Journal of Innovative Image Processing (ISSN: 2582-4252)
www.irojournals.com/iroiip/

CNN-LSTM Model for Effective Glaucoma Detection from Fundus Images

Raghda M. Nasr[1], Manal A. Abdel-Fattah[2], Ahmed Samir Montasser[3]

[1]Computers and Information Systems Department, Sadat Academy for Management Sciences, Cairo, Egypt.
[2]Faculty of Computer Science and Artificial Intelligence, Capital University (formerly Helwan University), Cairo, Egypt.
[3]Faculty of Medicine, Capital University (formerly Helwan University), Cairo, Egypt.

E-mail: [1]raghda.nasr2811@gmail.com, [2]manal_8@hotmail.com, [3]asmontasser@gmail.com

Orcid ID: [1]0009-0008-0759-6878, [2]0000-0002-2888-0367, [3]0009-0006-7338-2435

Abstract

Glaucoma is one of the main causes of irreversible blindness, where delayed detection may increase the risk of permanent damage to the optic nerve. Developing accurate and computationally efficient computer-aided diagnostic systems using low-cost retinal fundus images remains challenging, especially with multi-disease datasets and limited clinical metadata. This paper proposes a hybrid CNN-LSTM multi-model for detecting glaucoma and six others ophthalmic diseases by merging retinal fundus images with clinically annotated patient symptoms. The model was trained and evaluated using the ODIR-5K dataset and expert-guided symptom annotation. A lightweight convolutional neural network was employed to extract visual features, while a Long Short-Term Memory network processed textual symptom descriptions. The extracted features were fused for classification across seven classes. The model achieved an accuracy of 92%, a macro-average F1-score of 74.7%, a weighted-average F1-score of 92%, and a macro-averaged AUC of 96.8% across seven predicted classes. Then, Explainable AI (XAI) techniques were applied to verify that diagnostic decisions were mainly driven by clinically relevant retinal regions. These results reveal that integrating visual and symptom-based information can improve retinal disease classification while maintaining computational efficiency, making the proposed framework suitable for deployment in limited-resource environments.

Keywords: Image classification, Deep learning, Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), Glaucoma Detection.

1. Introduction

Glaucoma is a multifactorial neurodegenerative disease that causes progressive vision loss, and is one of the most common causes of permanent, incurable blindness. The delay in detecting Glaucoma is the first reason for permanent damage to the optic nerve. The main objective is to develop a deep neural network model capable of early detection of glaucoma, which may help reduce optic nerve deterioration. Some studies have predicted that the number

DOI: https://doi.org/10.36548/jiip.2026.3.008
Received: 06.05.2026, received in revised form: 13.06.2026, accepted: 02.07.2026, published: 11.07.2026

of patients diagnosed with glaucoma will reach around 111.8 million people by 2040, compared to 76.5 million people in 2020 [1].

As the number of patients increases, governments will face greater financial pressure to address the problem, potentially reducing the efficiency of the diagnostic process. Consequently, there has been a trend has begun toward diagnosing glaucoma through fundus images, given their moderate cost compared to other imaging methods. The idea behind using fundus images is that they are more accurate than visual field tests, yet much cheaper than OCT.

This paper proposes a deep learning model that can support glaucoma detection using a more reliable fundus image dataset, as the dataset includes several eye diseases, not only glaucoma and normal classes. Although the main objective of the research was to propose a model able to enhance glaucoma detection in resource-constrained environments, the decision was to rely on a multi-label ophthalmic disease dataset to approximate the practical reality in low-cost diagnostic centers.

The paper is organized as follows: Section 2 provides background on glaucoma, discusses deep learning techniques used in computer-aided diagnosis systems, and analyses the most relevant previous work. Section 3, the methodology section, presents the dataset selection, preprocessing, training phase, results review, and the evaluation matrics used to assess prediction accuracy. Section 4 discusses the results and evaluation metrics. Section 5 presents the conclusion and discusses future research directions.

2. Background

The last period witnessed a strong impact of artificial intelligence (AI) across all fields of medicine. AI technology will have a major impact on ophthalmology applications in the near future. Several studies show high interest in using deep learning techniques in computer-aided diagnosis (CADx) or computer-aided detection (CADe) systems to improve the analysis and detection of medical imaging and scanning. AI is applied in medicine from two perspectives: "virtual and physical". The virtual perspective includes informatics approaches based on machine learning and deep learning techniques to support medical information management, such as controlling health management systems and electronic health records, and to provide active guidance to support physicians in their treatment decisions. The physical perspective is best represented by robots used in some medical cases, such as assisting patients or the attending surgeon.

Deep learning could be considered the fast-growing approach of the AI field. This leads to the use of deep learning in different application areas, such as automatic speech recognition, natural language processing, image recognition, and medical applications. Deep learning shows a major impact on ophthalmology applications. There are many aspects of deep learning that could be helpful in ophthalmology, especially in medical imaging applications such as [2]:

- Object detection; example: determine tumour location.
- Object segmentation; for example, determine the glaucoma type or stage.
- Object classification; example: glaucoma eye vs normal eye.

There are several deep learning architectures. Figure 1 shows some of the most used architectures. A Convolutional Neural Network (CNN) is considered the most common and

most used type of feedforward neural network it consists of convolutional and pooling layers and is designed to recognize visual patterns [3]. The pooling layer lies between convolutional layers. Convolution is a mathematical operation on two functions that produces a third function. It is the fundamental building block that extracts features from input images by applying a set of filters to the image. After extracting features, the pooling layer aggregates the inputs using aggregation functions such as max or mean [3]. Recent studies show a rise in the use of different CNN architectures in medical imaging, such as DenseNet, ResNet (Residual Network), GoogleNet, AlexNet, MobileNet, InceptionV3, and VGGNet [4]. These models have become a baseline architecture for solving various problems in image recognition and visual feature extraction.

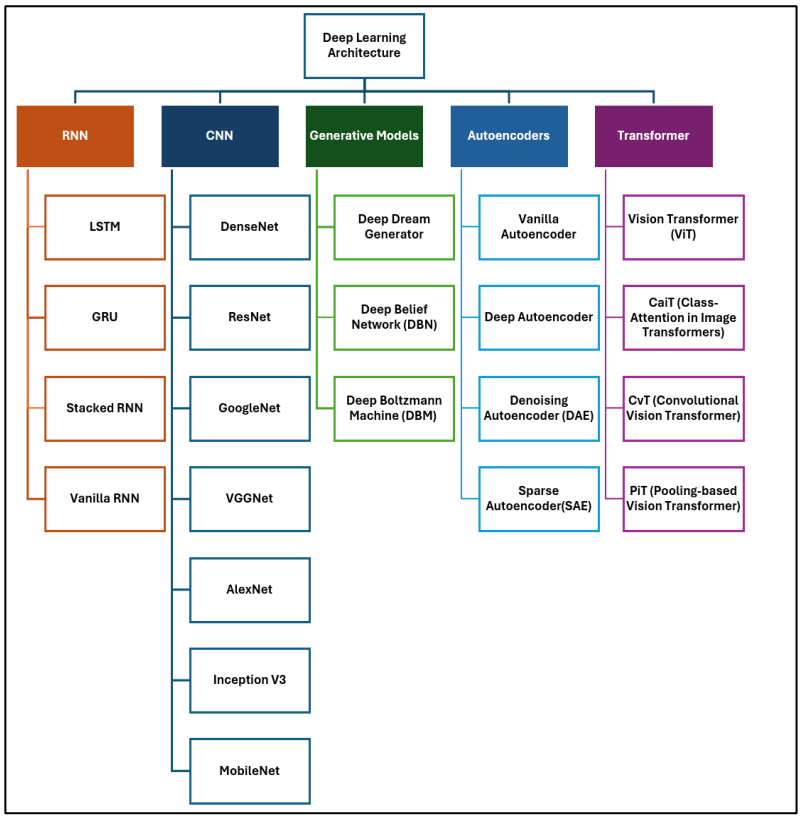

Figure 1. Major architectures of deep learning

Recurrent Neural Networks (RNNs) are models composed of artificial neurons with one or more feedback loops, used to learn and extract features from sequential and text data. RNNs were primarily introduced due to the limitations of CNNs in extracting features from sequential data. There are different RNN architectures such as simple RNN, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Stacked RNN, and Vanilla RNN. A simple RNN architecture consists of an input layer, hidden layers (which are also recurrent layers), and an output layer. An RNN minimizes the difference between the output and target values by optimizing the network weights. The LSTM network contains a more complex structure in the hidden layer, called the "LSTM unit", which comprises three gates: the input, forget, and output gates. These gates help to add or retain information in the cell state, which functions to store value or state [5].

