HematoVision

Blood Cell Classification Using Transfer Learning

By

MAMIDI SAISANTHI (S200873,RGUKT-Srikakulam)

GOWREESH KINTADA (S200854,RGUKT-Srikakulam)

Date:
June 27, 2025

1 Abstract

HematoVision is a deep learning-based platform designed to automate the classification of blood cells using advanced transfer learning techniques. Using a dataset of 12,000 labeled microscopic images, the model accurately distinguishes four key cell types: eosinophils, lymphocytes, monocytes, and neutrophils. Built using pre-trained CNN architectures like VGG16 and deployed via Flask, the system supports real-time classification with broad applicability in diagnostics, telemedicine, and medical training. This report presents a complete breakdown of the dataset, model design, training pipeline, deployment architecture, and evaluation metrics.

2 Introduction

Blood cell classification is fundamental in the diagnosis and monitoring of hematological and immunological disorders. Manual examination by hematologists is time-consuming and subject to human error. To address this, HematoVision leverages the power of deep learning to automate the identification of blood cells with high accuracy and efficiency.

This project aims to:

- Automate the classification of blood cells from microscopic images.
- Leverage transfer learning for fast, accurate, and generalizable results.
- Provide a lightweight, deployable web interface for real-time use.

3 Objective

The primary objective of this project is to develop a reliable and efficient image classification system to automatically identify four types of white blood cells—eosinophils, lymphocytes, monocytes, and neutrophils—from microscopic images. Key goals include:

Achieving high classification accuracy using transfer learning.

Building a user-friendly web application for image upload and prediction.

Ensuring fast, real-time inference for practical use in diagnostics and training.

4 Dataset

The model is trained on a labeled dataset from Kaggle containing around 12,500 augmented images of blood cells divided into four main types: Eosinophil, Lymphocyte, Monocyte, and Neutrophil, with about 3,000 images per type. Labels are provided in CSV format. Additionally, another folder includes 410 original images (before augmentation) along with bounding boxes and subtype labels in XML files. A second dataset folder adds 2,500 more images with extra subtype labels. This rich dataset is useful for training and testing blood cell classification models effectively.

- **Eosinophils**: Eosinophils are a type of white blood cell (WBC), part of the body's immune system. They play a crucial role in fighting infections, especially parasitic infections, and are also involved in allergic reactions.

- **Lymphocytes**: Lymphocytes are a type of white blood cell that help your body fight infections and diseases. They are an important part of your immune system.
- **Monocytes**:Monocytes are a type of white blood cell that help your body fight germs and clean up damaged tissues. They are part of your immune system and are like the body's clean-up crew.
- **Neutrophils**: Neutrophils are the most common type of white blood cell in your body. They are the first line of defense against infections — like the body's soldiers that quickly attack germs.

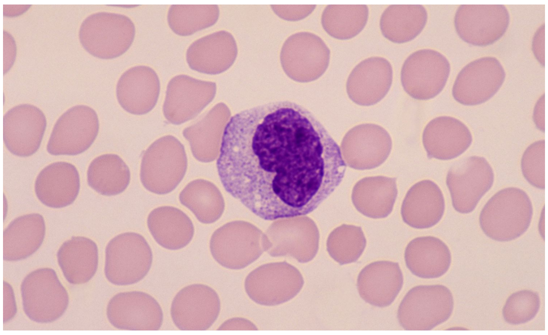
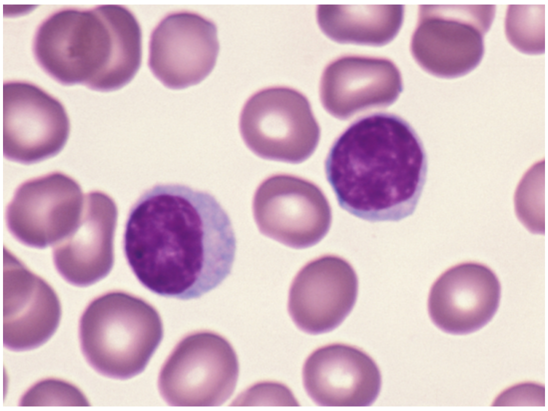


