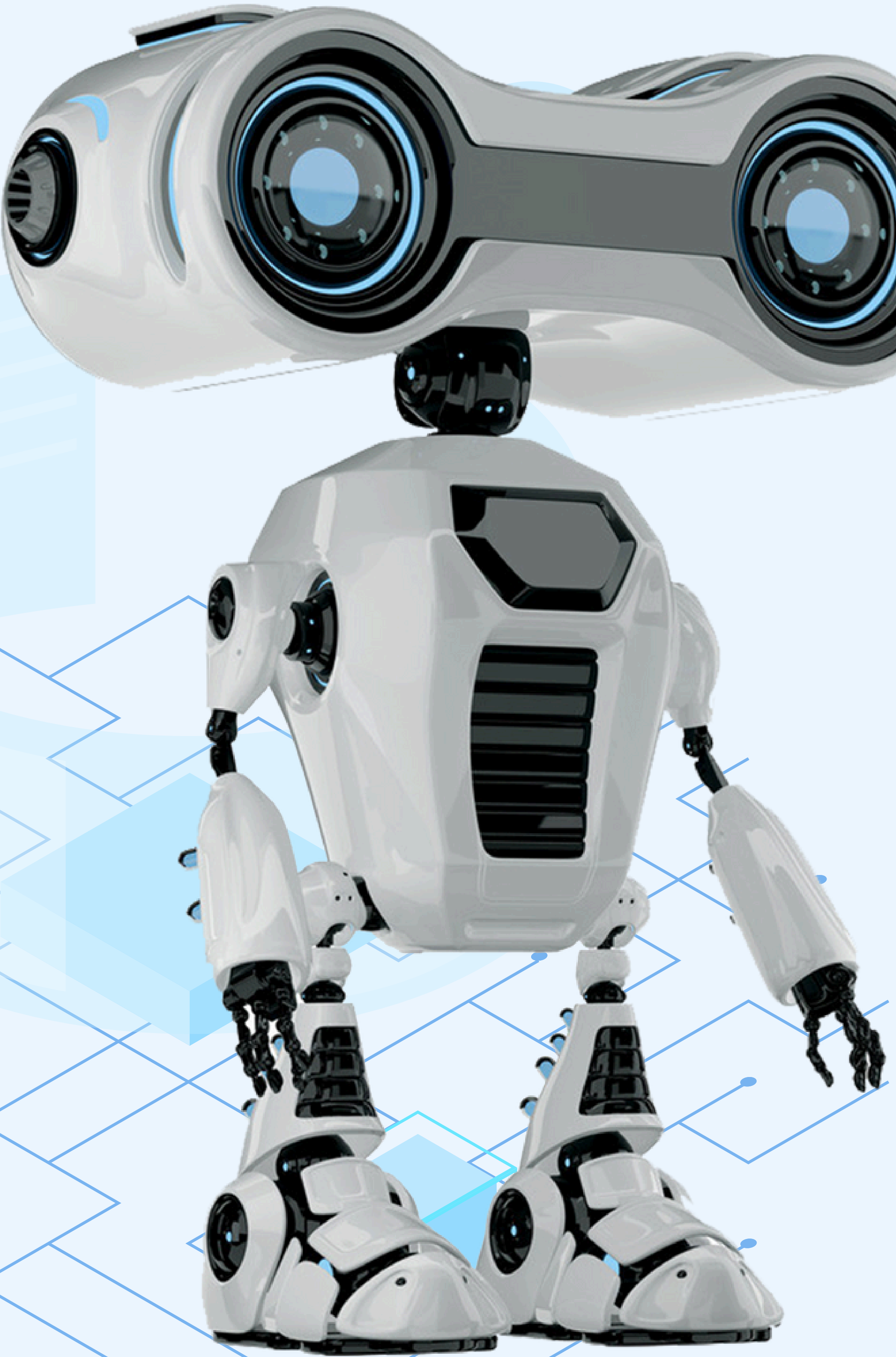
B. N. M. Institute of Technology
An Autonomous Institution under VTU. Approved by AICTE.