Generative AI (GenAI) primarily uses deep learning and neural network models to learn patterns and generate new data that resembles previously learned data. Generative AI (GenAI) combines both semi-supervised and unsupervised deep learning techniques to understand the

underlying patterns of data, instead of depending on labels [6]. GenAI can capture the structure and relationships within a dataset using an ANN with trainable weights to build a parametric model that approximates the learned probability distribution of the dataset. Generative models have high potential to improve pattern recognition capabilities. as GenAI can play a significant role in medical image recognition through its abilities in data augmentation, enhancing image resolution, and identifying anomalies.

Autoencoders are neural network models that learn representations of data. Autoencoders are used for dimensionality reduction, feature learning, and data reconstruction. An autoencoder compresses the input data into a smaller representation and reconstructs the original data from it. The output data is the input data after reconstruction [7]. There are different autoencoder architectures, such as Deep Autoencoders, Sparse Autoencoders, and Denoising Autoencoders, which are efficient at reconstructing corrupted inputs into uncorrupted outputs.

Transformers are neural network architectures primarily developed for Natural Language Processing (NLP) to perform tasks such as text classification, machine translation, and question answering. Transformer models for NLP include the Bidirectional Encoder Representations from Transformers (BERT) and the Text-to-Text Transfer Transformer. Subsequently, several studies focused on how transformers adapt for vision and image recognition tasks [8]. Some transformer models that have been developed to support vision tasks include Class-Attention in Image Transformers (CaiT), Convolutional Vision Transformers (CvTs), Pooling-based Vision Transformer (PiT), and Vision Transformer (ViT). Most of them work well with large-scale datasets, but their performance suffers on small datasets; in addition, they require substantial computational resources.

2.1 Literature Review

This section will present some recent research focused on using ML and AI techniques to support the detection and monitoring of glaucoma from binary or multi-disease datasets.

Daniel Milad et al. [9] presented a Code-Free Deep Learning (CFDL) binary model that enhances accuracy through ensemble voting based on the Rotterdam EyePACS AIROGS dataset of colored fundus images for glaucoma detection. The CFDL model achieved an AUC of 0.994 and an accuracy of 94-97%.

Kumari Jyoti et al. [10]; implemented a hybrid model that merges the 2D-FBSE-EWT (Fourier-Bessel Series Expansion-based Empirical Wavelet Transform) with a method based on MCA for feature extraction from retinal fundus images. This framework attained an accuracy of 94.15%.

Thisara Shyamalee et al. [11] created an integrated model based on U-Net and ResNet50 for segmentation and a modified Inception V3 architecture for classification with an explainability mechanism, as improving model performance and interpretability was the main goal. The model was tested on several datasets using open-source code and applied to five different datasets, showing AUC values for each dataset ranging from 83% to 99%.

Benton Chuter et al. [12] developed an EfficientNet-B0-based model for automated fundus image quality that effectively filters low-quality images for primary open-angle glaucoma detection. The model resulted in high accuracy in quality assessment by achieving an AUROC of 0.97.

Yen-Ying Chiang et al. [13] achieved good performance (AUC 0.894, F1 score 81.92%) by applying ConvNeXt_Base+CBAM architecture to a dataset consisting of 3088 fundus images from patients with high myopia (≤-6.0 D). However, the model is restricted to Asian populations.

Additionally, Nora A. Alkhaldi et al. [14] used EfficientNetB0 with transfer learning, reaching almost 100% accuracy; however, this raises overfitting issues owing to dependence on limited-size datasets.

Ruben Hemelings et al. [15] merged U-Net and ResNet-50 for identifying glaucoma characteristics outside the optic nerve, incorporating regression and classification. However, both researche efforts are limited to binary classification.

Xingyuan Ou et al. [16] introduced a deep learning framework called BFENet to enhance multi-label classification of ophthalmic diseases using the ODIR-5K dataset. The proposed framework used a two-stream CNN simultaneously to process bilateral retinal images and enable interaction between both streams to capture complementary diagnostic information, similar to clinical ophthalmic practice. The model incorporated attention-based feature enhancement and multiscale dilated convolutions to extract both global and local retinal features and improve lesion detection at different scales. Experimental results achieved an AUC of 0.912, an F1-score of 0.892, and a final score of 0.780 on the off-site test set.

Ajitha S et al. [17] suggested a 13-layer CNN with SoftMax to mine vital features from fundus images, and these features are classified into either glaucomatous or normal during testing. This model achieved an accuracy of 93.86%, although the limited dataset (1113 images) raises an overfitting concern.

Muhammad Bajwa et al. [18] introduced a two-stage model that integrates R-CNN for locating the optic disc and CNNs for classification of the optic disc (healthy or glaucomatous). The proposed two-stage model resulted in an AUC equal to 0.874 for classification on the ORIGA dataset (780 images).

Andres Y. Diaz Pinto et al. [19] evaluated various fine-tuned CNN architectures. The team compared the performance of five CNNs through two experiments. The first applied over cropped images of the optic disc, while the second applied an ensemble model that uses four CNN architectures. The first experiment's results show that the Xception architecture achieved a better AUC of 0.9605, While the ensemble model used in the second experiment obtained an AUC of 0.94.

3. Methodology

Figure 2 illustrates the overall methodology workflow. First, the ODIR-5K fundus image dataset was collected and clinical symptom annotations were generated. This was followed by a second stage where both image and annotated data were pre-processed and then split into training, validation, and testing sets. Next, a CNN-based branch extracted visual representations from the pre-processed fundus images, while the LSTM branch learned representations from the annotated symptoms. This was followed by a concatenation layer that fused the extracted representations from both images and text. The classification of seven disease categories then took place based on the fused features. In the final stage, the model

performance was evaluated using different measurements and techniques, and explainable AI techniques were applied to assess both predictive performance and model interpretability.

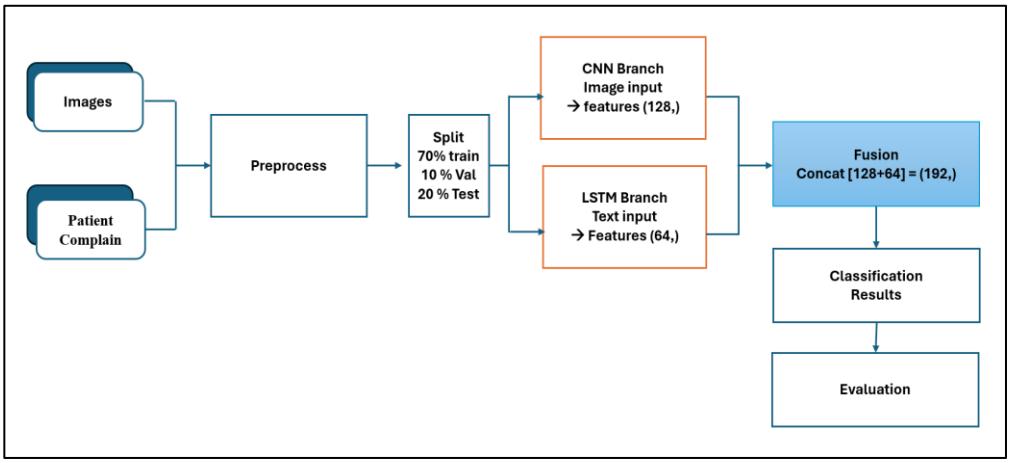

Figure 2. Methodology workflow

3.1 Dataset Description

The ODIR-5K dataset [20], released by a Chinese team, is a multi-label classification dataset of fundus images. This real-world dataset of patient information was collected by Shanggong Medical Technology Co., Ltd. from different hospitals/medical centers in China. The dataset consists of a total of 5000 fundus images (for both right and left eyes) for 2500 patients, in addition to their demographic data such as age and gender. The images were classified into eight classes: "normal (N), diabetes (D), glaucoma (G), cataract (C), Age-related Macular Degeneration [AMD] (A), hypertension (H), Pathological Myopia (M), and other diseases/abnormalities (O)".