Figure 1: Eosinophil



.jpeg

Figure 2: Lymphocyte

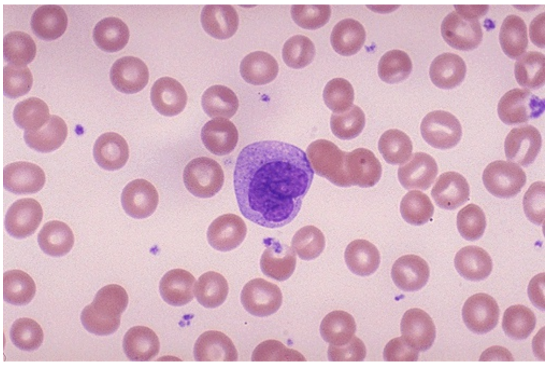


Figure 3: Monocyte

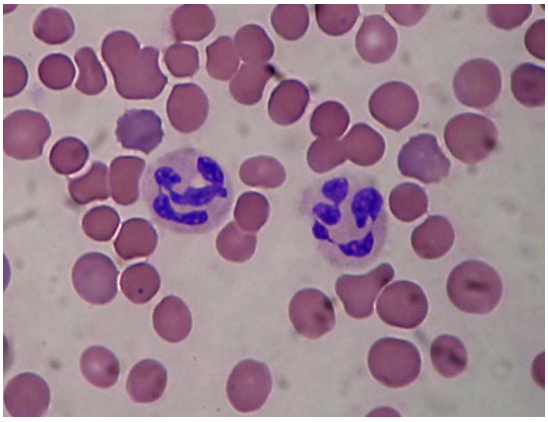


Figure 4: Neutrophil

5 Methodology Steps taken

Preprocessing

- Image Resizing All images are resized to a fixed size (e.g., 224x224 pixels) to match the input size of pre-trained models like VGG16, ResNet50, etc.

- Normalization Pixel values (0–255) are scaled to a smaller range (like 0 to 1) to help the model learn better.
- Used to increase the number and variety of images atificially by applying:

  Rotation

  Flipping (horizontal/vertical)

  Zooming

  Shifting

  Brightness changes

  Helps prevent overfitting and improves generalization

Data Split

- 80% training (9600 images)
- 20% testing (2400 images)

6 Model Architecture

In this project, we use transfer learning, which means we take a pre-trained model (already trained on millions of images) and fine-tune it for our task — classifying blood cells.

Training Pipeline

- 1. Data Collection Load the blood cell image dataset (e.g., BCCD dataset).

  Organize images into folders by cell type (Neutrophil, Lymphocyte, etc.).
- 2. Data Preprocessing Resize all images to 224x224 pixels.

  Normalize pixel values to 0–1.

  Augment the data (rotate, flip, zoom).

  Split into:

  Training Set

  Validation Set

  Test Set
- 3.Model Building (Using Transfer Learning) Load a pre-trained CNN model (e.g., VGG16, ResNet50).

  Remove its original output layer.

  Add new custom layers for blood cell classification.

  Set early layers as non-trainable (freeze them).

  Compile the model with:

  Loss function: categorical crossentropy

Optimizer: Adam

Metrics: accuracy

- 4. Model Training Train the model using the training data.

  Use the validation set to monitor performance.

  Apply EarlyStopping or ModelCheckpoint to avoid overfitting.

- 5. Model Evaluation Test the trained model on the test set.

  Evaluate performance using:

  Accuracy

  Precision

  Recall

  F1-score

  Confusion matrix

- 6. Model Saving Save the trained model for future use (e.g., .h5 file).

- 7. Prediction Load a new blood smear image.

  Preprocess it.

  Use the trained model to predict the type of blood cell.

