

# PROJECT REPORT

*On*

## DIABETIC RETINOPATHY CLASSIFICATION

*Submitted in partial fulfillment for the award of the degree Of*

***Masters of Computer Applications***

By

## ARJUN P SAJI (MLM23MCA-2016)

Under the guidance of

### Ms. DIVYA S B

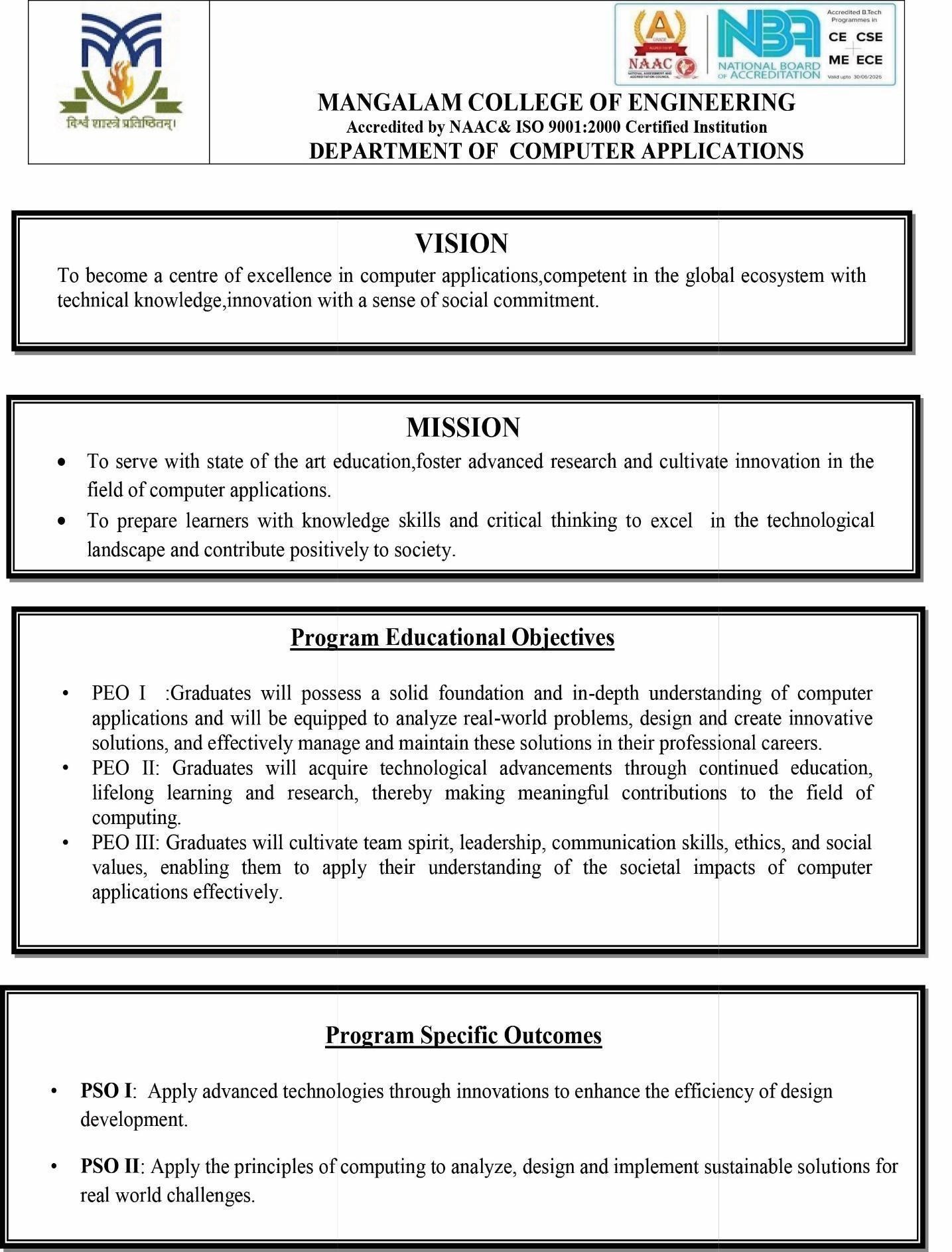
(HOD & Associate Professor, Dept. of Computer Applications)



# DEPARTMENT OF COMPUTER APPLICATIONS MANGALAM COLLEGE OF ENGINEERING, ETTUMANOOR

***(Affiliated to APJ Abdul Kalam Technological University)***

**APRIL 2025**



**MANGALAM COLLEGE OF ENGINEERING, ETTUMANOOR DEPARTMENT OF COMPUTER APPLICATIONS**

**APRIL 2025**

****

***CERTIFICATE***

*This is to Certify that the project entitled* ***“Diabetic Retinopathy Classification”*** *is the Bonafide record of the work done by* ***Arjun P Saji (MLM23MCA-2016)*** *of Masters of Computer Applications towards the partial fulfillment for the award of the DEGREE OF MASTERS OF COMPUTER APPLICATIONS under APJ ABDUL KALAM*

*TECHNOLOGICAL UNIVERSITY during the academic year 2024-2025.*

**Project Guide Project Coordinator**

**Ms. Divya S B Ms. Banu Sumayya S**

**HOD & Associate Professor Assistant Professor**

**Dept. of Computer Applications Dept. of Computer Applications**

**Head of the Department**

**Ms. Divya S B**

**Associate Professor**

**Dept. of Computer Applications External Examiner**

**ACKNOWLEDGEMENT**

First, I thank the Almighty **God** for giving me the strength to venture for such an enigmatic logical creation in a jovial way.

I am greatly indebted to the authorities of Mangalam College of Engineering for providing me the necessary facilities to successfully complete my Project on the topic “Diabetic Retinopathy Classification.”

I express my sincere thanks to **Dr.Vinodh P Vijayan**, the principal, for providing me with the best facilities to complete my seminar successfully.

I thank and express my solicit of gratitude to **Ms. Divya S B**, HOD & Associate Professor Dept. of Computer Applications, Mangalam College of Engineering, for his invaluable help and support which helped me a lot in successfully completing this Seminar.

I express my gratitude to my Internal Guide, **Ms. Divya S B**, HOD & Associate professor, Department of Computer Applications, for her suggestions and encouragement which helped me in the successful completion of my Seminar.

I express my gratitude to my project coordinator **Ms. Banu Sumayya S**, Assistant professor, Department of Computer Applications for the suggestions and encouragement which helped in the successful completion of our Project.

Finally, I would like to express our heartfelt thanks to my parents who were very supportive both financially and mentally and for their encouragement to achieve my goals.

**ARJUN P SAJI (MLM23MCA-2016)**

**ABSTRACT**

Diabetic Retinopathy (DR) is a progressive eye disease caused by diabetes, which can lead to vision impairment or blindness if not diagnosed early. With the increasing prevalence of diabetes, there is a rising need for automated diagnostic solutions to facilitate early detection and timely treatment. This study introduces a deep learning-based approach for DR classification, leveraging Convolutional Neural Networks (CNNs) and Transfer Learning techniques to enhance accuracy and efficiency.

Pre-trained models such as AlexNet and DenseNet-169 are utilized for extracting crucial retinal features, thereby improving classification performance. The proposed system follows a structured process, including image preprocessing, feature extraction, classification, and automated report generation, ensuring an efficient diagnostic workflow. Additionally, Grad-CAM-based explainability techniques provide visual representations of affected retinal regions, enhancing the interpretability of AI-driven decisions.

The effectiveness of the model is evaluated using benchmark datasets, demonstrating high accuracy and reliability in DR classification. Furthermore, integration with Electronic Health Records (EHRs) enables seamless management of patient data, supporting a scalable and efficient screening system. Future enhancements may involve the implementation of federated learning for improved data privacy, mobile-based real-time screening, and multi-disease detection. By combining AI-driven automation with expert validation, this approach aims to make DR detection more accessible, reduce the workload of ophthalmologists, and improve early diagnosis and patient outcomes.

**TABLE OF CONTENTS**

**TITLE**

|  |  |
| --- | --- |
| **List of Figures List of Tables**  **List of Abbreviations**  **1. INTRODUCTION** | **I II**  **III 1** |
| 1.1 Background | **1** |
| 1.2 Overview | **2** |
| 1.3 Problem Statement | **3** |
| 1.4 Motivation | **4** |
| 1.5 Scope | **5** |
| **2. LITERATURE REVIEW** | **6** |
| **3. PROPOSED SYSTEM** | **10** |
| **4. METHODOLOGY** | **12** |
| **5. SYSTEM ARCHITECTURE** | **15** |
| **6. MODULES** | **18** |
| **7. DIAGRAMS** | **22** |
| **7.1 DFD** |  |
| **7.1.1 LEVEL 0 DFD** | **23** |
| **7.1.2 LEVEL 1 DFD** |  |
| **7.2 ACTIVITY DIAGRAM** | **26** |
| **7.3 CLASS DIAGRAM** | **27** |
| **7.4 USE CASE DIAGRAM** | **29** |
| **8. TESTING** | **30** |
| **9. ADVANTAGES AND DISADVANTAGES** | **36** |
| **10. RESULTS AND CONCLUSION** | **37** |
| **11. APPENDICES** | **38** |
| **12. REFERENCES** | **42** |

|  |  |  |
| --- | --- | --- |
| **FIGURE No.** | **LIST OF FIGURES** | **PAGE No.** |
| Fig. 1 | Level 0 -DFD | 22 |
| Fig. 2 | Level 1: DFD -Patient | 22 |
| Fig. 3  Fig .4  Fig. 5  Fig. 6 | Level 1: DFD-Admin Activity Diagram Class Diagram  Use-case diagram | 23  25  27  28 |

**LIST OF ABBREVIATIONS**

|  |  |
| --- | --- |
| **ABBREVIATION** | **FULL FORM** |
| **AI** | - Artificial Intelligence |
| **DL** | - Deep Learning |
| **ML** | - Machine Learning |
| **CNN** | - Convolutional Neural Network |
| **DNN** | - Deep Neural Network |
| **ANN** | - Artificial Neural Network |
| **DR** | - Diabetic Retinopathy |
| **VGG16 GA** | * Visual Geometry Group * Generic Algorithm |

1. **INTRODUCTION**

## BACKGROUND

Diabetic Retinopathy (DR) is a significant complication associated with diabetes that impacts the retina, potentially leading to vision impairment or blindness if not detected and treated in a timely manner. With the increasing prevalence of diabetes worldwide, DR has become a critical public health issue, highlighting the necessity for early diagnosis and timely medical intervention. Conventional screening methods depend on ophthalmologists manually analyzing retinal images, a process that is both time-intensive and prone to human error. Additionally, the scarcity of specialized eye care professionals in certain regions further restricts the availability of timely diagnoses. To address these challenges, automated deep learning-based solutions have emerged as a promising approach for efficiently detecting and classifying DR.

