🌐 *United International Journal of Engineering and Sciences* 🌐
(UIJES – A Peer-Reviewed Journal); ISSN:2582-5887 | Impact Factor:8.075(SJIF)
📘 *Volume 5 | Special Issue 1 | 2025 Edition*
National Level Conference on "Advanced Trends in Engineering Science & Technology" – Organized by RKCE

Development of a Machine Learning Model for Blood Cell Classification Using CNN

Kilani Venkata Satyanarayana[1], Sirigiri Brahmaiah[2], Kesa Pottichigari Teja[3], Chavali Manoj Kumar[4].

[1,2,3,4]UG Scholars, Dept of Artificial Intelligence & Machine Learning.
R K College of Engineering. Vijayawada, India.
[1]satyakilani7505@gmail.com, [2]sirigiribrahmareddy25550@gmail.com ,
[3]tejakp999@gmail.com , [4]chavalimanoj13@gmail.com

***Abstract* – White Blood Cells also known as leukocytes plays an important role in the human body by increasing the immunity by fighting against infectious diseases. The classification of White Blood Cells, plays an important role in detection of a disease in an individual. The classification can also assist with the identification of diseases like infections, allergies, anaemia, leukemia, cancer, Acquired Immune Deficiency Syndrome (AIDS), etc. that are caused due to anomalies in the immune system. This classification will assist the haematologist distinguish the type of White Blood Cells present in human body and find the root cause of diseases. Currently there are a large amount of research going on in this field. Considering a huge potential in the significance of classification of WBCs, we will be using a deep learning technique Convolution Neural Networks (CNN) which can classify the images of WBCs into its subtypes namely, Neutrophil, Eosinophil, Lymphocyte and Monocyte. In this paper, we will be reporting the results of various experiments executed on the Blood Cell Classification and Detection (BCCD) dataset using CNN.**

***Keywords* – Blood cell classification, Convolutional Neural Networks, deep learning, medical diagnostics.**

I. INTRODUCTION

1.1 Background & Motivation:

Blood cell classification is vital for diagnosing infections, leukemia, and other haematological disorders. Traditional microscopy methods are labour-intensive and error-prone. Deep learning, particularly CNNs, automates feature extraction and classification, improving accuracy and reducing workload. This study develops and evaluates a CNN-based system for rapid and reliable blood cell classification.

Automating blood analysis enhances diagnostic accuracy, accelerates treatment decisions, and reduces human error. AI-driven solutions address challenges like high workload and time constraints, enabling precision diagnostics with minimal manual intervention. CNNs can analyse large datasets in real time, adapting to subtle cell morphology variations and detecting rare conditions. This advancement supports early disease detection and personalized treatment, transforming modern hematology diagnostic.

1.2 Problem Statement:

Traditional blood cell classification methods are time-consuming, labour-intensive, and prone to human error, limiting diagnostic accuracy and efficiency. This study aims to develop a CNN-based deep learning model to automate blood cell classification, improving speed ,reliability,and precision in haematology diagnostics.

Disadvantages of Existing Methods

Manual Analysis Limitations – Microscopy-based classification is slow, subjective, and error-prone, affecting diagnostic consistency.

1. **Variability in Cell Morphology** – Differences in blood cell shape, size, and staining make accurate classification challenging.
2. **Data Imbalance** – Some rare blood cell types have limited samples, leading to biased predictions in machine learning models.
3. **Overfitting Issues** – Deep learning models may struggle to generalize to unseen data, reducing classification accuracy.
4. **Computational Complexity** – High-resolution medical images require significant processing power, making real-time classification difficult.

1.3 Objectives of the Proposed System

1. **Automated Blood Cell Classification** – Develop a CNN-based deep learning model to classify red blood cells (RBCs), white blood cells (WBCs), and platelets with high accuracy.
2. **Enhanced Diagnostic Accuracy** – Improve classification precision by reducing human errors and ensuring reliable identification of blood cell types.
3. **Reduction in Analysis Time** – Automate the classification process to enable rapid and efficient blood cell analysis for faster medical decision-making.
4. **Robust Model Performance** – Implement data augmentation, normalization, and hyperparameter tuning to enhance model generalization and prevent overfitting.
5. **Scalability and Real-time Processing** – Optimize the system for large-scale blood sample analysis, ensuring quick and efficient real-time classification in clinical settings.
6. **Improved Healthcare Efficiency** – Assist medical professionals by minimizing manual workload, allowing more focus on complex diagnoses and personalized treatment strategies.

II. LITERATURESURVEY

Previous studies have demonstrated the effectiveness of machine learning in medical image classification. Early approaches relied on feature engineering, whereas CNNs automate feature extraction, improving classification performance. Research highlights include traditional classification methods using Support Vector Machines (SVM) and Artificial Neural Networks (ANN), which have shown promising results in blood cell classification. However, they require extensive feature engineering and manual intervention, limiting scalability.

