History of Medicine, 2021, 7(1): 71-84

DOI:10.48047.v7.1.2021.17

# Deep Learning Approaches for Enhanced White Blood Cell Subtype Classification

Sai Kumar Rapolu[1], Chenagoni Nagaraju[1], Uma Rani Koppula[1]
[1]Department of Computer Science and Engineering

[1]Sree Dattha Group of Institutions, Sheriguda, Telangana.

**ABSTRACT**

White blood cells (WBCs), or leukocytes, are a crucial component of the immune system, characterized by their nuclei and absence of hemoglobin. They play a vital role in defending the body against foreign microorganisms, such as bacteria and viruses, through processes like phagocytosis and antibody production. Leukocytes are categorized into five main types: neutrophils, eosinophils, lymphocytes, monocytes, and basophils. Neutrophils, the most abundant type, are primarily responsible for combating bacterial and fungal infections. Eosinophils (2–4% of WBCs) respond to allergies and parasitic infections, while lymphocytes are essential for the specific recognition and elimination of foreign agents. Monocytes facilitate the direct destruction of pathogens and assist in debris cleanup at infection sites.

The differential counting of WBCs is a critical diagnostic tool in clinical settings, serving as an indicator of hidden infections and alerting hematologists to conditions such as leukocytosis, which signifies an abnormal increase in WBC count. This counting also aids in monitoring the effectiveness of cancer treatments like chemotherapy and radiation. However, the manual differentiation of leukocytes under a microscope is labor-intensive and inefficient, highlighting the need for automated classification based on WBC images.

Typically, automated classification involves several key steps: preprocessing, segmentation, feature extraction, and classification. Preprocessing aims to eliminate noise and artifacts to enhance image contrast. Segmentation involves isolating WBCs from the background in smear images, followed by creating representative feature vectors for each WBC type. Classification is performed based on these vectors, with hematologists sometimes assisting in feature determination. Traditional classification methods often suffer from time inefficiencies and accuracy compromises.

In recent years, deep learning has revolutionized various applications, particularly in computer vision. Convolutional neural networks (CNNs) have gained prominence for image classification tasks, leveraging raw images as inputs to learn hierarchical feature representations in an end-to-end manner. This project aims to implement a deep learning-based approach using CNNs for the detection and classification of subtype blood cells.

**Keywords:** WBC classification, Historical images, CNN.

## 1. INTRODUCTION

Blood is a specialized body fluid. Its main components are red blood cells, plasma, platelets, and white blood cells. WBCs protect the body from infections, accounting for about 1% of human blood [1]. Basophils, Eosinophils, Lymphocytes, Monocytes, and Neutrophils are the types of white blood cells, Basophils white blood cells accounting only around 1%, they are important in mounting a nonspecific immune response to pathogens. Eosinophils play an important role in fighting bacteria and responding to infections with parasites. Lymphocytes are also very important in the immune system, they are 2 types: B and T lymphocytes, with B cells producing antibodies, T cells being responsible for directly killing many foreign invaders. Monocytes are responsible for cleaning up dead cells. Roughly half of the white blood cells are Neutrophils; they are usually the first cells of the immune system to respond to an invader such as a bacterium or a virus [2]. White blood cells (WBCs) classification is an important step because it can assist hematologists in the diagnosis of several blood disorders, such as leukemia, some immunological disorders, and certain types of cancer. The analysis procedure can be done by automatic and manual approaches to count and classify WBC. The manual classification of WBC has many medical difficulties, including error in the accuracy of results due to sampling errors and statistical probabilities and poor sensitivity, specificity, and predictive values. Furthermore, some automatic approaches in the laboratories have used instruments, such as flow cytometry and automatic counting machine to detect and classify WBC. These instruments do not make use of image processing techniques, and they can count and classify WBCs quantitatively not qualitatively. Therefore, it is necessary to design an automatic system which includes computer-based systems for classification of WBCs.

Researchers are increasingly interested in the development of algorithms for automated analysis of medical images such as microscopic blood smear images. They are using different correlated techniques like; image processing, computer vision, artificial neural networks, machine learning algorithms, etc. [3]. To overcome all these problems, we added the help of CNN to image processing. In this framework, we present a RBC image analysis with the convolution neural network (CNN). CNN is a strong image classifer tool, in which image is taken as input, classify it under certain categories based on their features. In CNN, an individual unit is called a neuron. Neurons are in a series of layers. Neurons of one layer are connected to the neurons of the next layer. Each neuron or node of one layer perform mathematical calculation and pass the results to the next node. The last layer of the neural network has increased computational power due to the accumulation of experience.

## 2. LITERATURE SURVEY

Alzubaidi et al. [4] introduced a new robust and effective deep Convolutional Neural Network to classify Red Blood Cells (RBCs) in three classes namely: normal (‘N’) abnormal (sickle cells anemia type (‘S’)) and miscellaneous (‘M’). To improve the results further, we have used this model as features extractor then this work applied an error-correcting output codes (ECOC) classifier for the classification task. This model with ECOC showed outstanding performance and high accuracy of 92.06%.

Rahman et al. [5] experimented the existing standard pre-processing techniques from the literature. In addition, several other complex architectures have been implemented and tested to pick the best performing model. A holdout test has also been conducted to verify how well the proposed model generalizes on unseen data. This best model achieved an accuracy of almost 97.77%.

Roopa et al. [6] demonstrated classification of white blood cells into six types namely lymphocytes, monocytes, neutrophils, eosinophils, basophils and abnormal cells. This work provided the comparison of traditional image processing approach and deep learning methods for classification of white blood cells. This work also evaluated neural network classifier results for hand-crafted features and obtained the average accuracy of 99.8%. And used full training and transfer learning approaches of convolutional neural network for the classification. An accuracy around 99% was obtained for full training CNN.

Malkawi et al. [7] classified the microscopic WBCs images using a hybrid system where Convolutional Neural Network (CNN) used as features extractor and different machine learning algorithms used as classifiers, then the performances of these classifiers were evaluated to recognize the best of them. These algorithms included Support Vector Machine (SVM), k-Nearest Neighbor (KNN) and Random Forest, for training and test parameters this framework used five features that were extracted from the images. According to results of performance, the RF performed better than the other methods with a testing accuracy reached 98.7%.