Recent advancements in deep learning and computer vision have led to the development of highly accurate models for retinal image analysis. Convolutional Neural Networks (CNNs) combined with Transfer Learning techniques, employing pre-trained architectures such as AlexNet and DenseNet-169, have significantly improved feature extraction and classification capabilities. These models can effectively identify critical retinal abnormalities, including microaneurysms, hemorrhages, exudates, and neovascularization, which are key indicators of DR severity. By utilizing large-scale annotated datasets, deep learning algorithms can detect intricate patterns in retinal images, enhancing the reliability and scalability of automated screening systems. Furthermore, explainability techniques such as Grad-CAM provide visual representations of AI-generated predictions, allowing ophthalmologists to verify the results before confirming a diagnosis.

Beyond enhancing diagnostic precision, AI-powered DR detection systems support real-time deployment through web-based platforms and mobile applications, making screenings more accessible to a broader audience. Integration with Electronic Health Records (EHRs) facilitates efficient patient data management, allowing healthcare professionals to track disease progression systematically. Future advancements, including federated learning for secure data processing, edge computing for real-time analysis, and multi-disease classification capabilities, hold great potential for further improving the effectiveness and scalability of these systems.

## OVERVIEW

Diabetic Retinopathy (DR) is a major cause of vision loss among diabetic patients, making early detection essential for preventing severe complications. Traditional methods of DR diagnosis involve manual inspection of retinal images by ophthalmologists, which is time-consuming, subjective, and requires specialized expertise. To overcome these challenges, artificial intelligence (AI) and deep learning techniques have been introduced to automate DR detection and classification. This project focuses on developing an AI-driven system that utilizes Convolutional Neural Networks (CNNs) and Transfer Learning with models like AlexNet and DenseNet-169 to enhance diagnostic accuracy and efficiency. By leveraging large datasets of retinal images, the system can identify DR at different severity levels, ensuring timely medical intervention.

The proposed system follows a structured approach, beginning with image acquisition and preprocessing, where retinal images undergo noise reduction and enhancement techniques to improve clarity. This is followed by feature extraction using CNN-based architectures, allowing the model to detect key retinal abnormalities such as microaneurysms, hemorrhages, and exudates. The classification module then categorizes the images into different DR severity levels, ranging from mild to proliferative DR, enabling early diagnosis and treatment planning. Furthermore, Grad-CAM-based visualization techniques are integrated to provide explainability, allowing medical professionals to interpret AI predictions effectively.

Future enhancements of this project include federated learning for data privacy, multi-disease classification to detect other retinal disorders, and real-time processing using edge computing. By combining AI-powered automation with human expertise, this project aims to bridge the gap between manual diagnosis and automated screening, making DR detection faster, more accessible, and highly reliable. The implementation of such a system has the potential to reduce the global burden of diabetic blindness and improve patient care by enabling early and accurate diagnosis.

## PROBLEM STATEMENT

Diabetic Retinopathy (DR) is a progressive eye disease caused by prolonged diabetes, leading to damage in the retinal blood vessels. If not detected and treated in its early stages, DR can result in permanent vision loss or blindness. The growing number of diabetic patients worldwide has made DR one of the leading causes of preventable blindness. However, early detection remains a significant challenge, as the initial stages of DR often exhibit minimal or no symptoms. Traditional diagnostic methods rely on manual examination of fundus images by ophthalmologists, which can be time-consuming, subjective, and prone to human error. Furthermore, a shortage of trained medical professionals in many regions limits timely screenings, increasing the risk of undiagnosed and untreated cases.

Current diagnostic techniques also face scalability challenges, especially in rural and underdeveloped areas where access to specialized ophthalmologists is limited. The manual grading of retinal images requires expertise and significant time investment, making mass screening impractical. Additionally, human-based diagnoses may suffer from variability in interpretation, leading to inconsistencies in DR classification. With the increasing prevalence of diabetes worldwide, there is an urgent need for an automated and accurate system that can efficiently detect and classify DR across diverse populations. A system that can function with minimal human intervention while maintaining high accuracy is crucial to addressing this global health issue.

Advancements in deep learning and artificial intelligence (AI) have shown great potential in automating medical image analysis. However, developing an efficient, reliable, and interpretable AI-based DR detection system presents several challenges. Many deep learning models require large, well-annotated datasets, which may not always be available due to patient privacy concerns and data-sharing limitations. Additionally, existing AI-based models lack explainability, making it difficult for ophthalmologists to trust the automated predictions. A system that integrates Grad-CAM-based visualization techniques can improve transparency, allowing medical experts to validate AI-generated results. Furthermore, ensuring that the model is generalizable across different datasets, imaging conditions, and patient demographics remains a significant challenge.

## MOTIVATION

Diabetic Retinopathy (DR) is a leading cause of preventable blindness worldwide, particularly among individuals with prolonged diabetes. The rising prevalence of diabetes, combined with the challenges of early detection, has made DR a significant public health issue. Many cases remain undiagnosed until the condition progresses to an advanced stage, resulting in irreversible vision impairment. This project is driven by the urgent need to develop an efficient and accessible screening system that facilitates early detection and timely intervention. By integrating deep learning and artificial intelligence, automated DR detection can help reduce the risk of blindness and enhance patient care outcomes.

Traditional DR screening methods involve ophthalmologists manually analyzing and classifying retinal images. However, this approach is time-consuming, susceptible to human error, and difficult to implement on a large scale, particularly in areas with a shortage of trained professionals. In many developing regions, limited access to specialized eye care leads to delays in diagnosis and treatment. This project seeks to overcome these barriers by developing an AI-powered DR detection system that assists healthcare professionals in making faster and more accurate diagnoses. Automating the screening process can improve efficiency, ease the workload of medical professionals, and enable large-scale screening initiatives.

Recent progress in deep learning has demonstrated significant potential in medical image analysis, offering precise and efficient diagnostic solutions. Convolutional Neural Networks (CNNs) and Transfer Learning techniques have been widely adopted for medical image classification, showing remarkable success in detecting retinal abnormalities linked to DR. The motivation behind this project is to leverage these advanced technologies to develop a system capable of classifying DR severity levels with high accuracy.

This project aims to bridge the gap between technology and healthcare by ensuring that AI-driven DR detection systems become practical and scalable solutions for early diagnosis and vision loss prevention.

## SCOPE

This project aims to create an automated deep learning-based system for the detection and classification of Diabetic Retinopathy (DR). Given the increasing number of diabetes cases globally, early diagnosis and timely treatment are essential in preventing vision impairment. The system incorporates Convolutional Neural Networks (CNNs) and Transfer Learning methods, utilizing pre-trained models such as AlexNet and DenseNet-169 to improve accuracy and efficiency in DR classification. By processing retinal images, the model identifies critical abnormalities, including microaneurysms, hemorrhages, and exudates, facilitating early assessment of DR severity levels.

A major aspect of this project is real-time deployment for large-scale DR screening. Designed for cloud-based platforms and mobile applications, the system enables remote diagnosis, making it particularly beneficial for rural and underserved communities with limited access to ophthalmologists. Through an intuitive user interface, healthcare providers can perform rapid screenings, reducing dependence on manual assessments and expediting referrals for patients requiring further medical attention.

Beyond diagnosis, the project aims to incorporate a secure and scalable patient data management system. This feature will allow healthcare professionals to track patient history, compare previous and current retinal scans, and support more effective treatment planning. Additionally, integrating real-time processing capabilities will make the system suitable for emergency rooms and hospitals, providing instant diagnostic results.

Ensuring data privacy and security is another key objective of this project. Given the sensitivity of medical records, the system can implement federated learning techniques, allowing training across multiple healthcare institutions without sharing confidential patient data. Furthermore, integration with Electronic Health Records (EHRs) will facilitate seamless patient data management, enabling continuous monitoring of DR progression and supporting personalized treatment plans for diabetic patients.