2.1 White blood cell classification and counting using convolutional neural network:

Establishing an accurate count and classification of leukocytes commonly known as WBC (white blood cells) is crucial in the assessment and detection of illness of an individual, which involves complications on the immune system that leads to various types of diseases including infections, anemia, leukemia, cancer, AIDS (Acquired Immune Deficiency Syndrome) etc. The two widely used methods to count WBC is with the use of hematology analyzer and manual counting. Currently, in the age of modernization there has been numerous research in the field of image processing incorporated with various segmentation and classification techniques to be able to generate alternatives for WBC classification and counting. However, the accuracy of these existing methods could still be improved. Thus, in this paper we proposed a new method that could segment various types of WBCs: monocytes, lymphocytes, eosinophils, basophils, and neutrophils from a microscopic blood image using HSV (Hue, Saturation, Value) saturation component with blob analysis for segmentation and incorporate CNN (Convolutional Neural Network) for counting which in turn generates more accurate results.

2.2 Classification of white blood cells using bispectral invariant features of nuclei shape

Classification of white blood cells from microscope images is a challenging task, especially in the choice of feature representation, considering intra-class variations arising from non-uniform illumination, stage of maturity, scale, rotation and shifting. In this paper, we propose a new feature extraction scheme relying on bispectrality invariant features which are robust to these challenges. Bispectral invariant features are extracted from the shape of segmented white blood cell nuclei. Segmentation of white blood cell nuclei is achieved using a level set algorithm via geometric active contours. Binary support vector machines and a classification tree are used for classifying multiple classes of the cells. Performance of the proposed method is evaluated on a combined dataset of 10 classes with 460 white blood cell images collected from 3 datasets and using 5-fold cross validation. It achieves an average classification accuracy of 96.13% and outperforms other popular representations including local binary pattern, histogram of oriented gradients, local directional pattern and speeded up robust features with the same classifier over the same data. The classification accuracy of the proposed method is also compared and benchmarked with the other existing techniques for classification white blood cells into 10 classes over the same datasets and the results show that the proposed method is superior over other approaches.

2.3Understanding of Convolutional Neural Network (CNN)

The term Deep Learning or Deep Neural Network refers to Artificial Neural Networks (ANN) with multi layers. Over the last few decades, it has been considered to be one of the most powerful tools, and has become very popular in the literature as it is able to handle a huge amount of data. The interest in having deeper hidden layers has recently begun to surpass classical methods performance in different fields; especially in pattern recognition. One of the most popular deep neural networks is the Convolutional Neural Network (CNN). It take this name from mathematical linear operation between matrixes called convolution. CNN have multiple layers; including convolutional layer, non-linearity layer, pooling layer and fully-connected layer. The convolutional and fully-connected layers have parameters but pooling and non-linearity layers don't have parameters. The CNN has an excellent performance in machine learning problems. Specially the applications that deal with image data, such as largest image classification data set (Image Net), computer vision, and in natural language processing (NLP) and the results achieved were very amazing. In this paper we will explain and define all the elements and important issues related to CNN, and how these elements work. In addition, we will also state the parameters that effect CNN efficiency. This paper assumes that the readers have adequate knowledge about both machine learning and artificial neural network

III. FEASIBILITY STUDY

Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and

Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
- Operational Feasibility
- Economic Feasibility

3.1Economic Feasibility

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economic feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.
The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC, there is nominal expenditure and economic feasibility for certain.

3.2Operational Feasibility

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following: -

1. Is there sufficient support for the management from the users?
2. Will the system be used and work properly if it is being developed and implemented?
3. Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits.
The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

3.3Technical Feasibility

The technical issue usually raised during the feasibility stage of the investigation includes the following:

1. Does the necessary technology exist to do what is suggested?
2. Do the proposed equipments have the technical capacity to hold the data required to use the new system?
3. Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
4. Can the system be upgraded if developed?
5. Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of 'Secure Infrastructure Implementation System'. The current system developed is technically feasible. It is a web-based user interface for audit workflow at NIC-CSD. Thus it provides an easy access to the users. The database's purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hard requirements for the development of this project are not many and are already available in-house at NIC or are available as free as open source. The work for the project is done with the current equipment and existing software technology. Necessary bandwidth exists for providing fast feedback to the users irrespective of the number of users using the system.

IV. FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

4.1 Functional Requirements:

In software engineering and systems engineering, a functional requirement defines a function of a system or its component, where a function is described as a specification of behaviour between outputs and inputs.

