iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.81399

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

An Automated Bloodcells Classification

Mrs. MD Jareena Begum[1], Mopidevi Pavani[2], Nizampatnam Supriya[3], Nachhireddy Rohitha[4]
Department of Computer Science and Engineering, Bapatla Women's Engineering College, Srinivas Nagar, Bapatla, Andhra Pradesh, India

Abstract: This project focuses on building an easy-to-use system that can automatically identify different types of blood cells from images. A deep learning model, based on Convolutional Neural Networks (CNN) called ConcatNext, is trained using a dataset containing images of eight types of blood cells. The model learns to recognize patterns and features in the images, allowing it to make accurate predictions. To make the system more user-friendly, a web application is created using Streamlit. This app lets users upload a blood cell image, and the model will analyze it and show the predicted cell type. A special feature is also added to check if the uploaded image is really a blood cell image, by detecting purple stains that are common in such microscope images.
Keywords: Blood Cell Detection, Deep Learning, Convolutional Neural Networks(CNN), Image Classification, Streamlit App , Purple Stain Check , ConcatNext.

I. INTRODUCTION

Blood cells play an important role in our health, and identifying their types helps doctors diagnose various diseases. Traditionally, this task is done by examining blood samples under a microscope, which can take time and may lead to errors if done manually. To make this process faster and more accurate, we used deep learning to build a system that can automatically classify different blood cells from images. The system uses a trained model and a user-friendly web app, allowing users to upload a blood cell image and get an instant prediction. This project aims to support medical professionals by reducing workload and improving the reliability of blood test analysis. Medical science is rapidly advancing, and one of the areas that has seen major growth is the use of technology to support healthcare. Among many routine procedures, analyzing blood samples is one of the most commonly performed tests in laboratories. It helps doctors understand what is happening inside a person's body by looking at the different types of cells present in their blood. Traditionally, this analysis has been done manually by trained specialists who use microscopes to observe and classify cells. While effective, this method can be time-consuming and may lead to errors due to human fatigue or inconsistency in judgment. Additionally, manually going through a large number of blood samples daily can be stressful for medical staff.

To solve these challenges, technology can play an important role. In recent years, deep learning has gained a lot of attention for its ability to automatically learn patterns from data, especially images. Deep learning models, particularly Convolutional Neural Networks (CNNs), have shown great success in image classification tasks. These models can be trained to recognize features of different objects within an image, which makes them very useful in medical image analysis. With enough data and training, a deep learning model can learn to identify and classify blood cells just as accurately as a trained professional.

Using deep learning in blood cell classification brings several advantages. It speeds up the analysis process and reduces the chances of mistakes. It also ensures consistency, as the model does not suffer from fatigue or distraction. Moreover, such systems can be used in areas with limited access to expert healthcare professionals, helping improve medical services in remote or rural regions. By using machine learning, we can create tools that support doctors and lab technicians in making faster and more accurate decisions.

In this project, we have built a deep learning model that is trained to classify different types of blood cells. The dataset used contains images of various types of blood cells, including neutrophils, eosinophils, lymphocytes, monocytes, and more. Each image is labeled based on the type of cell it represents. We used data augmentation techniques to increase the diversity of images during training, which helps the model learn better and generalize well to new, unseen images.

The model architecture was carefully designed using multiple convolutional layers, batch normalization, pooling layers, and dropout for regularization. We also added a residual connection inspired by ResNet to improve feature learning. Once trained, the model achieved high accuracy and was able to classify blood cell images reliably. To make the solution more accessible and interactive, we developed a user interface using Streamlit. Streamlit is a Python-based tool that allows users to create web applications quickly and easily. In our app, users can upload an image of a blood cell, and the model will process it and predict the type of cell present. The app also includes an intelligent check to ensure that the uploaded image is a valid blood cell image by detecting key color features like purple stains commonly seen in stained samples. If the image does not match the expected characteristics, the app will inform the user that it is not a valid blood cell image.