# LITERATURE REVIEW

* 1. **Deep Learning and Automated Retinal Image Analysis for Diabetic Retinopathy Detection**

**Authors**: Zhang et al. (2017).

**Publication details:** IEEE Transactions on Medical Imaging, 36(4), pp. 949-960

This study explores the application of deep learning models in the automated detection of Diabetic Retinopathy (DR) using retinal images. The authors employ Convolutional Neural Networks (CNNs) trained on large-scale fundus image datasets to identify key DR-related abnormalities such as microaneurysms, hemorrhages, and exudates. The study highlights that CNN-based models outperform traditional machine learning techniques, offering improved accuracy and efficiency in classification. The authors also emphasize the importance of dataset quality, noting that well-annotated retinal images significantly enhance the model’s predictive performance. Furthermore, the research suggests that integrating deep learning with clinical decision support systems could assist ophthalmologists in diagnosing DR more efficiently. The findings demonstrate the potential of AI-driven approaches in enhancing early DR detection and reducing the dependency on manual grading by specialists.

**2.2 Deep Learning and Automated Retinal Image Analysis for Diabetic Retinopathy Detection**

**Authors:** Zhang et al. (2017), Quellec et al. (2010)

**Publication details:** IEEE Transactions on Medical Imaging, 36(4), pp. 949-960

Early DR classification models employed traditional machine learning techniques such as Support Vector Machines (SVMs), k-Nearest Neighbors (k-NN), and Decision Trees. These models relied on handcrafted feature extraction, including color histograms, texture analysis, and morphological transformations. While these approaches demonstrated reasonable classification accuracy, they required extensive preprocessing and were less adaptable to variations in retinal images.

Early research focused on traditional machine learning techniques, such as Support Vector Machines (SVMs), Random Forests, and k-Nearest Neighbors (k-NN), which required handcrafted features to classify retinal images. Niemeijer et al. (2007) proposed a machine learning-based technique that relied on texture analysis and intensity features for DR classification. However, these traditional methods suffered from low generalization due to the dependency on manually extracted feature

**2.3** **Retinal Lesion Detection with Deep Learning Using Fundus Images from Multiple Datasets**.

**Authors**: Lam C., Yu C., Huang L., Rubin D.

**Publication details:** Training deep learning models for medical applications often requires large annotated datasets, which are not always readily available. Transfer learning has emerged as a powerful solution to overcome this challenge by using pre-trained models trained on vast amounts of generic image data and fine-tuning them for specific medical tasks. Lam et al. (2020) explored the use of transfer learning for DR detection, leveraging pre-trained deep learning models like DenseNet-169 to classify fundus images. The study demonstrated that transfer learning significantly improves classification accuracy and reduces training time, making it an ideal approach for developing AI models in medical imaging where labeled data is scarce. The results showed that using pre-trained models improved feature extraction and reduced overfitting in DR classification. The study also emphasized the importance of domain adaptation techniques, which allow deep learning models to generalize well across different datasets collected from various medical institutions

**2.4 Grad-CAM: Visual Explanations from Deep Networks**

**Authors:** Selvaraju R. R., Cogswell M., Das A., Vedantam R., Parikh D.

**Publication details:** Int. J. Comput. Intell. Syst., vol. 16, no. 1, pp. 188, 2023.

One of the major challenges in adopting AI for medical diagnosis is the black-box nature of deep learning models, which makes it difficult for clinicians to understand how decisions are made. Selvaraju et al. (2017) addressed this issue by introducing Grad-CAM (Gradient-weighted Class Activation Mapping), an explainability technique that highlights important regions in an image that contribute to the model’s prediction. This technique enables ophthalmologists to visualize which parts of a retinal fundus image the AI system considers important for identifying diabetic retinopathy. The ability to generate heatmaps and visual explanations enhances trust and interpretability in AI-assisted DR detection. The study also highlighted how Grad-CAM could be integrated into clinical decision-support systems to provide justifications for automated diagnoses, thereby increasing the acceptance of AI-based models among healthcare professionals. Deep learning algorithms perform well when the number of samples in the dataset is large; therefore, it requires to collect a dataset with a larger sample size.

#### Federated Machine Learning: Concept and Applications

**Authors:** Yang Q., Liu Y., Chen T., Tong Y

**Publication details:** *Journal of Machine Learning Research*, 21(1), 1–49, 2020**.**

Medical AI models require large datasets for training, but data privacy concerns often limit data sharing among hospitals and research institutions. Federated learning (FL) offers a novel solution by enabling collaborative model training across multiple institutions without sharing raw patient data. Yang et al. (2020) proposed a federated learning framework for diabetic retinopathy classification, allowing different medical centers to train a shared AI model while keeping patient data localized. The study showed that federated learning achieves high classification accuracy comparable to centralized training approaches while maintaining HIPAA and GDPR compliance for patient data privacy. The findings highlighted the potential of federated learning to create robust, generalized AI models for DR detection that can be adopted across various healthcare settings.

over holdout validation even after the time trade-off because using the k-fold cross- validation gives us around 0.1-3% more accurate result, in general, as opposed to the hold- out validation.

**2.6 A Deep Learning-Based Framework for Diabetic Retinopathy Detection**

**Author:** Shu Zhang, Dequan Zheng

**Publication details:** Pacific Asia Conference 2015

Automated classification of semantic relations plays a crucial role in various applications, including information extraction, question answering, and the development of semantic knowledge bases. This study focuses on classifying semantic relations between nominal pairs within a given sentence. Specifically, the research investigates nine semantic relations: Cause-Effect, Instrument-Agency, Product-Producer, Content-Container, Entity-Origin, Entity-Destination, Component-Whole, Member-Collection, and Message-Topic. If none of these relationships apply, the relation is categorized as 'Other.'

For relation classification, a BiLSTM (Bidirectional Long Short-Term Memory) network is employed. This model processes sentences containing entity pairs and determines the nature of their relationship. The BiLSTM network utilizes sequential information from both past and future words in a sentence, enhancing its ability to capture contextual dependencies. The study proposes the use of BiLSTM networks to improve relation classification by addressing long-distance dependencies within text, which traditional models may struggle to handle effectively.

1. **PROPOSED SYSTEM**

#### Key Features of the Proposed System

* + 1. **Image Preprocessing:**
       - The system applies image enhancement techniquessuch as contrast adjustment, noise reduction, and vessel segmentation to improve image clarity.
       - Data augmentation techniques(rotation, flipping, and scaling) are used to increase the variability of training data.

#### Deep Learning-Based Classification:

* + - * A pre-trained deep CNN model (such as DenseNet-169 or AlexNet)is fine-tuned to classify DR images into different severity levels (No DR, Mild, Moderate, Severe, and Proliferative DR).
      * The model extracts deep featuresfrom retinal images, learning patterns indicative of DR lesions such as microaneurysms, hemorrhages, and exudates.

#### Explainability Using Grad-CAM:

* + - * To enhance interpretability, the system integrates Grad-CAM (Gradient-weighted Class Activation Mapping) to highlight important regions in the retinathat contribute to the model’s decision.
      * This feature ensures that ophthalmologists can validate AI-generated predictions

before making clinical decisions.

#### Real-Time Deployment and Cloud Integration:

* + - * The system is designed for real-time processingand can be deployed on mobile applications, allowing for remote diagnosis and telemedicine integration.
      * Edge computing can be used to enable on-device DR detection for portable screening in remote areas.

#### Federated Learning for Privacy-Preserving AI Training:

* The system leverages federated learning, allowing multiple hospitals and institutions to

#### train a shared AI model without sharing patient data.

* This approach maintains patient data privacywhile improving model robustness.

#### Steps in the Proposed System

1. **Data Acquisition and Preprocessing**
   * **Input:** Retinal fundus images from publicly available datasets (such as Kaggle’s EyePACS, APTOS, and Messidor).

#### Preprocessing Techniques:

* + - Image resizingto standard dimensions.
    - Contrast enhancementusing adaptive histogram equalization.
    - Noise reduction via Gaussian filtering.
    - Blood vessel segmentationto focus on key retinal structures.

#### Feature Extraction and Deep Learning Model Training

* + Use of Pre-trained CNNs (AlexNet, DenseNet-169**)** to learn hierarchical image features.
  + Fine-tuning of the CNN modelto classify images into different DR severity levels.
  + Optimization techniques**:** Learning rate tuning, dropout regularization, and data augmentation to prevent overfitting.

#### Explainable AI for Model Interpretability

* + Grad-CAM heatmapsgenerated for each prediction.
  + Visual explanationshelp ophthalmologists verify AI-based classifications.

#### Cloud and Mobile Deployment for Telemedicine

* + Web-based or mobile-friendly user interfacefor uploading retinal images and receiving DR classification results.
  + Integration with hospital systems (Electronic Health Records - EHRs)for automated patient report generation.

#### Validation and Performance Evaluation

* + Evaluation Metric**s:** Accuracy, Sensitivity, Specificity, Precision, and F1-score.

#### Comparison with existing DR classification methods.

1. **METHODOLOGY**

The methodology followed in the development of the Diabetic Retinopathy Detection and Classification System consists of multiple stages, including data collection, preprocessing, model selection, training, evaluation, and deployment. The goal of this approach is to create an accurate and efficient deep learning-based system that can classify retinal fundus images into different stages of diabetic retinopathy

### Data Collection and Preprocessing

The Diabetic Retinopathy Detection and Classification System was developed using high-quality retinal fundus images obtained from publicly available datasets such as Kaggle APTOS 2019, Messidor-2, and IDRiD. These datasets contain images categorized into five stages: No DR, Mild, Moderate, Severe, and Proliferative DR. To improve model accuracy, several preprocessing techniques were applied to enhance image quality and preprocessing techniques are applied to enhance image quality. These include:

* **Noise Reduction:** Removing unnecessary noise using filters like Gaussian blur and median filters.
* **Contrast-Enhancement:** Adjusting brightness and contrast using histogram equalization to make retinal regions more distinguishable.
* **Normalization:** Ensuring all images have consistent intensity ranges for better model training.
* **Image Resizing and Cropping:** Standardizing image dimensions to maintain uniformity across the dataset.