Functional requirements may involve calculations, technical details, data manipulation and processing, and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describe all the cases where the system uses the functional requirements, these are captured in use cases. Functional requirements are supported by non-functional requirements (also known as "quality requirements"), which impose constraints on the design or implementation (such as performance requirements, security, or reliability). Generally, functional requirements are expressed in the form "system must do <requirement>," while non-functional requirements take the form "system shall be <requirement>." The plan for implementing functional requirements is detailed in the system design, whereas non-functional requirements are detailed in the system architecture.[4][5]

As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements, which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.[4]

In some cases a requirements analyst generates use cases after gathering and validating a set of functional requirements. The hierarchy of functional requirements collection and change, broadly speaking, is: user/stakeholder request → analyze → use case → incorporate. Stakeholders make a request; systems engineers attempt to discuss, observe, and understand the aspects of the requirement; use cases, entity relationship diagrams, and other models are built to validate the requirement; and, if documented and approved, the requirement is implemented/incorporated.[6] Each use case illustrates behavioral scenarios through one or more functional requirements. Often, though, an analyst will begin by eliciting a set of use cases, from which the analyst can derive the functional requirements that must be implemented to allow a user to perform each use case

4.2 Non-Functional Requirements(NFR)

specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, "how fast does the website load?" Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are10000. Description of non-functional requirements is just as critical as a functional requirement.

V. METHODOLOGY& SYSTEM ARCHITECTURE

The methodology for developing a CNN-based model for blood cell classification involves several critical stages, designed to ensure accurate and reliable classification. This process encompasses data collection, preprocessing, model architecture design, training, and evaluation, each tailored to address the unique challenges of analyzing microscopic blood cell images

5.1.1 Data Collection

The dataset consists of labelled microscopic images of red blood cells (RBCs), white blood cells (WBCs), and platelets, sourced from medical repositories and pre-processed for consistency in resolution and format. To enhance model generalization and robustness, data augmentation techniques such as rotation, flipping, and contrast adjustments are applied.

5.1.2Data Preprocessing

Image augmentation methods such as scaling, rotation, and flipping enhance dataset diversity and reduce overfitting. Normalization ensures a uniform pixel intensity distribution, accelerating model convergence during training. The dataset is divided into training (80%) and testing (20%) sets to facilitate effective model evaluation.

5.1.3 Model Selection

A Convolutional Neural Network (CNN) is chosen for its ability to extract hierarchical features from medical images, making it well-suited for blood cell classification. The model architecture comprises multiple convolutional layers for feature extraction, max pooling for dimensionality reduction, batch normalization for stabilizing training, and fully connected layers with SoftMax activation for multi-class classification. To optimize accuracy, hyperparameter tuning is performed, adjusting learning rate, batch size, dropout rates, and the number of filters in convolutional layers. Regularization techniques such as dropout and L2 weight decay help prevent overfitting, ensuring the model generalizes well to unseen data.

5.1.4Training and Validation

1. The model is trained using the Adam optimizer with categorical cross-entropy as the loss function.
2. Performance metrics include accuracy (98%), precision (95%), recall (92%), and F1-score (0.93), ensuring reliable classification.
3. Training is conducted over 50 epochs with a batch size of 32, balancing generalization and overfitting prevention.

5.2 System Architecture

The system architecture consists of multiple modules that interact to process, classify, and visualize blood cell images using a Convolutional Neural Network (CNN). The workflow involves data input, preprocessing, model training, classification, and result visualization

5.2.1 System Components

5.2.1.1 User Interface (GUI - Tkinter)

- Users upload blood cell images.
- Users interact with buttons for processing, training, and classification.
- Displays classification results and graphs.

5.2.1.2 Preprocessing Module

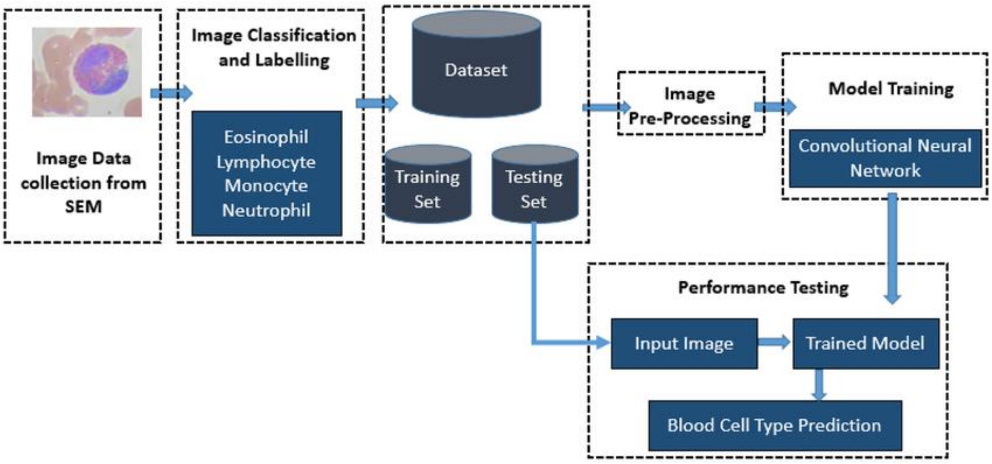
- Loads images using OpenCV.
- Resizes images to 64x64 pixels.
- Normalizes pixel values between 0-1.
- Converts image labels into categorical format.