Matek et al. [8] compiled an annotated image dataset of over 18,000 white blood cells, use it to train a convolutional neural network for leukocyte classification and evaluate the network's performance by comparing to inter- and intra-expert variability. The network classified the most important cell types with high accuracy. It also allows us to decide two clinically relevant questions with human-level performance: (1) if a given cell has blast character and (2) if it belongs to the cell types normally present in non-pathological blood smears. This framework approach holds the potential to be used as a classification aid for examining much larger numbers of cells in a smear than can usually be done by a human expert. This will allow clinicians to recognize malignant cell populations with lower prevalence at an earlier stage of the disease.

Sadafi et al. [9] presented an active learning framework that identifies the most relevant samples from a large set of not annotated data for further expert annotation. Applied to brightfield images of red blood cells with seven subtypes, this work trained a faster R-CNN for single cell identification and classification, calculate a novel confidence score using dropout variational inference and select relevant images for annotation based on (i) the confidence of the single cell detection and (ii) the rareness of the classes contained in the image. This framework showed that this approach leads to a drastic increase of prediction accuracy with already few annotated images. This original approach improves classification of red blood cell subtypes and speeds up the annotation. This important step in diagnosing blood diseases will profit from our framework as well as many other clinical challenges that suffer from the lack of annotated training data.

Parab et al. [10] utilized the algorithm which can extract the feature of each segmented cell image and classify it into 9 various types. Images of blood slides were collected from the hospital. The overall accuracy was 98.5%. The system has been developed to provide accurate and fast results that can save patients' lives.

Paravil et al. [11] tried to devise a methodology for automation by using feature fusion. For feature extraction, various fusion techniques using transfer-learning approaches such as Densely connected convoluted neural networks (DenseNet201) and VGG16 (Visual Geometry Group 2016) were proposed. The classification results are compared using various performance metrics such as Accuracy, Precision, Recall, and F1-Score. The maximum accuracy of 89.75% was obtained with the help of feature fusion combined with the Convolutional Neural Network (CNN) classifier.

Yildirim et al. [12] proposed one of the most popular neural networks, convolutional neural network (CNN) is selected to differentiate between different types of white blood cells, namely, eosinophil, lymphocyte, monocyte and neutrophil. The CNN was coupled with Alexnet, Resnet50, Densenet201 and GoogleNet in turn, and trained with the Kaggle Dataset. Then, Gaussian, and median filters were applied separately to the images in the database. The new images were classified again by the CNN with each of the four networks. The results obtained after applying the two filters to the images were better than the results obtained with the original data. The research results make it easier to diagnose blood related diseases.

### 3. PROPOSED SYSTEM

To implement this project, we have designed following modules

1) Upload WBC Dataset: using this module we will upload entire dataset to application

2) Preprocess Dataset: using this module we will read train and test images and then resize all images to equal size, shuffle and normalize images
3) Train Decision Tree: using this module we will train decision tree algorithm on training dataset and then test its performance using TEST images
4) Train Deep CNN model: using this module we will train CNN algorithm on training dataset and then test its performance using TEST images
5) Classification: using this module we will input test images and then CNN will classify its subtype blood cell.
6) Performance Evaluation: using this module we will plot accuracy comparison graph between both algorithms

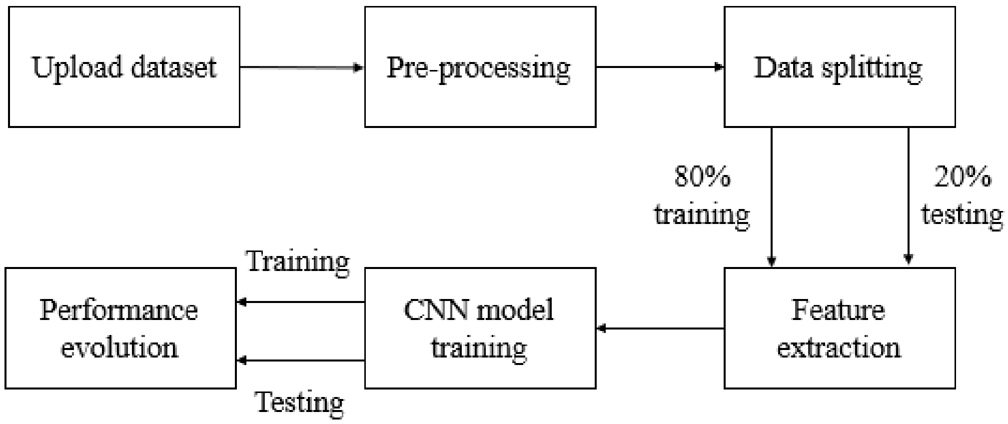



Fig. 1: Block diagram of proposed system.

**3.1 Pre-processing**

Data pre-processing is a process of preparing the raw data and making it suitable for a machine learning model. It is the first and crucial step while creating a machine learning model.

When creating a project, it is not always a case that we come across the clean and formatted data. And while doing any operation with data, it is mandatory to clean it and put in a formatted way. So, for this, we use data pre-processing task.

***Why do we need Data Pre-processing?***

A real-world data generally contains noises, missing values, and maybe in an unusable format which cannot be directly used for machine learning models. Data pre-processing is required tasks for cleaning the data and making it suitable for a machine learning model which also increases the accuracy and efficiency of a machine learning model.

- Getting the dataset
- Importing libraries
- Importing datasets
- Finding Missing Data
- Encoding Categorical Data
- Splitting dataset into training and test set
- Feature scaling

**3.1.1 Splitting the Dataset into the Training set and Test set**

In machine learning data pre-processing, we divide our dataset into a training set and test set. This is one of the crucial steps of data pre-processing as by doing this, we can enhance the performance of our machine learning model.

Supposeif we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models.

