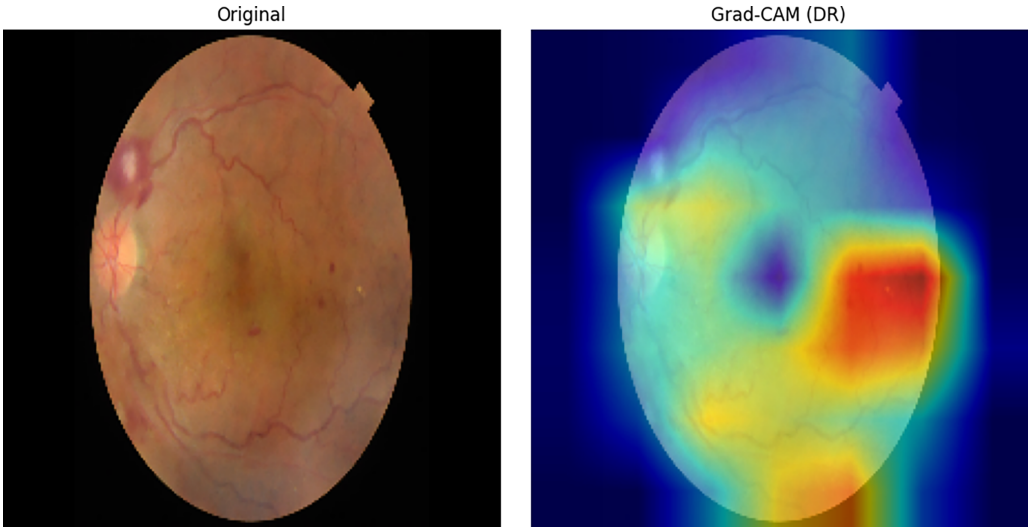

Figure 10: DR original vs. Grad-CAM.

b. MH: Fig. 11 shows the Grad-CAM for MH. It reveals diffuse activation across obscured retinal regions, which is consistent with 100% sensitivity. The model focuses on global haze and texture degradation. It does not focus on localized lesions, as MH is a diffuse image-quality abnormality.

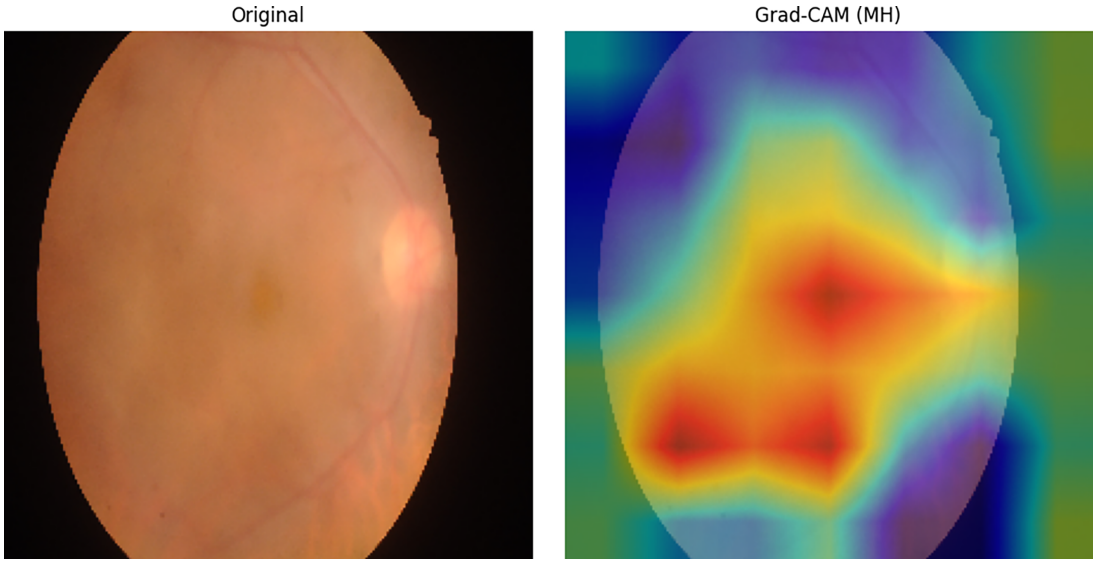

Figure 11: MH original vs. Grad-CAM.

c. ODC: Fig. 12 demonstrates the attention maps for ODC. These maps identify the optic disc contour. They compute the cup-to-disc ratio to confirm the model can detect structural changes. Grad-CAM emphasizes disc-centered features, confirming a focus on clinically relevant ODC regions.

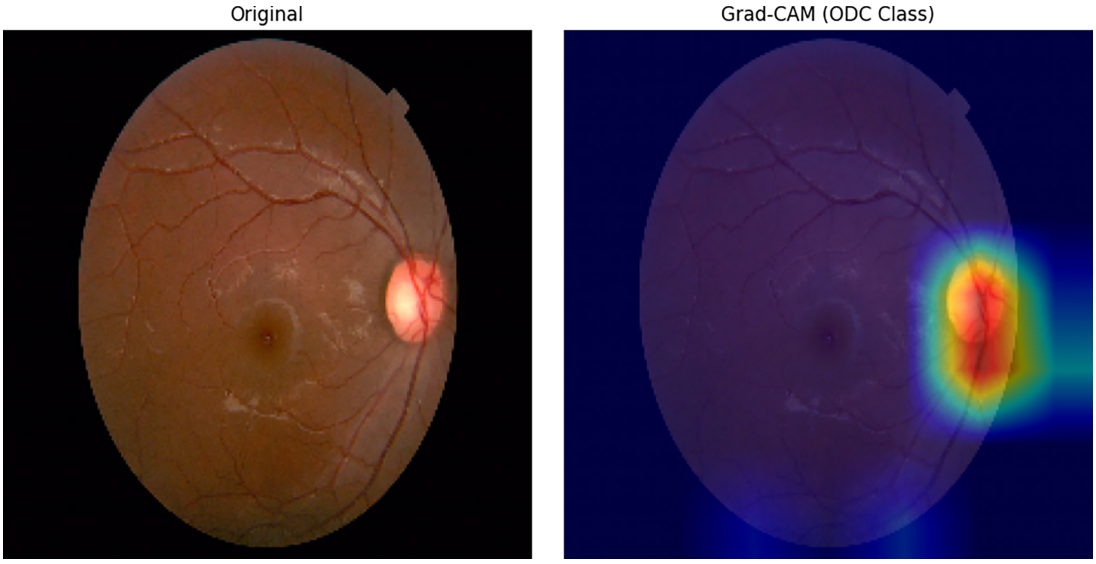

Figure 12: ODC original vs. Grad-CAM.