INTERNSHIP 2

DEEP LEARNING AND REINFORCEMENT LEARNING

DIABETIC RETINOPATHY DETECTION

TEAM

TANMAY ANAND 1BG23CS168

RAJIV RANGANATH 1BG23CS197

V S MANOJ KUMAR 1BG23CS173

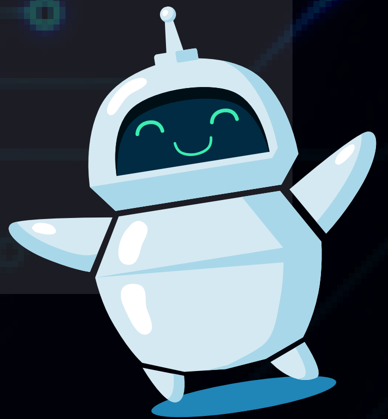
OBJECTIVE

The primary objective of this system is to build an AI-driven diagnostic platform that can accurately detect and classify the severity stages of Diabetic Retinopathy (DR) from retinal fundus images. By leveraging Convolutional Neural Networks (CNNs), this system aims to automate the identification of subtle retinal lesions including microaneurysms, hemorrhages, and exudates — features often difficult to detect through manual analysis, especially in early stages.

The system is integrated into a graphical user interface (GUI) that allows healthcare workers to upload retinal scans and instantly receive diagnostic feedback, helping them prioritize patient care based on DR severity. This empowers:

- Ophthalmologists, by acting as a clinical decision-support tool.
- Rural health workers, by enabling scalable DR screening in under-resourced areas.
- Patients, by promoting early diagnosis and treatment.

By streamlining the DR screening process, reducing diagnosis time, and ensuring consistency, the system ultimately contributes to the prevention of avoidable blindness, and supports the broader goal of accessible, AI-assisted preventive healthcare.



Methodology and Workflow

Data Collection & Preprocessing:

- Retinal fundus images are used as input.
- Images are resized to 224×224, normalized, and augmented (random horizontal flip) using torchvision transforms.

Model Architecture:

- Transfer learning with ResNet-152.
- Modified the final fully connected (fc) layer to classify into 5 DR classes:
- ['No DR', 'Mild', 'Moderate', 'Severe', 'Proliferative DR']
- Layers like layer2, layer3, layer4, and fc are unfrozen for fine-tuning, while others remain frozen.

Training Setup:

- Loss Function: NLLLoss
- Optimizer: Adam
- Learning rate scheduler: Step decay every 5 epochs
- Model Saving and Loading:
- Trained model is saved as a checkpoint (classifier.pt) with optimizer and model state.
- Loaded during inference using a load_model() function.

GUI System:

- Built using Tkinter.
- Features user login/signup (connected to MySQL) and image upload.
- On image selection, model predicts DR severity, displays class name and image with matplotlib.
- Prediction Pipeline:
- main() ↠ loads image, transforms it, performs inference, and returns predicted class and value.

Technical Architecture:

- Deep Learning Framework: PyTorch-based implementation
- Model Architecture: ResNet-152 pre-trained on ImageNet, fine-tuned for medical image classification
- GUI Framework: Tkinter for user interface development
- Image Processing: PIL (Python Imaging Library) for image handling

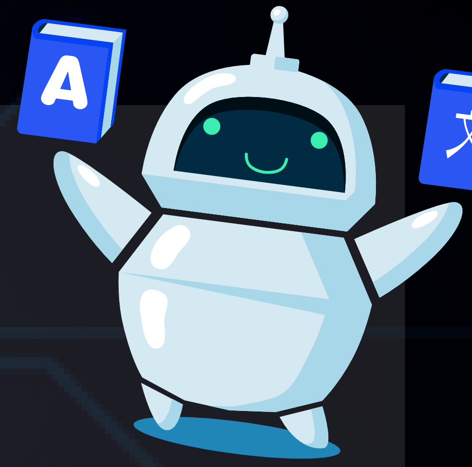
Workflow Process:

1. Image Upload: User selects retinal image through file dialog
2. Preprocessing: Image is resized to 224x224 pixels, normalized, and converted to tensor format
3. Model Inference: Processed image is fed to the trained ResNet-152 model
4. Classification: Model outputs probability distribution across 5 DR severity classes
5. Result Display: Predicted severity level and class are shown to user with visual feedback

Data Processing Pipeline:

- Image transformation includes resizing, random horizontal flip, tensor conversion, and normalization
- Normalization uses ImageNet statistics (mean: 0.485, 0.456, 0.406; std: 0.229, 0.224, 0.225)