If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So we always try to make a machine learning model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:

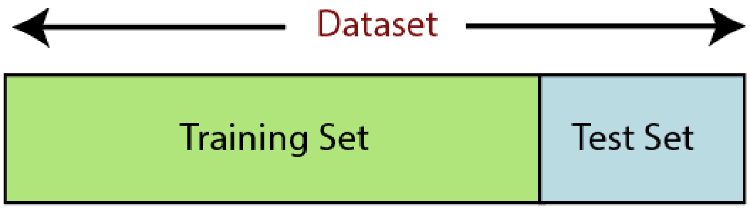



Fig. 2: Data splitting.

**Training Set**: A subset of dataset to train the machine learning model, and we already know the output.

**Test set**: A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

### 3.2 DL-CNN

According to the facts, training and testing of any deep neural network or transfer learning involves in allowing every source image via a succession of convolution layers by a kernel or filter, rectified linear unit (ReLU), max pooling, fully connected layer and utilize SoftMax layer with classification layer to categorize the objects with probabilistic values ranging from [0,1].

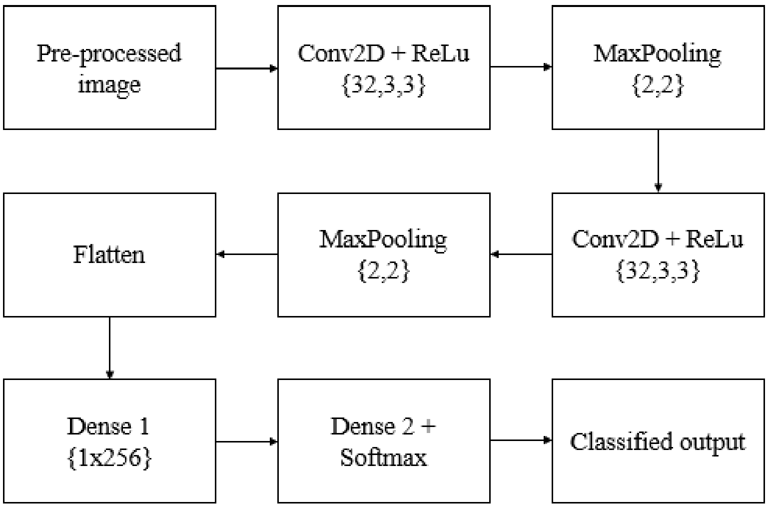



Fig. 3: CNN architecture.

Convolution layer as is the primary layer to extract the features from a source image and maintains the relationship between pixels by learning the features of image by employing tiny blocks of source data. It's a mathematical function which considers two inputs like source image $I(x, y, d)$ where $x$ and $y$

denotes the spatial coordinates i.e., number of rows and columns. $d$ is denoted as dimension of an image (here $d = 3$, since the source image is RGB) and a filter or kernel with similar size of input image and can be denoted as $F(k_x, k_y, d)$.

The output obtained from convolution process of input image and filter has a size of $C\left((x - k_x + 1), (y - k_y + 1), 1\right)$, which is referred as feature map. Let us assume an input image with a size of $5 \times 5$ and the filter having the size of $3 \times 3$. The feature map of input image is obtained by multiplying the input image values with the filter values.

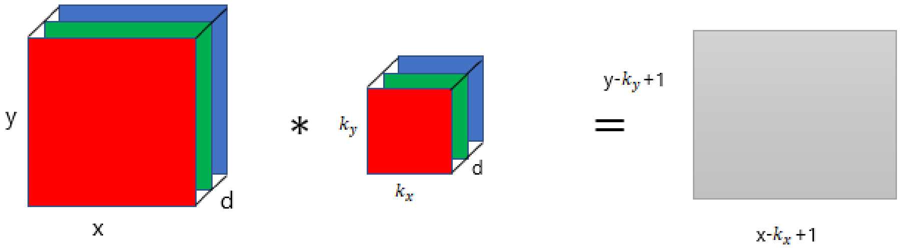



Fig. 4: Representation of convolution layer process.

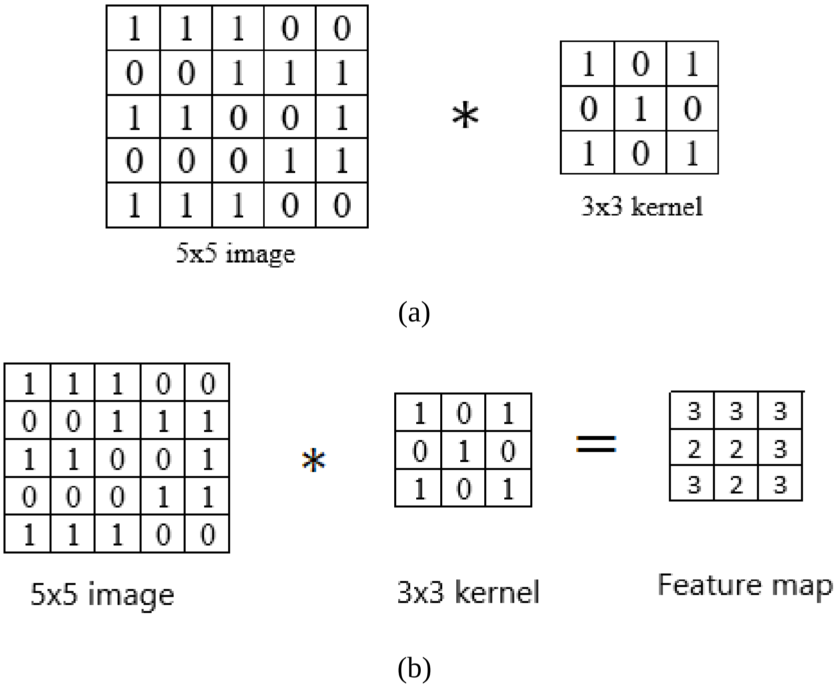



Fig. 5: Example of convolution layer process (a) an image with size $\mathbf{5} \times \mathbf{5}$ is convolving with $\mathbf{3} \times \mathbf{3}$ kernel (b) Convolved feature map.

