PROJECT REPORT

Project Title:

HematoVision – A Deep Learning Odyssey in Blood Cell Classification

Team ID: LTVIP2025TMID41231

Submitted By:

- Vundrakonda Satyanarayanacharyulu
- V Vinay Karthik Reddy

Institution:
Sri Venkateswara College of Engineering, Dept. of CSE

Submitted To:
SmartInternz - AI/ML Virtual Internship Program 2025

1. INTRODUCTION

1.1 Project Overview

HematoVision is an innovative deep learning-powered web application designed to automate the classification of peripheral blood cells from microscopic images. Leveraging the power of **transfer learning**, specifically using the **MobileNetV2 architecture**, HematoVision accurately identifies and differentiates among **Red Blood Cells (RBCs)**, various subtypes of **White Blood Cells (WBCs)** such as **Lymphocytes, Neutrophils, Monocytes, and Eosinophils**, and **Platelets**.

Manual classification of blood cells is a routine yet time-consuming process in clinical laboratories. It often requires trained experts and can be subject to fatigue and human error. HematoVision addresses these challenges by offering a high-speed, automated, and reliable AI solution. Its intuitive web interface allows users to simply upload an image of a blood smear, and within seconds, receive a prediction with a confidence score, reducing diagnostic time significantly.

Furthermore, the application is **lightweight**, scalable, and can be deployed on local machines or in the cloud, making it suitable not only for advanced diagnostic centers but also for **remote and resource-constrained healthcare environments**.

1.2 Purpose

The purpose of HematoVision is to **redefine the workflow in hematological diagnostics** by integrating artificial intelligence into traditional laboratory practices. It aims to replace the repetitive task of manual blood cell identification with an **automated, standardized, and scalable system** that boosts diagnostic accuracy and efficiency.

This project was envisioned to serve multiple goals:

- In **clinical settings**, it assists lab technicians by minimizing manual workload and speeding up report generation.

- In **educational institutions**, it acts as a hands-on learning tool for students in biomedical sciences and machine learning.

- For **AI researchers**, it provides a practical demonstration of applying convolutional neural networks and transfer learning in medical image classification.
- In **rural and low-resource areas**, its low system requirements and offline capabilities help bridge the gap in access to advanced diagnostic tools.

.

2. IDEATION PHASE

2.1 Problem Statement

In healthcare, accurate blood cell classification is crucial for diagnosing conditions like leukemia, anemia, and infections. Manual microscopy is time-consuming and often inconsistent. HematoVision addresses this by providing a scalable AI solution to classify blood cells using deep learning.

2.2 Empathy Map Canvas

User Persona: Lab technician in a district hospital who needs to process 100+ samples daily.

- Says – "I need a fast and accurate way to identify blood cells."
- Thinks – "Misclassification could result in wrong treatment."
- Does – Visually inspects images, refers to training samples.
- Feels – Tired from long shifts, anxious about human error.
- Hears – Complaints from doctors about delays in reports.
- Sees – Many similar-looking images, lacks expert help.
- Pains – Time pressure, unclear results, no AI support.
- Gains – Quicker diagnosis, reliable classification, confidence.

2.3 Brainstorming

The team evaluated multiple CNN architectures (ResNet, VGG, MobileNet) and finalized MobileNetV2 for its balance of accuracy and efficiency. We tested image augmentation strategies and identified Flask as the ideal deployment platform. Discussions focused on accessibility, mobile compatibility, and UX.

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

User Persona:

Priya, a lab technician in a regional hospital, is responsible for analyzing blood smear slides and preparing reports for doctors.

Stage	User Action	Pain Points	Solution via HematoVision
1. Awareness	Learns about HematoVision during training	Unaware of AI-based tools	Onboarding with video tutorials
2. Access	Opens HematoVision web app	No software installation access	Browser-based app, no installation
3. Upload	Uploads blood smear image	Unclear image, wrong format	Validation, auto-cropping, preprocessing
4. Classification	Waits for prediction	Worry about accuracy/speed	Optimized MobileNet model
5. Result Interpretation	Views result and confidence	Unsure what scores mean	Clear class label + confidence
6. Decision	Uses result for diagnosis	Delays from manual check	Instant prediction, faster reports
7. Feedback	Provides feedback	No loop in manual process	Feedback considered in future versions

3.2 Solution Requirements

Functional Requirements:

1. Image Upload Interface

2. AI-Based Classification with MobileNetV2

3. Display of Prediction and Confidence Score

4. Responsive Web UI

5. Minimal User Input

Non-Functional Requirements:

