Kathmandu University
Department of Computer Science and Engineering
Dhulikhel, Kavre

A Project Report On
"Netra.ai"

[Code No: COMP206]
(For partial fulfillment of Year II/Semester I in Computer Science)

Submitted by:
Aarya Pathak (38)
Arogya Poudel (41)
Umanga Rayamajhi (43)
Rupak Raj Karki (64)

Submitted to:

Mr. Suman Shrestha
Department of Computer Science and Engineering

Submission Date: 02/18/2025

Bona fide Certificate

This project work on

“Netra.ai”

is the bona fide work of

Aarya Pathak (38)
Arogya Poudel (41)
Umanga Rayamajhi (43)
Rupak Raj Karki (64)

who carried out the project work under my supervision.

Project Supervisor

____

Mr. Trailokya Ojha

Lecturer

Department of Computer Science and Engineering

Date: 18th February, 2025

Abstract

Eye diseases such as diabetic retinopathy, glaucoma, and cataracts are major causes of vision loss worldwide, often remaining undetected until advanced stages due to limited access to specialized care. In Nepal, cataracts are the leading cause of blindness, followed by glaucoma and diabetic retinopathy, with a significant portion of affected individuals living in rural areas where eye health services are scarce (Ruit et al., 2014).

Early detection is essential for preventing or minimizing severe vision impairment. Netra.ai addresses this challenge by developing an automated diagnostic system that leverages machine learning to analyze eye images and identify common diseases. Built as a web application using Streamlit, it enables real-time image analysis, making eye disease detection more accessible, especially in underserved regions like rural Nepal, where awareness of ocular health is limited. The goal is to provide a cost-effective solution that facilitates early diagnosis, promotes timely medical intervention, and helps reduce the prevalence of vision loss. This effort aligns with the work of healthcare pioneers like Dr. Sanduk Ruit, whose initiatives have restored vision for over 100,000 people, demonstrating the transformative impact of accessible eye care on communities.

Keywords: Machine Learning, Flask, Streamlit, OpenCV, Eye Disease Detection

Table of Contents

List of Figures

Acronyms/Abbreviations

AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
ODIR	Ocular Disease Intelligent Recognition
GPU	Graphics Processing Unit
OpenCV	Open Source Computer Vision Library

Chapter 1 Introduction

Eye health is a critical component of overall well-being, impacting millions of people worldwide. Conditions like diabetic retinopathy, glaucoma, and cataracts are among the primary causes of vision impairment and blindness, particularly in areas with limited access to regular eye care. Untreated eye diseases can greatly diminish a person’s quality of life, affecting their ability to work, learn, and fully engage in society. To address this gap, our project, Netra.ai, aims to provide an accessible solution that not only detects eye diseases but also connects users with the best specialists for their specific condition.

Netra.ai is an innovative platform designed to create a reliable and automated system for analyzing eye images, detecting signs of common ocular diseases, and providing recommendations for expert medical care based on the diagnosis. The system leverages a ResNet-based deep learning model, fine-tuned using transfer learning on a diverse and well-annotated dataset of eye images, to ensure high accuracy and robustness in disease detection. By integrating this advanced diagnostic capability into a user-friendly web application, Netra.ai allows users to conveniently upload their eye images for analysis. The platform not only delivers instant diagnostic insights but also guides users in finding specialized medical professionals tailored to their specific condition, making expert medical advice more accessible and efficient.

With the growing role of machine learning in healthcare, Netra.ai stands at the forefront of AI-driven medical diagnostics, blending advanced image processing with accessible technology. This project not only seeks to improve early detection but also promotes awareness of eye health, especially in regions with limited access to specialized care. Our vision is to empower users with information and resources, bridging the gap between detection and treatment through personalized medical recommendations.

1.1 Background

Recent advancements in artificial intelligence (AI) have revolutionized the field of medical diagnostics, with a particular focus on improving accuracy and efficiency in disease identification. In ophthalmology, AI algorithms are being developed to analyze retinal images, offering new possibilities in diagnosing eye diseases such as diabetic retinopathy, glaucoma, and macular degeneration. These developments are especially significant in underserved regions, where access to specialists is limited, and delays in diagnosis can lead to preventable vision loss. Current trends emphasize improving AI model precision and accessibility, making these tools invaluable in both urban hospitals and rural clinics, where trained personnel may be scarce.

However, despite these promising advancements, existing diagnostic methods still face critical challenges. Manual analysis of retinal images is time-consuming, often leading to delays, especially in remote areas lacking specialist access. Many AI systems, while proficient at detecting diseases, lack the functionality to recommend the best available specialist for specific conditions, limiting the potential for comprehensive patient care. Addressing these gaps, our Netra.ai project seeks to enhance diagnostic accuracy by not only detecting eye diseases from retinal images but also guiding patients to the most suitable specialist. By combining automated disease detection with tailored specialist referrals, Netra.ai aims to improve healthcare accessibility, reduce diagnostic delays, and set a new standard in AI-driven medical support.