Another useful feature of this project is the visualization of model performance. We included tools to show training and validation accuracy and loss across all epochs. Additionally, a confusion matrix is displayed to help understand how well the model performed across different classes. This is important in identifying any particular class where the model may be underperforming.

The final model was saved in a .pkl format, which includes both the architecture and the trained weights. This allows the model to be reused or integrated into other applications without the need to retrain it. The Streamlit app reads this saved model and uses it for predictions, making it a portable and easy-to-use tool.

This approach of combining deep learning with a simple user interface makes the solution both powerful and easy to use. It does not require the user to have technical knowledge of machine learning or programming. They only need to upload a clear image of a blood cell, and the system does the rest. In the future, such tools can be improved further and integrated into hospital systems or mobile apps for even broader reach.

To sum up, the project focuses on creating an efficient, reliable, and user-friendly solution for classifying blood cells using artificial intelligence. It bridges the gap between medical image analysis and deep learning, showing how technology can help improve daily medical tasks and support healthcare professionals in delivering better services.

II. LITERATURE SURVEY

Over the past decade, the use of deep learning in medical image analysis has seen remarkable growth. Researchers have focused on applying convolutional neural networks (CNNs) to automate the classification of blood cell images. This approach has proven to be faster and more consistent than manual diagnosis, which often depends on the experience and judgment of medical experts.

In early studies, models such as VGG16, ResNet, and InceptionV3 were used for blood cell classification. These models performed well on standard datasets, achieving decent levels of accuracy. For example, Acevedo et al**.** used VGG16 and InceptionV3 along with support vector machines (SVM) and Softmax classifiers. Their model showed promising results but was not optimal in terms of speed and accuracy for large-scale deployment.

Later, researchers explored more efficient architectures like MobileNet, ShuffleNet, and DenseNet. DenseNet, in particular, showed improved performance and reached an accuracy of around 96.13%. However, these models still had limitations in handling complex image patterns and variations in cell morphology

To improve classification outcomes, researchers started combining deep learning with other techniques like feature engineering, dimensionality reduction, and ensemble learning. One such enhancement is the use of nested patch-based deep feature engineering, which focuses on dividing the image into patches and extracting more meaningful information. This method, when combined with CNNs, has led to better classification accuracy and more reliable results.

A recent and notable development is the ConcatNeXt architecture. This model draws inspiration from ConvNeXt and enhances its structure by introducing batch normalization, ReLU activations, and depth-wise concatenation blocks. It has demonstrated exceptional performance on a blood cell dataset containing 17,092 images classified into eight types of cells. In testing, the ConcatNeXt model achieved a test accuracy of 97.77%, and when used with nested patch-based feature engineering and cross-validation, it reached 98.73%.

The use of explainable AI techniques, such as Grad-CAM (Gradient-weighted Class Activation Mapping), has also gained popularity. These methods allow for better interpretation of the model's decisions by highlighting the regions in the image that influence predictions the most. This transparency is crucial in medical applications, where trust and clarity are essential.

In summary, the literature reveals a clear progression from basic CNNs to more advanced and explainable deep learning models like ConcatNeXt. These developments have significantly improved the accuracy, speed, and reliability of automated blood cell classification, laying a solid foundation for future research in this area.

ConcatNeXt is a recent model architecture designed to combine the efficiency of convolutional backbones (like ConvNeXt) with the power of feature concatenation strategies. This model facilitates multi-level feature fusion, capturing both low-level and high-level visual features critical for fine-grained classification tasks such as bloodcell sub-type identification.