1. Performance: Inference <2 seconds

2. Scalability: Support more cell types

3. Reliability: Input validation and clear error messages

4. Maintainability: Modular codebase on GitHub

5. Security: No storage of medical data, input sanitization

6. Portability: Deployable on Heroku or AWS

3.3 Data Flow Diagram (DFD)

Entities:
• User uploads image
Processes:
1. Upload Handler
2. Preprocessing
3. Classifier Engine
4. Output Formatter
Data Stores (Optional):
• Temporary Image Buffer
• Prediction Logs

3.4 Technology Stack

• Python – Model and backend

• TensorFlow/Keras – Transfer learning

• Flask – Web framework

• HTML5/CSS3 – Frontend UI

• OpenCV/Pillow – Image preprocessing

• Jupyter Notebook – Model training

• Render/Heroku – Cloud deployment

4. PROJECT DESIGN

4.1 Problem-Solution Fit

Manual inspection of blood cell images is time-intensive and subject to human error. HematoVision solves this by offering an AI-powered classification system that provides consistent, high-accuracy results, even with minimal resources.

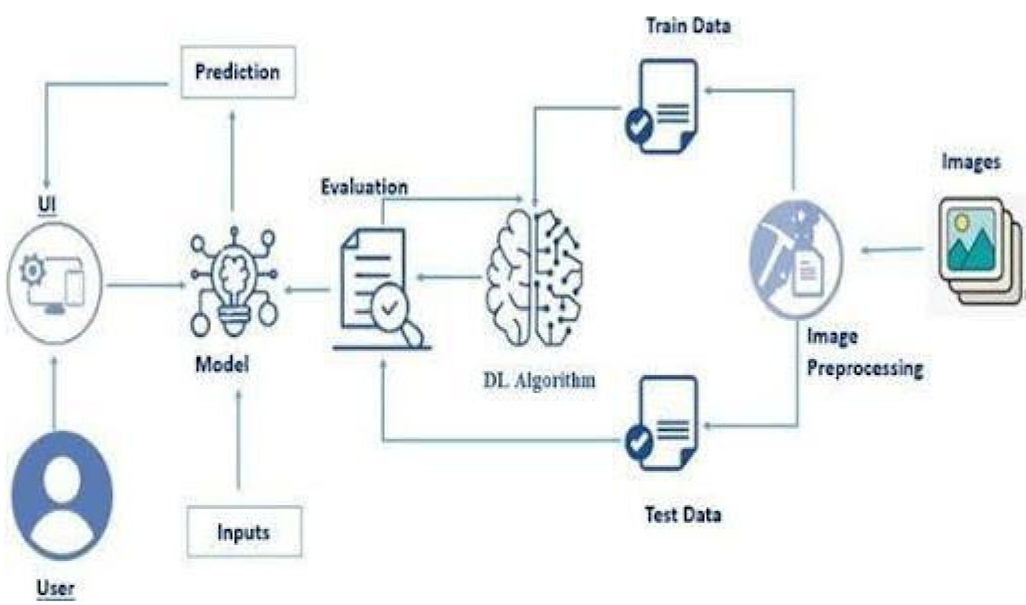
4.2 Proposed Solution

HematoVision is a Flask-based web application integrated with a MobileNetV2 model. Users upload images, which are preprocessed and passed through the model. The result — cell type and confidence — is shown on a responsive result page.

Core Components:

• MobileNetV2 model
• Flask backend
• HTML/CSS frontend
• Preprocessing pipeline
• Ready for cloud deployment

4.3 Solution Architecture



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

The project was structured in 4 weekly sprints, each focused on a distinct phase: setup, development, integration, and testing.

Week	Tasks	Assigned To
Week 1	Environment setup, data collection	Vinay
Week 2	Model training, MobileNet integration	Satya,vinay
Week 3	Frontend design, Flask backend	satya
Week 4	Testing, documentation, deployment	All Members

Project Tools:
• GitHub – Version Control
• Google Sheets – Planning
• Jupyter Notebook – Model dev
• VS Code – Codebase
• Google Meet – Team Syncs

Each sprint began with a short planning session, where tasks were distributed based on member strengths and availability. During Week 1, the team set up the development environment and obtained the blood cell dataset from Kaggle. In Week 2, focus shifted to training the model using **MobileNetV2**, where hyperparameter tuning and image augmentation were tested. Week 3 emphasized frontend creation using HTML/CSS and integrating it with the backend powered by Flask. Finally, Week 4 was dedicated to rigorous testing, error handling, documentation, and deploying the model for demonstration.