3.2 Data Preprocessing

3.2.1 Image Preprocessing

Image preprocessing focused on preparing images for classification into 7 classes. Different cleaning methods were used, such as discarding missing or corrupted images. One of the eight classes in the dataset, “other diseases/abnormalities (O), included” several diseases with very few images per disease and was therefore excluded from the experiment.

Image augmentation techniques were intentionally not applied to evaluate the model's performance using the original fundus images, as if acquired in routine clinical scenarios. Given that the ODIR-5K contains thousands of fundus images, which is a reasonable size and represents the natural distribution for a dataset, the decision was to preserve the dataset’s natural fundus image characteristics and minimize synthetically generated data for only the complaints annotation.

All fundus images were resized to 128*128 pixels to minimize the computational cost and resource consumption of the training process. This resolution helps reduce the computational burden without losing sufficient visual information for accurate disease detection. This provides an effective trade-off between model performance and computational efficiency, which allows the proposed model to be suitable for resource-constrained environments such as small ophthalmology diagnostic centers.

3.2.2 Text Annotation and Preprocessing

A novel annotation framework was followed to enrich the dataset with clinically generated patient complaints. The annotation process can be considered a research contribution, as it extends the original fundus image dataset by adding clinically meaningful symptom descriptions based on expected patient complaints derived from expert ophthalmologist assessment.

First, all existing labels were removed from images to ensure unbiased ophthalmologist evaluation. An ophthalmologist independently examined each fundus image and assigned the most likely patient complaint associated with the observed retinal findings. These annotations were added from the neutral symptoms list. Figure 3 shows a sample of the dataset used.

```
8,8_left.jpg,none
13,13_right.jpg,vision_blur
13,13_left.jpg,vision_blur
18,18_right.jpg,"field_loss, vision_blur"
18,18_left.jpg,"field_loss, vision_blur"
```

Figure 3. Dataset sample images and annotated complaints

The resulting dataset structure consists of fundus images, original disease categories (labels), and clinically assigned patient complaints. To ensure consistency and prevent information leakage, similar symptoms were standardized using unified terminology. A strategic generalization procedure was used to break the correlations between symptom phrases and their diagnostic class. Individual textual descriptors were successfully mapped across the classes by standardizing and abstracting the complaints into more comprehensive clinical symptom clusters. Thus, various distinct phrasings were mapped to generic groups. For example, cloudy vision, fuzzy vision, unclear vision, and blurry vision all mapped to one unified terminology, "vision_blur," within the seven categories. The data was successfully transformed into a more realistic representation of noisy, overlapping real-world patient complaints by this purposeful insertion of textual ambiguity. So, the complaints still provide meaningful clinical context to the model, but they are no longer predictive enough to permit shortcutting. As a result, the model is forced to rely on the fundus images to make its final diagnostic decisions. The text data preprocessing used several techniques, such as handling missing values, tokenization, and padding.

The added text annotations were used exclusively for research purposes to enrich the dataset and increase the generalization of the model. The annotation protocol followed will support the development of a multi-modal diagnostic model based on both visual and symptom-based information.

Dataset images were split randomly into 70% training, 10% validation, and 20% testing for each class. The training, validation, and test sets have the same distribution of original classes, as shown in Figure 4.

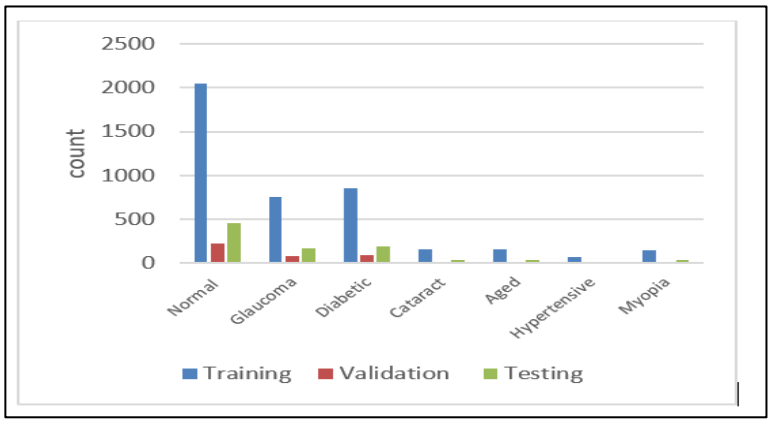

Figure 4. Dataset split per class

3.3 CNN-LSTM Architecture

The contribution of this research is the development of a hybrid multi-model based on CNN and LSTM architectures to detect various eye diseases from low-cost fundus images, achieving acceptable accuracy.

The CNN-LSTM architecture was selected for this study because of its effectiveness in extracting and modelling complex patterns from fundus images and patients complaints while maintaining computational efficiency for low-cost environments. Compared with other architectures such as Vision Transformer (ViT) and BERT, the CNN-LSTM approach requires fewer computational resources and is better suited to moderate-sized medical datasets like ODIR-5K. ViT and BERT-based architectures generally require substantially larger training datasets and higher computational resources. In this scenario, the CNN-LSTM model reduces the risk of overfitting while still achieving strong classification performance, which makes it a more appropriate and practical choice for glaucoma detection and multi-label ophthalmic disease classification in low-cost environments.

To provide a clear overview of the proposed architecture, Algorithm 1 summarizes the complete workflow, showing the five steps of data preprocessing, data pipeline, model building, training, and evaluation. Figure 5 shows the detailed architecture of the CNN branch, LSTM branch and Fusion branch within the proposed multi-model.

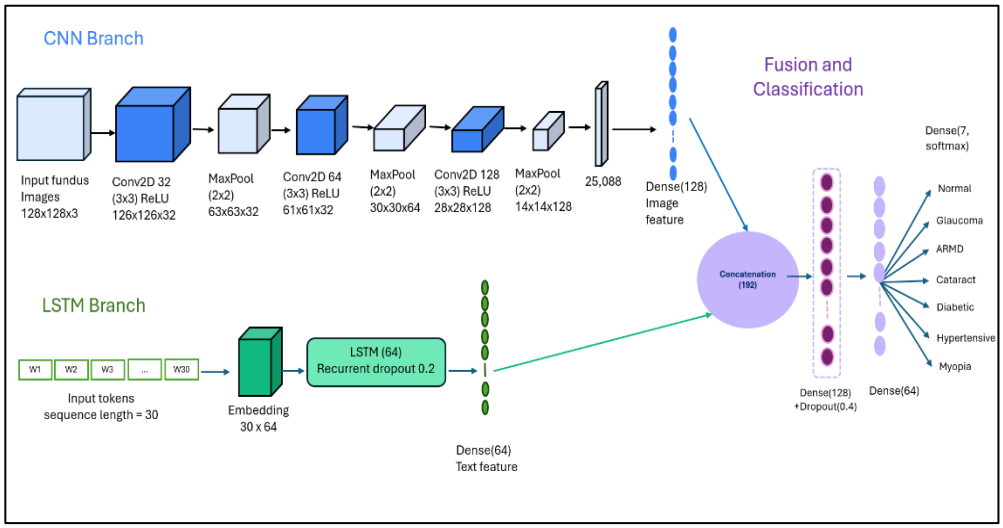

Figure 5. Architecture of CNN branch, LSTM branch and Fusion branch within the Proposed multi-model

Algorithm 1: CNN-LSTM-MultimodelEyeDiseaseClassifier

INPUT: retinal images + patient complaint text + disease labels
OUTPUT: trained model, evaluation metrics
BEGIN
Step 1: Load & Preprocess Data
1. Load CSV (4,652 samples retinal images + patient complaint), fill missing complaints
2. Tokenize complaint text →pad to fixed length 30
3. Encode 7 class labels →one-hot vectors
4. Stratified split: 70% train / 10% validtion / 20% test
5. Compute balanced class weights to handle imbalance

Step 2. Build Data Pipeline
6. For each split: load image → resize 128 x 128 x 3 →normalize [0,1]
7. Batch (size=24), shuffle train, prefetch all splits