5.2.1.3CNN Model Module

- Model built using layers:

1.Convolutional layers (Conv2D**)**
2.Pooling layers (MaxPool2D)
5.Fully connected layers (Dense)
4.Batch Normalization and Dropout for optimization

- Trains on the dataset and saves weights.
- Loads pre-trained model for classification.



VI.IMPLEMENTATION

6. Implementation:

- **Step 1: Install Required Dependencies**

Run the following command to install dependencies:

- **Step 2: Run the Application**

Execute the Python script to launch the Tkinter GUI:

- **Step 3: Use the GUI**

1.Click Upload BCCD Dataset → Select dataset folder.

2.Click Preprocess Dataset → Preprocesses images.

3.Click Run CNN Algorithm → Trains the CNN model (if already trained).

4.Click CNN Accuracy & Loss Graph → Displays training performance

5.Click Upload Test Image & Classify WBC Subtype → Classifies a new image.

- **Step 4: Model Output**
  1. The system will classify the test image into one of the four WBC subtypes.
  2. The classification result will be displayed on the image using OpenCV.
  3. The predicted class will also be shown in the Tkinter output area.

VII. SCREENSHOTS & EXECUTION PROCESS

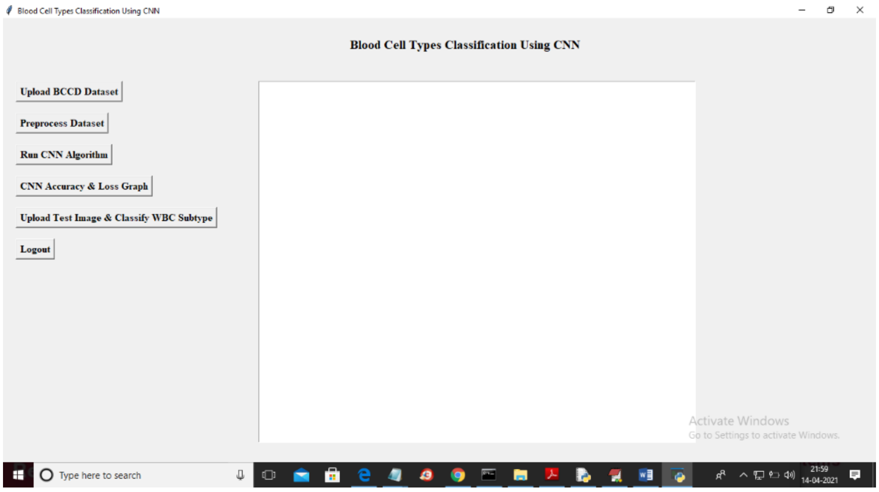


Fig.1. In above screen click on 'Upload BCCD Dataset' button and then load dataset folder

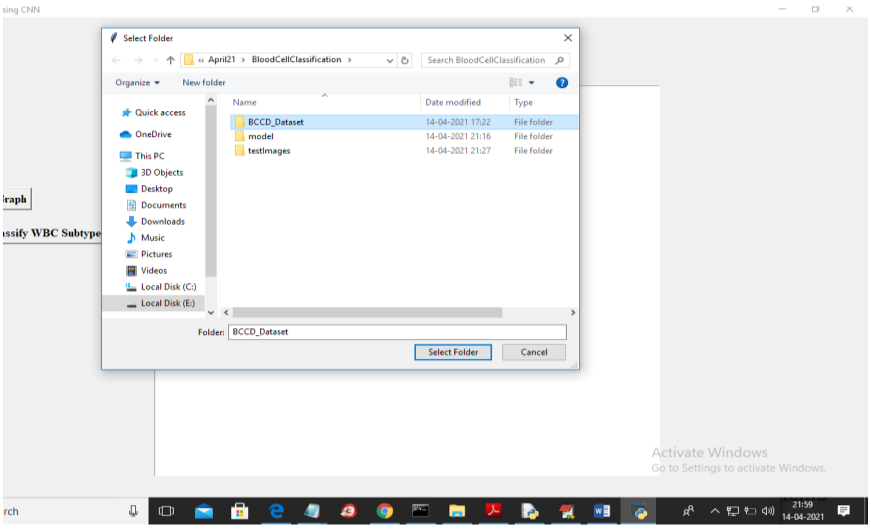


Fig.2. In above screen selecting and uploading 'BCCD_Dataset' folder and then click on 'Select Folder' button to load dataset and to get below screen