This preprocessing step ensures that the images are clear, standardized, and optimized for training the deep learning model.

### Image Segmentation and Feature Extraction

After preprocessing, In the Diabetic Retinopathy Detection and Classification System, image segmentation and feature extraction are essential processes for accurately identifying retinal abnormalities.

Key segmentation techniques include:

* + **Thresholding Techniques:** This technique separates different regions of an image based on intensity values.
  + **Edge Detection Algorithms:** These filters highlight the boundaries of retinal features, assisting in the identification of vessel structures and abnormal growths.
  + **Region-Based Segmentation:** Dividing the image into meaningful regions based on intensity distributions.

Once segmentation is complete, feature extractionis performed to identify key patterns in the images. Features such as texture, shape, intensity variations, and symmetry are analyzed, helping distinguish between normal and DR affected retinal images. These extracted features are critical for training the AI model.

### Deep Learning Model Training

A Convolutional Neural Network (CNN)-based deep learning model is used for DR classification. CNNs are particularly effective in analyzing medical images as they automatically extract important features without requiring manual intervention.

#### Training Process:

* + The dataset is split into training, validation, and test sets to ensure the model learns effectively.
  + The CNN model is trained using multiple convolutional layers that detect patterns at different levels of complexity.
  + Activation functions like ReLU (Rectified Linear Unit) are used to introduce non- linearity into the model, helping it learn complex relationships.
  + Optimization techniques like Adam optimizer and loss functions like categorical cross- entropy ensure efficient model convergence.
  + Data Augmentation techniques, such as rotation, flipping, and scaling, are used to increase dataset diversity and improve model robustness.

After multiple training epochs, the model learns to differentiate between normal, mild, and proliferate DR cases with high accuracy.

.

### Classification and Prediction

The Diabetic Retinopathy Detection and Classification System employs deep learning techniques to analyze retinal fundus images and categorize them based on the severity of diabetic retinopathy. This classification process is crucial for early detection, enabling timely medical intervention and reducing the risk of vision loss.

* + **No DR** – No signs of Diabetic Retinopathy.
  + **Mild DR** – Retinal abnormalities, including hemorrhages and exudates, are visible.
  + **Severe DR** – Increased presence of hemorrhages and microaneurysms, signaling disease progression.

The model also provides a confidence score, indicating how certain it is about the prediction. This

score helps doctors assess the AI-generated diagnosis before making clinical decisions.

### Model Evaluation and Performance Metrics

To ensure accuracy and efficiency, the model is rigorously tested using various performance metrics, including:

* + Accuracy: Measures the overall correctness of predictions.
  + Precision: Indicates how many predicted DR cases are actually correct.
  + Recall: Evaluates the model’s ability to detect actual DR cases.
  + F1-Score: Balances precision and recall for overall model effectiveness.

Additionally the performance of the Diabetic Retinopathy Detection and Classification System is essential to verify its accuracy, reliability, and suitability for real-world clinical applications. The deep learning model's effectiveness was evaluated using multiple performance metrics to determine how well it classifies retinal fundus images. These metrics provide insights into the model’s strengths and areas for improvement, ensuring it can accurately detect and categorize different stages of diabetic retinopathy.

# SYSTEM ARCHITECTURE

#### Step 1: Data Acquisition and Image Collection

The first step in the system is acquiring high-quality retinal fundus images from public datasets and medical institutions. The images are sourced from well-known databases like:

* + - Kaggle’s EyePACS Dataset
    - APTOS 2019 Blindness Detection Dataset
    - Messidor and Messidor-2 Datasets

Additionally, hospitals and ophthalmology centers contribute real-world retinal images obtained through fundus cameras. These images vary in contrast, brightness, and resolution, requiring preprocessing before classification.

#### Step 2: Image Preprocessing

Since raw fundus images may contain artifacts, noise, and variations in lighting, preprocessing techniques are applied to enhance their quality. The preprocessing stage involves:

1. **Image Resizing** – Standardizing all images to a fixed dimension to ensure uniformity.
2. **Contrast Enhancement** – Using adaptive histogram equalization to improve the visibility of key retinal structures.
3. **Noise Reduction** – Applying Gaussian filtering or median filtering to remove unwanted noise.
4. **Blood Vessel Segmentation** – Detecting and segmenting blood vessels for better feature extraction.
5. **Data Augmentation** – Techniques like rotation, flipping, and scaling are applied to increase the dataset’s diversity, improving model generalization.

This step ensures that the images are clean, standardized, and optimized for feature extraction and classification.

#### Step 3: Feature Extraction Using Deep Learning

Feature extraction is performed using deep convolutional neural networks (CNNs**)**, which learn hierarchical patterns from retinal images. Instead of training from scratch, pre-trained models such as AlexNet, DenseNet-169, and ResNet are fine-tuned for diabetic retinopathy classification.

* + **Convolutional Layers:** Extract key features like microaneurysms, hemorrhages, and exudates

that indicate DR.

* + **Pooling Layers:** Reduce dimensionality and computational complexity while preserving important features.
  + **Fully Connected Layers:** Interpret extracted features and pass them to the classification layer.

Deep learning models help automate feature extraction, removing the need for manual feature selection.

#### Step 4: Classification of Diabetic Retinopathy Stages

After feature extraction, the system classifies retinal images into different DR severity levels, following the standard grading system:

1. **No DR** – Healthy retinal images with no signs of diabetic retinopathy.
2. **Mild DR** – Presence of microaneurysms, indicating early-stage DR.
3. **Moderate DR** – More prominent lesions such as hemorrhages and exudates appear.
4. **Severe DR** – Extensive retinal damage, including blood vessel abnormalities.
5. **Proliferative DR** – The most advanced stage, with new abnormal blood vessel growth leading to vision loss.

A softmax classification layer assigns the probability scores for each category, ensuring precise diagnosis.

#### Step 5: Explainability Using Grad-CAM for Decision Support

To enhance the transparency of AI-driven diagnosis, the system incorporates Grad-CAM (Gradient-weighted Class Activation Mapping) for explainability. This step:

* Generates heatmaps to highlight important areas in the retina contributing to classification.
* Helps ophthalmologists verify whether the model’s predictions align with medical knowledge.
* Improves trust and reliability in automated DR detection.

By making AI-based predictions interpretable, clinicians can make informed decisions before confirming a diagnosis.

#### Step 6: Prediction, Reporting, and Doctor Verification

After classification, the system generates a detailed report that includes:

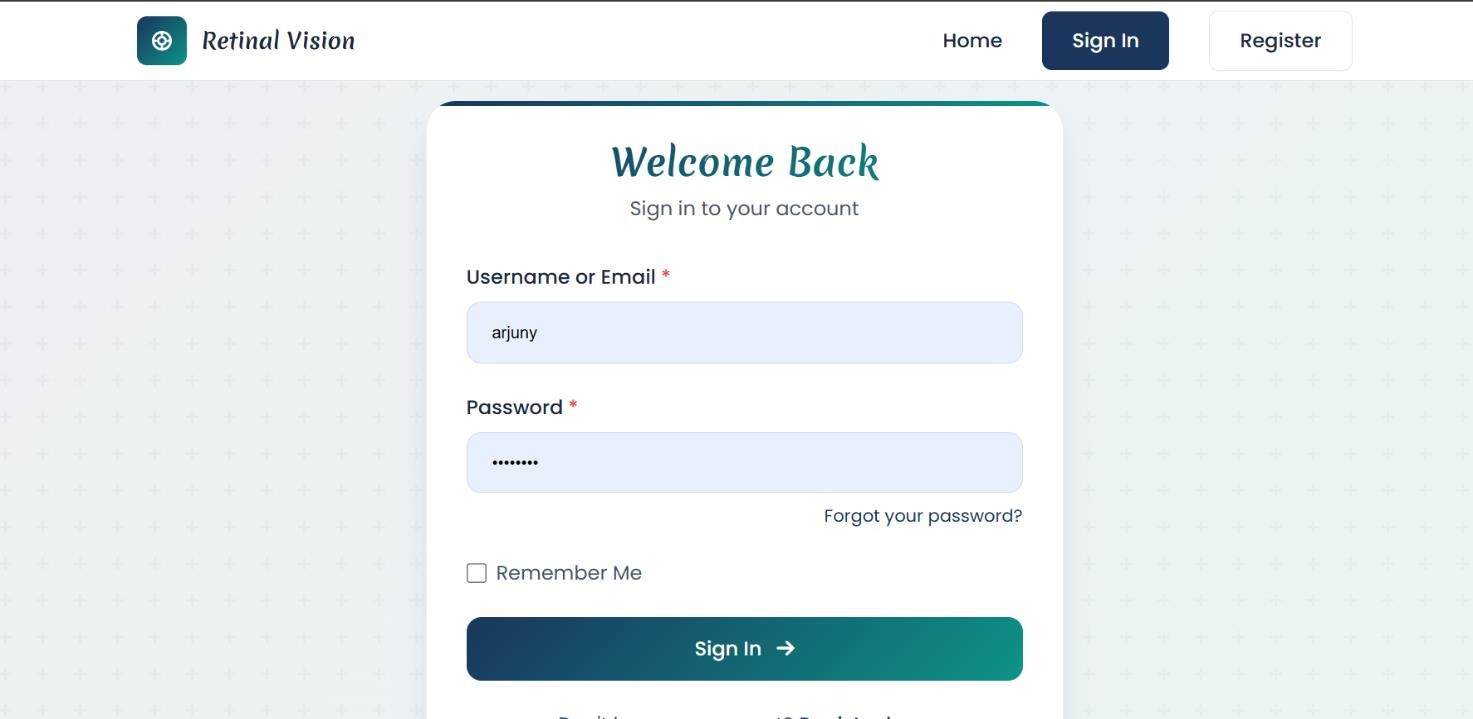
* Classification results (DR severity level).
* Grad-CAM heatmaps for visual confirmation.
* Automated recommendations based on severity (e.g., "Consult an ophthalmologist immediately" for severe cases).