### 3.2.1 ReLU layer

Networks those utilizes the rectifier operation for the hidden layers are cited as rectified linear unit (ReLU). This ReLU function $\mathcal{G}(\cdot)$ is a simple computation that returns the value given as input directly if the value of input is greater than zero else returns zero. This can be represented as mathematically using the function $max(\cdot)$ over the set of 0 and the input $x$ as follows:

$$\mathcal{G}(x) = \max\{0, x\}$$

### 3.2.2 Max pooing layer

This layer mitigates the number of parameters when there are larger size images. This can be called as subsampling or down sampling that mitigates the dimensionality of every feature map by preserving the important information. Max pooling considers the maximum element form the rectified feature map.

**3.2.3 Softmax classifier**

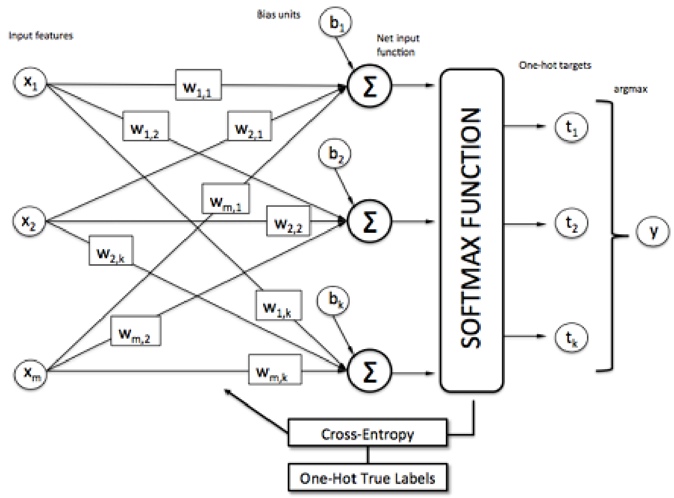



Fig. 6: WBC classification using SoftMax classifier.

Generally, as seen in the above picture softmax function is added at the end of the output since it is the place where the nodes are meet finally and thus, they can be classified. Here, X is the input of all the models and the layers between X and Y are the hidden layers and the data is passed from X to all the layers and Received by Y. Suppose, we have 10 classes, and we predict for which class the given input belongs to. So, for this what we do is allot each class with a particular predicted output. Which means that we have 10 outputs corresponding to 10 different class and predict the class by the highest probability it has.

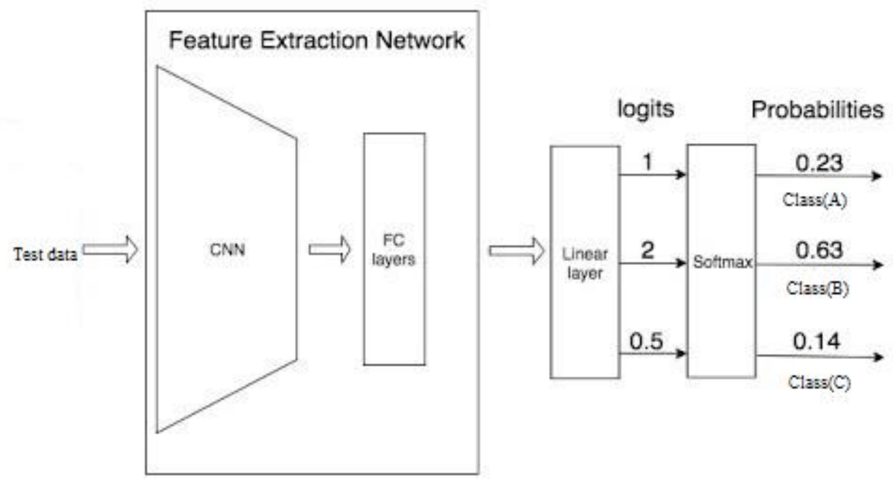



Fig. 7: Example of SoftMax classifier.

In Fig. 7, and we must predict what is the object that is present in the picture. In the normal case, we predict whether the crop is A. But in this case, we must predict what is the object that is present in the picture. This is the place where softmax comes in handy. As the model is already trained on some data.

So, as soon as the picture is given, the model processes the pictures, send it to the hidden layers and then finally send to softmax for classifying the picture. The softmax uses a One-Hot encoding Technique to calculate the cross-entropy loss and get the max. One-Hot Encoding is the technique that is used to categorize the data. In the previous example, if softmax predicts that the object is class A then the One-Hot Encoding for:

Class A will be [1 0 0]

Class B will be [0 1 0]

Class C will be [0 0 1]

From the diagram, we see that the predictions are occurred. But generally, we don't know the predictions. But the machine must choose the correct predicted object. So, for machine to identify an object correctly, it uses a function called cross-entropy function.

So, we choose more similar value by using the below cross-entropy formula.

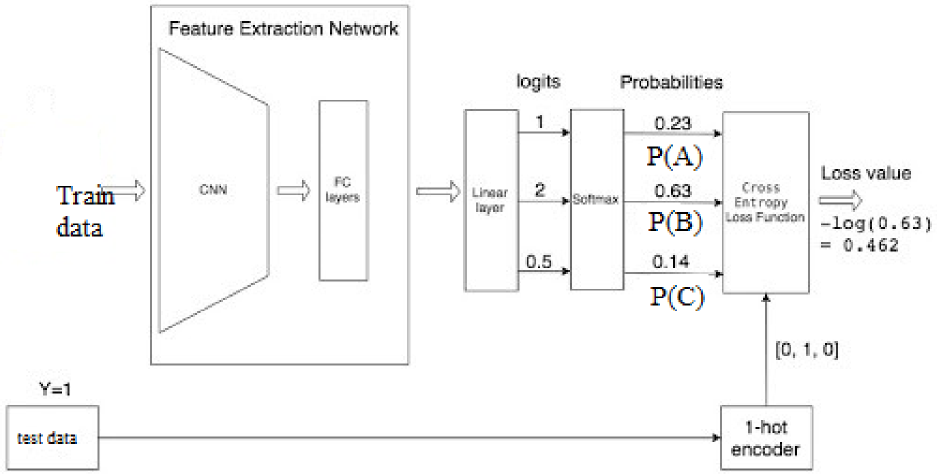



Fig. 8: Example of SoftMax classifier with test data.