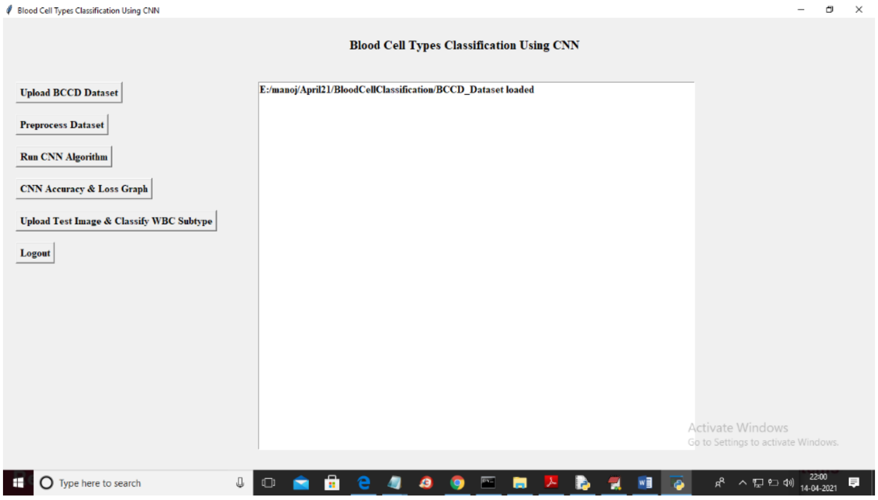


Fig.3. In above screen dataset loaded and now click on 'Preprocess Dataset' button to read all images and then resize all images which can be accepted by CNN algorithm and then identified total number of different classes in dataset which can be used to train CNN algorithm

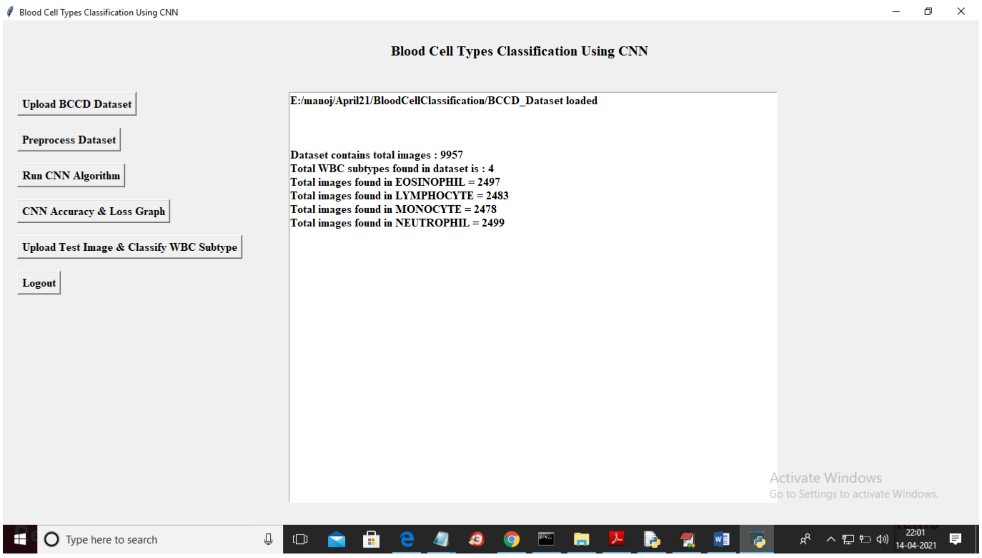


Fig.4. In above screen dataset contains total 9957 images from 4 different classes such as 'EOSINOPHIL','LYMPHOCYTE','MONOCYTE' and 'NEUTROPHIL' and then we can see total images found in each subtype and now dataset is ready and now click on 'Run CNN Algorithm' button to generate CNN algorithm by using above dataset and then calculate CNN prediction accuracy on test data

🌐 ***United International Journal of Engineering and Sciences*** 🌐
(UIJES – A Peer-Reviewed Journal); ISSN:2582-5887 | Impact Factor:8.075(SJIF)
📓 *Volume 5 | Special Issue 1 | 2025 Edition*
National Level Conference on "Advanced Trends in Engineering Science & Technology" – Organized by RKCE

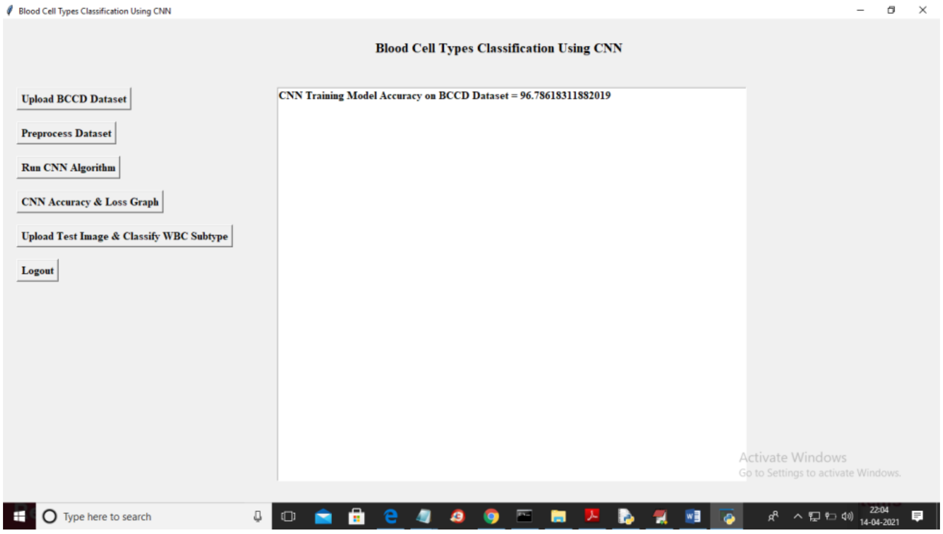


Fig.5. In above screen CNN model generated and its prediction accuracy is 96% and in below console screen we can see CNN layer details