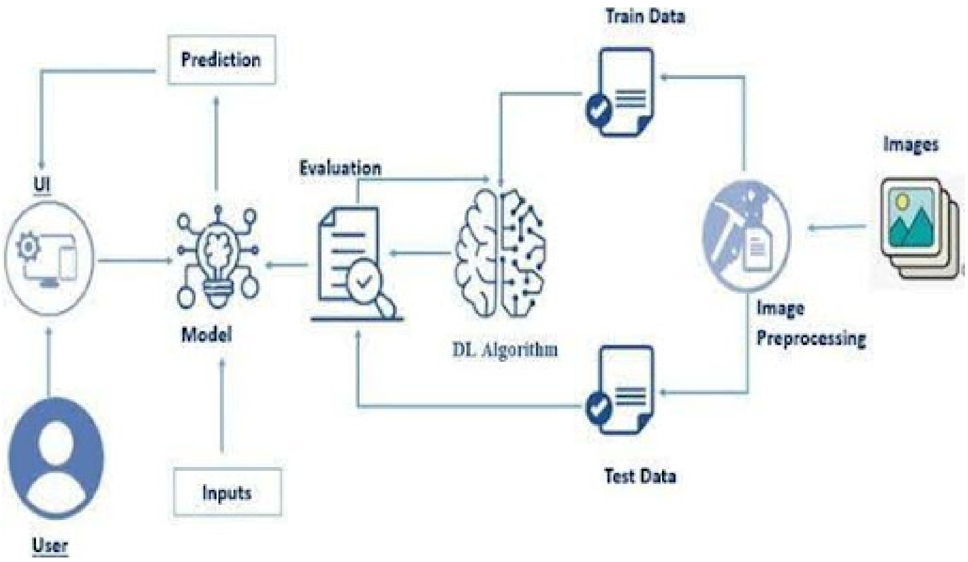


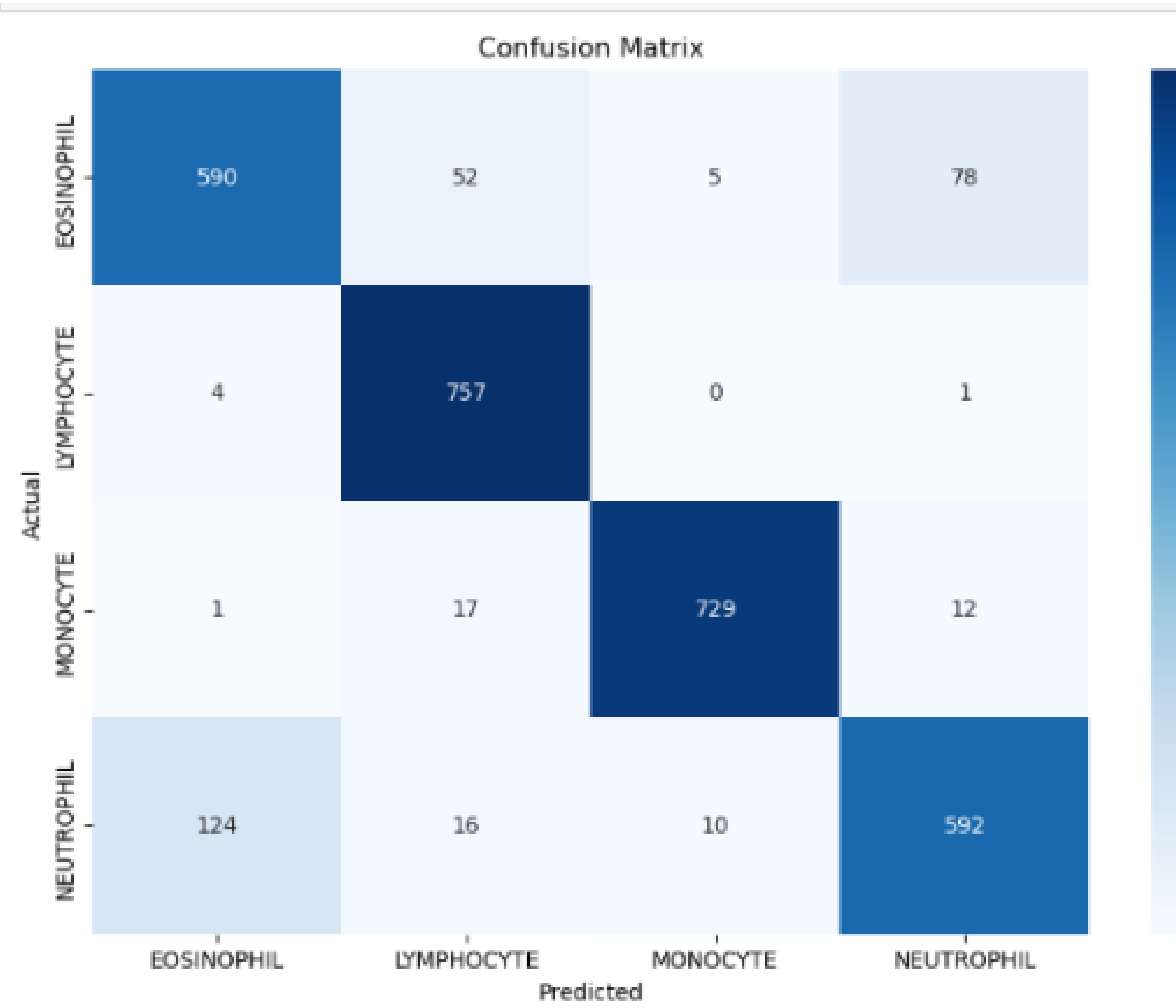
Figure 5: Model arhitecture diagram

7 Results and Evaluation

The MobileNetV2 model achieved high classification accuracy. Key performance metrics:

Table 1: Model Evaluation Metrics

Class	Precision	Recall	F1-score	support
Eosinophil	0.82	0.81	0.78	725
Lymphocyte	0.90	0.99	0.94	762
Monocyte	0.98	0.96	0.97	759
Neutrophil	0.87	0.80	0.83	742
accuracy			0.89	2988
macro avg	0.89	0.89	0.89	2988
weighted avg	0.89	0.89	0.89	2988



Accuracy of the model: 85.0

8 Applications

HematoVision has practical use cases in:

- **Automated Blood Report Analysis**:

  The system can quickly identify and classify white blood cells from microscopic images, helping in generating blood reports faster.