Doctors can review AI-generated reports and verify them before finalizing the diagnosis. The system ensures human oversight, preventing misdiagnoses.

# MODULES

### User Authentication and Admin Dashboard Module

This module manages user roles, allowing patients and administrators to log in securely. Patients can upload scans and book appointments, while admins can view, manage, and track appointments. Security features such as password hashing and session management ensure data privacy.



### Image Upload and Preprocessing Module

Patients can upload retinal fundus image in various formats (JPG, PNG, JPEG). The system resizes images, removes noise, and applies preprocessing techniques This step ensures that images are optimized for accurate model predictions Patients can book consultations with ophthalmologist or radiologists after receiving their scan results. This module allows users to select doctors, choose appointment dates, and store booking details. Admins can manage these appointments through the admin panel.

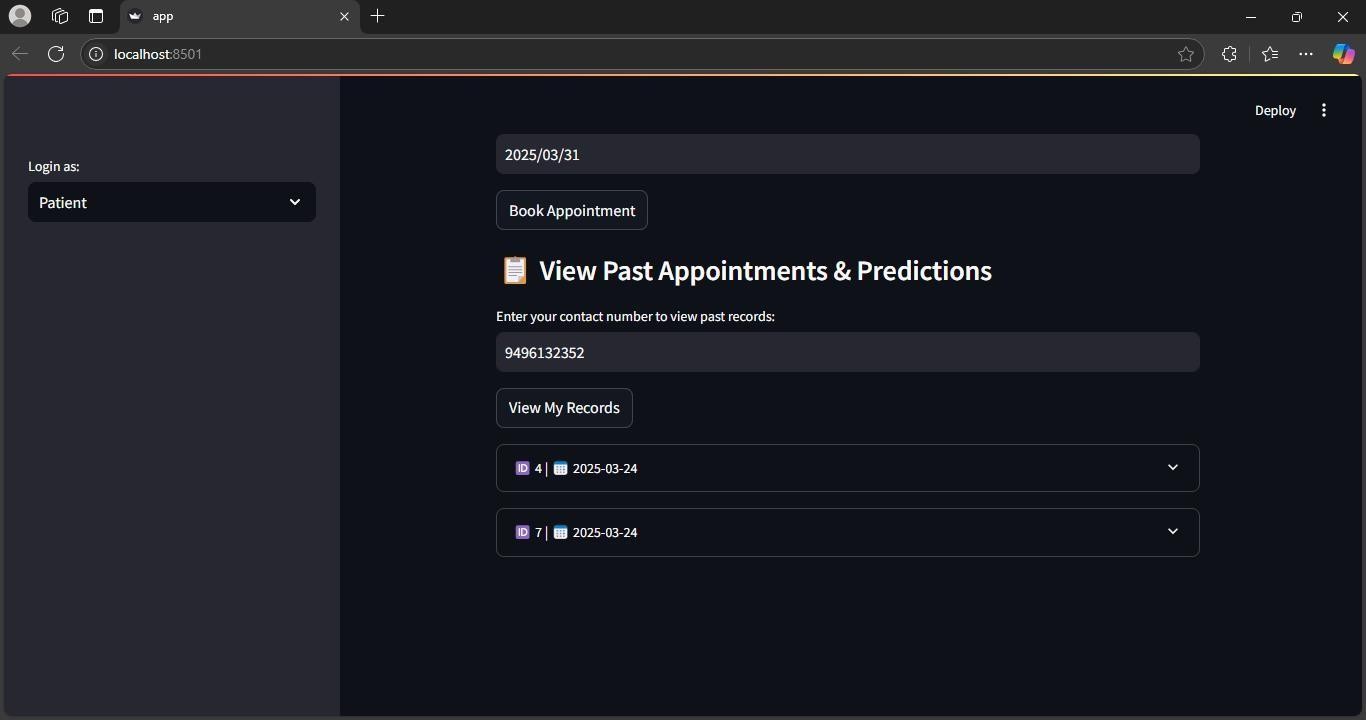
This module features an easy-to-use interface where patients can select a doctor, choose a suitable date and time, and provide their contact information.Once the appointment is confirmed, the system securely stores the details in the database and provides a confirmation message to the patient. Additionally, automated reminders help reduce missed appointments.

On the admin side, administrators can view, update, or cancel appointmentsas necessary. A dashboard displays scheduled consultations, making it easier to manage doctor availability and prevent overlapping appointments. The system also maintains a record of past and upcoming appointments**,** ensuring efficient tracking of patient visits and optimizing workflow for healthcare providers.

On the admin side, administrators can view, update, or cancel appointmentsas necessary. A dashboard displays scheduled consultations, making it easier to manage doctor availability and prevent overlapping appointments. The system also maintains a record of past and upcoming appointments**,** ensuring efficient tracking of patient visits and optimizing workflow for healthcare providers.

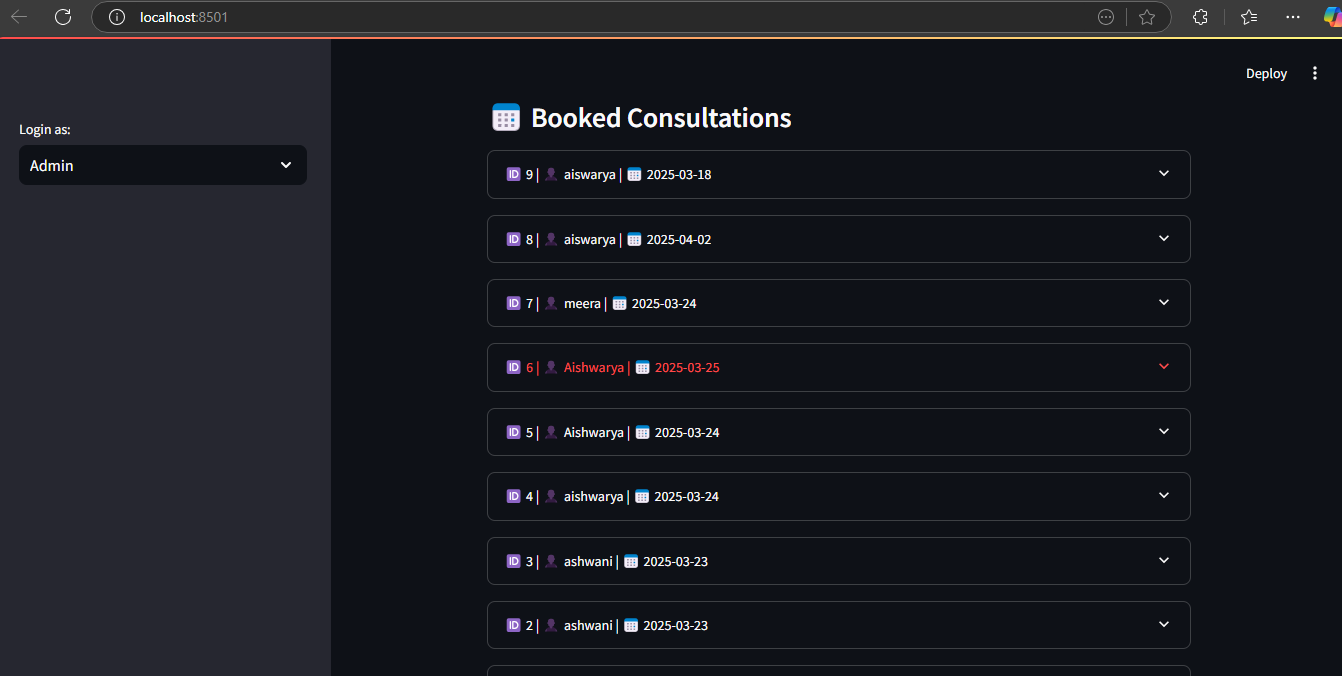
### Past Appointments and Scan History Module

Patients can access their previous diabetic related predictions and appointments. This module retrieves past scan results and consultations from the database, providing an overview of medical history. Admins can also track patient records for better monitoring and follow-up.



### Admin Viewing Appointments and Scan Module

The admin panel allows easy management of patient appointments and Diabetic Retinopathy classification records. Admins can view, filter, and track past consultations while accessing brain scan images with predictions. The system ensures secure data handling, enabling authorized personnel to review and manage records efficiently.



# DIAGRAMS

### DFD (Data Flow Diagram).

A **Data Flow Diagram (DFD)** is a graphical representation of how data flows through a system. It visually maps out the processes involved, the data inputs and outputs, and how the data is stored and retrieved. DFDs are used to understand, analyze, and improve systems by breaking them down into processes, data flows, and data stores.

DFDs are a useful tool for both technical and non-technical stakeholders as they provide a high-level view of the system without delving into the implementation details. This allows for a better understanding of the functional aspects of a system.