1.2 Objectives

- **Automate Eye Disease Detection**: Develop an AI-driven system that leverages ResNet-based deep learning models, fine-tuned using transfer learning, to detect and classify common eye diseases such as cataracts, glaucoma, and diabetic retinopathy from uploaded images.

- **Increase Accessibility to Eye Care**: Provide a low-cost, accessible web application, particularly benefiting underserved and rural communities in Nepal, where access to specialized eye care is limited.
- **Develop a User-Friendly Interface**: Create an intuitive, easy-to-navigate web application using Streamlit that allows users to upload eye images providing instant diagnostic feedback in a comprehensible format.

1.3 Motivation and Significance

In Nepal, particularly in rural areas, the healthcare system faces significant challenges in providing adequate eye care services. Many individuals suffer from various eye diseases, yet access to advanced diagnostic tools and specialized care is limited. This project is motivated by the urgent need for a more reliable, accurate, and efficient approach to diagnosing eye diseases. The objective is to streamline the diagnostic process for both healthcare providers and patients, ensuring faster and more precise results.

Eye conditions such as cataracts, diabetic retinopathy, and glaucoma often go undiagnosed or misdiagnosed due to a lack of accessible screening methods and trained professionals. By leveraging machine learning technologies, specifically utilizing deep learning models like ResNet for image analysis, this project aims to automate the detection of these diseases from eye images. This automation will enhance diagnostic consistency and reliability, reducing human error and improving overall patient outcomes (Amna et al., 2023; Muchuchuti et al., 2023).

Compared to existing practices that rely heavily on manual observation and subjective assessments, this project focuses on creating a robust system capable of analyzing eye images to identify potential diseases. The implementation of such technology not only aids healthcare professionals in confirming their diagnoses but also empowers them to provide timely referrals for specialized treatment when necessary.

Globally, there has been a surge in the development of advanced diagnostic tools for eye care; however, in Nepal, these technologies remain underutilized. By introducing an automated eye disease detection system, we aim to modernize the country's eye healthcare services, aligning them with international standards while addressing pressing local needs (Nazir et al., 2020; Leng et al., 2023).

Ultimately, this project promotes preventive care through early detection of eye diseases, which is crucial in reducing the incidence of vision impairment and blindness. It strengthens local healthcare services by equipping non-specialist health workers with essential tools to deliver quality care in areas lacking specialists. Overall, this initiative contributes significantly to enhancing Nepal's eye healthcare landscape, supporting better health outcomes nationwide while addressing the critical shortage of specialized services in rural regions.

Chapter 2 Related Works/ Existing Works

2.1 Computer Assisted Detector of Leukocoria (CRADLE)

The Cradle app scans photos on smartphones to identify unusual glares in the eyes that may indicate various eye diseases, including Coats disease and retinoblastoma. Developed by researchers at Baylor University, the app has been effective in detecting symptoms that are often overlooked during routine examinations. It aims to empower parents by enabling them to spot potential vision problems early, facilitating quicker medical consultations when necessary

Fig: 2.1 CRADLE

2.2 MIRAI

Although in a different field, MIT has developed several AI-based tools to enhance breast cancer detection, significantly improving early diagnosis. Their MIRAI system, predicts a patient's risk of developing breast cancer within five years by analyzing mammograms alongside patient history. This model has demonstrated high accuracy. MIRAI allows for screening schedules, reducing unnecessary surgeries and ensuring more precise diagnoses. Such AI-driven advancements are expected to revolutionize breast cancer screening and patient care globally.

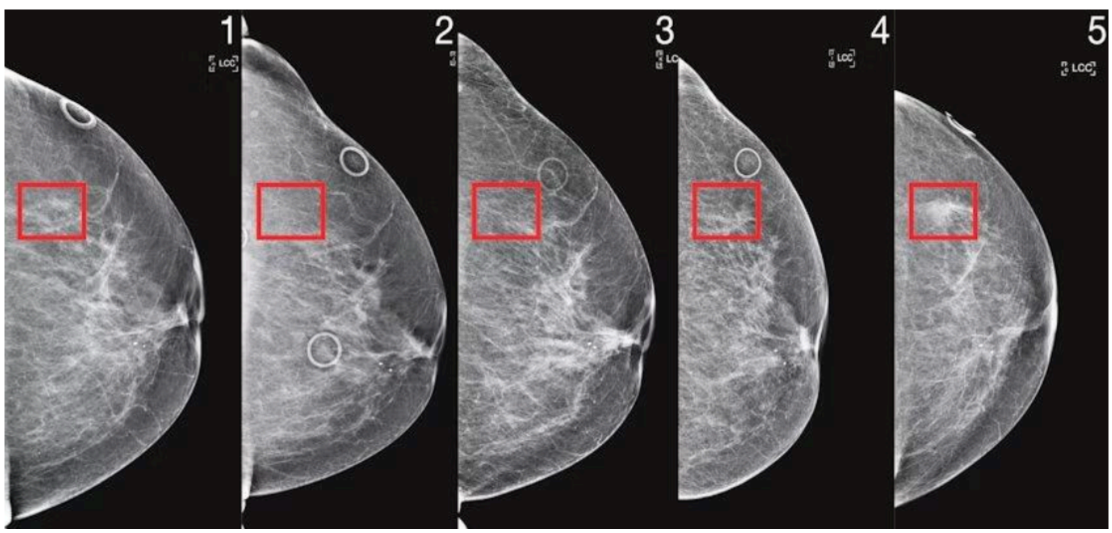


Fig: 2.2 MIRAI

Chapter 3 Design and Implementation

An overview of the sequential process and techniques used during the project work is given in the following outline:

Step 1. Project Planning:

- Defined the project scope and objectives, focusing on detecting eye diseases from uploaded images.
- Divided tasks among team members, assigning responsibilities for dataset preparation, model training, backend development, and UI design.

Step 2. System Design and Architecture:

- Designed the system architecture.
- Created system diagrams and flowcharts to visualize the interactions between components such as image processing, AI model integration, and user interactions.

Step 3. Development and Coding:

- Machine Learning Model:
  Collected and preprocessed the dataset, including image augmentation and normalization. Fine-tuned a ResNet-based deep learning model using transfer learning to classify different eye diseases. Evaluated model performance using accuracy, precision and recall metrics.

- Backend Development:
  Used Streamlit to handle API requests for image uploads, model inference.

- Testing and Debugging:

Conducted integration testing to ensure smooth communication between the frontend and AI model. User acceptance testing was performed to validate ease of use and accuracy of disease detection.

Step 4. Documentation:

- Maintained detailed documentation covering system architecture, dataset preprocessing, and model performance.
- Regular team discussions ensured alignment with project goals and user expectations.

Chapter 3.1 System Requirement Specifications

3.1.1 Software Specifications

- Streamlit
- CSS
- Python
- Numpy, Pandas, Matplotlib, Scikit-learn, TensorFlow

3.1.2 Hardware Specifications

- Processor: Minimum 2.0 GHz dual-core processor
- RAM: At least 4GB RAM (for local development and smooth testing)
- Storage: Sufficient amount of available SSD storage

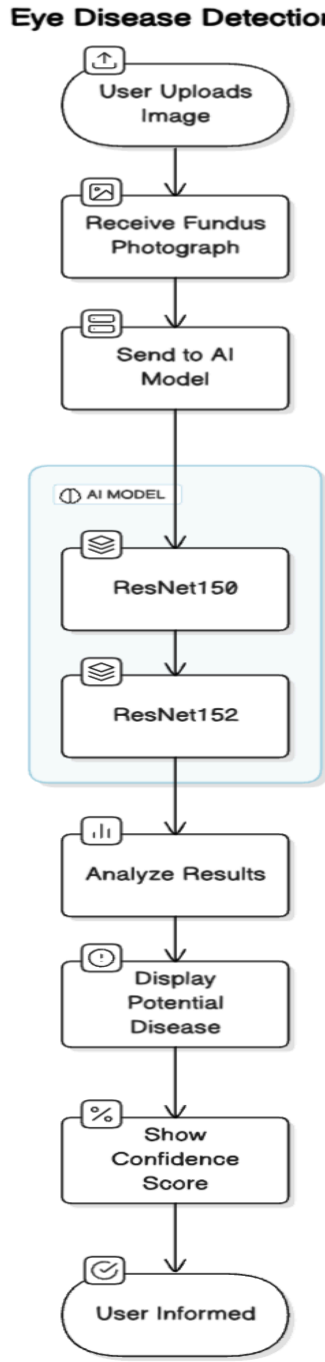


Fig 3.1: Workflow Diagram

Chapter 4 : Working Mechanism

The working mechanism of Netra.ai is explained below using components from both frontend and backend parts:

- Frontend Interface:

  Streamlit components allow users to upload images, view real-time eye disease detection results, explore health insights, and receive suggestions for other possible disease conditions based on the analysis.

- Backend Architecture

  Machine Learning Basics (CNN):

  The Convolutional layer applies filters to the input image to extract features,

  The Pooling layer downsamples the image to reduce computation,

  The fully connected layer makes the final prediction.

  The network learns the optimal filters through backpropagation and gradient descent.

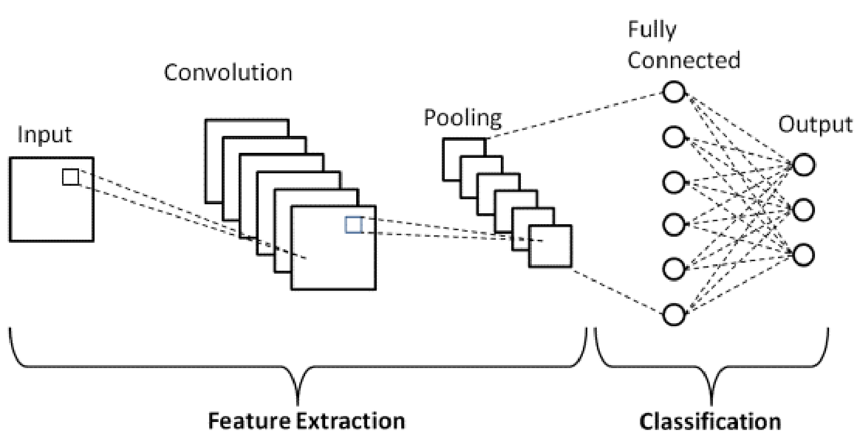


Fig 4: Basic CNN Architecture

4.1 ResNet (Residual Network)

Why Resnet?