Step 3. Build Model
8. CNN branch: Conv2D (32, 3x3, RelU) →MaxPool2D(2x2) → Conv2D (64, 3x3, RelU) → MaxPool2D(2x2) → Conv2D (128, 3x3, RelU) → MaxPool2D(2x2) → Flatten → Dense(128) → output: 128 dimensional image feature vector
9. LSTM branch: Embedding(64) → LSTM(64 recurrent_droupout = 0.2) → Dense(64) → output: 64 dimensional text feature vector
10. Fusion: Concatenate [image (128) + text (64)] → output: 192 dimensional fused feature vector
11. Classification: Dense(128) → Dropout (0.4) →Dense (64) →Dense(7, Softmax)
12. Compile with AdamW optimizer + label-smoothed cross-entropy loss

Step 4. Train
13. Fit up to 40 epochs with class weights and callbacks: → Model checkpointing (save best model), stop early, reduce LR on plateau

Step 5. Evaluate
14. Predict on test set → compute accuracy, precision, recall , F1-score, and AUC per class
15. Plot confusion matrices and ROC curves
16. RETURN model and evaluation metrics

END

3.4 Experimental Setup

The CNN-LSTM multi-model was implemented and trained on a computing platform equipped with an Intel Core i7-12700H processor operating at up to 4.6 GHz, 40 GB of RAM, and Intel Iris Xe Graphics (Alder Lake-P GT2). The CNN branch consisted of three lightweight convolutional layers with 32, 64, and 128 filters, respectively. Each convolutional layer applied a 3 × 3 kernel, a stride of 1, and valid padding, followed by a 2 × 2 max-pooling operation with a stride of 2 for spatial downsampling. A 3 × 3 kernel size was chosen to maintain a balance between feature extraction and computational efficiency. Valid padding was used to rely only on real image pixels for feature extraction and to help reduce computational cost, balancing the stride effect. A stride of 1 was applied because disease indicators can be very small, despite the higher computational cost. A pooling stride of 2 was used to reduce memory usage and help prevent overfitting.

The text branch processed patient symptoms using Keras's word-level Tokenizer, configured to retain the 2,000 most frequent terms. An out-of-vocabulary token () was used to map unseen or infrequent words. Complaints were transformed into integer-encoded sequences with a fixed length of 30 tokens using post-padding and post-truncation. This was followed by an embedding layer with a dimension of 64, enabling dense semantic representations to be learned. The resulting embeddings were processed by the LSTM branch to capture contextual relationships among symptom terms before fusion with the image branch output.

Model training was performed using the AdamW optimizer with a learning rate of 3×10^{-4} and a weight decay coefficient of 1×10^{-4}. AdamW combines learning rates with weight decay to improve generalization and reduce overfitting, while the small learning rate reduces the risk of overshooting despite the longer learning time. A batch size of 24 was chosen to improve generalization and maintain lower memory usage. Categorical cross-entropy with a label smoothing factor of 0.05 was adopted as the optimization objective. This configuration is designed to balance computational efficiency and classification performance, enabling the multi-model to achieve good generalization, stable training, and minimize overfitting in a low-cost environment.

3.5 Learning Models and Training Procedures

The proposed deep learning model architecture leverages the concatenation of Convolutional Neural Networks (CNNs) with a Long Short-Term Memory (LSTM) network to achieve high accuracy in glaucoma detection. This architecture integrates different techniques to enable the model to better learn and generalize in eye disease detection.

Given the limitations of the available dataset, the team prepared a clinical annotation to describe the several different patient complaints for each class. In this experiment, a hybrid-modal was used to process two different formats of input data. As shown in Table 1, images are handled using a Convolutional Neural Network (CNN). The CNN branch is used for deeper feature extraction to produce a 128-dimensional image feature output vector describing the image content. The text data is handled using a Long Short-Term Memory (LSTM) network. The LSTM branch is used for text feature extraction, producing a compact 64-dimensional text feature vector.

This is followed by the fusion branch, which concatenates the image and text feature vectors, resulting in a single multi-modal representation of 192 feature vectors. The model comprises multiple layers, each contributing uniquely to the model's performance. As shown in Table 1, the model consists of 18 layers in total, 10 of which are training layers.

The input layer has no computations; it only defines the image input size and format, passing (128, 128) images to the first convolutional layer. The first convolutional layer (conv2d) extracts low-level features using 32 filters; each filter learns to detect basic patterns and features such as edges, corners, and textures in images, producing an output shape of (126, 126, 32) with 896 parameters. The number of filters increased to 64 in the second convolutional layer (conv2d_1) to extract intermediate features, producing an output shape of (61, 61, 64) with 18,496 parameters. The third convolutional layer (conv2d_2) increases the number of filters to 128 to extract more abstract and complex features, thereby enhancing the model's ability to make more accurate classifications. The conv2d_2 layer produces 73,856 parameters with an output shape of (28, 28, 128). Each convolutional layer is followed by a pooling layer

to retain significant features and avoid overfitting by downsampling to approximately half the dimensions.

By the end of the CNN model, the Flatten layer prepares the data by converting it into a 1D vector of size 25,088. This is followed by a dense layer, which is a fully connected layer of 128 neurons, each connected to the 25,088 inputs previously received from the flattened layer to produce a compact 128-dimensional representation of the image, yielding 3,211,392 parameters. The LSTM model begins with an input layer, which passes the text data "patients complain" to an embedding layer that converts the input into "64 features" learned by the model, producing an output of 4,736 parameters. This is followed by an LSTM Layer that mainly captures temporal dependencies and contextual meaning across the available text, using memory cells to retain information and remember previous context. This is followed by a dense layer to prepare the text features for the fusion layer.

The Fusion Branch concatenates a 128-dimensional image feature vector with a 64-dimensional text feature vector, resulting in a total 192-dimensional feature vector. This concatenation helps the model learn relationships between image and text information and derive higher-level interactions between the two modalities. This is followed by a dense layer, a fully connected layer of 128 neurons, each connected to 192 features, to produce an output of 24,704 parameters. A dropout layer is used to deactivate 50% of neurons, which prevents overfitting or memorization and enhances model generalization. Then, two dense layers are applied to reach a fully connected layer of 7 neurons that represent the seven predicted classes.

Table 1. CNN-LSTM model layers & output shape

Layer (type)	Output Shape	Param #
image_input (InputLayer)	(None, 128, 128, 3)	0
conv2d (Conv2D)	(None, 126, 126, 32)	896
max_pooling2d (MaxPooling2D)	(None, 63, 63, 32)	0
conv2d_1 (Conv2D)	(None, 61, 61, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_2 (Conv2D)	(None, 28, 28, 128)	73,856
text_input (InputLayer)	(None, 30)	0
max_pooling2d_2 (MaxPooling2D)	(None, 14, 14, 128)	0
embedding (Embedding)	(None, 30, 64)	4,736
flatten (Flatten)	(None, 25088)	0
lstm (LSTM)	(None, 64)	33,024
dense (Dense)	(None, 128)	3,211,392
dense_1 (Dense)	(None, 64)	4,160
concatenate (Concatenate)	(None, 192)	0
dense_2 (Dense)	(None, 128)	24,704
dropout (Dropout)	(None, 128)	0
dense_3 (Dense)	(None, 64)	8,256
output (Dense)	(None, 7)	455

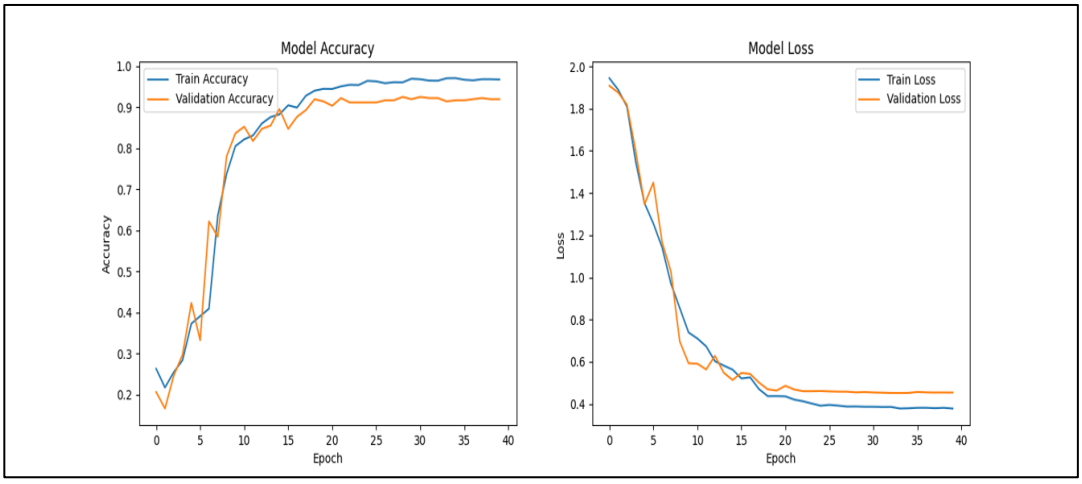

Figure 6. Model training and validation curves

The training phase of the CNN-LSTM multi-model over 30 epochs resulted in a training accuracy of 0.961 and a validation accuracy of 0.922. Figure 6 shows that the model is learning effectively and generalizes well, with no signs of overfitting, as the gap between both learning and validation curves is considered small (3–5%) and remains relatively stable.