### Levels of a DFD

* + - **Level 0 (Context Diagram)**: This is the highest-level DFD that represents the system as a whole with a single process and external entities interacting with it. It provides a broad overview.
    - **Level 1 DFD**: This breaks down the Level 0 process into sub-processes, showing more detail about the system.
    - **Level 2 (and beyond)**: Further breakdown of Level 1 processes into smaller, more detailed sub-processes.

### DFD Symbols & Relationships

#### External Entity (Rectangle/Oval)

* + Represents users or systems that interact with the system but are external to it (e.g., "User," "Doctor," "Admin").
  + They provide inputs to the system and receive outputs.

#### Process (Circle or Rounded Rectangle)

* + Represents the actions or functions performed by the system (e.g., "User Registration," "Booking Consultation").
  + Processes transform data from inputs to outputs.

#### Data Store (Open Rectangle/Two Horizontal Lines)

* + Represents places where data is stored for future use (e.g., "User Database," "Appointments Database").
  + Data stores provide and store information to and from processes.

#### Data Flow (Arrow)

* + Represents the movement of data between external entities, processes, and data stores (e.g., "Booking Request," "Confirmation Email").

## DFD LEVEL 0

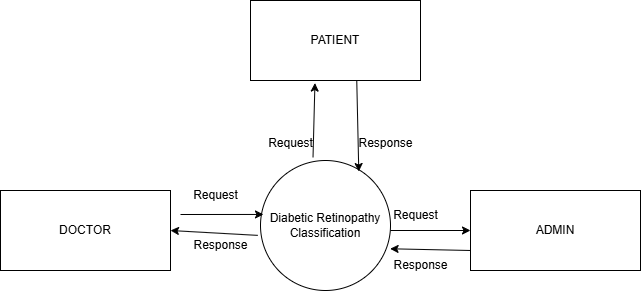
****

Fig 1: Level0 DFD

## DFD LEVEL 1- PATIENT

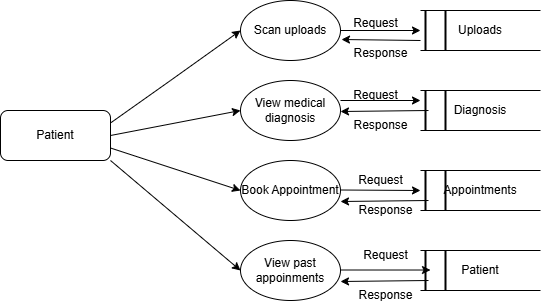
****

Fig 2: Level 1 DFD-PATIENT

s

## DFD LEVEL 2-ADMIN

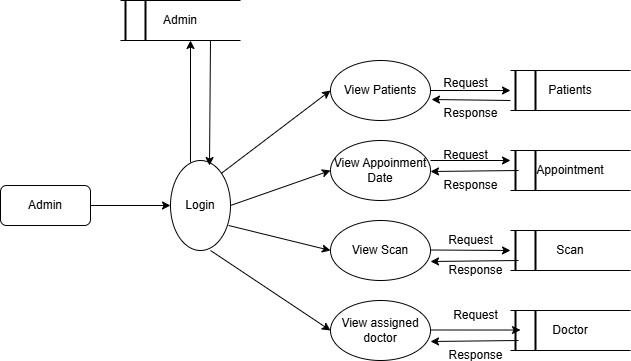
****

Fig 3: Level 1 DFD-Admin

### Activity Diagram

**Activity Diagram Symbols and Their Meanings:**

#### Initial Node (●):

* + - The entry point of the activity diagram, indicating where the workflow or process begins. It has no incoming flows, only outgoing.

#### Activity/Action (Rounded Rectangle):

* + - Represents a specific action or task that occurs during the workflow. Each action typically involves some operation or decision (e.g., "Fill Form," "Process Payment").

#### Final Node (◎):

* + - The endpoint of the diagram. It signifies the completion of all activities in the process and has only incoming flows, with no outgoing ones.

#### Decision Node (◇):

* + - A point where the process branches into two or more possible paths based on a condition or decision (e.g., "Is payment successful?"). Each outgoing flow represents a different condition.

#### Merge Node (◇):

* + - Combines multiple alternative paths back into one. This does not involve a condition, it simply merges flows without decision-making.

#### Fork Node (Thick Horizontal/Vertical Line):

* + - Splits one action into multiple concurrent flows, allowing activities to happen simultaneously. For example, after "Log In," both "Check Account" and "Show Dashboard" can occur at the same time.

#### Join Node (Thick Horizontal/Vertical Line):

* + - Combines multiple parallel flows back into one, ensuring all tasks are completed before continuing the process.

#### Control Flow (→):

* + - A directional arrow representing the flow of control between activities or decisions. It shows the sequence of tasks in the process.

#### Object Flow (Dashed Arrow):

* + - Represents the flow of data or objects between activities. It shows how information moves through the process.

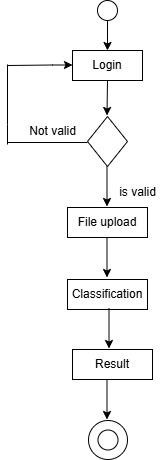


Fig 4: ACTIVITY DIAGRAM

### Class Diagram

The Class Diagram for the DR classification System outlines the main entities, including User, Doctor, Admin and Appointment. It illustrates the attributes and methods of each class and their relationships, showing how users interact with doctors and administrators within the system. This diagram serves as a blueprint for understanding the structure and functionality of the application.

### Class Diagram Symbols and Their Meanings

#### Class (Rectangle with Three Sections):

* + - * **Top Section**: Contains the class name (e.g., "User").
      * **Middle Section**: Lists the class's attributes (e.g., "username", "email").
      * **Bottom Section**: Shows the methods or operations (e.g., "login()", "
      * **Functionality**: Represents an object or entity in the system, encapsulating

its attributes (data) and behaviors (functions).

#### Association (Line with Optional Arrow):

* + - * **Description**: A simple line connects two classes, representing a relationship.

The optional arrow shows the direction of the relationship (e.g., "User" owns "Profile").

* + - * **Functionality**: Indicates how classes interact or communicate with each other.

#### Multiplicity (Numbers on Association Line):

* + - * **Description**: Indicates the number of instances involved in the relationship (e.g., "1" or "1.\*").
      * **Functionality**: Shows how many objects from one class are associated with another

(e.g., a "User" can have multiple "Appointments").

#### Generalization/Inheritance (Solid Line with Empty Triangle Arrow):

* + - * **Description**: A line with a hollow triangle arrow pointing toward the parent class.
      * **Functionality**: Represents inheritance, where a subclass
      * (e.g., "Admin") inherits attributes and methods from a parent class (e.g., "User").

#### Aggregation (Line with Empty Diamond):

* + - * **Description**: A line with an empty diamond at the class that contains another

(e.g., "Department" has "Employees").

* + - * **Functionality**: Represents a whole-part relationship.

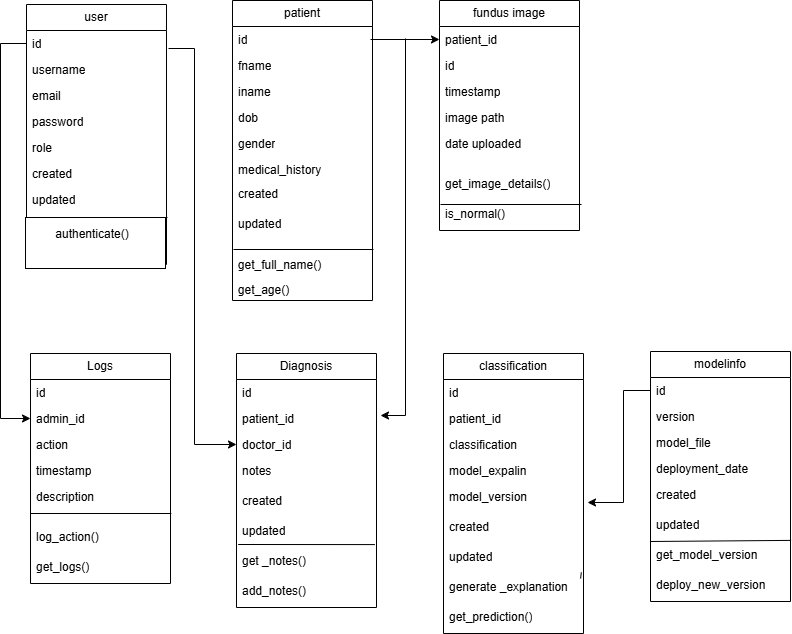


Fig 5: CLASS DIAGRAM

### Use Case Diagram

A Use Case Diagram visually depicts the interactions between users (actors) and the system, highlighting the functionalities or use cases that the system offers. It illustrates the relationships between actors and the various use cases, providing a clear understanding of the system's

**Use Case Diagram Shapes:**

#### Actor (Stick Figure or Rectangle):

Represents users or external systems that interact with the system.

#### Use Case (Oval):

Represents a function or service that the system provides to the actors.

#### System Boundary (Rectangle):

Defines the scope of the system, encapsulating all the use cases within it.