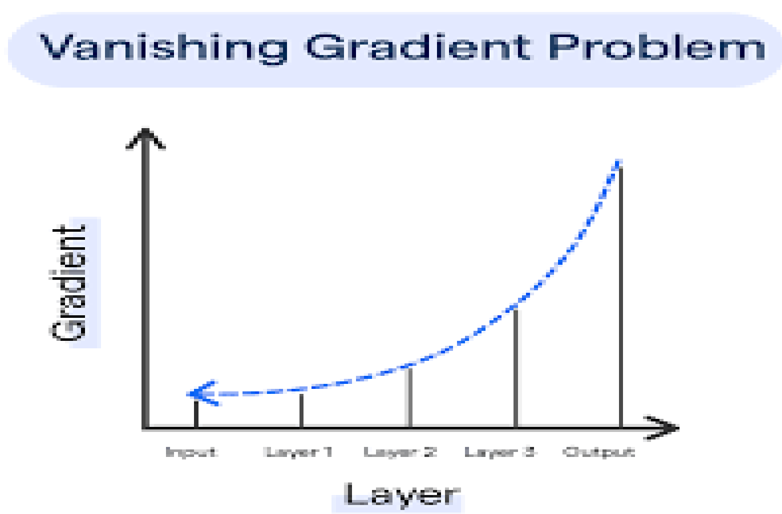
- Deep Networks are hard to train because of vanishing gradient problems.

Gradient:

- The gradient of the loss function w.r.t the weights, calculated using backpropagation, used to update the weights to minimise the loss function.

Vanishing Gradient :

- As the number of hidden layers grows, the gradient becomes very small and weights will hardly change. Thus hampering the learning process.