- **Early Disease Detection**:

  Helps in early diagnosis of diseases like leukemia, infections, anemia, and allergic conditions by analyzing abnormal cell types or counts.

- **Medical Training and Education**:

  Used as a teaching tool to help medical students and trainees learn how to identify different types of blood cells using real images.

9 Flask Deployment

Flask is a tool in Python that helps you create a website or web app.

System Flow

1. Upload an image (blood cell image)
2. Use your trained model to predict what type of cell it is
3. Show the result on the webpage
4. Model prediction is generated using Keras.
5. The result is displayed with the image using base64 encoding.

Code Overview

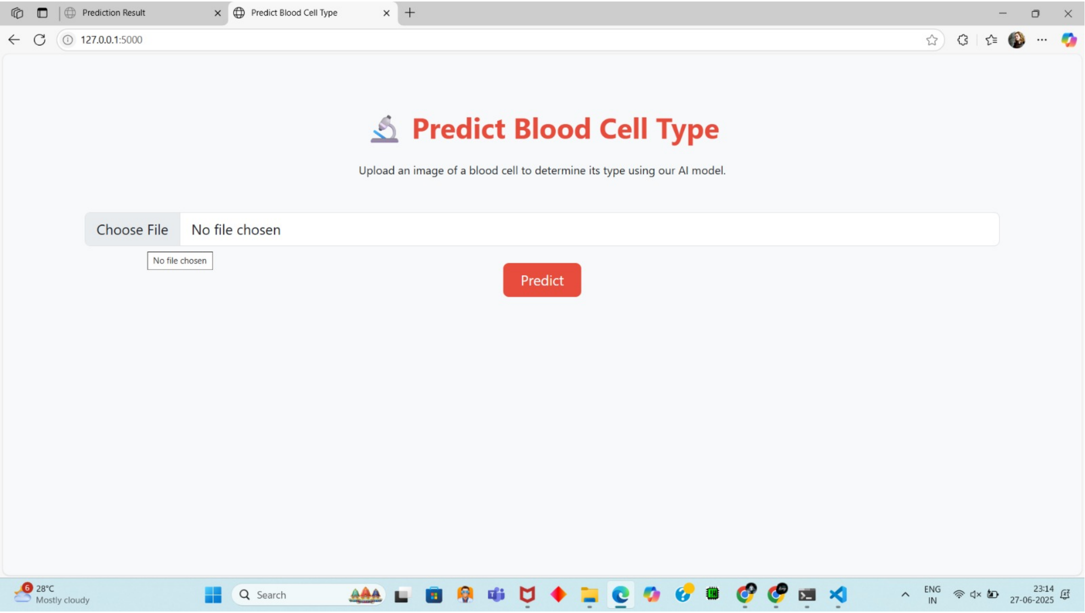
The backend includes:

- `app.py`: Flask logic for file handling and prediction
- `home.html` & `result.html`: Frontend templates
- `blood_cell.h5`: Trained CNN model

Image preprocessing uses OpenCV and MobileNetV2-specific normalization. The final label is inferred using `np.argmax()` and mapped to a class.

Working Demo

Steps



Predict Blood Cell Type

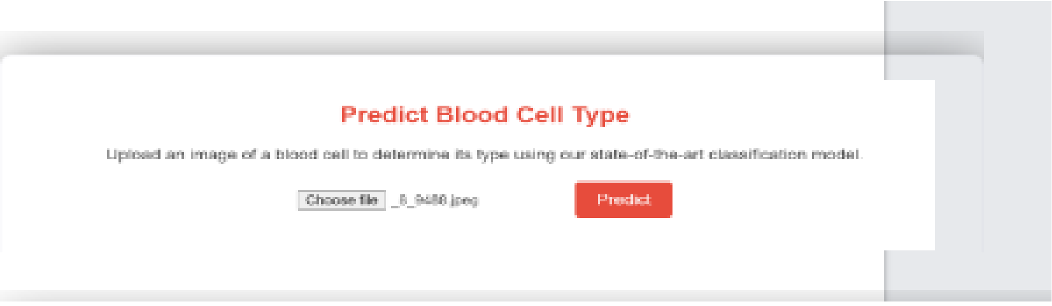
Upload an image of a blood cell to determine its type using our AI model.