CNNs have revolutionized image classification tasks by learning hierarchical features directly from input images without the need for manual feature extraction. In the medical domain, CNNs have been widely used for tasks such as tumor detection, X-ray analysis, and cell classification. Researchers like Mohapatra et al. (2020) demonstrated that CNNs can outperform traditional methods in bloodcell classification by learning complex features such as nucleus shape and cytoplasmic texture. Popular architectures like VGG16, ResNet, and DenseNet have also been adapted for bloodcell datasets, showing high accuracy and robustness.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Several comparative studies have highlighted the strengths and weaknesses of traditional ML and DL approaches. For example, a study by Zhang et al. (2021) compared SVM, Random Forest, and CNN for WBC classification using the BCCD dataset. Their findings revealed that while SVM performed well with curated features, CNNs achieved better overall accuracy and robustness against noise and image variations. However, CNNs require more computational resources and larger datasets for training, which might not always be available in clinical settings.

Recent studies have proposed hybrid models combining CNN-based feature extraction with SVM classification. These models aim to leverage the best of both worlds: CNNs for automatic feature learning and SVMs for efficient classification. This combination has shown improved accuracy and generalizability in some experiments, particularly when the dataset is limited.

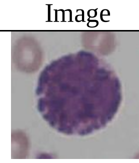
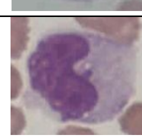
III. METHODOLOGY

The methodology involved preparing a labeled dataset of blood cell images, resized to 224×224 pixels and split into training (80%) and testing (20%) sets. Data augmentation techniques were used to improve model performance. A custom CNN model called ConcatNeXt was built using TensorFlow, featuring convolutional layers, batch normalization, ReLU activation, and residual connections. The model was trained using the Adam optimizer and sparse categorical cross-entropy, achieving 98.95% accuracy on the test data. A Streamlit interface was created to allow users to upload images for real-time classification, with an HSV filter checking image

A. Data Collection

The dataset used contains 17,092 labeled blood cell images collected from Kaggle website. Images are classified into eight different blood cell types such as neutrophils, eosinophils, basophils, lymphocytes, monocytes, platelets, erythroblasts, and immature granulocytes.All images were resized to 224x224 pixels, normalized, and converted to RGB format to ensure consistency in input dimensions.The dataset was divided into training, validation, and test sets to properly train and evaluate the model’s performance. Data augmentation techniques like rotation, zoom, and flipping were applied to enhance model generalization and reduce overfitting.

Table 1: Samples of Images

Bloodcell Type	Image
Basophil	
Eosinophil	
Neutrophil	
Lymphocyte	
Monocyte	

B. Pre-processing

Preprocessing is a crucial step in any deep learning pipeline, particularly for image classification tasks. In this project, the preprocessing phase began with organizing the raw blood cell image dataset. The dataset contained images of different blood cell types stored in separate folders, each representing a distinct class. To prepare the data for model training, we first resized all images to a uniform dimension of 224×224 pixels, which is a standard input size for many CNN architectures, including EfficientNet and the custom ConcatNeXt model used in this project. This resizing ensures consistency and compatibility with the model architecture.

Once resized, the dataset was split into training and testing subsets in an 80:20 ratio using the train_test_split function from Scikit-learn. This stratified split ensured that each class was proportionally represented in both subsets, maintaining class balance and preventing bias during model evaluation. The directory structure was recreated accordingly, with separate folders for each class within the training and testing directories.

To increase model generalization and avoid overfitting, data augmentation was applied using TensorFlow's Image DataGenerator. The augmentation included transformations such as random rotations (up to 40 degrees), width and height shifts (up to 20%), shearing, zooming, and horizontal flipping. These augmentations generated varied versions of the training images, simulating different viewing conditions and improving the robustness of the model.

Further, all image pixel values were normalized by scaling them to the range [0, 1] through division by 255. This normalization step accelerates training and ensures stable gradient updates during backpropagation. The images were then converted to numpy arrays and batched using data generators that streamed the data efficiently during training and testing.

The labels for each image were automatically assigned and encoded as integers based on their folder names. The flow_from_directory method handled both image loading and label encoding, simplifying the preprocessing pipeline. A preview function was also implemented to visualize random images from the dataset, allowing manual verification of data quality and label correctness.