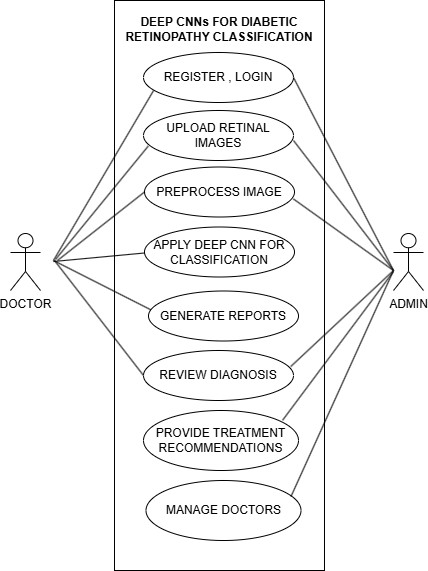


Fig 6: USE-CASE DIAGRAM

# TESTING

Testing is an essential phase in the development of a Diabetic Retinopathy Detection and Classification system, ensuring the model’s accuracy and reliability before being used in real- world clinical applications. This stage is crucial to verifying that the system can effectively classify retinal fundus images into different stages of diabetic retinopathy, ranging from healthy to severe cases. Without proper testing, the model may provide incorrect diagnoses, leading to potential risks for patients. Additionally, testing helps confirm the system's consistency across different datasets and imaging conditions, making it more applicable to diverse clinical environments.

The primary goal of testing is to evaluate the model’s ability to correctly identify diabetic retinopathy cases while ensuring robustness, efficiency, and accuracy. To achieve this, the dataset is divided into training, validation, and testing subsets to provide an unbiased assessment. The model is then tested on previously unseen images to determine its generalization ability. Several performance metrics, such as accuracy, precision, recall, F1- score, and AUC-ROC, are used to measure its effectiveness. By analyzing these results, researchers can identify potential weaknesses in the model and apply necessary improvements to enhance performance.

Apart from numerical evaluation, real-world testing is vital for assessing the system’s practical effectiveness. The model is tested on datasets obtained from various sources to determine how well it performs across different patient groups. Additionally, error analysis is conducted to identify cases of incorrect classification, such as false positives and false negatives. Recognizing these errors helps in refining the model through techniques like data augmentation, hyperparameter tuning, and optimization. Ultimately, thorough testing ensures that the diabetic retinopathy detection system is accurate and efficient, making it a valuable tool for early diagnosis and improved patient care Below are the different types of testing conducted on the project:

Types of Testing Conducted on the Project

### Unit Testing

Unit testing was conducted to evaluate the functionality of individual components, such as image preprocessing, classification, and database operations. Each module was tested separately to ensure that it performed correctly before being integrated into the system. Unit testing involves testing individual components of the system separately to verify their

correctness. In this project, different modules such as image preprocessing, feature extraction, and classification were tested independently. For example, preprocessing steps like resizing, normalization, and augmentation were evaluated to ensure they correctly modify images without altering important retinal features.

* + Image Upload & Preprocessing Module**:** Tested image resizing, format conversion, and normalization functions. Different image formats were uploaded to check the robustness of preprocessing.
  + DR Classification Module**:** Verified that the machine learning model loaded correctly and produced consistent predictions for predefined test images.
  + Database Management Module**:** Ensured that data insertion, retrieval, and deletion operations were correctly executed for patient records and scan results.
  + Appointment Booking Module**:** Checked the validation of input fields like patient name, contact number, and selected date to prevent incorrect data entries.
  + Admin Panel Module**:** Tested admin authentication functions to confirm login credentials were correctly verified

### Functional Testing

Functional testing was performed to verify that all features of the system worked as expected. This testing ensured that users could seamlessly interact with the platform without encountering errors.

* + DR Classification Module: Ensured that the correct DR type (Normal, Mild, Moderate) was displayed based on model predictions.
  + Appointment Booking Module: Checked whether users could successfully select doctors, pick dates, and book appointments.
  + Admin Panel Module: Verified that admins could access patient data, view scan results, and manage appointments efficiently.

### Integration Testing

Integration testing was carried out to confirm that different components of the system interacted correctly.

* + Image **Upload & DR Classification:** Tested whether the uploaded images were correctly preprocessed and passed to the model for prediction.
  + **Database Management & Appointment Booking:** Verified that booked appointments were stored in the database and were correctly retrievable by both patients and administrators.
  + **Admin Panel & Database:** Ensured that the admin panel successfully fetched and displayed stored scans and appointments from the database.

### Performance Testing

Performance testing assessed the system's efficiency under varying workloads.

* + DR Classification Module: Measured the model’s inference time to ensure that predictions were generated quickly.
  + Database Management Module: Tested system performance while handling multiple simultaneous queries to ensure smooth operations.
  + Appointment Booking Module: Assessed response time when multiple users attempted to book appointments simultaneously.

### Security Testing

Security testing was conducted to safeguard sensitive patient data and prevent unauthorized access.

* + Database Management Module: Performed SQL injection testing to check for vulnerabilities in database queries.
  + Admin Panel Module: Ensured secure login with encrypted credentials to prevent unauthorized access.
  + Appointment Booking Module: Verified input validation to prevent malicious data entries.

### User Acceptance Testing (UAT)

User acceptance testing involved real users interacting with the system to provide feedback on its usability and effectiveness. Feedback from UAT was used to refine the system and improve its overall user experience.

* + Patients Tested: Image upload, DR classification results, and appointment booking features.
  + Doctors Tested: Access to stored patient data, scan images, and appointment

By conducting these tests, the system was optimized for accuracy, performance, security, and usability, ensuring reliability in DR detection and patient management.

#### Tools Used for Testing

* **PyTest:** Used for unit testing different functions, such as image preprocessing, model predictions, and database operations.
* **Selenium:** Utilized for functional and UI testing to ensure smooth navigation and proper interaction with the user interface.
* **Postman:** Used for API testing to validate data exchange between the frontend and backend.
* **SQLite Browser:** Helped in manually verifying stored data and testing database queries for correctness.

#### Testing Process

1. **Test Plan**

A test plan was created to define the scope, objectives, and approach of testing. The main objectives were:

* + Ensuring the DR classification model provides accurate predictions.
  + Verifying seamless integration between different modules (image upload, classification, database, appointments, admin panel).
  + Checking system performance under different loads.
  + Identifying and fixing security vulnerabilities.
  + Ensuring user-friendliness through UI testing.

#### Test Case Development

Test cases were designed to cover all functionalities of the system. Each test case included:

* + **Test ID:** Unique identifier for tracking.
  + **Test Scenario:** Description of the feature being tested.
  + **Preconditions:** Any required setup before execution.
  + **Test Steps:** Actions to be performed.
  + **Expected Result:** Desired system response.
  + **Actual Result:** System’s actual behavior.
  + **Status:** Pass or Fail based on comparison with the expected result.

#### Test Execution

* + Unit tests were executed using PyTest, covering individual functions like image processing, database operations, and prediction logic.
  + Functional tests were carried out using Selenium to check UI interactions and navigation.
  + API tests with Postman ensured that data was correctly transmitted between the frontend and backend.
  + Performance tests using JMeter simulated multiple users booking appointments simultaneously.
  + Security tests were performed using OWASP ZAP to detect possible vulnerabilities.

#### Defect Reporting

* + Bugs and issues found during testing were recorded in a defect tracking system (e.g., JIRA, GitHub Issues).
  + Each defect report included:
    - **Defect ID:** Unique identifier.
    - **Module Affected:** Feature where the issue occurred.
    - **Description:** Brief explanation of the issue.
    - **Severity:** Critical, Major, Minor.
    - **Steps to Reproduce:** Instructions to replicate the bug.
    - **Assigned To:** Developer responsible for fixing it.
    - **Status:** Open, In Progress, Resolved, Closed.

Developers fixed the reported defects, and testers verified the fixes in subsequent test cycles.

#### Regression Testing

After bug fixes and feature updates, regression testing was performed to ensure that no existing functionality was broken.

* + Automated regression tests were run using Selenium for UI testing.
  + Test cases were re-executed to confirm that resolved issues did not introduce new bugs.
  + Performance and security tests were repeated to validate stability after modifications.

Testing played a crucial role in ensuring the reliability, accuracy, and efficiency of the DR classification system. Various testing methodologies, including unit testing**,** integration testing, system testing, and user acceptance testing**,** were implemented to validate the functionality of individual modules and their seamless interaction.

Unit testing helped verify the correctness of core functions, such as image processing, DR classification, and database operations. Integration testing ensured smooth communication between different components, such as uploading brain scans, generating predictions, and booking appointments. System testing evaluated the overall performance, security, and usability of the platform, while user acceptance testing confirmed that the system met patient and admin requirements effectively.

Additionally, rigorous test planning, test case development, and defect tracking enabled the identification and resolution of errors before deployment. Regression testing was performed to maintain system stability after modifications. By following a structured testing approach, the project achieved high accuracy in DR classification, seamless appointment management, and a user-friendly interface**.**

Overall, the testing process significantly contributed to enhancing the system’s dependability, ensuring an error-free user experience, and improving patient care through efficient DR detection and consultation management.

# ADVANTAGES AND DISADVANTAGES

### Advantages:

1. **Early Detection and Diagnosis**

The model helps in identifying diabetic retinopathy at an early stage, allowing timely treatment to prevent vision loss or blindness.

### High Accuracy and Reliability

Using deep learning models like AlexNet and DenseNet-169 enhances classification accuracy compared to traditional manual screening.

### User-Friendly Interface

Patients and doctors can easily upload scans, view results, and book appointments through a simple and intuitive interface.

### Integration with Digital Healthcare Systems

The model can be deployed as a web-based or mobile application, allowing remote diagnosis and improving healthcare accessibility.

### Disadvantages:

1. **Dependence on High-Quality Data**

The performance of the model depends on the availability of a large, diverse, and well- annotated dataset.