d. WNL: Fig. 13 shows the Grad-CAM heatmaps for the WNL class. The augmented model highlights healthy landmarks. These include the macula and optic disc. The model avoids artifacts and illumination issues, indicating that it defines normality by the absence of lesions. It does not rely on background noise. Furthermore, Grad-CAM visualizations provide a visual comparison between ODC and WNL.

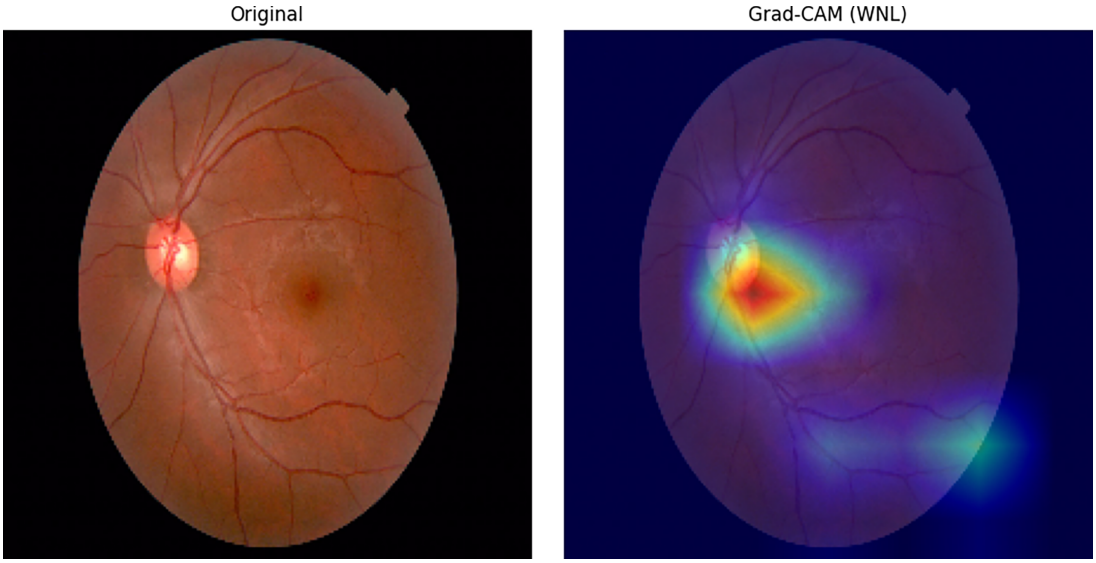

Figure 13: WNL original vs. Grad-CAM.

e. Grad-CAM Spatial Attention Maps: Fig. 14 reveals the model's spatial focus during classification. For DR, the attention map shows highly concentrated, localized activation over a specific lesion. The MH

map highlights a broad, central region corresponding to the macula. In contrast, the ODC map exhibits scattered, diffuse activations. It fails to isolate the optic disc, which indicates poor feature localization. The model cannot consistently pinpoint subtle structural changes in ODC, which directly explains the degraded F1-score of 51.28%. Scattered attention leads to poor precision and sensitivity. Conversely, the WNL attention map lacks central focal points. It shows only peripheral edge activations, which are consistent with a healthy retina. It lacks distinct pathologies. This absence of conflicting focal features is beneficial. It enables the model to verify healthy baselines confidently and directly contributes to the robust 91.48% accuracy for the WNL class.

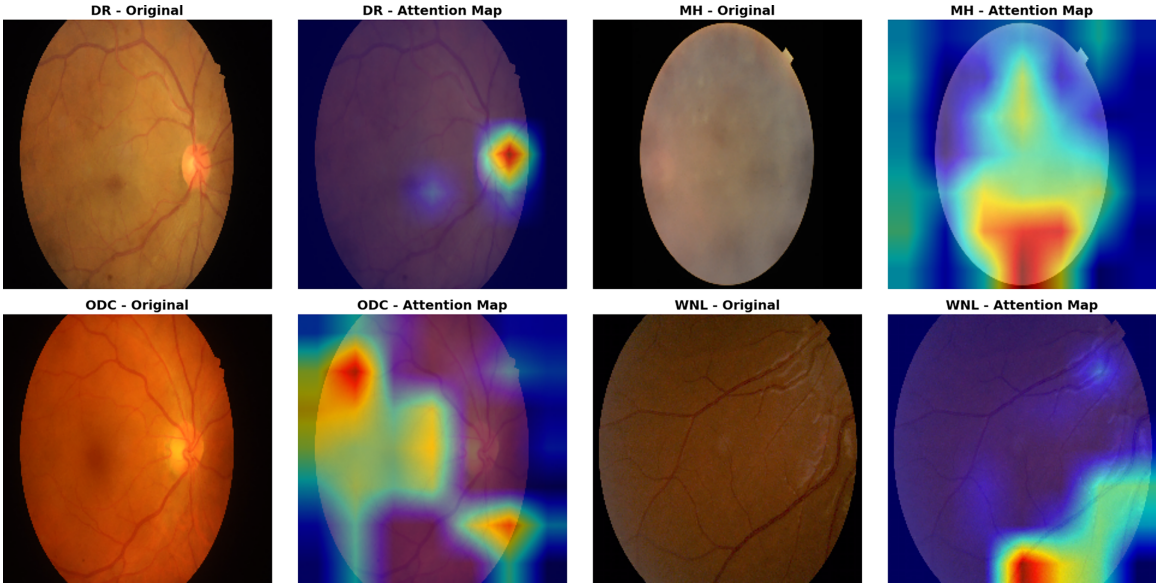

Figure 14: 4 original vs. 4 Grad-CAM.