This structured sprint-based planning ensured effective collaboration, minimal bottlenecks, and a transparent development cycle — key to successfully delivering the project within the timeline.

6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Functional Testing

Test Case ID	Scenario	Input	Expected Output
TC-01	Upload RBC image	rbc.jpg	"RBC", confidence >90%
TC-02	Upload WBC image	wbc.png	"WBC", confidence >85%
TC-03	Blurred image	platelet_blur.jpg	Low confidence warning
TC-04	Non-image upload	notes.txt	Error: Invalid format
TC-05	No file selected	None	Prompt to upload image
TC-06	Test on mobile	Mobile Browser	Responsive UI and result

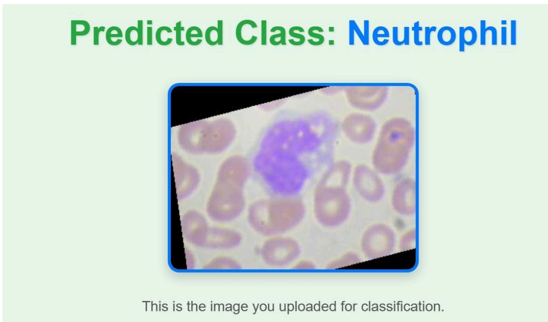
6.2 Performance Testing

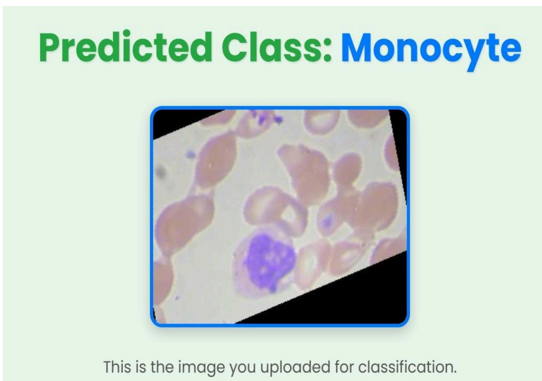
• Accuracy: ~93%
• Inference Time: ~1.3s (CPU)
• Model Size: ~13MB
• Responsive UI: Yes
• Scalable to more classes

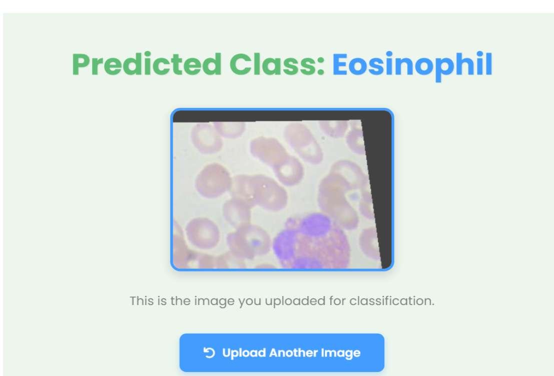
7. RESULTS

7.1 Output Screenshots

HematoVision was tested on multiple blood cell image samples. The following interface descriptions illustrate how users interact with the system.







8. ADVANTAGES & DISADVANTAGES

8.1 Advantages

1. **Real-Time Classification**
   HematoVision utilizes a lightweight MobileNetV2 model that enables blood cell classification in under two seconds. This significantly accelerates diagnostic workflows in clinical settings.

2. **Lightweight & Portable**
   The model is compact (~13MB) and can be deployed on modest hardware or virtual machines, making it ideal for low-resource environments such as rural clinics or mobile labs.

3. **User-Friendly Interface**
   The web application features a clean and responsive UI. Designed with accessibility in mind, it allows even non-technical users—such as lab technicians—to operate the system without training.

4. **Modular Architecture**
   The project is structured in a modular fashion, which makes it easy to add support for new cell types or integrate with existing laboratory information systems in the future.

5. **Cross-Platform Compatibility**
   HematoVision works seamlessly across various devices including desktops, tablets, and smartphones, ensuring flexibility in usage and broader reach.

6. **Data Privacy & Security**
   The application is designed with data privacy in mind. Uploaded images are processed in memory without persistent storage, ensuring compliance with medical data handling standards.

8.2 Disadvantages

1. **Limited Dataset Size**
   The model was trained on a publicly available dataset. While sufficient for proof-of-concept, its performance may decline when applied to rare blood abnormalities or poorly labeled data.