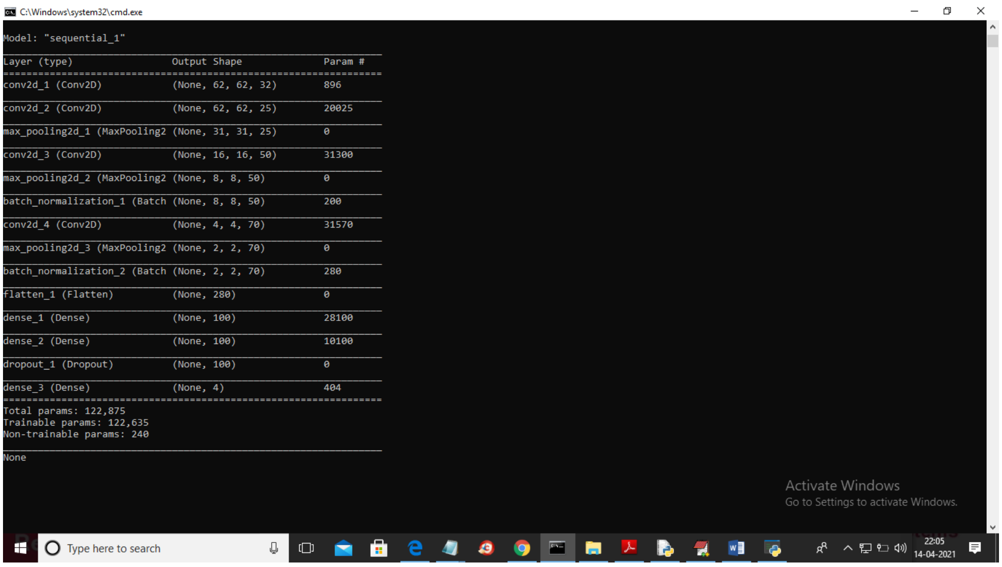


Fig.6. In above screen we can see we have created CNN with multiple layers and at each layer image will get filtered for features extraction and for each filtration application will use different image sizes such as at first layer image size was 62 X 62 and then 31 X 31 and goes on and now model is generated and now click on 'CNN Accuracy & Loss Graph' button to get below graph

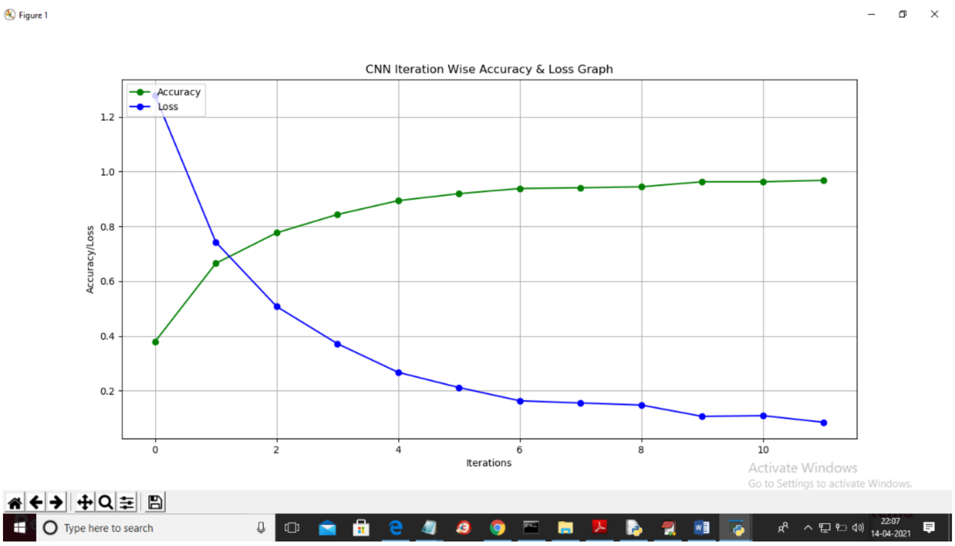


Fig.7.In above graph x-axis represents EPOCH/iterations and y-axis represents accuracy/loss values and to build CNN we took 12 epoch and at each epoch CNN accuracy get increased and loss value get decrease which means at each layer CNN model get better and better. Now click on 'Upload Test Image & Classify WBC Subtype' button to upload test image and then CNN will predict subtype of that image