In summary, YOLOv11 exhibits significant majority-class bias on imbalanced datasets, disproportionately failing on minority classes with subtle anomalies. For example, the DR-FID "Mild" class achieved 0% accuracy, while only 2 of 22 "ODC" cases in RFMiD/RFMiD2.0 were correctly predicted (16 were misclassified as WNL). Although dominant classes inflate overall accuracy, reliable classification requires mitigating this bias through data augmentation, aggressive minority-class oversampling, or cost-sensitive loss functions.

5 Results Analysis and Discussion

5.1 Model Explainability

This framework advances beyond standard binary classification. It integrates multi-class triage (DR, MH, ODC, WNL). It uses a single, computationally efficient YOLOv11 architecture. Grad-CAM visualizations confirm diagnostic transparency. The model's attention mechanisms accurately localize true clinical biomarkers. For example, it targets microaneurysms in DR and avoids spurious background artifacts. Quantitative metrics indicate high performance. However, medical diagnostic models also require transparency to build trust. We needed to validate that high accuracy came from valid feature extraction. We employed Grad-CAM to visualize the regions of interest (ROI). These ROIs drive the model's predictions. The visual explanations match the quantitative gains.

5.2 Rationale for YOLOv11 Selection and Addressing Research Questions

While traditional lightweight CNNs (e.g., MobileNet, EfficientNet) reduce computational overhead, their aggressive downsampling frequently erases subtle ophthalmic biomarkers, such as isolated microaneurysms. YOLOv11 overcomes this limitation by integrating advanced spatial pyramid pooling and enhanced attention mechanisms in its neck architecture. This design preserves fine-grained, multi-scale spatial resolution and effectively captures minute pathological features, while avoiding the heavy parameter burden of two-stage detectors or Vision Transformers. Consequently, YOLOv11 delivers an optimal algorithmic balance. It provides robust feature extraction for complex medical imaging. It pairs this with low-latency, single-stage efficiency. This efficiency is required for resource-constrained clinical deployment.

a. Answering RQ1 (Classification Efficacy): A highly lightweight, single-stage architecture can achieve robust diagnostic performance. We demonstrated this successfully. We applied the YOLOv11 Nano framework. Our model achieved an overall test accuracy of 89.5%. It also achieved an 89.6% validation accuracy. Single-pass global feature extraction is highly capable of multi-class retinal triage. It delivers highly accurate classifications for prominent classes like WNL and DR. It achieves this without heavier ensemble methods.
b. Answering RQ2 (Impact of Pathology Isolation): Isolating single pathologies before training is effective for distinct conditions. However, limitations remain for highly subtle structural changes. We used custom data filtering and augmentation to isolate features for DR and WNL. We achieved near-perfect true positive rates for these classes. Still, confusion matrix analysis revealed a challenge. Distinguishing ODC from WNL remains difficult. Our isolation prevents spatial feature-confusion for most classes. Yet, highly localized structural pathologies need more focus. Future iterations may require targeted class balancing or specialized cropping techniques.
c. Answering RQ3 (Edge Deployment Viability): The study establishes an exceptionally efficient baseline. It heavily fulfills the deployment requirement. The YOLOv11 Nano method avoids massive parameter counts and GPU bottlenecks. The architecture has only 1.53 million parameters. It operates at just 3.2 GFLOPs. It yields a final fused weight file of just 3.2 MB. It maintains an optimal balance between diagnostic accuracy and ultra-lightweight processing. These results confirm the framework's fundamental structure. It is ready for translation to resource-constrained edge devices. Examples include portable fundus cameras, Jetson Nano platforms, or mobile applications, which are ideal for rural clinical deployment.

5.3 Analysis of Training Dynamics and Overfitting Limits

Training logs demonstrate highly efficient convergence. They also show stable learning dynamics. The augmentation and optimization pipeline yielded a steadily decreasing training loss. It achieved a minimum validation loss of 0.19. It also stabilized validation accuracy at approximately 89.6%. The framework's early stopping mechanism prevented classic overfitting. Training halted automatically at epoch 22, which occurred after 10 epochs without improvement. The most optimal, generalizable weights were captured at epoch 12. Current augmentation prevents massive feature memorization. However, the model shows diminished sensitivity to the minority ODC class, suggesting a risk with heavy, generalized augmentation. It can cause feature washout for subtle pathologies. Future work will address this issue. We will apply targeted minority-class oversampling. We will also use localized structural enhancements.

5.4 Performance Interpretation and Strengths

The YOLOv11 Nano architecture achieved robust 89.5% overall test accuracy, demonstrating the efficacy of single-stage global feature extraction for retinal triage. We implemented "Pathology Isolation" preprocessing, which was highly successful for prominent classes. It yielded exceptional sensitivity for DR and WNL baselines. The framework's primary strength is extreme efficiency. It requires only 1.53 million parameters and 3.2 GFLOPs. The final fused model is only 3.2 MB, which allows high-fidelity triage in a single forward pass. It is fundamentally structured for offline edge deployment. Target devices include portable fundus cameras or NVIDIA Jetson platforms, which are highly suitable for resource-constrained environments.

5.5 Clinical Applicability

The model is a highly efficient automated triage tool. It identifies prevalent conditions such as DR and distinct anomalies such as MH. However, it exhibits notable limitations. Confusion matrix analysis reveals a specific challenge. The model struggles to distinguish ODC from WNL baselines. Generalized global feature extraction may miss subtle details. It fails to capture the localized structural changes of ODC. Furthermore, the baseline relied on isolated single-pathology images. Therefore, the current framework fails to account for complex clinical presentations. It cannot handle overlapping pathologies.