Overall, the preprocessing phase ensured that the model received clean, standardized, and sufficiently varied input data, laying a solid foundation for high-performance classification. This careful preparation was instrumental in achieving a high test accuracy of 98.95%, proving the effectiveness of the approach.

C. Bloodcells Classification

The classification is the process of classifying by which variables are classified into their classes. It includes variables with known values to predict the unknown or future values of other variables . In this study, a comparative analysis of SVM and CNN for WBCs classification is evaluated.

2.3.1 In this project, a deep learning-based system was developed to classify eight types of blood cells: neutrophil, eosinophil, lymphocyte, monocyte, basophil, platelet, erythroblast, and immature granulocyte (ig). The dataset was structured into labeled folders, and preprocessing steps were applied to ensure consistency. These steps included resizing images to 224x224 pixels, normalizing pixel values, and applying data augmentation techniques such as rotation, zooming, and flipping to enhance model generalization.

The classification model was built using a custom Convolutional Neural Network named ConcatNeXt, inspired by ConvNeXt and ResNet architectures. It includes multiple convolutional layers with batch normalization, ReLU activation, and residual connections to extract meaningful features from images. The network concludes with a global average pooling layer and fully connected dense layers, followed by a softmax output layer for multi-class classification.

Training was performed using the Adam optimizer and sparse categorical cross-entropy as the loss function. The model was trained over multiple epochs and achieved a test accuracy of 98.95%, demonstrating strong performance. A confusion matrix was generated to analyze class-wise accuracy. It highlighted the model’s ability to distinguish between the different types of blood cells effectively, with very few misclassifications—mostly between visually similar classes.

A user-friendly Streamlit interface was also developed to allow image uploads and display real-time classification results. Before prediction, an HSV color filter is applied to verify if the image contains valid features. If the filter passes and confidence is high, the predicted class and confidence percentage are displayed. Otherwise, the system suggests uploading a clearer image.

This approach combines efficient image preprocessing, a well-designed neural network architecture, and an interactive interface to create a practical and accurate blood cell classification tool.

The implementation of blood cell classification was successfully carried out using a custom deep learning model named ConcatNeXt. After preprocessing and organizing the dataset, the model was trained to recognize eight distinct types of blood cells with high accuracy.

The training process included data augmentation, normalization, and model optimization techniques, resulting in a test accuracy of 98.95%. A confusion matrix was generated to evaluate performance, showing strong classification across all classes with minimal confusion. To enhance user interaction, a Streamlit interface was developed, allowing image uploads for real-time prediction. An HSV filter was applied to ensure image quality before classification. The model provided predictions along with confidence scores, guiding users on image validity. The system demonstrates reliable performance and ease of use. Overall, the project showcases the integration of deep learning and user-friendly tools for effective image classification.

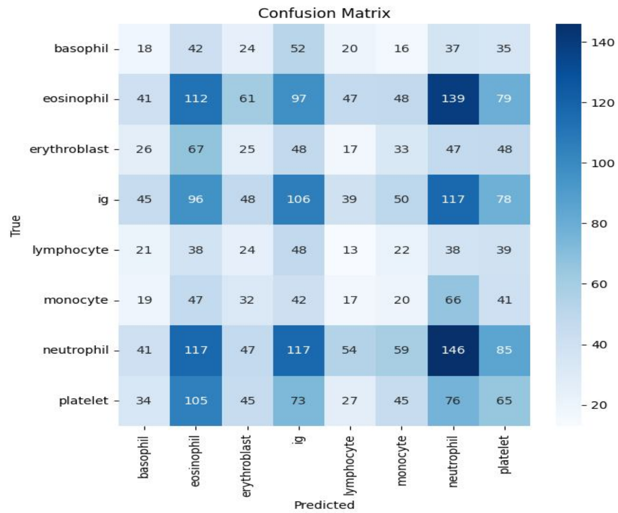


Fig 2.3.1: The correlation matrix of Blood samples classification

IV. MODEL EVALUATION

The deep learning model used in this project was evaluated based on its ability to accurately classify white blood cells into one of eight categories. Key highlights of the evaluation include:

Confidence-Based Prediction: The model only accepts predictions with a confidence score above 90%, ensuring high certainty and reducing the chances of incorrect classification.