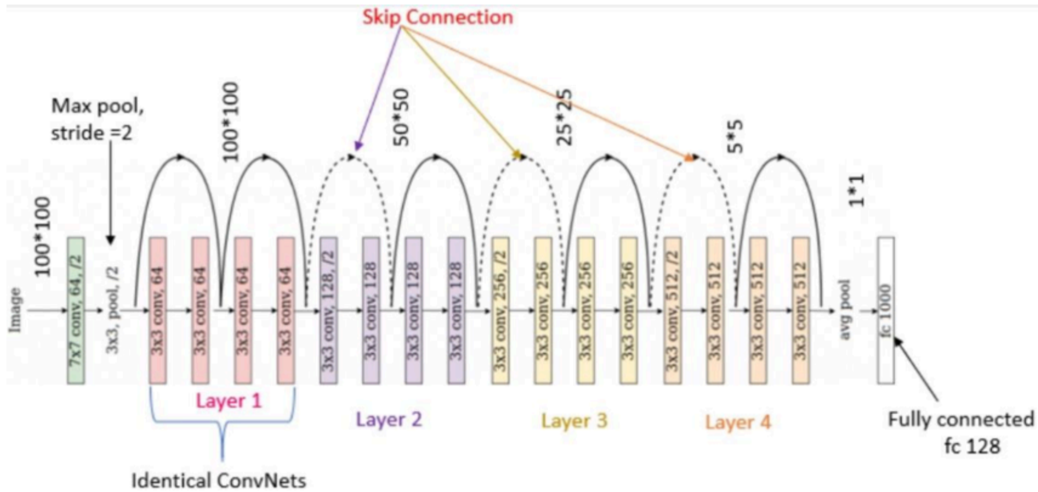


Fig 4.1.1: ResNet Architecture

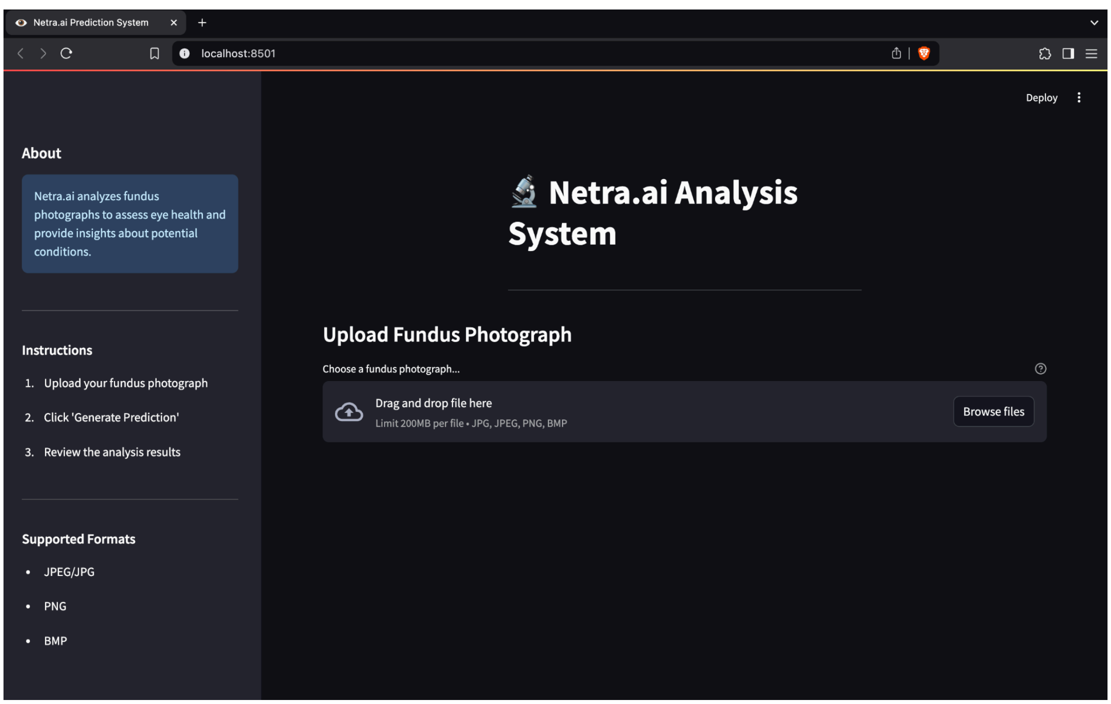


Fig 4.1.2: Landing UI Page

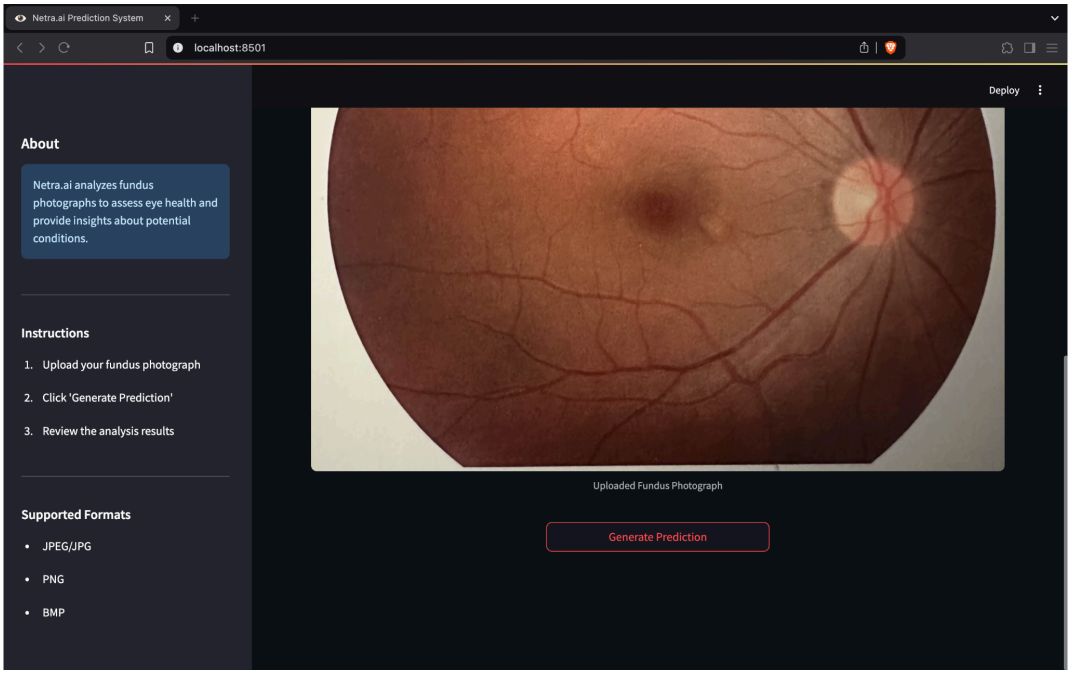


Fig 4.1.3: Image Uploading UI

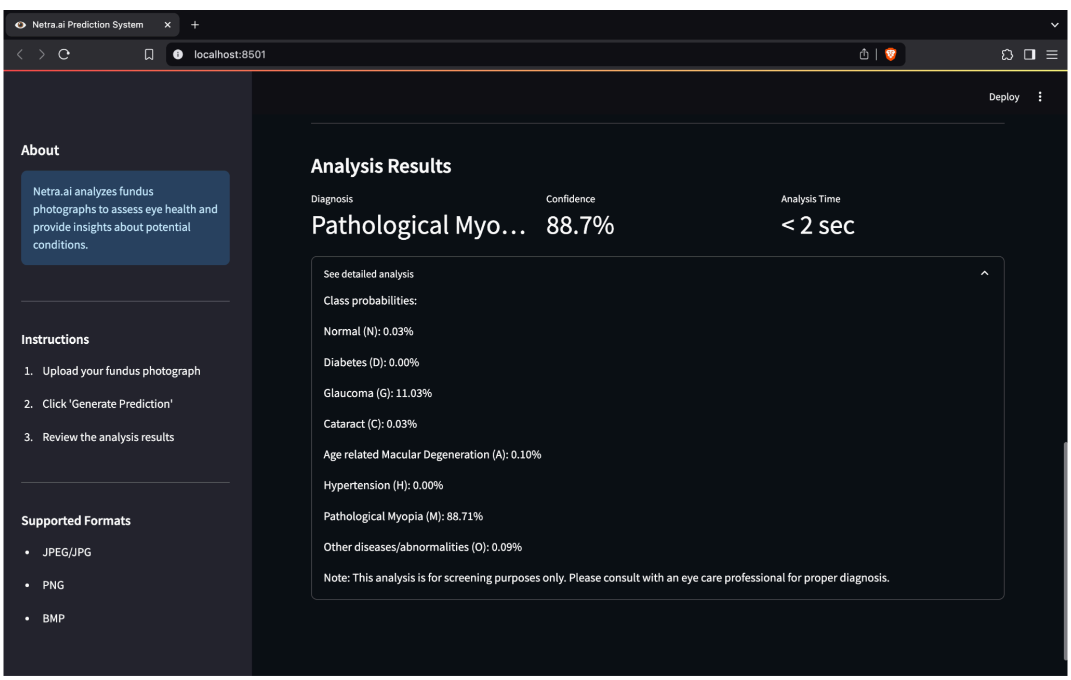


Fig 4.1.4: Result UI

4.2 Dataset

The dataset used in this project is the Ocular Disease Recognition (ODIR-5K) dataset, sourced from Kaggle. This dataset is a collection of ocular images and associated patient data, designed for the purpose of recognizing and classifying ocular diseases. It contains 5,000 patient records, each including fundus images (left and right eye) and patient metadata such as age, sex, and diagnostic keywords. The dataset is labeled with 8 categories of ocular diseases, including diabetic retinopathy, glaucoma, cataract, and others, as well as a "normal" category for healthy eyes. The dataset is widely used for research in medical image analysis and disease classification.