5.6 Efficacy of Data Augmentation

Inter-class similarity and imaging artifacts complicate the classification of retinal disease. We evaluated our framework using official dataset partitions. We avoided custom internal splits. This choice ensures a robust assessment. It provides reproducible metrics of real-world performance. Fig. 5 presents a comparative analysis. It contrasts baseline and augmented training runs. The augmentation strategy smoothed the training dynamics. It reduced both training and validation loss. It achieved an overall accuracy of 89.5% on the independent test set, confirming that our pipeline prevented data memorization. It genuinely enhanced the model's generalization. The model effectively handles unseen, standardized data. This results in a reliable clinical triage system.

5.7 Comparative Analysis for CPU Environments

The primary purpose of this experiment is to evaluate CPU utilization during the validation phase for YOLOv11 and ResNet50. YOLOv11 demonstrates significant advantages over ResNet50 across multiple metrics. It achieves superior memory efficiency by processing four times larger batch sizes (64 vs. 16) while consuming nearly 2 GB less VRAM. In Table 8, inference efficiency shows a clear advantage for YOLOv11 under Intel Xeon processor conditions. On an Intel Xeon processor, YOLOv11 processes each image in 97.8 ms, whereas ResNet50 requires 1409.3 ms per image, indicating substantial computational overhead. Consequently, YOLOv11 completes the validation of 364 images in approximately 3.5 min, compared to over 11 min for ResNet50. These findings highlight a critical operational advantage.

Fig. 15 presents a comparative analysis of YOLOv11 and ResNet50 across training epochs and accuracy. We monitored the training process using an early stopping patience of 10 epochs. YOLOv11 achieved its highest validation accuracy at epoch 12. Training concluded at epoch 22 when early stopping triggered, significantly reducing overall training time. In contrast, ResNet50 completed the full 40-epoch cycle without triggering early stopping, reaching its optimal validation accuracy at epoch 31. To ensure a fair and rigorous evaluation, we selected the best-performing validation checkpoint for each model (Epoch 12 for YOLOv11 and Epoch 31 for ResNet50) to report the final test results. This difference in training duration indicates a significant trend: YOLOv11 converges more efficiently than ResNet50 under identical experimental conditions.

Table 8: Architecture, CPU hardware utilization, and performance results comparison.

Metric	YOLOv11	ResNet50
Architecture & Complexity		
Layers (Fused)	47	91
Parameters (Fused)	1,531,148	26,109,316
Computational Complexity	3.2 GFLOPs	68.8 GFLOPs
Hardware Utilization & Speed		
Hardware Environment	Intel Xeon CPU @ 2.20GHz	
Inference Latency (per image)	97.8 ms	1409.3 ms
Total Validation Time	3 min 35 s	11 min 11 s
Validation Results (364 Images)		
Top-1 Validation Accuracy	89.56%	86.81%

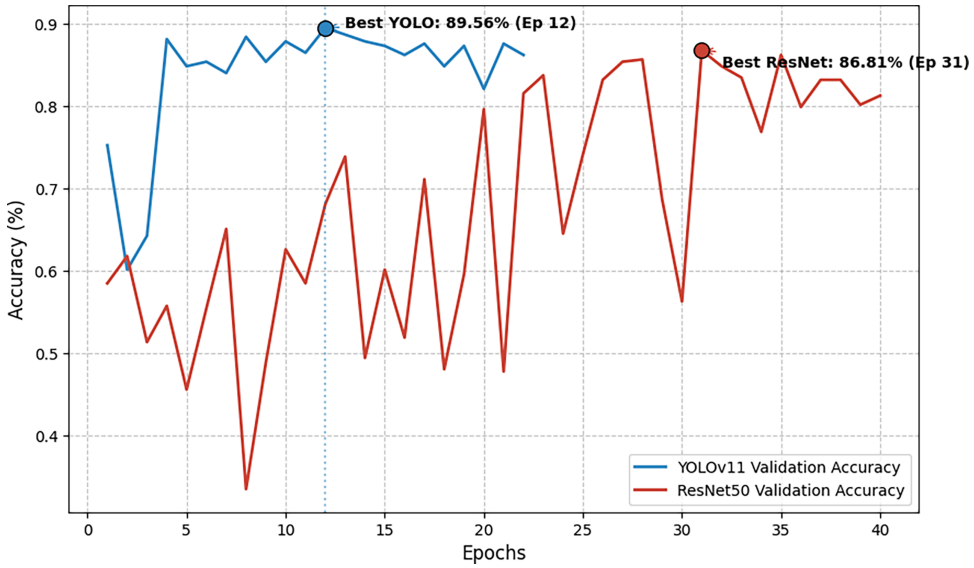

Figure 15: Top-1 accuracy comparison: YOLOv11 vs. ResNet50.

Notably, although significantly smaller and faster, YOLOv11 achieves a Top-1 accuracy of 89.56%, surpassing ResNet50 at 86.81%. Furthermore, both models report a 100% Top-5 accuracy. Because the dataset used for this hardware benchmark contains exactly four classes, the true class is mathematically guaranteed to fall within the top five predicted probabilities. Ultimately, YOLOv11 provides an optimal trade-off between diagnostic accuracy and resource consumption, making the framework ideal for real-time applications on CPU-only hardware.