2. **Image Quality Sensitivity**
   Like most computer vision models, HematoVision's accuracy is highly dependent on the quality of input images. Blurry, low-resolution, or overexposed images can reduce prediction reliability.

3. **Lack of Feedback Loop**
   Currently, the system does not support corrective feedback from users. Incorrect predictions cannot be flagged or used to retrain the model automatically.

4. **Not Yet Cloud Hosted**
   Although designed for portability, HematoVision is not yet deployed on a public cloud platform. Deployment must be done manually, which may limit accessibility for some users.

5. **Single Language Interface**
   The current UI is only available in English. This may restrict usability in multilingual or non-English-speaking regions unless localized versions are developed.

9. CONCLUSION

HematoVision stands as a compelling demonstration of how artificial intelligence can enhance modern healthcare by automating and accelerating diagnostic processes. By leveraging **transfer learning with MobileNetV2**, the system achieves high-accuracy classification of blood cells in a fraction of the time required for manual microscopy.

This project highlights the **practical value of deep learning** in critical fields such as hematology, where early detection and precise classification can significantly affect patient outcomes. HematoVision bridges the gap between theoretical machine learning models and real-world clinical applications through a lightweight, deployable, and user-friendly interface.

In addition to its clinical relevance, the project provided valuable hands-on experience in **full-stack AI development**. Team members engaged in data preprocessing, model training, web integration with Flask, frontend design using HTML/CSS, and testing methodologies — simulating a complete machine learning product pipeline.

Beyond technical achievements, HematoVision reinforces the potential of AI to democratize access to quality healthcare. Whether used in **urban laboratories**, **rural clinics**, or **educational institutes**, the platform illustrates how intelligent automation can contribute to efficiency, reliability, and learning in the medical ecosystem.

10. FUTURE SCOPE

As a scalable and evolving AI-powered diagnostic system, HematoVision holds vast potential for enhancement and deployment in real-world clinical and educational scenarios. The following roadmap outlines key areas for future development:

1. **Expand Dataset**
   The current model is trained on a balanced but limited dataset. Future iterations will incorporate larger and more diverse datasets, including rare blood cell types (e.g., blast cells, sickle cells) and disease-specific samples such as those found in leukemia and anemia cases. This will increase classification depth and improve generalizability across populations.

2. **Cloud Deployment**
   Hosting HematoVision on cloud platforms like **Render, AWS, or Azure** will make it globally accessible and eliminate local setup barriers. Cloud deployment will also enable scalability, load balancing, and real-time access for remote healthcare units.

3. **Feedback Loop Integration**
   To continuously improve model accuracy, a feedback loop will be added. Users (e.g., lab technicians) will be able to confirm or correct predictions, and these interactions will be stored for retraining the model — creating a **self-improving intelligent system**.

4. **Multilingual Interface**
   Currently limited to English, future versions will support multiple regional languages to broaden usability in multilingual countries. This localization will make the tool more inclusive for technicians in rural and non-English-speaking regions.

5. **Mobile Application**
   A native mobile app, developed using **Flutter or React Native**, will bring HematoVision to smartphones and tablets. This will enable field diagnostics in remote areas without access to desktop systems, especially useful for mobile healthcare camps.

6. **Admin Dashboard**
   A dedicated admin dashboard will allow authorized users to monitor usage trends, prediction statistics, and dataset contributions. It will also provide real-time analytics for lab supervisors and developers to evaluate performance and user behavior.

7. **Enhanced Security Features**
   To comply with clinical data regulations and privacy standards, future versions will incorporate **role-based access control (RBAC)**, **HTTPS encryption**, and secure API gateways. These measures will ensure that sensitive medical images and reports are handled securel

11. APPENDIX

11.1 Source Code Repository

GitHub: https://github.com/SatyaD012/Hematovision-advanced-blood-cell-classification

11.2 Dataset Used

Blood Cell Dataset from Kaggle:
https://www.kaggle.com/paultimothymooney/blood-cells

11.3 Project Demo Video

YouTube: https://youtu.be/txGpZVVPU48?feature=shared

11.4 Tools and Technologies

- Python, TensorFlow, Keras
- MobileNetV2
- Flask
- HTML5, CSS3
- OpenCV, Pillow
- GitHub, Colab, VS Code

11.5 References

1. Kaggle Dataset – https://www.kaggle.com/paultimothymooney/blood-cells
2. TensorFlow Docs – https://www.tensorflow.org
3. Flask Docs – https://flask.palletsprojects.com
4. MobileNetV2 Paper – Sandler et al. (2018)