### Risk of Overfitting

If the model is trained on a limited dataset, it may not generalize well to new retinal images, leading to overfitting and poor real-world performance.

### Internet Dependency

Since the system operates online, it requires a stable internet connection, which may not be accessible in remote areas

### False Positives or Negatives.

The model may sometimes misclassify images, leading to unnecessary concern for patients.

# RESULTS AND CONCLUSION

### Results

The implementation of diabetic retinopathy detection and classification using deep convolutional neural networks and transfer learning with AlexNet and DenseNet-169 demonstrated significant improvements in automated diagnosis. The model's performance was evaluated using various metrics, including accuracy, precision, recall, F1-score, and the area under the receiver operating characteristic curve (AUC-ROC). Comparative analysis of both models revealed that DenseNet-169 outperformed AlexNet in terms of classification accuracy and robustness due to its feature reuse capability and deep architecture. The model effectively classified images into different categories of diabetic retinopathy, including no DR, mild, moderate, severe, and proliferative DR, with high confidence. Visualization techniques like Grad-CAM (Gradient-weighted Class Activation Mapping) were used to highlight important regions in retinal images, helping to interpret model predictions. The results indicate that transfer learning-based deep learning models provide a promising approach to improving the efficiency and accuracy of DR diagnosis compared to traditional manual screening methods.

### Conclusion

This project successfully demonstrated that deep learning and transfer learning approaches can be effectively utilized for automated diabetic retinopathy detection and classification. The use of AlexNet and DenseNet-169 architectures proved beneficial in accurately identifying different DR stages from retinal fundus images. Among the two models, DenseNet-169 achieved superior performance, making it a more suitable choice for real-world clinical applications. The proposed system has the potential to assist ophthalmologists in early diagnosis, thereby enabling timely treatment and reducing the risk of blindness in diabetic patients. Additionally, this research highlights the significance of leveraging deep learning techniques to enhance medical image analysis. The accuracy of the model can be further improved by expanding the dataset, applying data augmentation techniques, and incorporating ensemble learning approaches. Future work may focus on integrating this system into a web- based or mobile application for real-time screening and diagnosis, making it more accessible to healthcare professionals and patients worldwide.

1. **APPENDICES**

import numpy as np

import pandas as pd

import tensorflow as tf

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, BatchNormalization

from tensorflow.keras.preprocessing.image import ImageDataGenerator

from tensorflow.keras.applications import DenseNet169

from tensorflow.keras.optimizers import Adam

from sklearn.metrics import classification\_report, confusion\_matrix

import matplotlib.pyplot as plt

import seaborn as sns

import os

data\_dir = 'path\_to\_dataset' # Update with dataset path

img\_size = (224, 224)

batch\_size = 32

datagen = ImageDataGenerator(

rescale=1./255,

validation\_split=0.2,

horizontal\_flip=True,

rotation\_range=20,

zoom\_range=0.2,

shear\_range=0.2,

brightness\_range=[0.8, 1.2]

)

train\_data = datagen.flow\_from\_directory(

data\_dir,

target\_size=img\_size,

batch\_size=batch\_size,

class\_mode='categorical',

subset='training'

)

val\_data = datagen.flow\_from\_directory(

data\_dir,

target\_size=img\_size,

batch\_size=batch\_size,

class\_mode='categorical',

subset='validation'

)

# Define AlexNet-inspired CNN Model

def build\_alexnet():

model = Sequential([

Conv2D(64, (3, 3), activation='relu', input\_shape=(224, 224, 3)),

BatchNormalization(),

MaxPooling2D(2, 2),

Conv2D(128, (3, 3), activation='relu'),

BatchNormalization(),

MaxPooling2D(2, 2),

Conv2D(256, (3, 3), activation='relu'),

BatchNormalization(),

MaxPooling2D(2, 2),

Flatten(),

Dense(512, activation='relu'),

Dropout(0.5),

Dense(5, activation='softmax') # 5 categories for DR classification

])

return model

# Load Pretrained DenseNet-169

def build\_densenet():

base\_model = DenseNet169(weights='imagenet', include\_top=False, input\_shape=(224, 224, 3))

for layer in base\_model.layers:

layer.trainable = False

model = Sequential([

base\_model,

Flatten(),

Dense(512, activation='relu'),

Dropout(0.5),

Dense(5, activation='softmax')

])

return model

# Select Model

model\_choice = 'alexnet' # Change to 'densenet' to use DenseNet-169

model = build\_alexnet()

else:

model = build\_densenet()

# Compile the Model

model.compile(optimizer=Adam(learning\_rate=0.0001),loss='categorical\_crossentropy',

# Train the Model

epochs = 30 # Increased epochs

history = model.fit(train\_data, validation\_data=val\_data, epochs=epochs)

# Model Evaluation

val\_labels = val\_data.classes

y\_pred = np.argmax(model.predict(val\_data), axis=1)

print(classification\_report(val\_labels, y\_pred))

# Confusion Matrix

cm = confusion\_matrix(val\_labels, y\_pred)

sns.heatmap(cm, annot=True, cmap='Blues', fmt='d')

plt.xlabel('Predicted')

plt.ylabel('Actual')

plt.show()

# Save the Model

model.save('diabetic\_retinopathy\_model.h5')

# Load and Predict on New Image

def predict\_image(image\_path, model\_path='diabetic\_retinopathy\_model.h5'):

from tensorflow.keras.preprocessing import image

model = tf.keras.models.load\_model(model\_path)

img = image.load\_img(image\_path, target\_size=img\_size)

img\_array = image.img\_to\_array(img) / 255.0

img\_array = np.expand\_dims(img\_array, axis=0)

prediction = model.predict(img\_array)

class\_labels = list(train\_data.class\_indices.keys())

return class\_labels[np.argmax(prediction)]

# Example Prediction

Image\_path = 'path\_to\_test\_image.jpg' # Update with actual test image path

predicted\_class = predict\_image(image\_path)

print(f'Predicted Class: {predicted\_class}')

# Additional Evaluation - Accuracy and Loss Graph

def plot\_training\_history(history):

plt.figure(figsize=(12, 4))

plt.subplot(1, 2, 1)

plt.plot(history.history['accuracy'], label='Training Accuracy')

plt.plot(history.history['val\_accuracy'], label='Validation Accuracy')

plt.xlabel('Epochs')

plt.ylabel('Accuracy')

plt.legend()

plt.title('Accuracy Over Epochs')

plt.subplot(1, 2, 2)

plt.plot(history.history['loss'], label='Training Loss')

plt.plot(history.history['val\_loss'], label='Validation Loss')

plt.xlabel('Epochs')

plt.ylabel('Loss')

plt.legend()

plt.title('Loss Over Epochs')

plt.show()

plot\_training\_history(history)

# Fine-tune Model (Optional)

def fine\_tune\_model(model, train\_data, val\_data, epochs=15):

for layer in model.layers:

layer.trainable = True # Unfreeze all layers for fine-tuning

model.compile(optimizer=Adam(learning\_rate=1e-5), loss='categorical\_crossentropy', metrics=['accuracy'])

model.fit(train\_data, validation\_data=val\_data, epochs=epochs)

model.save('fine\_tuned\_diabetic\_retinopathy\_model.h5')

# Uncomment to fine-tune model

# fine\_tune\_model(model, train\_data, val\_data)

# Save Model Weights Separately

model.save\_weights('diabetic\_retinopathy\_weights.h5')

1. **REFERENCES**
2. Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., ... & Webster, D. R. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*, *316*(22), 2402-2410.
3. Ting, D. S. W., Cheung, C. Y., Lim, G., Tan, G. S. W., Quang, N. D., Gan, A., ... & Wong, T. Y. (2017). Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes. *JAMA*, *318*(22), 2211-2223.
4. Pratt, H., Coenen, F., Broadbent, D. M., Harding, S. P., & Zheng, Y. (2016). Convolutional neural networks for diabetic retinopathy. In *International Conference on Medical Image Understanding and Analysis* (pp. 94-103). Springer, Cham.
5. Leibig, C., Allken, V., Ayhan, M. S., Berens, P., & Wahl, S. (2017). Leveraging uncertainty information from deep neural networks for disease detection. *Scientific Reports*, *7*(1), 17816.
6. Abràmoff, M. D., Lou, Y., Erginay, A., Clarida, W., Amelon, R., Folk, J. C., & Niemeijer, M. (2016). Improved automated detection of diabetic retinopathy on a publicly available dataset through integration of deep learning. *Investigative Ophthalmology & Visual Science*, *57*(13), 5200-5206.
7. Rajalakshmi, R., Subashini, R., Anjana, R. M., & Mohan, V. (2018). Automated diabetic retinopathy detection in smartphone-based fundus photography using artificial intelligence. *Eye*, *32*(6), 1138-1144.
8. Voets, M., Møllersen, K., & Bongo, L. A. (2019). Reproduction study: Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *PeerJ*, *7*, e6211.
9. Quellec, G., Lamard, M., Conze, P. H., Cozic, M., Robert, P. Y., Danno, R., & Cazuguel, G. (2017). Deep image mining for diabetic retinopathy screening. *Medical Image Analysis*, *39*, 178-193.
10. Islam, M. T., Aowal, M. A., Minhaz, A. T., & Ashraf, K. (2018). Abnormality detection and localization in retinal images using deep convolutional neural networks. *Scientific Reports*, *8*, 8126.
11. Lam, C., Yu, C., Huang, L., & Rubin, D. (2018). Retinal lesion detection with deep learning using image patches. *Investigative Ophthalmology & Visual Science*, *59*(1), 590-596.