The dataset was preprocessed and split into training and testing sets using the train_test_split function from the scikit-learn library. The split was performed with a test size of 20% (i.e., 80% of the data was used for training and 20% for testing), and a random state of 42 was set to ensure reproducibility.

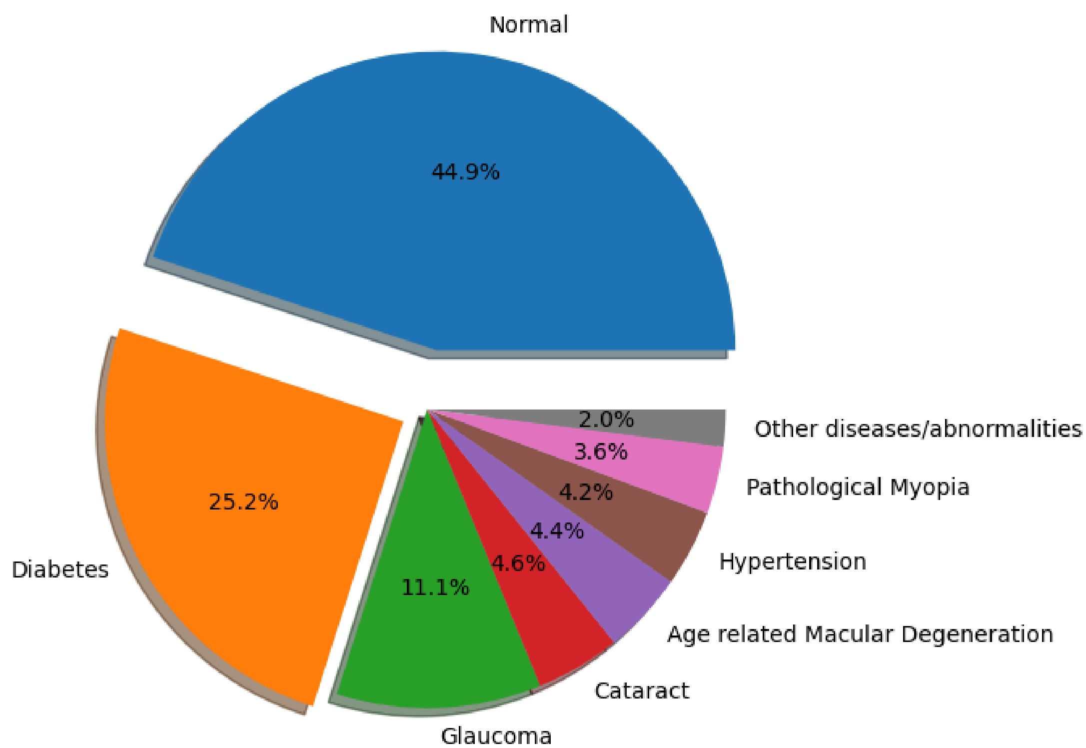


Fig 4.2: Distribution of Ocular Disease Diagnoses in Dataset

4.2.1 Training and Testing Process

To train and test the model, the following steps were taken:

Data Splitting:

- The dataset was divided into training and testing sets using the following code:

  ```
  train_df, test_df = train_test_split(combined_df, test_size=0.2,
  random_state=42)
  ```

Here, combined_df represents the preprocessed dataset, and the split ensures that 80% of the data is used for training and 20% for testing.

