Deep Learning Fundus Image Analysis for Early Detection of Diabetic Retinopathy

Using Transfer Learning (Xception) with Flask Web Application and IBM Cloudant Integration

1. Introduction & Objective

Diabetic Retinopathy (DR) is a severe complication of diabetes that damages the retina and may lead to blindness if not detected early.

This project develops a Deep Learning-based web application that detects and classifies DR severity using fundus retinal images.

The system uses Transfer Learning with Xception architecture and provides predictions through a Flask web interface.

Objectives:
• Develop an automated DR detection system
• Classify retinal images into five severity levels
• Implement transfer learning using Xception
• Build a secure Flask web application
• Integrate IBM Cloudant NoSQL database

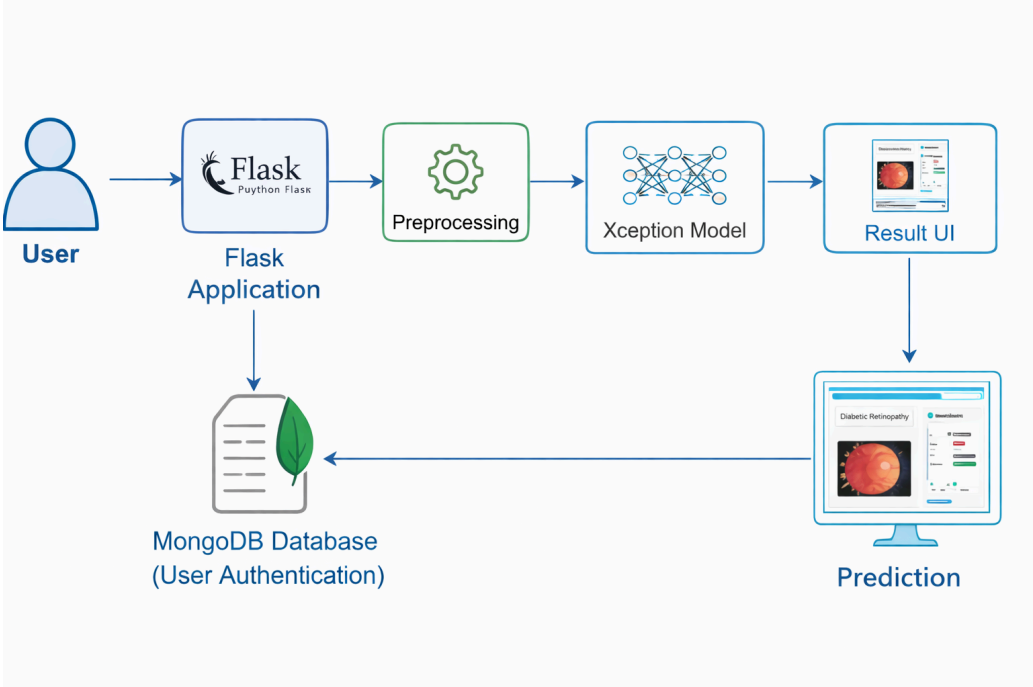
2. Technology Stack & Dependencies

Programming Language: Python
Deep Learning Framework: TensorFlow, Keras
Model Architecture: Xception (Pre-trained on ImageNet)
Web Framework: Flask
Frontend: HTML, CSS
Database: IBM Cloudant NoSQL

Key Libraries:
tensorflow, keras, flask, numpy, pandas, matplotlib, cloudant, werkzeug

3. System Architecture & Workflow

System Architecture:



Workflow:
1. User logs in.
2. User uploads retinal image.
3. Image is resized to 229x229 and preprocessed.
4. Model predicts probability scores.
5. Highest probability class is selected.
6. Result with confidence score is displayed.