In the above example we see that 0.462 is the loss of the function for class specific classifier. In the same way, we find loss for remaining classifiers. The lowest the loss function, the better the prediction is. The mathematical representation for loss function can be represented as: -

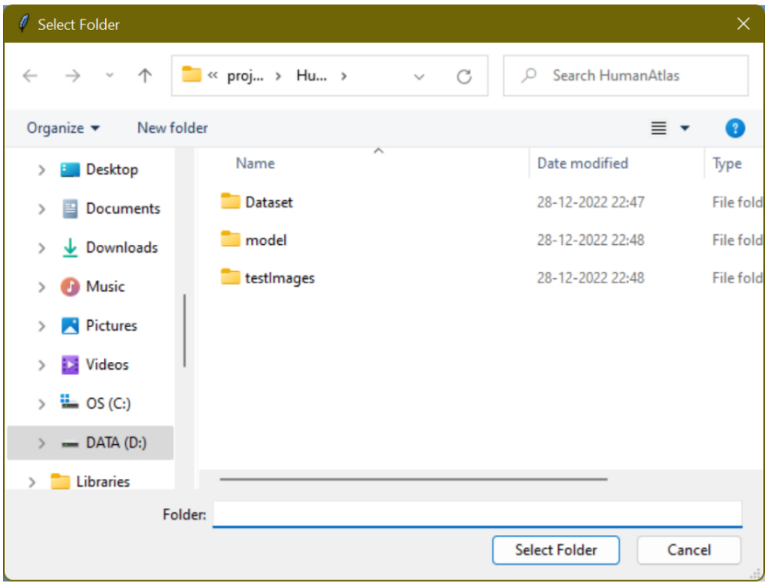
$$LOSS = np.sum(-Y \ * \ np.log(Y\_pred)$$

**Advantages of proposed system**

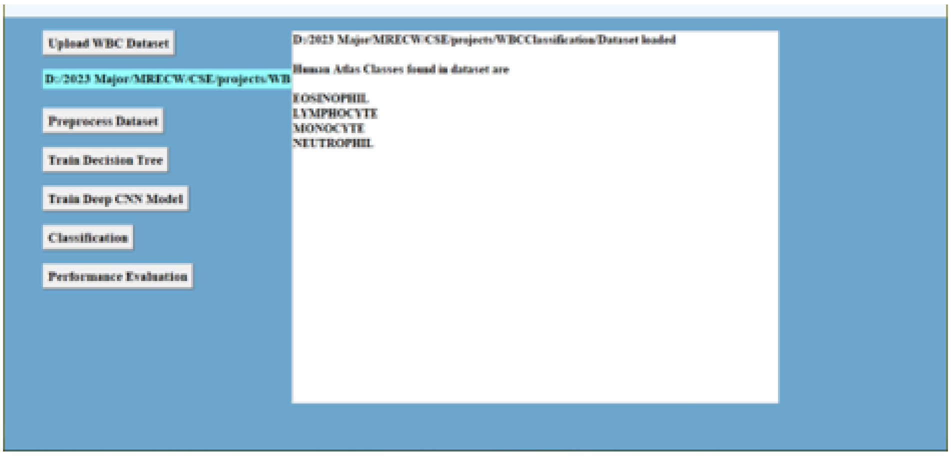
- CNNs do not require human supervision for the task of identifying important features.
- They are very accurate at image recognition and classification.
- Weight sharing is another major advantage of CNNs.
- Convolutional neural networks also minimize computation in comparison with a regular neural network.
- CNNs make use of the same knowledge across all image locations.

## 4. RESULTS AND DISCUSSION

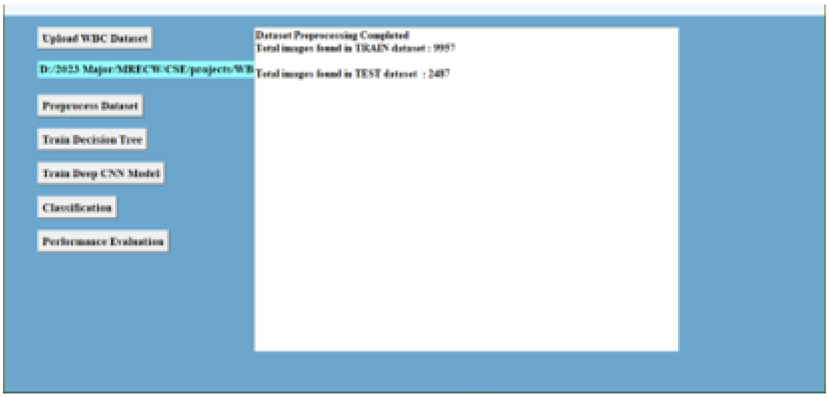

In above screen click on ‘Upload WBC Dataset’ button to load dataset and get below output



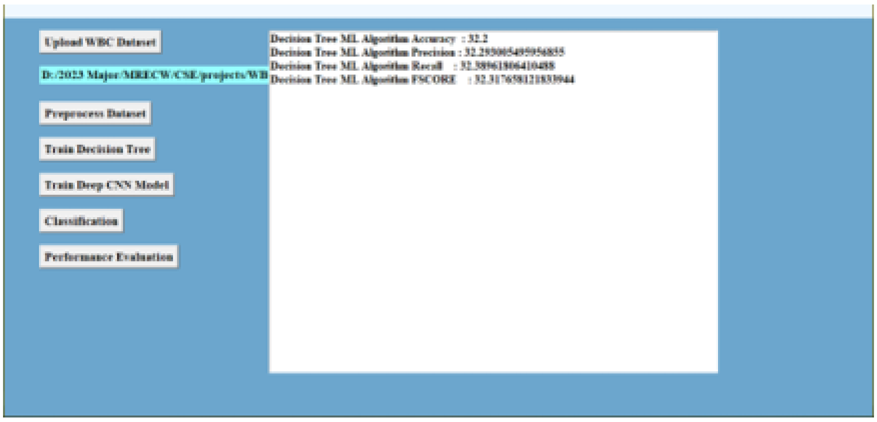

In above screen selecting and uploading ‘Dataset’ folder and then click on ‘Select Folder’ button to load dataset and get below output