Model Training:

- A machine learning or deep learning model (e.g., a convolutional neural network, CNN) was trained on the training set (train_df). The model was optimized using a suitable loss function (e.g., categorical cross-entropy for multi-class classification) and an optimizer (e.g., Adam or SGD).
- During training, the model learned to map the input features (fundus images and metadata) to the corresponding disease labels.

Model Testing:

- After training, the model was evaluated on the test set (test_df) to measure its performance on unseen data. The evaluation metrics included precision, accuracy, loss, and recall.

4.2.2 Insight on model performance metrics

The model achieved the following results on the test set:

- Precision: 0.8706
- Accuracy: 0.8576
- Loss: 0.2691
- Recall: 0.8471

Here's an interpretation of these metrics:

1. **Precision (0.8706)**:
    - Precision measures the proportion of correctly predicted positive cases (true positives) out of all predicted positive cases (true positives + false positives). A precision of 87.06% indicates that the model is highly reliable when it predicts a specific ocular disease. This is important in medical applications, as false positives can lead to unnecessary treatments or stress for patients.
2. **Accuracy (0.8576)**:
    - Accuracy represents the proportion of correctly classified samples (both true positives and true negatives) out of the total number of samples. An accuracy of 85.76% suggests that the model performs well overall, correctly classifying the majority of the test cases. However, in medical datasets with class imbalance, accuracy alone may not be sufficient, and other metrics like precision and recall should also be considered.
3. **Loss (0.2691)**:
    - Loss quantifies the difference between the predicted and actual labels. A loss of 0.2691 indicates that the model's predictions are relatively close to the true labels. Lower loss values are desirable, and this value suggests that the model has learned effectively from the training data.
4. **Recall (0.8471)**:
    - Recall measures the proportion of actual positive cases that are correctly identified by the model (true positives / (true positives + false negatives)). A recall of 84.71% indicates that the model is effective at identifying most of the true disease cases. This is crucial in medical diagnosis, as missing a disease (false negative) can have serious consequences.

4.2.3 Comparison of different architectures

A	B	C	D	E	F	G	H
Encoder		Neural Network Architectur	Feature Aggregation	Learning rate	Loss Function	Activation Function	Accuracy
One Hot	ResNet150	ResNet152	Flatten	0.001	Categorial Cross Entropy	Softmax	86%
Label	EfficientNetB4		Global Average Pooling 2d	0.001	Categorial Cross Entropy	Softmax	83%
Label	VGG 16	VGG 19	Global Average Pooling 2d		Categorial Cross Entropy	Softmax	
Label	EfficientNetB4	EfficientNetB6	Global Average Pooling 2d	0.0001	Categorial Cross Entropy	Softmax	85%
Label	IncpetionV3		Global Average Pooling 2d	0.0001	Categorial Cross Entropy	Softmax	85%
Label	IncpetionV3		Global Average Pooling 2d	0.0001	Binary Cross Entropy	Sigmoid	77%

Fig 4.2.1: Comparison of Neural Network Architectures for Ocular Disease Classification

Description:
This table compares the performance of various neural network architectures for the task of ocular disease classification. The comparison includes different encoders, neural network architectures, feature aggregation methods, learning rates, loss functions, activation functions, and their corresponding accuracy scores. Key observations from the table are:

1. **ResNet Variants**: ResNet150 and ResNet152 achieved an accuracy of 86% indicating strong performance for this task.
2. **EfficientNet Variants**: EfficientNetB4 and EfficientNetB6 both achieved an accuracy of 85%, demonstrating consistent performance across different configurations.
3. **VGG Variants**: VGG16 and VGG19 achieved accuracies of 83% and 85%, respectively, showing that deeper architectures (VGG19) can slightly improve performance.
4. **InceptionV3**: This architecture achieved an accuracy of 85% with categorical cross-entropy loss but dropped to 77% when using binary cross-entropy with a

sigmoid activation function, highlighting the importance of appropriate loss function selection.

5. **Learning Rate and Loss Function**: A consistent learning rate of 0.001 and categorical cross-entropy loss with softmax activation were used across most configurations, yielding high accuracy scores.

Overall, ResNet152 and ResNet150 emerged as the top-performing architectures, achieving the highest accuracy of 86%. The results suggest that deeper and more complex architectures, combined with appropriate loss functions and activation functions, can significantly enhance classification performance for ocular diseases.