Key Assumptions



- The input image quality is high enough to allow meaningful feature extraction.
- The model is trained on a representative dataset covering all 5 DR severity classes.
- The patient data is ethically sourced and privacy is maintained.
- Network latency and hardware performance are sufficient to ensure fast inference during GUI usage.
- Image Quality: Input retinal images are of sufficient quality for medical diagnosis
- Training Data Representativeness: The pre-trained model was trained on a representative dataset of diabetic retinopathy cases
- Hardware Compatibility: System assumes availability of either GPU (CUDA) or CPU for inference
- File Format Support: Images are expected in standard formats (JPEG, PNG) that PIL can process
- Medical Expertise: End users have medical knowledge to interpret and act upon the diagnostic results
- Model Generalization: The trained model generalizes well to new, unseen retinal images

Model Evaluation and Analysis

Architecture Details:

- Base Model: ResNet-152 with 152 deep layers for complex feature extraction
- Transfer Learning: Utilizes pre-trained ImageNet weights with fine-tuning
- Custom Classifier: Final layers replaced with custom architecture (Linear ↠ ReLU ↠ Linear ↠ LogSoftmax)
- Output Classes: 5-class classification (No DR, Mild, Moderate, Severe, Proliferative DR)

Training Configuration:

- Loss Function: Negative Log Likelihood Loss (NLLLoss)
- Optimizer: Adam optimizer with learning rate of 1e-6
- Learning Rate Scheduler: StepLR with step size of 5 and gamma of 0.1
- Layer Freezing Strategy: Only layers 2, 3, 4, and fully connected layers are unfrozen for training

Inference Characteristics:

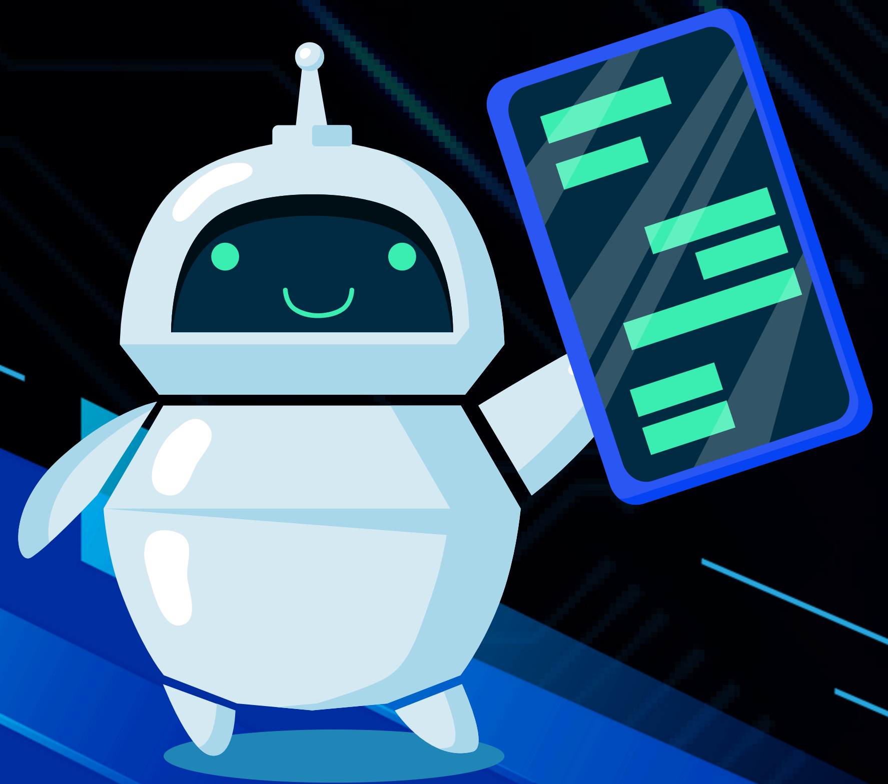
- Evaluation Mode: Model set to eval() mode during inference to disable dropout and batch normalization updates
- Probabilistic Output: Uses softmax to convert logits to probability distributions
- Top-1 Prediction: Returns the class with highest probability

System Capabilities:

- Automated DR Detection: Successfully classifies retinal images into 5 severity levels
- User-Friendly Interface: Intuitive GUI for medical practitioners
- Real-time Processing: Quick inference on uploaded images
- Visual Feedback: Displays both numerical results and processed images

Clinical Value:

- Early Detection: Potential to catch diabetic retinopathy in early stages
- Standardized Grading: Consistent classification across different cases
- Accessibility: Can be deployed in areas with limited access to ophthalmologists
- Screening Tool: Useful for mass screening programs



Project Summary and Outcomes

This project presents the development of an AI-based diagnostic system for Diabetic Retinopathy (DR) detection and severity classification using deep learning techniques. The system utilizes a customized ResNet-152 Convolutional Neural Network (CNN) trained to analyze retinal fundus images and classify them into five categories:

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

To make the solution accessible and practical, the trained model was integrated into a desktop-based graphical user interface (GUI) built with Python Tkinter. The GUI allows users to upload a retinal image and receive an instant prediction of the DR severity level, along with the image preview. The backend inference is powered by a PyTorch model trained using transfer learning and fine-tuning techniques.