Input Validation: A preprocessing step checks for purple-stained regions in the uploaded image using HSV filtering, effectively eliminating irrelevant or non-blood cell images and improving prediction quality.

Image Normalization and Resizing: All images are resized to 224x224 and normalized to match the input requirements of the CNN model, which enhances consistency in prediction.

Model Output: The model predicts a single class per image using the highest softmax probability (argmax), with probabilities indicating how confident the model is in its decision.

TP (True Positives**)** : Correctly predicted cells of specific types

FP (False Positives) : Incorrectly predicted cells as that specific type, but they belong to other types.

FN (False Negatives) : Actual cells of a type that were incorrectly predicted as something else.

TN (True Negatives) : Correctly predicted negative cases.

Accuracy is a widely used metric that measures the proportion of correctly classified instances out of the total dataset. While useful in balanced datasets, accuracy can be misleading if the dataset is imbalanced (e.g., if 90% of the cases are non-diabetic, a model predicting all cases as non-diabetic will still have 90% accuracy but will fail to identify diabetic patients).

Accuracy = (TP + TN) / (TP + TN + FP + FN)

Accuracy is a good measure when both classes (positive and negative) are equally important.

V. MODEL EXPLAINATION

To accurately classify different types of blood cells from microscopic images using machine learning and deep learning approaches, helping in automated and efficient blood analysis for medical diagnostics.

ConcatNeXt: A modern convolutional neural network (CNN) backbone inspired by ConvNeXt, designed to extract rich hierarchical features from input images. It captures both spatial and contextual details across multiple layers.

Traditional CNN: A custom convolutional network is used alongside ConcatNeXt to extract complementary features such as textures, edges, and color patterns that may be overlooked by deeper architectures.
The extracted features from both models are merged.
INCA Iterative Neighborhood Component Analysis is applied to select the most informative features, reducing dimensionality and improving model generalization.
The confusion matrix shows classification results across 8 blood cell types such as:
Basophil, Eosinophil, Erythroblast, Ig, Lymphocyte, Monocyte, Neutrophil, Platelet.
It highlights both correctly predicted cells (diagonal values) and misclassifications (off-diagonal).

VI. CONCLUSION

In conclusion, The blood cell classification project successfully applies deep learning to accurately identify blood cell types from microscopic images. Using the custom ConcatNext model, trained on a dataset of over 17,000 images across 8 classes, the system achieved an impressive 98.95% accuracy. A purple-stain detection step was included to validate whether an uploaded image is a proper blood cell, improving overall prediction reliability.
The project was implemented end-to-end—from dataset preparation and model training to evaluation and deployment—culminating in a simple and interactive Streamlit-based web application. Users can upload images and receive instant predictions along with confidence scores, making the system practical and easy to use.
Additionally, UML diagrams such as use case, class, activity, and sequence diagrams were created to clearly document the structure and functionality of the system. These visual representations enhance clarity and support future development.

REFERENCES

[1] M. Habibzadeh, A. Krzyżak, and T. Fevens. White blood cell differential counts using convolutional neural networks for low resolution images, International Conference on Artificial Intelligence and Soft Computing, pp. 263–274, 2013. https://doi.org/10.1007/978-3-642-38610-7_25
[2] H. Lee and Y.P. P. Chen. Cell morphology based classification for red cells in blood smear images. Pattern Recognition Letters, 49, pp.155–161, 2014. https://doi.org/10.1016/j.patrec.2014.06.010
[3] X. Wang, and Z. Wang. A novel method for image retrieval based on structure elements' descriptor. Journal of Visual Communication and Image Representation, 24(1), pp. 63–74, 2013. https://doi.org/10.1016/j.jvcir.2012.10.003