5.8 Baseline Evaluation Protocol and Dataset Stratification

In Table 9, the proposed YOLOv11 framework is benchmarked against recent state-of-the-art studies. Comparative metrics for the baseline models were acquired directly from their original publications. To ensure a rigorous evaluation, the comparative literature is stratified into two distinct benchmarks.

a. Direct Dataset Benchmarks: This category serves as our primary baseline for comparison. Ejaz et al. (2024) [13] used the same source datasets (RFMiD and RFMiD2.0) to classify the same overarching disease categories. However, the baseline study used a custom-built dataset structure that pooled and shuffled all images before the final split. In contrast, our YOLOv11 framework strictly enforces the official, predefined partitions. This comparison provides a dataset-matched evaluation of architectural performance, specifically testing how our model performs in a rigorous, leak-free environment compared to their custom-shuffling approach.
b. Architectural Benchmarks: Studies utilizing alternative datasets (such as DDR and APTOS) provide a broader architectural context. For instance, Butt et al. (2025) [34] utilized a hybrid classification pipeline. They combined traditional CNNs (GoogleNet) with standard ML classifiers. Although the underlying training datasets differ across these studies, including this benchmark remains necessary. It highlights the computational and diagnostic advantages of a unified, end-to-end framework like YOLOv11, which contrasts sharply with the fragmented, multi-stage pipelines traditionally used for fundus imaging tasks.

Table 9: Comparison with recent studies.

Author	Technique	Dataset	Images	Classes	Acc%	Pre%	Rec%	F1%
Kumar & Katal 2025 [7]	DL OD3-YOLO	SMDG, IDRID, REFUGE	1360	3	–	78	84	80
Mahapadi et al., 2026 [11]	YOLOv10 + CBAM + Hybrid Optimizer	DIARETDB1, MESSIDOR, APTOS	2985	5	88.7	93.8	94.1	93.9
Ardelean et al., 2025 [8]	YOLO + Attention + Transformer	AROI, OCT5k	2808	3	–	44.5	26.9	32.2
Lokesh et al., 2025 [32]	Ghost YOLO	Roboflow	427	2	–	98.1	91.9	94.9
Wang et al., 2025 [33]	SSD + VGG16	Shanxi Hospital	1733	3	95.7	94.7	95.1	94.7
Butt et al., 2025 [34]	CNN + GoogleNet + ML	DDR, APTOS	17,335	5	94.60	94.60	94.88	94.74
Islam et al., 2025 [21]	ResNet-18 + CNN	Bangladesh Eye Hospital	416	2	92	–	–	–
Liu et al., 2024 [39]	CRD-Net	MMC-AMD, APTOS-2021, GAMMA	2365	3	83.61	–	–	–
Ejaz et al., 2024 [13]	DL Framework	RFMiD, RFMiD2.0	9442	4	95.03	85.33	79.30	79.86

(Continued)

Table 9 (continued)

Author	Technique	Dataset	Images	Classes	Acc%	Pre%	Rec%	F1%
Elsayed & Rushdi 2024 [20]	GR-CNN	RFMiD	3200	28	–	67.66	77.36	77.16
Proposed Study	YOLOv11	RFMiD, RFMiD2.0	2201	4	94.78	79.61	83.61	81.14

Note: Acc = Accuracy, Pre = Precision, Rec = Recall, F1 = F1-Score.

5.9 Comparison with Recent Studies

Our method offers a highly efficient alternative to existing models, achieving competitive diagnostic performance while significantly reducing computational overhead, as detailed in Table 9. Key comparative insights against recent studies are as follows:

a. Model Efficiency and Computational Requirements: YOLOv11 requires a minimal memory footprint, comprising just 47 layers and 1,531,148 parameters, while operating efficiently at approximately 3.2 GFLOPs.
b. Performance vs. Complexity: Achieving an overall accuracy of 94.78% and an optimal F1-score of 81.14%, our YOLOv11 model demonstrates strong classification performance across primary clinical classes without the heavy computational burden of larger, slower architectures.
c. Dataset Efficiency: Despite using a moderate-sized dataset, the proposed model achieves robust generalization, highlighting the effectiveness of our data filtering and augmentation pipeline compared to models that rely on massive datasets (Butt et al., 2025 [34]) with 17,335 images.
d. Architectural Advantage: The evolution to the YOLOv11 architecture introduces improved feature extraction and localization mechanisms, enabling a real-time inference latency of just 16.3 ms on a standard CPU, a critical advantage for low-resource clinical deployments.

We compared our model with the baseline study by Ejaz et al. (2024) [13]. The comparative outcomes in Table 9 require a nuanced interpretation. Fig. 16 presents a comparative analysis between the proposed study and the baseline. These results highlight the trade-offs and strengths of our lightweight approach. The baseline reported high ODC sensitivity on a custom-built dataset augmented before the train-test split. If a dataset is augmented before splitting, the "parent" image and its "child" versions (rotated, flipped, or scaled copies) may be distributed across both the training and test sets. The model "memorizes" specific features of those images rather than learning generalizable diagnostic patterns. Ignoring the original training, validation, and test folders compromises the integrity of the "blind" test set. We recognized that custom splitting might artificially inflate performance metrics. Therefore, we revised our pipeline to ensure methodological integrity. We now use the official predefined partitions provided by the dataset authors (RFMiD and RFMiD2.0). This strategy eliminates image-level data leakage. Consequently, our test set differs from the custom split used in [13].

Our YOLOv11 model achieves 94.78% accuracy, which is comparable to the 95.03% reported in the baseline [13]. Several methodological factors account for the marginal 0.25% difference in global accuracy. First, our study utilizes official dataset partitions to eliminate image-level data leakage, a rigorous approach that naturally yields more conservative metrics. Second, we excluded 29 images (2.47%) due to unrecoverable file corruption, though this minor loss does not statistically undermine the evaluation. Despite these

constraints, our model demonstrates a significant 4.31% gain in recall. A custom filtering module drives these gains by isolating single-pathology cases and significantly reducing label noise. Furthermore, fine-tuning a lightweight YOLOv11 architecture with AMP training successfully prevents overfitting and sets a new benchmark for computational efficiency. While we state claims of superiority cautiously due to the lack of variance reporting in the baseline [13], our framework offers a distinct clinical advantage by prioritizing sensitivity over global accuracy. Although detecting subtle minority classes like ODC requires focus in future iterations, the current model delivers reliable, real-time automated triage for the most severe ocular diseases.