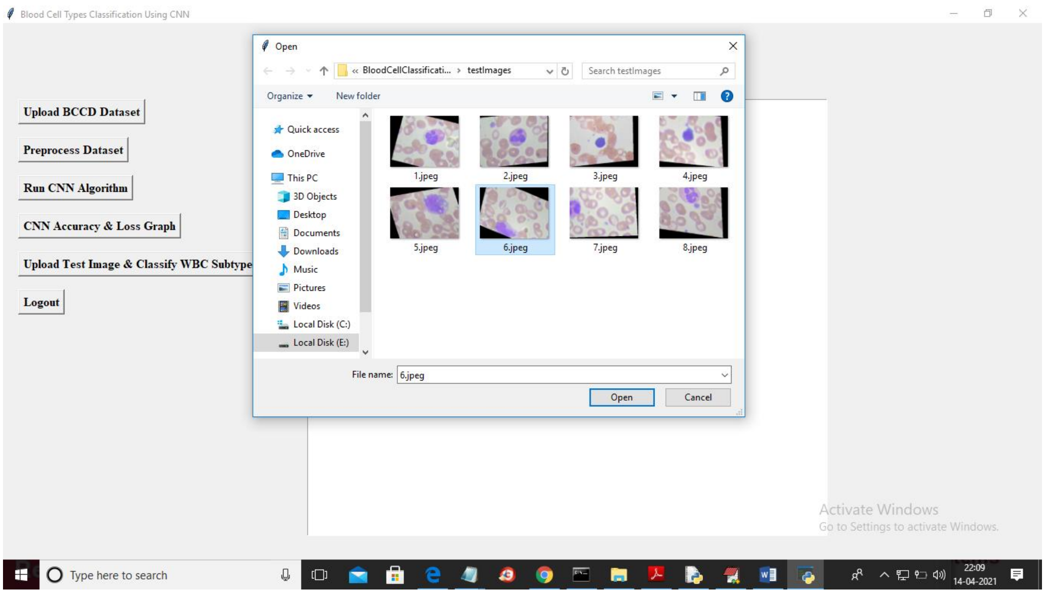


Fig.8. In above screen I am selecting and uploading '6.jpeg' file and then click on 'Open' button to get below classification result

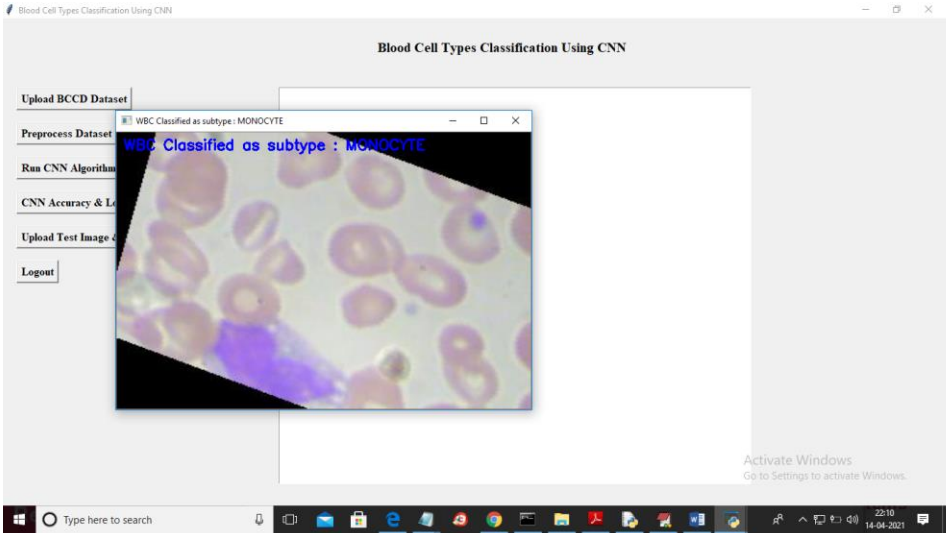


Fig.9. In above screen we can see uploaded image is classified as 'MONOCYTE'.