4. Results and Evaluation

Different metrics are used to evaluate the proposed model's performance, such as Accuracy, F-1 score, Precision, Recall, and Area Under Curve (AUC).

- Accuracy is a fundamental metric that represents the correct classification rate, i.e., the number of correctly classified instances out of the total number of instances. A higher accuracy rate indicates that the model is making correct predictions. However, it can be a misleading metric, especially when detecting minority classes.
- Precision: The ratio of true positives out of total predicted positives.
- Recall: Represents the ratio of true positive predictions out of the actual total (true positives + false negatives). Also called sensitivity, it is the model's ability to identify all relevant instances in the data.
- F-1 score: A score that represents the balance and harmony between the precision rate and the recall rate.
- AUC: The area under the Receiver Operating Characteristic (ROC) curve.

(ROC) curve: A curve that helps visualize the model's performance.

$$accuracy = \frac{\sum TP \ + \ TN}{\sum TP \ + \ TN + FP \ + \ FN} \tag{1}$$

$$precision = \frac{TP}{TP \ + \ FP} \tag{2}$$

$$recall = \frac{TP}{TP \ + \ FN} \tag{3}$$

$$F^{\beta}score = \frac{(\beta^2 + 1)TP}{(\beta^2 + 1)TP \ + \ \beta^2 \, FN \ + \ FP} \tag{4}$$

The confusion matrices in Table 2 compare the CNN-LSTM model's predictions against the actual labels per class. As for the Glaucoma class, the model shows effectiveness in capturing glaucomatous features. The Normal, Glaucoma, and Diabetic classes all achieved detection accuracy higher than 90%, followed by Myopia, Cataract, and Aged, respectively. However, the detection accuracy for the Hypertensive class was very poor due to the size imbalance.

Table 2. CNN-LSTM model confusion matrices and prediction results for predicted classes

Disease	True Negative	False Positive	False Negative	True Positive
ARMD	877	18	6	30
Cataract	885	10	8	28
Diabetic	722	19	23	167
Hypertensive	903	14	11	3
Glaucoma	758	6	17	150
Myopia	895	4	10	22
Normal	468	7	3	453

Statistical analysis was conducted at a 95% confidence level (CI). Bootstrap resampling with 2000 iterations was used to evaluate model stability by estimating 95% CIs. Confidence intervals provide a measure of variability and enable statistical comparisons among the CNN, LSTM, and CNN-LSTM models.

The proposed multi-model CNN-LSTM model achieved acceptable performance for major retinal diseases. Table 3 shows the precision, recall, and F1-score results for the seven classes applied in this experiment. The Normal class achieved the highest outcomes with 99% accuracy, 98% precision, 99% recall, and 99% F1-score. The Glaucoma and Diabetic classes showed F1-scores of 93% and 89%, respectively. Cataract and Myopia reached balanced outcomes with F1-scores of 76%. However, the Hypertensive class showed relatively poor detection effectiveness, achieving around 18% precision, 21% recall, and an F1-score of 19%, suggesting challenges in accurately recognizing this class due to weak support regarding the small class size.

Table 3. CNN-LSTM model accuracy, precision, recall & F1-score per each class

Disease	Accuracy	Precision	Recall	F1-score
ARMD	0.97	0.62	0.83	0.71
Cataract	0.98	0.74	0.78	0.76
Diabetic	0.95	0.90	0.88	0.89
Hypertensive	0.97	0.18	0.21	0.19
Glaucoma	0.98	0.96	0.90	0.93
Myopia	0.99	0.85	0.69	0.76
Normal	0.99	0.98	0.99	0.99

The statistical analysis shows that the CNN-LSTM model achieved robust performance for the target class (Glaucoma) because of the narrow confidence interval, which indicates stable and reliable detection. It also demonstrated strong performance for both Normal and Diabetic Retinopathy, with narrow confidence intervals reflecting highly reliable classification. For ARMD, Cataract, and Myopia, the model achieved moderate-to-high performance,

although their confidence intervals were wider because of smaller sample sizes. Hypertensive Retinopathy showed a very wide confidence interval, indicating uncertainty in the estimated performance due to the very small class size.

The model achieved 0.968 for the total macro-averaged AUC, as Figure 7 visualizes the Area Under the Receiver Operating Characteristic (Area Under-ROC). As shown, the hybrid model achieved exceptional performance in detecting Glaucoma with an AUC of 0.996.

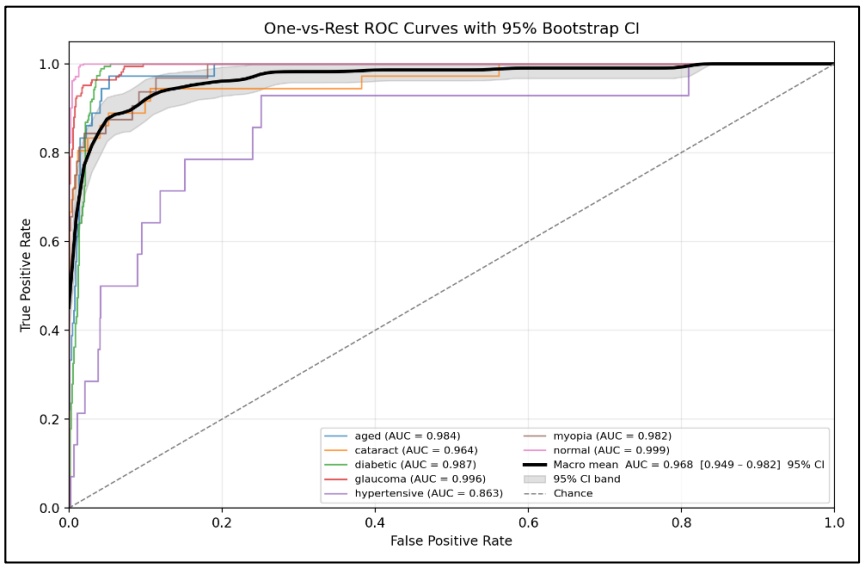

Figure 7. Area under ROC for 7 classes

Table 4 presents the baseline comparison of the CNN-LSTM model with the CNN model and the LSTM model. All models were trained and evaluated under the same experimental conditions using the same dataset, preprocessing procedures, and evaluation metrics using bootstrap resampling with 2000 iterations to estimate 95% CIs. The CNN-LSTM shows good performance stability as it has the narrowest interval, referring to less accuracy variation across the 2000 bootstrap resamples.

According to the baseline comparison, the CNN-LSTM multi-model performed better than the CNN model and the LSTM model, based on the dataset used. This indicates that the hybrid model effectively integrates the strengths of the two architectures. The LSTM model demonstrated the lowest accuracy with 0.32, indicating that symptom descriptions alone are insufficient for disease classification. In contrast, CNN-LSTM model results demonstrate that integrating retinal image features with patient complaint information enhanced the discriminative capability and provided more balanced performance across disease classes compared with the baseline CNN and LSTM models.