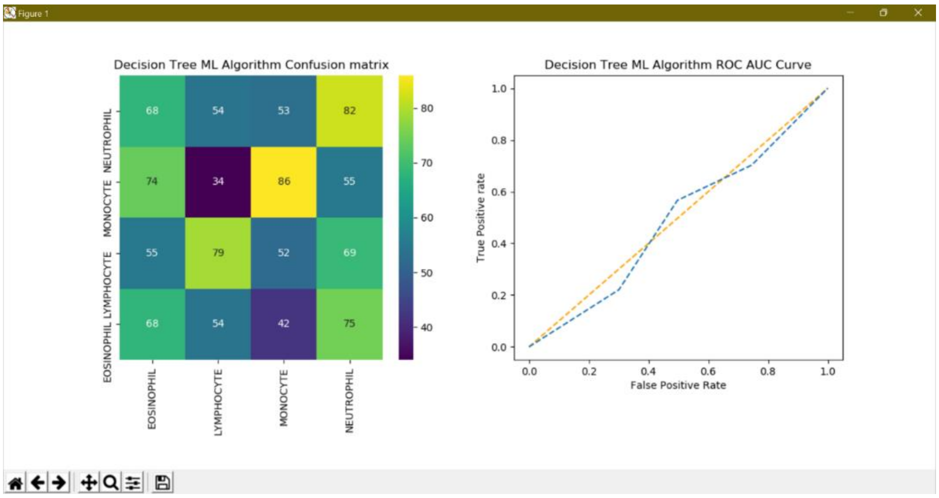

In above screen dataset loaded and then displaying different classes found in dataset and now click on ‘Preprocess Dataset’ button to process images and get below output



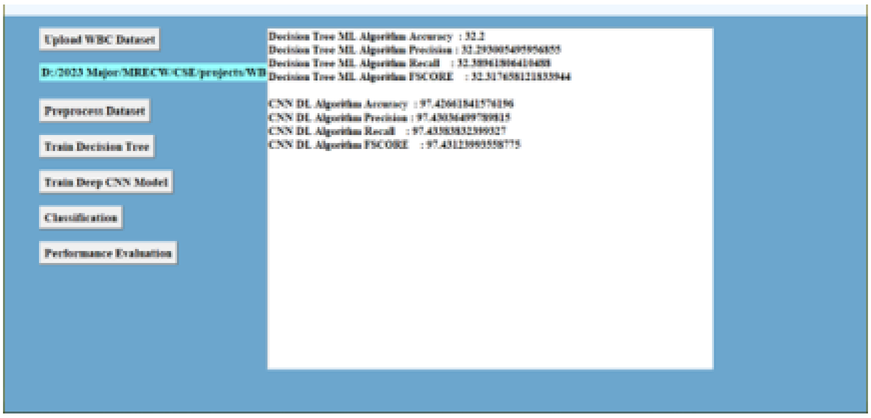

In above screen both train and test images loaded and then displaying sample processed images and now close above image and then click on ‘Train Decision Tree Algorithm’ button to train algorithm and get below output

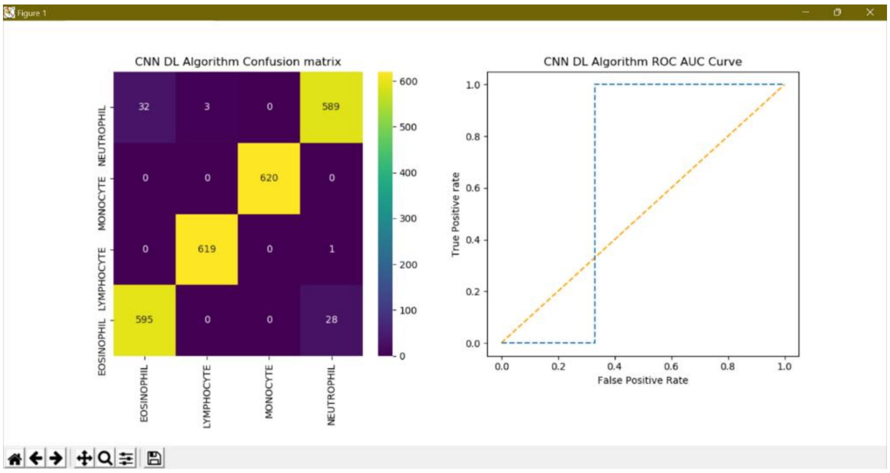



In above screen with decision tree, we got 32.2% accuracy as ML are not good enough for image classification.

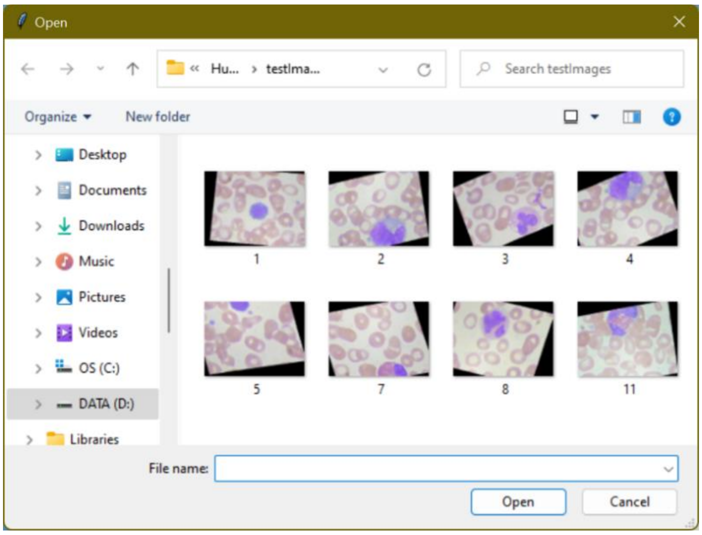



In above confusion matrix graph x-axis represents Predicted Labels and y-axis represents True labels and the count with same label in x and y-axis represents correct prediction count and other boxes represents incorrect prediction count. In above ROC graph x-axis represents False positive rate and y-axis represents True positive rate and if blue line comes below orange line, then prediction is False and if comes on top of orange line then prediction True. Now close above graph and then click on ‘Train CNN Algorithm’ button to train CNN and get below output