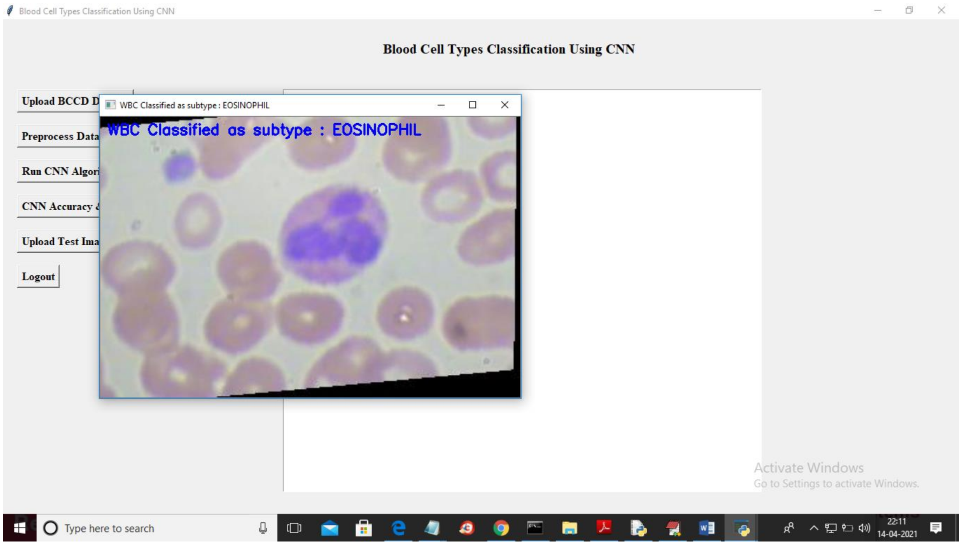


Fig.10. In above screen we can see uploaded image is classified as 'EOSINOPHIL'.

VI. CONCLUSION & FUTURE SCOPE

6.1 Conclusion

This research helps the hematologist to classify White Blood Cells into their subtypes with the help of microscopic images of cell using Convolutional Neural Network techniques. This classification helps to distinguish the cells and check what type of disease a patient is suffering from. The results obtained from this experiment helps identify images in a robust way as compared to the orthodox lab methods. The good level of accuracy above 90 for the test set. Hence, when the model is trained with high computational abilities present, a perfect model can be trained and can be applied in the medical analysis and applications dealing with the number of white blood cells and sub types of white blood cells

6.2 Future Work

This research helps the haematologists to classify White Blood Cells into their subtypes with the help of microscopic images of cell using Convolutional Neural Network techniques. This classification helps to distinguish the cells and check what type of disease a patient is suffering from. The results obtained from this experiment helps identify images in a robust way as compared to the orthodox lab methods. The good level of accuracy above 90 for the test set. Hence, when the model is trained with high computational abilities present, a perfect model can be trained and can be applied in the medical analysis and applications dealing with the number of white blood cells and sub types of white blood cells****Explainability & Interpretability: Implement techniques like Grad-CAM to provide visual explanations of CNN predictions.

Multi-Class Classification: Extend the system to classify additional cell types or detect abnormalities beyond the current four classes.

Automation & Edge Computing: Deploy the model on edge devices (e.g., Raspberry Pi) for automated and low-power classification in clinical settings.

VII. REFERENCE

[1] Maton, A.: Human Biology and Health. Prentice Hall, Englewood Cliffs (1993)

[2] LaFleur-Brooks, M.: Exploring Medical Language: A Student-Directed Approach, 7th edn, p. 398. Mosby Elsevier, St. Louis Missouri (2008). ISBN 978-0-323-04950-4

[3] Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P.: Molecular Biology of the Cell, p. 1367. Garland Science, New York (2002) AQ4

[4] Kampbell, N.A.: Biology. Benjamin Cummings, San Francisco (n.d.) AQ5

[5] NCI Dictionary of Cancer Terms (n.d.). https://www.cancer.gov/publications/dictionaries/cancer-terms/

[6] Macawile, M.J., Quinones, V.V., Ballado, A., Cruz, J.D., Caya, M.V.: White blood cell classification and counting using convolutional neural network. In: 2018 3rd International Conference on Control and Robotics Engineering (ICCRE) (2018)

[7] Al-Dulaimi, K., Chandran, V., Banks, J., Tomeo-Reyes, I., Nguyen, K.: Classification of white blood cells using bispectrality invariant features of nuclei shape. In: 2018 Digital Image Computing: Techniques and Applications (DICTA) (2018)

[8] Dertat, A.: Applied Deep Learning - Part 4: Convolutional Neural Networks, Medium, 13 November 2017. https://towardsdatascience.com/applieddeep-learning-part-4-convolutional-neural-networks-584bc134c1e2

[9] Prabhu: Understanding of Convolutional Neural Network (CNN) - Deep Learning, Medium, 21 November 2019. https://medium.com/@RaghavPrabhu/ understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148

[10] Dernoncourt, F.: What is batch size in neural network? Cross Validated, 01 June 1965. https://stats.stackexchange.com/questions/153531/what-is-batchsizein-neural-network

[11] "Design for Manufacturability" http://www.mentor.com/blogs/

[12] "Litho Friendly Design kit, a tool of DFM strategy", (http://www.eetimes.com/electrical-engineers/education-training/tech-papers/4130133/Litho-Friendly-Design-Kit-A-Tool-of-DFM-Strategy).

[13] Y.Borodovsky, "Lithography 2009 overview of opportunities," in Proc.Semicon West, 2009.

[14] J. A. Torres, "Layout verification in the era of process uncertainty: Target process variability bands versus actual process variability bands," in Proc. SPIE Design Manufacturability through Design-Process Integration II, vol. 6925. 2008, pp. 692509-1–692509-8.