Table 4. Performance comparison between CNN–LSTM multi-model and the baseline models at (95% CI)

Model	Accuracy	Precision	Recall	F1-Score	Macro-avg AUC
CNN	0.633 [0.6015 – 0.6660]	0.539 [0.4866 – 0.6005]	0.469 [0.4244 – 0.5137]	0.49 [0.4493 – 0.5310]	0.82 [0.7901 – 0.8476]
LSTM	0.7411 [0.7143 – 0.7691]	0.441 [0.4200 – 0.4605]	0.5706 [0.5687 – 0.5714]	0.488 [0.4718 – 0.5016]	0.992 [0.9892 – 0.9954]
CNN-LSTM proposed model	0.92 [0.9001 – 0.9324]	0.747 [0.7050 – 0.7928]	0.755 [0.766 – 0.8036]	0.747 [0.7009 – 0.7901]	0.968 [0.9493 – 0.9824]

The performance of the three models was evaluated using 95% bootstrap confidence intervals and pairwise McNemar's tests. The CNN-LSTM multi-model achieved the highest performance with an accuracy of 92% (95% CI: 89.8–93.3%), followed by the LSTM base model with 74.1% (95% CI: 71.3–76.9%), and lastly, the CNN base model with 63.3% (95% CI: 60.2–66.6%). Since the confidence intervals of the models are non-overlapping, this indicates statistically significant differences between the baseline models and the CNN-LSTM model. McNemar's test was performed to support p-value analysis to evaluate the CNN-LSTM against the CNN model.

Table 5. McNemar's Test (CNN-base vs CNN-LSTM)

	CNN-LSTM Correct	**CNN-LSTM Wrong**
CNN Correct	n_{00} = 576	n_{01} = 13
CNN Wrong	n_{10} = 277	n_{11} = 65

According to Table 5, the discordant pairs:

- n01 = 13; CNN correct, CNN-LSTM wrong.
- n10 = 277; CNN wrong, multi-model correct

The used method; x^2 with continuity correction

$$\text{(Chi-Square) } x^2 = \frac{(|n01 - n10| - 1)^2}{n01 + n10} \tag{5}$$

$$x^2 = \frac{(|\,13 - 277\,|\, - 1\,)^2}{13 + 277} = 238.5$$

x^2 statistic >> 200, p = 1.32 × 10-65

The CNN-LSTM multi-model correctly classified 277 samples that CNN misclassified, while it made 13 misclassifications that CNN classified correctly. This drives an extremely small p-value of 1.32 × 10-65. The CNN-LSTM multi-model is overwhelmingly more accurate than CNN-base ($p < 0.001$), indicating statistical significance.

For better assessment of the CNN-LSTM model's robustness and generalization, a 5-fold cross-validation was conducted. 5-fold cross-validation was utilized as an additional validation approach to evaluate the model's effectiveness across different data segments. In this method, the dataset was split into five equal sections (four sections utilized for training and one section for testing during each fold). This offers a broader evaluation of the model's generalization across various segments of the dataset.

Table 6. CNN-LSTM multi-model 5-fold cross-validation results

Fold	**Accuracy**	**Precision**	**Recall**	**F1**	**AUC**
Fold 1	0.6430	0.5821	0.6700	0.6083	0.9105
Fold 2	0.3898	0.3558	0.4968	0.3558	0.8265
Fold 3	0.9355	0.8052	0.8073	0.8040	0.9797
Fold 4	0.8575	0.7199	0.7282	0.7100	0.9719
Fold 5	0.9247	0.8121	0.7728	0.7824	0.9811
Mean	0.7501	0.6550	0.6950	0.6521	0.9339
±Std	±0.2333	±0.1912	±0.1221	±0.1824	±0.0668

According to Table 6, the CNN-LSTM multi-model achieved an F1-Score of 0.6521 ± 0.1824 and an AUC of 0.9339 ± 0.0668 across the five stratified folds. The best fold used for

test evaluation was Fold 3. The high inter-fold variance (accuracy 0.7501 ± 0.233) reflects the challenges associated with the difficulty of the ODIR-5k dataset and the severe class imbalance, especially for the limited number of minority hypertensive retinopathy cases. The AUC, being less sensitive to imbalance, reflects more stable performance (0.9339 ± 0.0668), confirming the model's consistent discriminative ability. The final evaluation on the independent held-out test achieved 92.9% accuracy and 98.7% macroAUC, confirming that the selected model generalizes well to unseen data, while the cross-validation results provide a more conservative estimate of average performance across different data partitions.

Table 7. Comparison with pervious work utilize ODIR-5K dataset

Papers				Used Architecture	Performance Measures			
Ref.	**Authors**	**Year**	**Targeted Class**		**Accuracy**	**Precision**	**Recall**	**F1-score**
[21]	Ram, Ajna et al	2020	[N, C, AMD, M]	CNN (2 Fully Connected Layers)	0.883	0.769	0.769	0.384
[22]	Du, Fanyu et al	2024	[N, D, G, C, AMD, H, M]	Two models (ResNet50 and ResNet101) by fused by D-S theory	0.9237	0.945	0.89	0.914
[23]	Hanfi, Rabiya et al	2025	[N, D, G, C, AMD, H, M, O]	Hybrid DL model [EfficientNetV2 and Vision Transformers]	0.953	0.962	0.971	0.967
This Paper			[N, D, G, C, AMD, H, M]	CNN-LSTM	0.92	Macro-avg: 0.747 Weighted-avg: 0.92	Macro-avg: 0.755 Weighted-avg: 0.92	Macro-avg: 0.747 Weighted-avg: 0.92
Class Labels: N: Normal, D: Diabetic, G: Glaucoma, C: Cataract AMD: Age-related Macular Degeneration, H:Hypertesniv, M: Myopa, O: Others								

Table 7 shows a comparison of CNN-LSTM model results with other models utilizing the ODIR-5K dataset. This comparison shows that integrating the complaint annotations with the fundus images improved the detection performance of retinal diseases, especially for glaucomatous eyes.

These results show that the CNN-LSTM model achieved strong overall classification performance with high discriminative capability. The difference between macro-average and weighted-average metrics is mentioned to show how the model performed better on majority classes (Normal, Glaucoma, and Diabetic Retinopathy) while maintaining balanced moderate-to-high performance for (ARMD, Cataract, and Myopia). This reflects the reliability of the model. The model performance is strongly reliable for majority classes and moderate to high for minority classes. However, for hypertensive retinopathy, reliability decreases due to class imbalance.

4.1 Explainable AI (XAI) Analysis

Explainable Artificial Intelligence (XAI) techniques were applied to validate the resulting diagnoses, improve the interpretability of the CNN-LSTM multi-model, and support the transparency of the proposed diagnostic system. Grad-CAM was applied to the CNN branch

to visualize retinal regions influencing disease predictions, while symptom importance scores were extracted from the text branch.

Grad-CAM is used to generate heatmaps that highlight the specific regions of the eye fundus images which support the model predictions. Figure 8 illustrates how the Grad-CAM heatmaps highlighted areas of the retina that affected the diagnoses for various diseases.

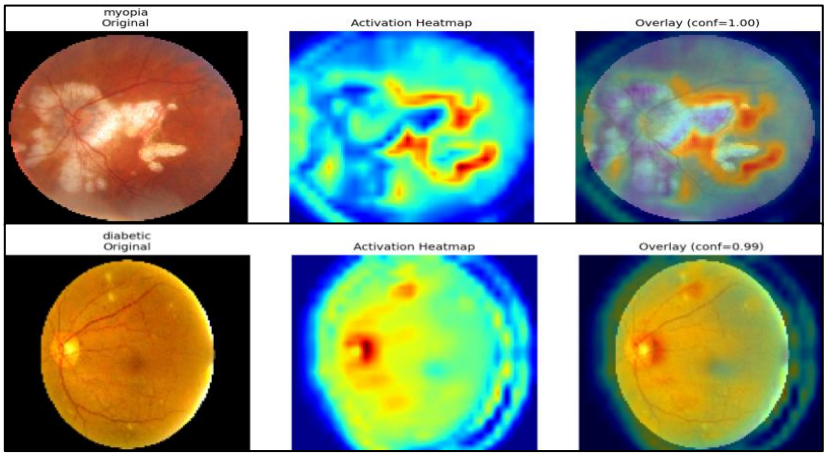

Figure 8. Grad-CAM Visualizations for Fundus Disease Classification

To interpret the textual branch of the CNN-LSTM multi-model, the Gradient × Embedding technique was employed. This method estimates the contribution of each token. Higher values indicate a stronger effect on the prediction. Table 8 shows the top symptom importance scores for different classes.