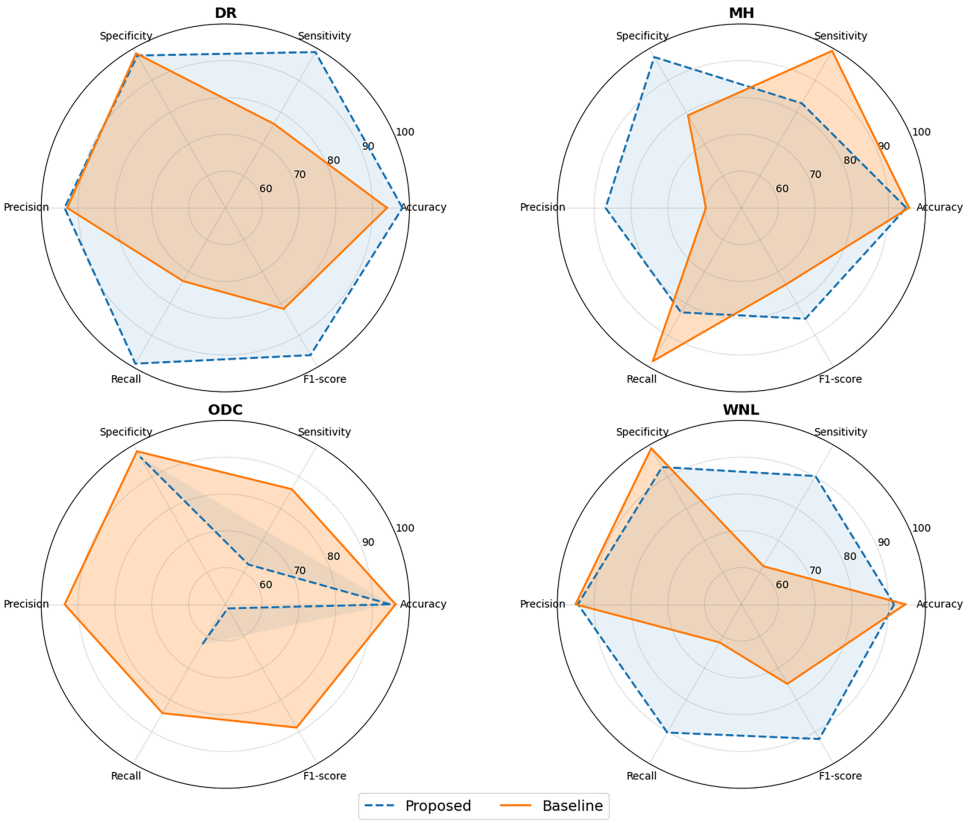

Figure 16: Comparison with base study.

5.10 Limitations and Future Work

5.10.1 Limitations

The proposed YOLOv11 framework shows strong baseline performance. However, several limitations exist. These must be addressed before achieving clinical-grade autonomy.

a. Single-Pathology Isolation: The preprocessing module strictly isolated single-pathology images to establish a pure baseline for feature extraction. Consequently, while the model is highly effective for frontline triage, it currently cannot manage complex clinical presentations where patients exhibit multiple concurrent conditions.

b. Severe Class Imbalance: The datasets contained significant class imbalances. For example, the "Mild DR" class in the DR-FID dataset required oversampling just 6 original images to 300. Although we mitigated the associated risks using transfer learning and controlled augmentations, extreme oversampling can lead to overfitting by causing the model to memorize patient-specific features rather than generalizing.

c. Validation Constraints: Due to computational and resource limitations, this study did not utilize n-fold cross-validation or multi-run variance testing. While the model showed stable convergence and achieved 89.6% accuracy in the primary validation phase, broader statistical robustness will require future validation across multiple independent runs with varying random seeds.

5.10.2 Future Work

To bridge the gap between this retrospective baseline and real-world clinical deployment, future research will focus on the following key areas:

a. Multi-Label Detection & Granular Interpretability: Future iterations will expand the network's capabilities to detect and localize concurrent, overlapping pathologies (e.g., simultaneous DR and Glaucoma). Furthermore, we aim to move beyond image-level classification by integrating pixel-level lesion segmentation, providing clinicians with much deeper diagnostic interpretability.
b. Architectural & Edge Optimization: We plan to evaluate other lightweight architectures, such as MobileNet, EfficientNet, and MobileViT. Additionally, applying hardware-level optimizations, including post-training INT8 quantization and model pruning, will allow us to deploy these compressed weights directly onto low-power edge devices, such as portable fundus cameras or the NVIDIA Jetson Nano, for offline rural use.
c. Clinical Decision Support System (CDSS) Integration: Inspired by recent studies (Shyamalee et al., 2024 [40]), we will integrate the inference engine into a graphical interface to facilitate clinical adoption. This CDSS will display real-time disease predictions, bounding-box localizations, and Grad-CAM heatmaps, enabling immediate clinical verification.
d. Prospective Clinical Trials: The optimized system will undergo prospective, human-in-the-loop clinical trials alongside board-certified ophthalmologists. Testing the model on live, heterogeneous demographic cohorts will assess real-world classification accuracy and mitigate domain shift. Crucially, experts must verify that the model's spatial attention aligns with true physiological anomalies to satisfy regulatory standards.

6 Conclusion

This work validates an efficient DL pipeline for multi-class retinal disease classification by integrating a single-pathology filter and a balanced sampling scheme with targeted augmentations, effectively reducing label noise and mitigating class imbalance. We fine-tuned a lightweight YOLOv11 model using mixed-precision training, achieving an inference time of 16.3 ms on a standard CPU and demonstrating its viability for real-time clinical deployment. Overall, the pipeline achieved 94.78% accuracy, 96.12% specificity, and an 81.14% F1-score. The model demonstrated exceptional performance in detecting DR (98.08% accuracy, 98.89% sensitivity) and normal cases (92.27% F1-score). While identifying morphologically subtle conditions, such as ODC, remains an ongoing challenge (62.50% sensitivity), the proposed method establishes a robust, highly efficient baseline for automated ophthalmic screening in resource-constrained environments.