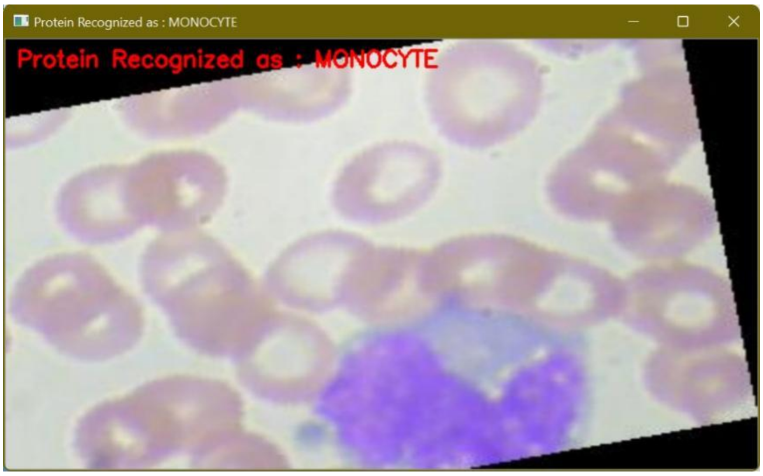


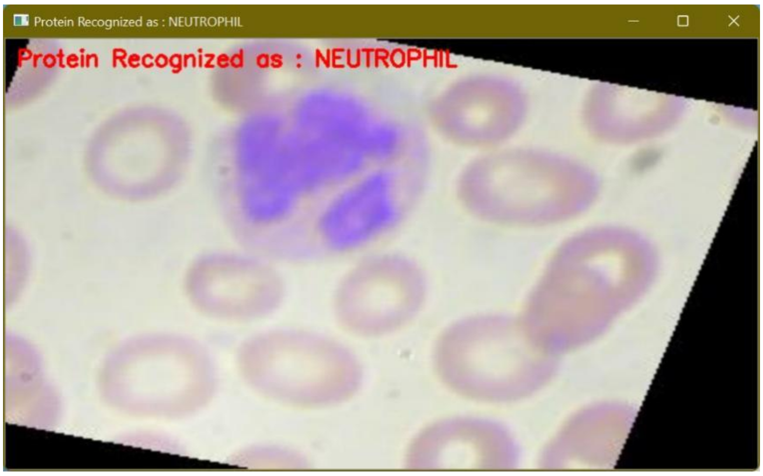

In above screen with CNN, we got 97% accuracy, and we can see confusion and ROC graph also. Now click on 'Classification' button to upload test image and get prediction

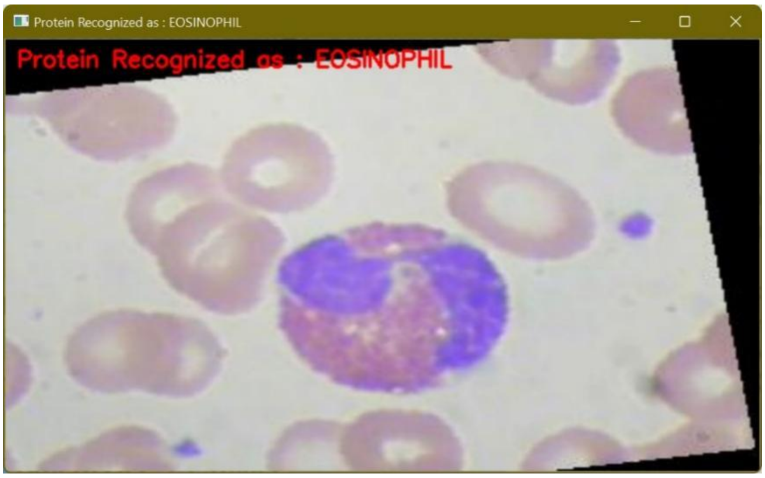


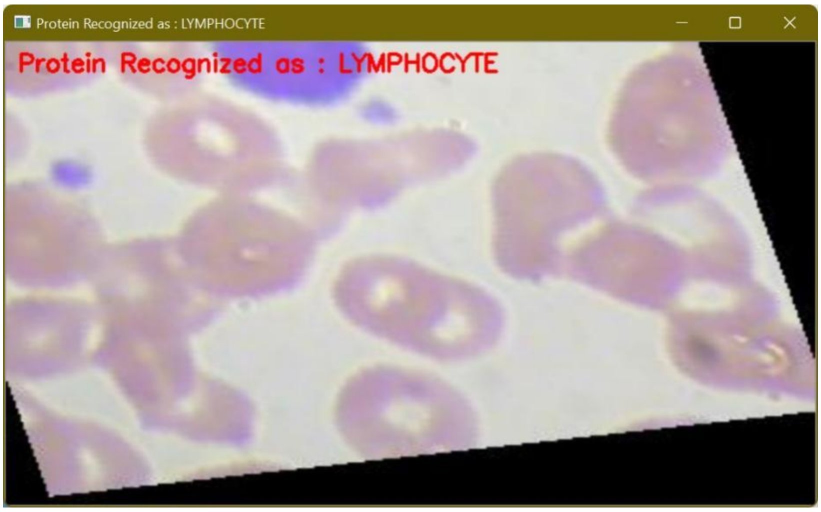

In above screen selecting and uploading '2.jpeg' and then click on 'Upload' button to get below output