Table 8. Top symptoms importance score

Class	Token	Importance
hypertensive	Blur	0.002719
diabetic	Blur	0.00011
cataract	Blur	7.00E-06
hypertensive	discomfort	0.01124
glaucoma	discomfort	5.00E-06
Aged	Field	2.90E-05
Aged	Loss	2.50E-05
hypertensive	Sensitivity	0.004839
cataract	Sensitivity	1.30E-05

Integrated Gradients explanations for the CNN-LSTM multi-model are shown in Figure 9. The optic disc region is the model's primary focus, according to the IG attribution map, which indicates that this feature has the greatest influence on the diagnosis of "Glaucoma." The overlay visualization verifies that attention is concentrated around the selected clinical features rather than other image regions. The patient's complaint of "discomfort, vision blur" reflects a moderate attribution score for the textual modality, indicating that symptom information also affects the prediction. These findings demonstrate that the model integrates and bases its decision on both visual and textual inputs.

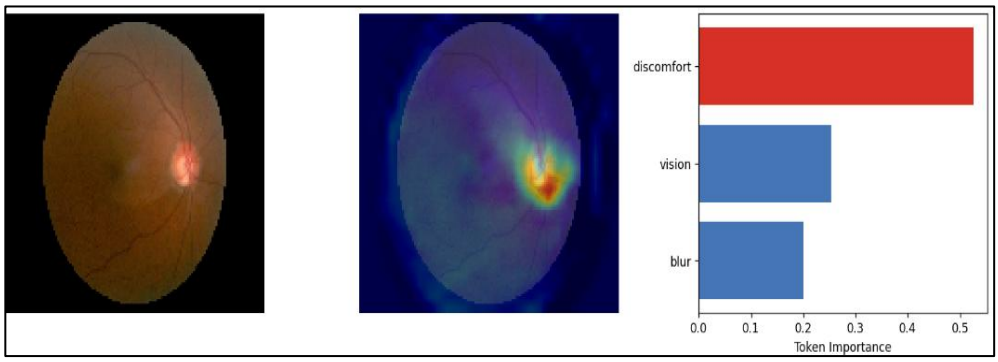

Figure 9. Integrated gradients for CNN-LSTM multi-model

The faithful metrics of Grad-CAM were used to validate the explanations. Insertion AUC and Deletion AUC were 0.73 and 0.76, respectively. Deletion AUC shows that when the highlighted regions are removed, prediction confidence decreases, while Insertion AUC increases prediction confidence as important pixels are gradually added back. This indicates that highlighted regions contribute meaningfully to the model's decision. The low Average Drop of 5.75% indicates that the regions identified by the Explainable AI (XAI) represent most of the predictive information. The increase in confidence, positively at 2.40%, shows that revealing the highlighted regions enhances the model's confidence, supporting the reliability and faithfulness of the generated explanations.

Table 9. CNN vs. LSTM modality contribution per class

Label	CNN contribution	LSTM contribution	Correct
AMD	1.000	0.000	0.833
Cataract	0.999	0.001	0.806
Diabetic	1.000	0.000	0.932
Glaucoma	0.998	0.002	0.898
Hypertensive	0.993	0.007	0.143
Myopia	0.999	0.001	0.781
Normal	0.993	0.007	0.996

To quantify the relative importance of the image and text modalities in the CNN-LSTM multi-model, modality contribution analysis was conducted. As per the results presented in Table 9, the modality contribution analysis showed that the CNN branch was dominant, contributing approximately 99.74% of the overall attribution, whereas the LSTM branch contributed only 0.026%. This is strong evidence that the model's predictions were driven primarily by retinal image features. The text modality provides complementary information that enhances overall performance despite its relatively low attribution scores.

5. Discussion

5.1 Complexity Analysis

Table 10 shows the computational complexity of the CNN-LSTM multi-model compared with CNN-only and LSTM-only baselines, evaluated from different perspectives: model size, trainable parameter count, floating-point operations (FLOPs), inference latency for batch sizes (1 and 32), and training convergence.

Table 10. Comparison of computational complexity of CNN-only, LSTM-only & CNN-LSTM

Metric	CNN-only	LSTM-only	CNN-LSTM
Test Accuracy (%)	63.3	74.11	92
Trainable Parameters	14,767,047	42,439	3,375,879
Model Size (MB)	112.70	0.53	38.72
FLOPs per Sample	0.137 B	0.002 B	0.289 B
Inference Latency (Batch = 1)	64.05 ms	74.02 ms	76.75 ms
Inference Latency (Batch = 32)	2.15 ms	2.24 ms	2.67 ms
Throughput (Batch = 32)	465.6 samples/s	446.5 samples/s	374.1 samples/s
Training Epochs (Early Stopping)	30	11	37

CNN-LSTM contains around 3.38 million trainable parameters, fewer than the CNN-only model's 14.77 million. This reduction affected the model size of 38.72 MB, approximately one-third of the CNN-only model. This size reduction allows CNN-LSTM to be more memory efficient without degrading predictive performance, demonstrating the efficiency of fusion. The computational cost of CNN-LSTM is 0.289 billion FLOPs per forward pass, with convolutional layers accounting for the majority of computations, while the LSTM branch contributes only a small fraction of the total workload. CNN-LSTM shows a latency of 76.75 ms for single-sample prediction and 2.67 ms for a batch size of 32. It also achieved a throughput of 374 samples/s on an NVIDIA RTX 3060 GPU, which means it is computationally efficient for real-time clinical applications. The LSTM-only model is the most lightweight and computationally efficient but lacks diagnostic performance due to the absence of fundus images. The CNN-only model has the largest parameter count and lowest classification accuracy. This comparative complexity analysis demonstrates that CNN-LSTM achieved the best results in balancing computational cost and classification performance.

The CNN-LSTM multi-model was designed to balance classification performance with computational efficiency, to support low-resource centers. The CNN-based branch contains three convolutional layers followed by lightweight fully connected layers, while the text branch employs one LSTM layer with a maximum length of 30 tokens. Compared to other transformer-based architectures that require substantially larger datasets and higher computational resources, the proposed CNN-LSTM architecture maintains a relatively low computational burden while effectively learning complementary visual and textual representations. In addition, the resizing of fundus images to (128 × 128) pixels reduces memory consumption and inference time without affecting or degrading diagnostic performance. This CNN-LSTM multi-model design enables efficient model training, making the framework practical for small ophthalmology clinics and low-cost screening systems.

5.2 Clinical Relevance

The main relevance is to provide a multi-model that has clinical efficiency to support glaucoma diagnosis in the presence of other retinal diseases. CNN-LSTM enables the integration of clinician-annotated patient symptoms, which provides additional contextual information that better reflects routine ophthalmic practice, where diagnosis relies on both retinal examination and patient-reported complaints.

The explainable AI and modality contribution analysis enhance the model's clinical trust by demonstrating that the model primarily focuses on clinically meaningful retinal regions, particularly around the optic disc, while using symptom information as only complementary evidence, not as a primary source for detection. Moreover, the high glaucoma detection performance (AUC = 0.996, F1-score = 0.93) indicates that the proposed system could serve

as an effective primary decision support tool to assist ophthalmologists in prioritizing high-risk patients for further expert examination.

As the model relies on low-cost fundus imaging and modest computing requirements, it has the potential to support the improvement of community clinics and limited healthcare facilities, where advanced imaging techniques or expert specialists may not always be available.

5.3 Research Limitations

There are some limitations the research faced regarding the dataset, model training, and interpretation. The chosen dataset contains 5,000 fundus images, which is considered relatively small for deep learning algorithms. Another limitation is that the classes are highly imbalanced, with dominant and rare classes. The dataset also shows a number of quality image issues. There is no available clinical metadata, such as IOP measurements, lab results, or visual field test results; the dataset only contains fundus images and simple demographic data, such as age and sex. The team manually built the metadata by describing patients' different complaints in natural language. The annotation process was another very difficult and time-consuming limitation the team faced because it was done blindly from the class label, "as if it were a diagnosis process." Then, the most common complaints that patients with this diagnosis normally mention were annotated for each image.