Choose File No file chosen

No file chosen

Predict

Figure 6: (a) Opening page of the website



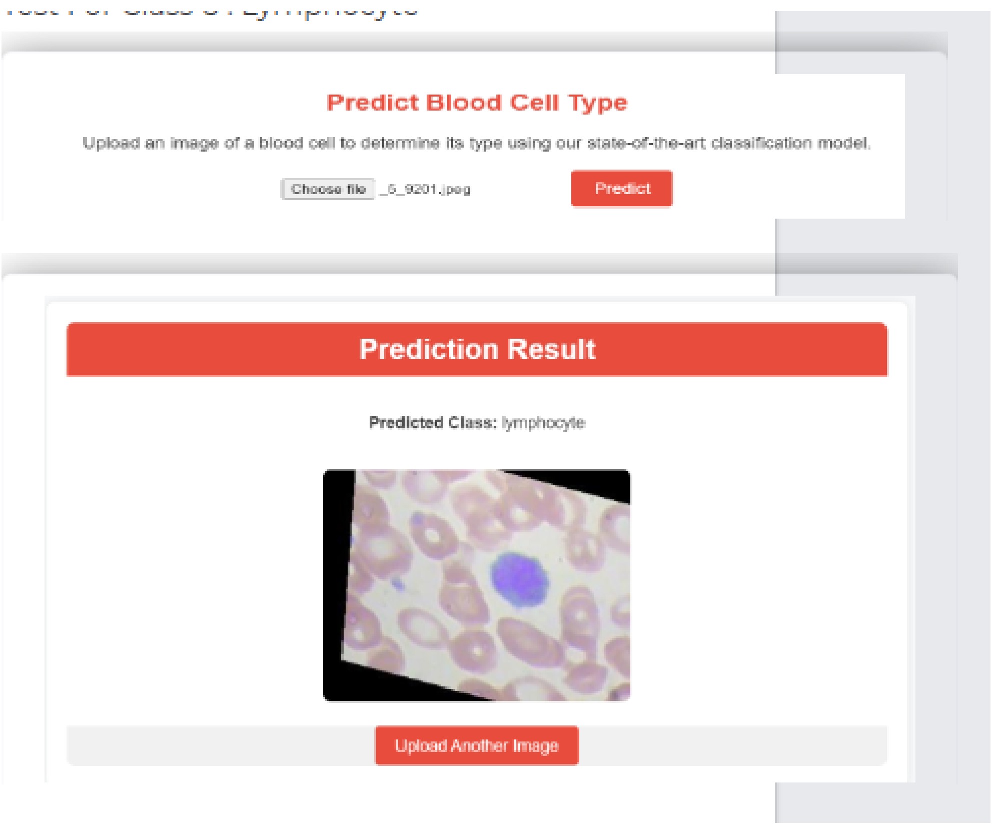
Predict Blood Cell Type

Upload an image of a blood cell to determine its type using our state-of-the-art classification model.

Choose file _6_9488.jpeg

Predict

Figure 7: (b) File uploading section



Predict Blood Cell Type

Upload an image of a blood cell to determine its type using our state-of-the-art classification model.

Prediction Result

Predicted Class: lymphocyte

Figure 8: (c) Updating the output

10 Conclusion and Future Work

In this project, we successfully developed HematoVision, a deep learning-based system for the automatic classification of white blood cells using transfer learning techniques. By utilizing pre-trained models and applying them to microscopic blood smear images, the system achieved efficient and accurate identification of different types of white blood cells: neutrophils, lymphocytes, monocytes, and eosinophils.

The model helps reduce manual effort and improves the speed and consistency of blood cell analysis. The integration with a Flask web application makes it user-friendly and accessible for real-time use in clinical or remote environments.

- Mobile Deployment: Convert the model to work on mobile apps or handheld devices using TensorFlow Lite or ONNX, making it useful in field settings.
- Larger and Diverse Dataset: Train the model on a larger, more diverse dataset to improve generalization across various imaging conditions and laboratories.
- Extend to More Cell Types: Include red blood cells (RBCs), platelets, and abnormal cells like blast cells or leukemia variants for more comprehensive diagnosis.

11 References

- Kaggle Blood Cell Dataset: `https://www.kaggle.com/api/v1/datasets/download/paultimothymooney/blood-cells`
- GitHub Link:

  `https://github.com/sai-santhi/HEMATOVISION-BLOOD-CELL-CLASSIFICATION-USING-TRANS git`