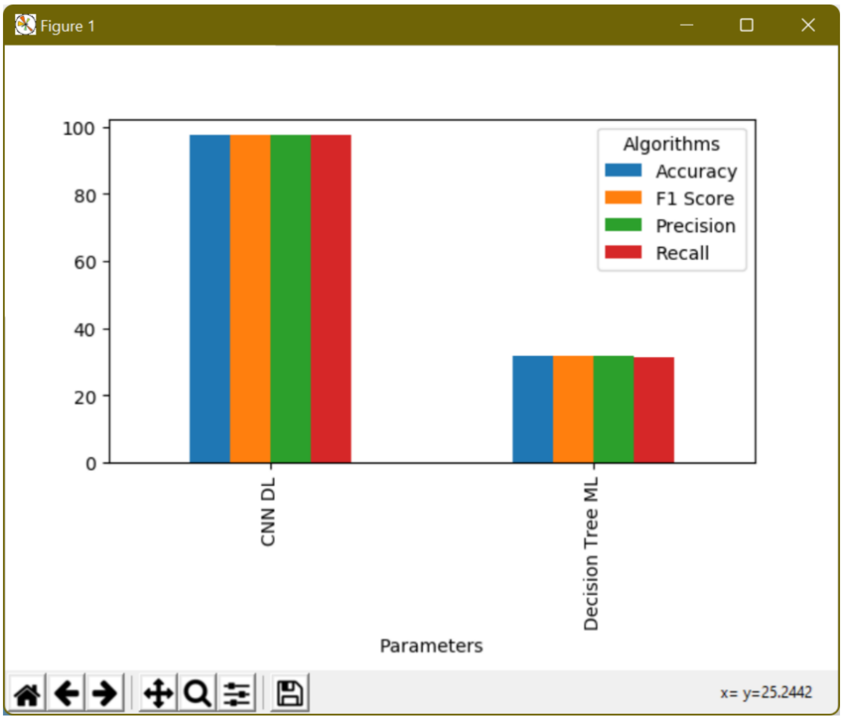

In above screen protein recognized as 'Monocyte' and similarly you can upload and test other images






Now click on ‘Comparison Graph’ button to get below graph




In above graph x-axis represents algorithm names and y-axis represents accuracy and other metrics in different colour bars.

## 5. CONCLUSION AND FUTURE WORK

This work implemented the detection of subtype blood cells using the advancement of neural networks known as deep learning CNN. Compared to machine learning algorithms, which use hand-crafted features as inputs, CNNs typically take raw images as inputs and learn hierarchical feature representations in an end-to-end fashion. By using CNN, we got 97% accuracy. The future work on the proposed word that we can compare the cropped with segmented WBCs images with different input sizes to find which is the best input type (cropped or segmented) and size that can provide more accurate classification of WBCs.

## REFERENCES


[1] C. Shorten and T. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning", Journal of Big Data, vol. 6, no. 1, 2019. Available: 10.1186/s40537-019-0197-0.

[2] "The American Society of Hematology", Hematology.org, 2020. [Online]. Available: https://www.hematology.org. [Accessed: 28- Nov- 2019].

[3] AL-Dulaimi, Khamael, et al. "Classification of white blood cell types from microscope images: Techniques and challenges." Microscopy Science: Last Approaches on Educational Programs and Applied Research. Vol. 8. Formatex Research Center, 2018.

[4] Alzubaidi, L., Al-Shamma, O., Fadhel, M.A., Farhan, L., Zhang, J. (2020). Classification of Red Blood Cells in Sickle Cell Anemia Using Deep Convolutional Neural Network. In: Abraham, A., Cherukuri, A., Melin, P., Gandhi, N. (eds) Intelligent Systems Design and Applications. ISDA 2018 2018. Advances in Intelligent Systems and Computing, vol 940. Springer, Cham. https://doi.org/10.1007/978-3-030-16657-1_51.

[5] Rahman, A., Zunair, H., Rahman, M. S., Yuki, J. Q., Biswas, S., Alam, M. A., ... & Mahdy, M. R. C. (2019). Improving malaria parasite detection from red blood cell using deep convolutional neural networks. arXiv preprint arXiv:1907.10418.

[6] Roopa B. Hegde, Keerthana Prasad, Harishchandra Hebbar, Brij Mohan Kumar Singh, Comparison of traditional image processing and deep learning approaches for classification of white blood cells in peripheral blood smear images, Biocybernetics and Biomedical Engineering, Volume 39, Issue 2, 2019, Pages 382-392, ISSN 0208-5216, https://doi.org/10.1016/j.bbe.2019.01.005.

[7] A. Malkawi, R. Al-Assi, T. Salameh, B. Sheyab, H. Alquran and A. M. Alqudah, "White Blood Cells Classification Using Convolutional Neural Network Hybrid System," 2020 IEEE 5th Middle East and Africa Conference on Biomedical Engineering (MECBME), 2020, pp. 1-5, doi: 10.1109/MECBME47393.2020.9265154.

[8] Matek, C., Schwarz, S., Spiekermann, K. et al. Human-level recognition of blast cells in acute myeloid leukaemia with convolutional neural networks. Nat Mach Intell 1, 538–544 (2019). https://doi.org/10.1038/s42256-019-0101-9

[9] Sadafi, A. et al. (2019). Multiclass Deep Active Learning for Detecting Red Blood Cell Subtypes in Brightfield Microscopy. In: , et al. Medical Image Computing and Computer Assisted Intervention – MICCAI 2019. MICCAI 2019. Lecture Notes in Computer Science(), vol 11764. Springer, Cham. https://doi.org/10.1007/978-3-030-32239-7_76.

[10] Parab, M.A., Mehendale, N.D. Red Blood Cell Classification Using Image Processing and CNN. SN COMPUT. SCI. 2, 70 (2021). https://doi.org/10.1007/s42979-021-00458-2.

[11] S. Parayil and J. Aravinth, "Transfer Learning-based Feature Fusion of White Blood Cell Image Classification," 2022 7th International Conference on Communication and Electronics Systems (ICCES), 2022, pp. 1468-1474, doi: 10.1109/ICCES54183.2022.9835815.

[12] Yildirim, Muhammed & Çinar, Ahmet. (2019). Classification of White Blood Cells by Deep Learning Methods for Diagnosing Disease. Revue d'Intelligence Artificielle. 33. 335-340. 10.18280/ria.330502.