Acknowledgement: Not applicable.

Funding Statement: This work was partly supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP)-Innovative Human Resource Development for Local Intellectualization program grant funded by the Korea government (MSIT) (IITP-2026-RS-2023-00259678) and by the IITP (Institute of Information & Communications Technology Planning & Evaluation)-ITRC (Information Technology Research Center) grant funded by the Korea government (MSIT) (IITP-2026-RS-2024-00438335).

Author Contributions: Conceptualization, Jaffar Hussain, Tahira Nazir and Junaid Rashid; methodology, Jaffar Hussain, Tahira Nazir and Junaid Rashid; software, Jaffar Hussain; validation, Junaid Rashid and Jungeun Kim; formal analysis, Jungeun Kim; investigation, Jaffar Hussain; resources, Junaid Rashid; data curation, Jaffar Hussain and Tahira Nazir; writing—original draft preparation, Jaffar Hussain; writing—review and editing, Tahira Nazir, Junaid Rashid and Jungeun Kim; visualization, Jaffar Hussain; supervision, Tahira Nazir; project administration, Tahira Nazir, Junaid Rashid and Jungeun Kim. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: RFMiD is openly available in a public repository [15] sourced from IEEE DataHub. RFMiD2.0 is openly available in a public repository [16] sourced from Zenodo. DR Fundus Image Dataset (DR-FID) [17] acquired from the Department of Ophthalmology of the Hospital de Clínicas, Facultad de Ciencias Médicas, Universidad Nacional de Asunción, Paraguay.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Anvesh K, Reshmi BM, Hariharan S, Reddy HV, Krishnamoorthy M, Kukreja V, et al. A novel approach deep learning framework for automatic detection of diseases in retinal fundus images. Comput Model Eng Sci. 2025;143(2):1485–517. doi:10.32604/cmes.2025.063239.
2. Elmannai H, Hamdi M, Meshoul S, Alhussan AA, Ayadi M, Ksibi A. An improved deep learning framework for automated optic disc localization and glaucoma detection. Comput Model Eng Sci. 2024;140(2):1429–57.
3. International Diabetes Federation. Diabetic macular edema clinical practice recommendations. 2019 [cited 2025 Sep 11]. Available from: https://idf.org/media/uploads/2019/09/IDF-DME-CPR.pdf.
4. Bayer AG. Diabetic macular edema (DME); 2024 [cited 2025 Sep 11]. Available from: https://www.bayer.com/en/pharma/diabetic-macular-edema-dme.
5. Kropp M, Golubnitschaja O, Mazurakova A, Koklesova L, Sargheini N, Vo TKS, et al. Diabetic retinopathy as the leading cause of blindness and early predictor of cascading complications—risks and mitigation. EPMA J. 2023;14(1):21–42. doi:10.1007/s13167-023-00314-8.
6. World Health Organization. Diabetes fact sheet. 2024 [cited 2025 Sep 11]. Available from: https://www.who.int/news-room/fact-sheets/detail/diabetes.
7. Kumar A, Katal N. A lightweight YOLO model for detection of disease from optic disc region of eye fundus imagery. Sens Imaging. 2025;26(1):1–27. doi:10.1007/s11220-025-00575-9.
8. Ardelean AI, Ardelean ER, Marginean A. Can YOLO detect retinal pathologies? A step towards automated OCT analysis. Diagnostics. 2025;15(14):1823. doi:10.3390/diagnostics15141823.
9. Bodapati JD, Balaji BB. Self-adaptive stacking ensemble approach with attention based deep neural network models for diabetic retinopathy severity prediction. Multimed Tools Appl. 2024;83(1):1083–102. doi:10.1007/s11042-023-15120-7.
10. Lalithadevi B, Krishnaveni S. Diabetic retinopathy detection and severity classification using optimized deep learning with explainable AI technique. Multimed Tools Appl. 2024;83(42):89949–90013. doi:10.1007/s11042-024-18863-z.
11. Mahapadi AA, Shirsath V, Pundge A. Real-time diabetic retinopathy detection using YOLO-v10 with nature-inspired optimization. Biomed Mater Devices. 2026;4(2):2164–86. doi:10.1007/s44174-025-00343-z.
12. Hemal MM, Saha S. Explainable deep learning-based meta-classifier approach for multi-label classification of retinal diseases. Array. 2025;26:100402.
13. Ejaz S, Baig R, Ashraf Z, Alnfiai MM, Alnahari MM, Alotaibi RM. A deep learning framework for the early detection of multi-retinal diseases. PLoS One. 2024;19(7):e0307317. doi:10.1371/journal.pone.0307317.
14. Meedeniya D, Shyamalee T, Lim G, Yogarajah P. Glaucoma identification with retinal fundus images using deep learning: systematic review. Inform Med Unlocked. 2025;56(18):101644. doi:10.1016/j.imu.2025.101644.

