Project Report — Diabetic Retinopathy Detection using Deep Learning

Hackathon: GDG on Campus PIEAS AI/ML Boot camp & Hackathon GDG Community
GitHub Repository: *ZayanRashid295/GDGOC-PIEAS-AI-ML-HACKATHON*
Author: (Muhammad Jawad Ahsan)
Date: (15-12-2025)

1. Introduction

Diabetic Retinopathy (DR) is a severe eye disease caused by complications of diabetes that can lead to vision impairment and blindness if not detected early. It is characterized by damage to the retinal blood vessels and is traditionally diagnosed by specialists analyzing retinal fundus images. Manual diagnosis is time-consuming and requires medical expertise, making automatic detection through artificial intelligence a valuable tool in healthcare.

1.1 Current Working

At this time our model is not completely trained because the model is developed from scratch. So the model is not completed trained on all the pictures. We trained images and make a model and predict on this.

2. Problem Statement

The goal of this project is to build a deep learning model that classifies retinal images into five classes of diabetic retinopathy severity (0–4) using a publicly available balanced fundus image dataset. The model should:

- Learn discriminative retinal patterns.
- Predict severity level accurately.
- Be trained quickly within hackathon constraints.

3. Dataset

We used the **Diabetic Retinopathy Balanced dataset** from Kaggle. It contains labeled retinal images grouped into five DR classes. The data was split into:

- **Training set:** ~27,834 images (80%)
- **Validation set:** ~6,958 images (20%)

Images were resized to **128×128** and augmented (rotation, zoom, flip) to enhance generalization.

4. Methodology

4.1 Data Preprocessing

- Rescaled pixel values to [0, 1].
- Applied augmentation to reduce overfitting.
- Used training and validation splits with ImageDataGenerator.

4.2 Model Architecture

A **Convolutional Neural Network (CNN)** was built with the following layers:

- 3 Convolution + MaxPool blocks
- A Flatten layer
- Dense layer with 128 neurons
- Dropout for regularization
- Output Dense layer with 5 neurons and softmax activation

Model summary:

```
Conv2D(32) → MaxPool2D
Conv2D(64) → MaxPool2D
Conv2D(128) → MaxPool2D
Flatten → Dense(128) → Dropout → Dense(5) (softmax)
```

5. Training Setup

- **Optimizer:** Adam (learning rate 1e-4)
- **Loss:** Categorical Crossentropy
- **Metrics:** Accuracy
- **Callbacks:** EarlyStopping (patience 3), ModelCheckpoint
- **Epochs:** 20 (stopped early due to training behavior)
- Only 7 Epochs are completed at this time.

6. Results

6.1 Training Performance

Although training progressed, validation accuracy remained modest (~45%), indicating challenges in classification performance with a simple CNN architecture:

```
Epoch 7/20 — val_accuracy: ~0.45
```

6.2 Classification Report

From model predictions on the validation set:

Class	Precision	Recall	F1-Score	Support
0	0.44	0.73	0.55	1400
1	0.35	0.38	0.36	1358
2	0.35	0.17	0.23	1400
3	0.53	0.43	0.48	1400
4	0.61	0.56	0.59	1400
Accuracy	—	—	**~0.46**	6958

Overall performance remained modest with lower precision and recall for mid-severity classes, showing the difficulty of the multi-class DR classification problem.

7. Analysis and Discussion

The results reveal:

- The model performs **best for class 0 and class 4**, likely because they have more distinctive features.
- Middle classes were harder to distinguish, leading to substantial misclassification.
- A simple CNN struggles with subtle retinal features and fine granularity between adjacent severity levels, as highlighted in published research in DR classification.

In research settings, advanced architectures such as **transfer learning (EfficientNet, ResNet)** significantly improve performance on biomedical image tasks.

8. Grad-CAM Visualization

A Grad-CAM heatmap was generated to interpret which regions contributed to the model's decision. The heatmap highlights areas the CNN attended to in the image, offering insight into model focus and identifying where improvements may be needed.

9. Conclusion

This project successfully developed a CNN model to classify diabetic retinopathy severity using retinal fundus images within a hackathon environment. While performance was limited, it provides a good baseline and shows the potential for improvement with more complex techniques and larger image input sizes.

10. Future Work

To improve performance:

- Use **transfer learning** with pre-trained models (e.g., EfficientNet)
- Increase image resolution (e.g., 224×224)
- Apply class-weighting and focal loss to handle class confusion
- Add more extensive preprocessing and lesion segmentation