4. Data Collection & Preprocessing

Dataset Source: Kaggle

Classes:
0 – No DR
1 – Mild
2 – Moderate
3 – Severe
4 – Proliferative DR

Preprocessing Steps:
• Resize images to 229x229
• Apply Xception preprocess_input normalization
• Use ImageDataGenerator for augmentation

5. Model Architecture

Input Layer: (229x229x3)
Xception Base Model (Frozen Layers)
Flatten Layer
Dense Layer (256, ReLU)
Output Layer (5 Classes, Softmax)

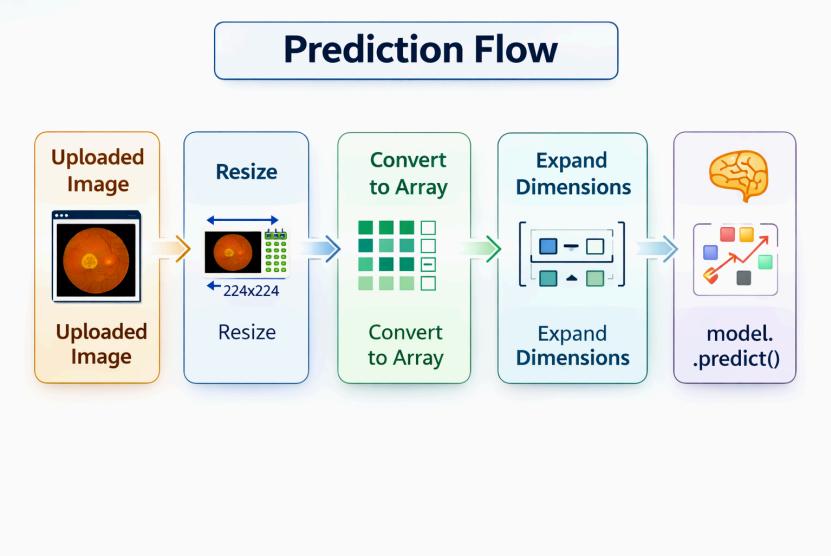
Loss Function: Categorical Crossentropy
Optimizer: Adam
Epochs: 30

6. Model Training Process

Transfer learning approach is used.
Base Xception layers are frozen.
Custom dense layers are trained on retinal dataset.
Model is trained for 30 epochs.
Best model saved as Updated-Xception-diabetic-retinopathy.h5

7. Flask Web Application

Features:
• User Registration and Login
• Secure Session Handling
• Image Upload
• Image Preprocessing
• Model Prediction
• Confidence Score Display



8. MongoDB Database

MongoDB is a NoSQL database used for storing application data in JSON-like documents.

In this project, MongoDB is used to:

- Store user registration details
- Validate login credentials during authentication
- Manage user session-related data

MongoDB provides fast, scalable, and flexible data storage, making it suitable for web applications integrated with Flask.

---

9. How to Run the Application

Step 1: Install Dependencies

pip install -r requirements.txt

Step 2: Place the trained model file

Place the trained .h5 model inside the project directory (or specified model folder).

Step 3: Start MongoDB Server

Ensure MongoDB service is running:

net start MongoDB

Step 4: Run Flask Application

python app.py

Step 5: Open in Browser

http://127.0.0.1:5000

---

10. Output & Results

The system classifies retinal fundus images into the following categories:

- No DR
- Mild
- Moderate
- Severe
- Proliferative DR

The application displays:

- Uploaded retinal image
- Predicted disease stage
- Confidence score percentage

---

11. Conclusion & Future Enhancements

Conclusion

This project demonstrates the effectiveness of Transfer Learning using the Xception Convolutional Neural Network for automated Diabetic Retinopathy detection.

Integration of Deep Learning with Flask and MongoDB enables a scalable, efficient, and user-

friendly diagnostic support system.

Future Enhancements

- Implement Grad-CAM visualization for model explainability
- Use EfficientNet or Vision Transformers for improved accuracy
- Deploy the application on cloud platforms (AWS, Azure, Render)
- Implement secure password hashing and encryption
- Develop a mobile-friendly or Android application version
- Format this properly for README.md
- Convert this into a project report section
- Prepare viva explanation points
- Create PPT content for presentation

Tell me what you need next.