15. Pachade S, Porwal P, Thulkar D, Kokare M, Deshmukh G, Sahasrabuddhe V, et al. Retinal fundus multi-disease image dataset (RFMiD). New York, NY, USA: IEEE Dataport; 2020.
16. Panchal S, Naik A, Kokare M, Pachade S, Naigaonkar R, Phadnis P, et al. Retinal fundus multi-disease image dataset (RFMiD) 2.0: a dataset of frequently and rarely identified diseases. Data. 2023;8(2):29.
17. Benítez VEC, Matto IC, Román JCM, Noguera JLV, García-Torres M, Ayala J, et al. Dataset from fundus images for the study of diabetic retinopathy. Data in Brief. 2021;36:107068. doi:10.1016/j.dib.2021.107068.
18. He J, Song J, Han Z, Cui M, Li B, Gong Q, et al. Multi-spectral transformer with attention fusion for diabetic macular edema classification in multicolor image. Soft Comput. 2024;28(7):6117–27. doi:10.1007/s00500-023-09417-w.
19. Liu Z, Gao A, Sheng H, Wang X. Identification of diabetic retinopathy lesions in fundus images by integrating CNN and vision mamba models. PLoS One. 2025;20(1):e0318264. doi:10.1371/journal.pone.0318264.
20. Elsayed TS, Rushdi MA. Computer-aided multi-label retinopathy diagnosis via inter-disease graph regularization. Biomed Signal Process Control. 2024;96(8):106516. doi:10.1016/j.bspc.2024.106516.
21. Islam S, Deo RC, Barua PD, Soar J, Acharya UR. Novel deep learning model for glaucoma detection using fusion of fundus and optical coherence tomography images. Sensors. 2025;25(14):4337. doi:10.3390/s25144337.
22. Zuo Q, Shi Z, Liu B, Ping N, Wang J, Cheng X, et al. Multi-resolution visual Mamba with multi-directional selective mechanism for retinal disease detection. Front Cell Dev Biol. 2024;12:1484880. doi:10.3389/fcell.2024.1484880.
23. Gao W, Rong F, Shao L, Deng Z, Xiao D, Zhang R, et al. Enhancing ophthalmology medical record management with multi-modal knowledge graphs. Sci Rep. 2024;14(1):23221. doi:10.1038/s41598-024-73316-9.
24. Breeyear JH, Mitchell SL, Nealon CL, Hellwege, Charest B, Khakharia A, et al. Development of electronic health record based algorithms to identify individuals with diabetic retinopathy. J Am Med Inform Assoc. 2024;31(11):2560–70. doi:10.1093/jamia/ocae213.
25. Chen X, Zhou C, Zhu Y, Luo M, Hu L, Han W, et al. Detecting glaucoma in highly myopic eyes from fundus photographs using deep convolutional neural networks. Clin Exp Ophthalmol. 2025;53(5):502–15. doi:10.1111/ceo.14498.
26. Bodapati JD, Veeranjaneyulu N. Adaptive ensembling of multi-modal deep spatial representations for diabetic retinopathy diagnosis. Multimed Tools Appl. 2024;83:68467–86.
27. Macsik P, Pavlovicova J, Kajan S, Goga J, Kurilova V. Image preprocessing-based ensemble deep learning classification of diabetic retinopathy. IET Image Process. 2024;18(3):807–28. doi:10.1049/ipr2.12987.
28. Shafiq M, Fan Q, Alghamedy FH, Obidallah WJ. DualEye-FeatureNet: a dual-stream feature transfer framework for multi-modal ophthalmic image classification. IEEE Access. 2024;12:143985–4008. doi:10.1109/access.2024.3469244.
29. Raghunathan T, Mishra A, Mahur AK, Balaji B. Multi-Modal AI/ML integration for precision glaucoma detection: a comprehensive analysis using optical coherence tomography, fundus imaging, RNFL, and vessel density. In: Proceedings of the 2024 2nd International Conference on Artificial Intelligence and Machine Learning Applications (AIMLA); 2024 Mar 15–16; Namakkal, India. New York, NY, USA: IEEE; 2024. p. 1–7.
30. Mehta P, Petersen CA, Wen JC, Banitt MR, Chen PP, Bojikian, et al. Automated detection of glaucoma with interpretable machine learning using clinical data and multimodal retinal images. Am J Ophthalmol. 2021;231(12):154–69. doi:10.1016/j.ajo.2021.04.021.
31. Benbakreti S, Benbakreti S, Ozkaya U. The classification of eye diseases from fundus images based on CNN and pretrained models. Acta Polytech. 2024;64(1):1–11.
32. Lokesh, Poola RG, Gorrepati, Yellampalli SS. Real-time cataract diagnosis with GhostYOLO: a GhostConv-enhanced YOLO model. Eng Technol Appl Sci Res. 2025;15(3):22945–52.
33. Wang N, Jin Y, Zhao Z, Wu Q, Li F, Wang X. Study on classification detection method of diabetic retinopathy based on SSD. Sens Imaging. 2025;26(1):1–19. doi:10.1007/s11220-025-00578-6.
34. Butt M, Iskandar A, Khan MA, Latif G, Bashar A. MEDCnet: a memory efficient approach for processing high-resolution fundus images for diabetic retinopathy classification using CNN. Int J Imaging Syst Technol. 2025;35(2):e70063. doi:10.1002/ima.70063.
35. Malviya R, Singh AK, Sundram S, Balusamy B, Kadry S. Blockchain with artificial intelligence for healthcare: a synergistic approach. Bristol, UK: IOP Publishing; 2023.

36. Al-Fahdawi S, Al-Waisy AS, Zeebaree DQ, Qahwaji R, Natiq H, Mohammed MA, et al. Fundus-deepnet: multi-label deep learning classification system for enhanced detection of multiple ocular diseases through data fusion of fundus images. Inf Fusion. 2024;102:102059.
37. Alam MNU, Bahadur EH, Masum AKM, Noori FM, Uddin MZ. SwAV-driven diagnostics: new perspectives on grading diabetic retinopathy from retinal photography. Front Robot AI. 2024;11:1445565. doi:10.3389/frobt.2024.1445565.
38. Ben-Kiki O, Evans C, döt Net I. YAML Ain't Markup Language (YAML^{TM}) Version 1.2, 2021, Revision 1.2.2. [cited 2025 Dec 21]. Available from: https://yaml.org/spec/1.2.2/.
39. Liu K, Si T, Huang C, Wang Y, Feng H, Si J. Diagnosis and detection of diabetic retinopathy based on transfer learning. Multimed Tools Appl. 2024;83(35):82945–61. doi:10.1007/s11042-024-18792-x.
40. Shyamalee T, Meedeniya D, Lim G, Karunarathne M. Automated tool support for glaucoma identification with explainability using fundus images. IEEE Access. 2024;12(1):17290–307. doi:10.1109/ACCESS.2024.3359698.