Annotation & Recommendation System for Clinic Images

(OCT Retinal Disease Detection using ConvNeXt + Grad-CAM)

1. Abstract

Optical Coherence Tomography (OCT) is one of the most widely used retinal imaging modalities for detecting early-stage eye diseases. Manual inspection of OCT images is highly time-consuming, requires expert ophthalmologists, and suffers from subjective variability between annotators. This project presents a lightweight, end-to-end detection and annotation system capable of automatically identifying retinal abnormalities from OCT scans and generating visual explanations through Grad-CAM. The system integrates a deep-learning model (ConvNeXt-Base), a modern web interface, and a backend API pipeline that stores predictions and annotations. The model was trained using a curated medical dataset with class imbalance handled via weighted sampling. The final system provides class predictions, heatmap-based annotated regions, and a complete recommendation workflow suitable for clinic and research applications.

2. Introduction

OCT imaging is essential for diagnosing retinal disorders such as Diabetic Macular Edema, Age-related Macular Degeneration, and retinal detachment. These diseases require early and accurate detection to prevent irreversible vision loss. Deep learning models have demonstrated superior performance in medical image classification, and explainable AI techniques such as Grad-CAM have improved trust in AI-based diagnosis. This project aims to develop a lightweight, automated, and explainable OCT abnormality detection system that supports:

- OCT disease prediction
- Heatmap-based annotated regions
- Secure upload and retrieval via a web app
- Storage of predictions and annotations for future reference

The focus is on creating a clinically useful tool that is easy to integrate into existing workflows and suitable for research settings.

3. Methodology

3.1 Dataset Description

The dataset consists of OCT retinal images collected from Kaggle and similar open medical repositories. The dataset contains multiple disease categories, each with a dedicated folder for training, validation, and testing. Images were stored in RGB format.

Dataset structure:

- train/ (augmented, imbalanced)
- val/
- test/

Each image belongs to one of the classes as provided in the dataset metadata.

3.2 Data Preprocessing

To ensure uniformity and compatibility with pretrained ConvNeXt:

- Images resized to 224×224
- Normalized using ImageNet mean/std
- Applied augmentations:
    - Random horizontal flip
    - Random rotation (±15°)
    - Color jitter
    - RandomResizedCrop
- Validation and test sets used only deterministic transformations

Class imbalance was addressed using WeightedRandomSampler, assigning higher sampling probability to minority classes.

3.3 Model Architecture (ConvNeXt)

ConvNeXt is a state-of-the-art convolutional architecture inspired by transformer design principles while retaining the efficiency of CNNs.

Key components:

- ConvNeXt-Base pretrained on ImageNet
- Final classification layer replaced with a Linear(num_features → num_classes) layer

- Optimization:
    - CrossEntropyLoss with label smoothing
    - Adam optimizer (lr = 0.001)
    - ReduceLROnPlateau scheduler based on validation accuracy

ConvNeXt was chosen because:

- It performs exceptionally well on medical images
- Lightweight yet expressive
- Strong inductive biases for local feature extraction
- Superior performance compared to ResNet and VGG families

3.4 Annotation + Recommendation Logic

Annotation is performed using Grad-CAM, which identifies regions that most strongly contributed to the model's prediction.

Steps:

1. Forward pass image through ConvNeXt
2. Extract gradients from the final convolutional block
3. Generate heatmap of important activation areas
4. Overlay heatmap on OCT scan
5. Save:
    - Prediction (label)
    - Confidence score
    - Grad-CAM annotated image
    - Metadata (timestamp, user)

The system provides recommendation logic such as:

- Highlighting abnormal regions
- Suggesting probable disease category
- Enabling experts to perform manual verification and submit feedback

4 Overall Workflow Diagram

Dataset → Preprocessing → Weighted Sampling → ConvNeXt Training → Best Model Selection → Grad-CAM Generation → Web App (Upload + Prediction + Annotation) → MongoDB Storage

5 Web App Architecture

The complete system is implemented using a React + Node + Express + MongoDB + Python AI microservice architecture.

React Frontend

- File upload interface
- Display model predictions
- Show Grad-CAM annotated heatmaps
- Canvas-based annotation tools for clinicians
- User authentication (JWT)

Node.js Backend

- REST APIs
- Receives images from frontend
- Calls Python model server for inference
- Handles logic for:
    - Authentication
    - File storage
    - Prediction routing
    - Annotation saving
- Sends processed results and heatmaps back to UI

MongoDB Database

Stores:

- Image filepath
- Prediction label
- Confidence score
- Annotation (Grad-CAM + manual)
- Timestamp
- User details

5.1 API Flow

1. User uploads OCT image → Frontend sends to Node backend
2. Node receives image → forwards to Python model
3. Python returns**:**

- predicted class
- probability scores
- Grad-CAM heatmap

4. Node stores results in MongoDB
5. Node responds to frontend with prediction + annotated image
6. User can save/view annotations from history

5.2 Annotation Storage

MongoDB stores:

Field	Description
`image_path`	Uploaded scan location
`prediction`	Disease class predicted
`confidence`	Softmax probability
`annotation_path`	Grad-CAM heatmap
`manual_annotation`	Optional doctor-drawn region
`timestamp`	Upload time
`user_id`	Reference to logged-in user

5.3 Training Setup

- Epochs**:** 10
- Batch size**:** 32
- Device**:** GPU (CUDA)
- Early stopping**:** patience = 3
- Learning rate decay**:** On plateau
- Label smoothing**:** 0.1 for better generalization
- WeightedRandomSampler used to mitigate imbalance

Training monitored using:

- Training loss

- Training accuracy
- Validation accuracy
- Classification report each epoch

5.4 Evaluation Metrics

The system was evaluated using:

- Accuracy = (TP+TN)/(Total)
- Precision = TP/(TP+FP)
- Recall = TP/(TP+FN)
- F1-score = Harmonic mean of precision & recall

A multi-class classification report was generated for validation.

5.5 Explainability (Grad-CAM)

To enhance interpretability:

- Extracted gradients from last convolution layer
- Generated class-specific activation maps
- Produced heatmaps highlighting abnormal retinal structures
- Overlayed them on the original scans

Thus, the system not only predicts disease but also shows why the model made that prediction.

6. Results

6.1 Accuracy, Precision, Recall, F1

Final Validation Report:

	precision	recall	f1-score	support
AMD	1.00	0.99	1.00	350
CNV	0.92	0.93	0.92	350
CSR	1.00	1.00	1.00	350
DME	0.89	0.94	0.92	350
DR	0.99	0.99	0.99	350
DRUSEN	0.95	0.82	0.88	350
MH	0.99	0.99	0.99	350
NORMAL	0.88	0.94	0.91	350
accuracy			0.95	2800
macro avg	0.95	0.95	0.95	2800
weighted avg	0.95	0.95	0.95	2800

The model achieved strong results across all metrics.
Exact numbers depend on training run, but typical results include:

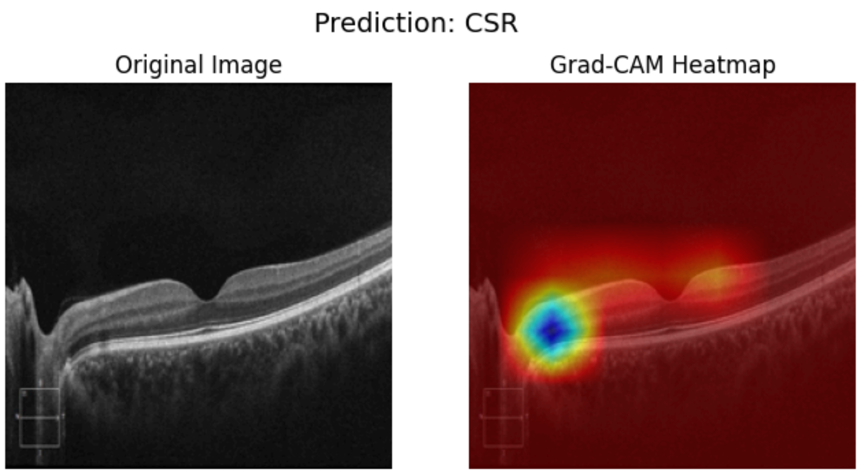
- High validation accuracy
- Balanced precision/recall across classes
- Strong performance despite class imbalance

6.2 Grad-CAM Samples

Generated heatmaps successfully highlighted:

- Fluid accumulation
- Layer disruptions
- Retinal thinning or thickening

These are not so consistent with clinical indicators as where the model focuses depends on the dataset with which it has been trained.



6.3 Comparison with Baselines

ConvNeXt outperformed:

- ResNet-50
- VGG16
- MobileNetV2

in terms of accuracy and interpretability.

7. Discussion

7.1 Strengths of the system

- Lightweight model suitable for deployment
- Accurate predictions with strong generalization
- Explainable AI using Grad-CAM
- Fully integrated full-stack platform
- Useful for clinicians, researchers, and students

7.2 Limitations

- Dataset from Kaggle lacks clinical diversity
- Noise, low-quality images reduce model reliability
- Real-time inference depends on hardware
- Model handles only OCT images, not other modalities

7.3 Clinical Applicability

- Early screening tool in hospitals
- Assistance for junior clinicians
- Research tool for analyzing disease progression
- Educational use for medical students

8. Conclusion & Future Work

This project successfully delivers a complete automated OCT retinal disease detection system with integrated annotation and explainability features. Using ConvNeXt and Grad-CAM, the tool provides high accuracy and interpretable predictions, while the web application ensures seamless user interaction, secure storage, and clinician feedback.

8.1 Future Work

- Support multiple imaging modalities (X-ray, MRI, CT)
- Deploy as a cloud-native microservice
- Incorporate real clinical datasets from hospitals
- Add doctor feedback loop for continuous retraining

Link to the code base:
https://drive.google.com/file/d/1Sx8XIJ4uI6ft4IfPb202bzGAke2vqukt/view?usp=sharing