6. Conclusion and Future Work

Glaucoma is a disease that causes permanent sight loss. Although early detection cannot cure optic nerve damage, it can help reduce future damage. This paper discusses how the use of deep learning techniques can support the enhancement of glaucoma detection. In this research, the dataset used has several limitations, so the patient's complaints were integrated as metadata to enrich the clinical data. A hybrid multi-model is proposed based on CNN and LSTM architectures. The proposed model extracts features from low-cost fundus images and patients' complaints to detect different ophthalmic diseases. The model achieved a test accuracy of 92% and a macro-avg AUC of 96.8% across seven predicted classes. The model's precision macro-avg and weighted-avg were 74.7% and 92%, respectively. Also, the recall macro-avg and weighted-avg were 75.5% and 92%. While the model's F1-Score macro-avg and weighted-avg were 74.7% and 92%, respectively. These results indicate overall balanced detection performance across the seven diseases. This research demonstrates the effectiveness of augmenting patients' complaints into the clinical data, leading to noticeable improvements in model results. Several avenues remain open for future experiments. Integrate more valuable metadata to study its effect on prediction and interpretation, such as medical history, medication use, progression data as severity grades, or more clinical measurements like intraocular pressure to demonstrate deterioration over time. Validating and evaluating the model on larger and independent datasets remains an important direction as it would provide stronger evidence of generalization. The proposed multi-model contains a large number of parameters for the dataset size, which may increase the overfitting risk. Several measures were adopted to mitigate this overfitting, such as dropout and early stopping based on validation loss. Furthermore, bootstrap confidence intervals and McNemar's statistical analysis demonstrate that the observed performance is stable and statistically significant.

References

[1] Baudouin, Christophe, Miriam Kolko, Stéphane Melik-Parsadaniantz, and Elisabeth M. Messmer. "Inflammation in Glaucoma: From the Back to the Front of the Eye, and Beyond." Progress in retinal and eye research 83 (2021): 100916.

[2] Currie, Geoff, K. Elizabeth Hawk, Eric Rohren, Alanna Vial, and Ran Klein. "Machine Learning and Deep Learning in Medical Imaging: Intelligent Imaging." Journal of medical imaging and radiation sciences 50, no. 4 (2019): 477-487.

[3] Sewak, Mohit, Md Rezaul Karim, and Pradeep Pujari. Practical Convolutional Neural Networks: Implement Advanced Deep Learning Models Using Python. Packt Publishing Ltd, 2018.

[4] Chai, Junyi, Hao Zeng, Anming Li, and Eric WT Ngai. "Deep Learning in Computer Vision: A Critical Review of Emerging Techniques and Application Scenarios." Machine Learning with Applications 6 (2021): 100134.

[5] Yunita, Ariana, MHD Iqbal Pratama, Muhammad Zaki Almuzakki, Hani Ramadhan, Emelia Akashah P. Akhir, Andi Besse Firdausiah Mansur, and Ahmad Hoirul Basori. "Performance Analysis of Neural Network Architectures for Time Series Forecasting: A Comparative Study Of RNN, LSTM, GRU, And Hybrid Models." MethodsX 15 (2025): 103462.

[6] Zhou, Hans Aoyang, Dominik Wolfschläger, Constantinos Florides, Jonas Werheid, Hannes Behnen, Jan-Henrik Woltersmann, Tiago C. Pinto, Marco Kemmerling, Anas Abdelrazeq, and Robert H. Schmitt. "Generative AI in Industrial Machine Vision: A Review." Journal of Intelligent Manufacturing 37, no. 4 (2026): 1447-1470.

[7] Voulodimos, Athanasios, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. "Deep Learning for Computer Vision: A Brief Review." Computational intelligence and neuroscience 2018, no. 1 (2018): 7068349.

[8] Khan, Salman, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. "Transformers in Vision: A Survey." ACM computing surveys (CSUR) 54, no. 10s (2022): 1-41.

[9] Milad, Daniel, Fares Antaki, David Mikhail, Andrew Farah, Jonathan El-Khoury, Samir Touma, Georges M. Durr et al. "Code-Free Deep Learning Glaucoma Detection on Color Fundus Images." Ophthalmology Science 5, no. 4 (2025): 100721.

[10] Jyoti, Kumari, Saurabh Yadav, Chandrabhan Patel, Mayank Dubey, Pradeep Kumar Chaudhary, Ram Bilas Pachori, and Shaibal Mukherjee. "Implementation of FBSE-EWT Method In Memristive Crossbar Array Framework for Automated Glaucoma Diagnosis from Fundus Images." Biomedical Signal Processing and Control 100 (2025): 107087.

[11] Shyamalee, Thisara, Dulani Meedeniya, Gilbert Lim, and Mihipali Karunarathne. "Automated Tool Support for Glaucoma Identification with Explainability Using Fundus Images." IEEe Access 12 (2024): 17290-17307.

[12] Chuter, Benton, Justin Huynh, and Christopher Bowd. "Deep Learning Identifies High-Quality Fundus Photographs and Increases Accuracy in." Translational Vision Science & Technology, 13 (1) (2024): 23-23.

[13] Chiang, Yen-Ying, Ching-Long Chen, and Yi-Hao Chen. "Deep Learning Evaluation of Glaucoma Detection Using Fundus Photographs in Highly Myopic Populations." Biomedicines 12, no. 7 (2024): 1394.

[14] Alkhaldi, Nora A., and Ruqayyah E. Alabdulathim. "Optimizing Glaucoma Diagnosis with Deep Learning-Based Segmentation and Classification of Retinal Images." Applied Sciences 14, no. 17 (2024): 7795.

[15] Hemelings, Ruben, Bart Elen, João Barbosa-Breda, Matthew B. Blaschko, Patrick De Boever, and Ingeborg Stalmans. "Deep Learning on Fundus Images Detects Glaucoma Beyond the Optic Disc." Scientific Reports 11, no. 1 (2021): 20313.

[16] Ou, Xingyuan, Li Gao, Xiongwen Quan, Han Zhang, Jinglong Yang, and Wei Li. "BFENet: A Two-Stream Interaction CNN Method for Multi-Label Ophthalmic Diseases Classification with Bilateral Fundus Images." Computer Methods and Programs in Biomedicine 219 (2022): 106739.

[17] Ajitha, S., John D. Akkara, and M. V. Judy. "Identification of Glaucoma from Fundus Images Using Deep Learning Techniques." Indian journal of ophthalmology 69, no. 10 (2021): 2702-2709.

[18] Bajwa, Muhammad Naseer, Muhammad Imran Malik, Shoaib Ahmed Siddiqui, Andreas Dengel, Faisal Shafait, Wolfgang Neumeier, and Sheraz Ahmed. "Correction to: Two-Stage Framework for Optic Disc Localization and Glaucoma Classification in Retinal Fundus Images Using Deep Learning." BMC Medical Informatics and Decision Making 19, (2019): 153.

[19] Diaz-Pinto, Andres, Sandra Morales, Valery Naranjo, Thomas Köhler, Jose M. Mossi, and Amparo Navea. "CNNs for Automatic Glaucoma Assessment Using Fundus Images: An Extensive Validation." Biomedical engineering online 18, no. 1 (2019): 29.

[20] Peking University International Competition on Ocular Disease Intelligent Recognition (ODIR-2019) 2020. Available from: https://odir2019.grand-challenge.org/dataset/.

[21] Ram, Ajna, and Constantino Carlos Reyes-Aldasoro. "The Relationship Between Fully Connected Layers and Number of Classes for the Analysis of Retinal Images." arXiv preprint arXiv:2004.03624 (2020).

[22] Du, Fanyu, Lishuai Zhao, Hui Luo, Qijia Xing, Jun Wu, Yuanzhong Zhu, Wansong Xu, Wenjing He, and Jianfang Wu. "Recognition of Eye Diseases Based on Deep Neural Networks for Transfer Learning and Improved DS Evidence Theory." BMC Medical Imaging 24, no. 1 (2024): 19.

[23] Hanfi, Rabiya, Harsh Mathur, and Ritu Shrivastava. "Hybrid Attention-Based Deep Learning for Multi-Label Ophthalmic Disease Detection on Fundus Images." Graefe's Archive for Clinical and Experimental Ophthalmology 263, no. 10 (2025): 2901-2914.