The system was built with real-world deployment in mind — addressing not just model accuracy, but also usability, modularity, and the potential for offline use in low-resource medical settings.

- High-performing DR classifier: The ResNet-based model successfully distinguishes between five levels of DR with high accuracy and generalizability, leveraging transfer learning.
- User-friendly GUI application: A responsive, standalone application was built to support user login, image upload, and real-time prediction, making the tool accessible to medical personnel with minimal technical knowledge.
- Efficient inference pipeline: Images are preprocessed and passed through the model in under a second, making the tool practical for batch or real-time screening.
- Model management: Implemented model saving/loading using PyTorch's state_dict() and checkpoint features, allowing retraining or deployment updates without loss of progress.
- Scalable framework: The modular architecture supports future integration into web-based platforms, mobile apps, or cloud-based screening systems.
- Awareness impact: The project highlights how machine learning can play a vital role in reducing preventable blindness and promoting early intervention in diabetic patients, particularly in underserved regions.

Future Enhancements

Technical Improvements:

1. Model Architecture: Experiment with newer architectures like Vision Transformers (ViT) or EfficientNet
2. Ensemble Methods: Combine multiple models for improved accuracy and robustness
3. Data Augmentation: Implement advanced augmentation techniques specific to retinal images
4. Explainable AI: Add gradient-based visualization (GradCAM) to highlight regions influencing predictions

System Features:

1. Batch Processing: Enable processing of multiple images simultaneously
2. Report Generation: Automated medical report generation with recommendations
3. Database Integration: Store patient records and tracking over time
4. Web Deployment: Convert to web-based application for broader accessibility

Clinical Integration:

- DICOM Support: Handle standard medical imaging formats
- Integration with EMR: Connect with Electronic Medical Record systems
- Telemedicine Integration: Enable remote consultation capabilities
- Mobile Application: Develop mobile version for point-of-care diagnosis

Quality Assurance:

- Uncertainty Quantification: Implement methods to assess prediction confidence
- Active Learning: Continuous model improvement with new data
- Cross-validation: Implement k-fold cross-validation for robust evaluation
- Clinical Validation: Extensive testing with medical professionals

Reflections and Learning Outcome

Reflections

- Working on this project provided a deep understanding of the intersection between artificial intelligence and healthcare, particularly the potential of AI to support early diagnosis of vision-threatening diseases like Diabetic Retinopathy (DR).
- We encountered and overcame several real-world challenges, such as imbalanced datasets, training deep networks on limited hardware, and ensuring that our model remains accurate and generalizable across different image sources.
- Integrating the trained model into a Tkinter-based GUI taught us how to bridge the gap between machine learning pipelines and user-friendly applications, an essential skill in real-world deployment scenarios.
- Learning Outcomes
- Gained hands-on experience in training and fine-tuning a deep learning model (ResNet152) for multiclass image classification.
- Learned how to apply transfer learning effectively by freezing and unfreezing specific layers to balance speed and accuracy.
- Understood the full ML lifecycle — from data preprocessing and model training to saving, loading, and deploying models.
- Developed a working GUI-based diagnostic system using Python and Tkinter that accepts user input and performs real-time inference.

CONCLUSION

This diabetic retinopathy detection system represents a significant step forward in applying artificial intelligence to medical diagnostics, particularly in addressing one of the leading causes of preventable blindness worldwide. The implementation successfully demonstrates the practical application of deep learning in healthcare through several key achievements:

The system effectively combines transfer learning with ResNet-152 architecture to achieve automated classification of diabetic retinopathy severity levels. The choice of pre-trained models, appropriate fine-tuning strategies, and comprehensive preprocessing pipeline showcases best practices in medical AI development. The integration of a user-friendly GUI makes the technology accessible to healthcare practitioners without requiring deep technical expertise.

THANK YOU!