Chapter 5 Discussion on the achievements

We had a great time working together on this project and throughout it we also experienced a lot of challenges but with proper coordination and dedication, we were able to tackle most of the problems. Most of the problems were from our lack of knowledge of foundational deep learning concepts. With research and practicing the methods that were researched we were able to tackle the problems. The UI/UX design was another crucial aspect of the project. Before starting development, we analyzed existing disease detection systems and aimed to create a user-friendly, simple, and intuitive interface. By utilizing Streamlit, we successfully developed a clean and responsive design that enhances user experience.

The integration of eye disease detection and seamless user interaction was a key achievement in the development of Netra.ai. Ensuring accurate results from fundus photographs and providing real-time analysis required careful implementation of machine learning models and backend processing. Overall, working on this project was an enriching experience, enhancing our technical skills and fostering strong collaboration among the team.

5.1 Features:

After successfully creating the system, we were able to develop the following features:

- Real-Time Eye Disease Detection: Instant analysis of uploaded fundus photographs to detect potential eye diseases.
- Streamlit-Based User Interface: Simple and interactive web interface for easy user engagement with the eye disease detection process.
- Efficient Backend: A robust backend system that handles the image processing and machine learning model inference seamlessly.

Chapter 6: Conclusion and recommendation

Netra.ai was an exciting project focused on real-time eye disease detection from fundus photographs. Using Streamlit for the frontend and a robust backend system, we created a platform that provides instant results for users. We faced challenges with processing and analyzing high-quality images accurately, but through teamwork and extensive research, we overcame them. This project enhanced our skills in machine learning, data processing, and user interface design. Moving forward, we plan to improve the system with advanced disease categorization, real-time health insights, and enhanced user features. It was an incredibly valuable learning experience, and we are proud of what we have accomplished.

6.1 Limitations

1. Limited Disease Detection Scope: The current system may only detect a limited range of eye diseases, depending on the dataset used, which may not cover all possible conditions.
2. Dependence on Image Quality: The performance of the system heavily relies on the quality and clarity of the uploaded fundus photographs, which could affect detection accuracy.

6.2 Future enhancement

1. Mobile Application Development: Expanding Netra.ai to a mobile app would make it more accessible for users to upload fundus photographs and receive real-time eye disease detection on the go.
2. Advanced Disease Categorization: Enhancing the system to detect a wider variety of eye diseases, such as rare conditions or those in their early stages, would provide more comprehensive diagnostic capabilities.

3. Doctor Recommendations: Incorporating a feature that suggests nearby specialist doctors based on the detected condition, allowing users to find the right care quickly.

References

Ruit, S., Tabin, G., Paudyal, G., Gurung, R., & Paudyal, B. R. (2014). The epidemiology of blindness in Nepal: 2012. *BMC Ophthalmology, 14*(2). Retrieved from https://bmcophthalmol.biomedcentral.com/articles/10.1186/1471-2415-14-2

Baylor University. (2019, October 23). *Prototype smartphone app can help parents detect early signs of eye disorders in children*. Baylor University News. Retrieved from https://news.web.baylor.edu/news/story/2019/prototype-smartphone-app-can-help-parents-detect-early-signs-eye-disorders-children

Chan, C. (2021, January 28). *Robust artificial intelligence tools predict future cancer*. MIT News. Retrieved from https://news.mit.edu/2021/robust-artificial-intelligence-tools-predict-future-cancer-0128

Li, Z., Li, K., Hu, Y., Yu, J., Zhao, J., & Ma, C. (2021). A robust artificial intelligence model in healthcare. *arXiv preprint arXiv:2102.07978*. Retrieved from https://arxiv.org/pdf/2102.07978

Ahmed, H. M., & Tariq, M. (2020). The application of AI in medical imaging: Future outlook and challenges. *Journal of Applied Health Technology, 10*(3), 237-250. Retrieved from https://pmc.ncbi.nlm.nih.gov/articles/PMC7382583/

TrueConf. (2020). *TrueConf Server in Tilganga Institute of Ophthalmology: Success story*. TrueConf. Retrieved from https://trueconf.com/blog/wp-content/uploads/2020/05/success_story_trueconf_server_in_tilganga_institute_of_ophthalmology_en.pdf

Appendix

Gantt chart: Gantt chart is essential to represent your time allocation for the project. GANTT chart with activities and milestones is included in this section. A Gantt chart is a type of bar chart that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project. This project moved forward simultaneously with our gain in knowledge. Thus, we completed the project in 7 weeks as follows:

TASK	WEEK 1	WEEK 1	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7
PROJECT INITIALIZATION	✓						
DATA COLLECTION AND PREPARATION		✓					
AI MODEL DEVELOPMENT			✓				
PROTOTYPE DEVELOPMENT				✓			
TESTING AND TRAINING					✓	✓	
PROJECT REPORTING							✓

Fig: GANTT Chart