



PAPER • OPEN ACCESS

WBC-KICNet: knowledge-infused convolutional neural network for white blood cell classification

To cite this article: Jeneessha P *et al* 2024 *Mach. Learn.: Sci. Technol.* **5** 035086

View the [article online](#) for updates and enhancements.

You may also like

- [In Vitro and in Vivo Measurements of ROS Scavenging Activity and White Blood Cells Activity by Chemiluminescence of a New Selena-Diazole Derivative Compare to Dipyrone Activity](#)
Nadheerah Faliq Neamah, Abdul-Razzak Naaem Khudair and Shaker A. N. Al-Jadaan
- [Domain knowledge-based deterministic graph traversal method for white blood cell classification](#)
Jeneessha P and Vinoth Kumar Balasubramanian
- [Investigating WBC margination in different microfluidic geometries: influence of RBC shape and size](#)
Sanjay Mane, Vadiraj Hemadri and Siddhartha Tripathi



PAPER

OPEN ACCESS

RECEIVED

16 June 2024

REVISED

19 August 2024

ACCEPTED FOR PUBLICATION

12 September 2024

PUBLISHED

1 October 2024

Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](#).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



WBC-KICNet: knowledge-infused convolutional neural network for white blood cell classification

Jeneessha P^{1,*} , Vinoth Kumar Balasubramanian¹ and M Murugappan^{2,3} ¹ Department of Information Technology, PSG College of Technology, Coimbatore 641004, India² Intelligent Signal Processing (ISP) Research Lab, Department of Electronics and Communication Engineering, Kuwait College of Science and Technology, Doha 13133, Kuwait³ Department of Electronics and Communication Engineering, School of Engineering, Vels Institute of Sciences, Technology, and Advanced Studies, Chennai 600117, India

* Author to whom any correspondence should be addressed.

E-mail: jen.it@psgtech.ac.in**Keywords:** convolutional neural networks, domain knowledge, feature fusion, white blood cells, classification**Abstract**

White blood cells (WBCs) are useful for diagnosing infectious diseases and infections. Machine learning and deep learning have been used to classify WBCs from blood smear images. Despite advances in machine learning, there has been little research on applying medical domain knowledge to convolutional neural networks (CNNs) to improve WBC classification. The existing models are often inaccurate, rely on manual input, and fail to incorporate external medical knowledge into decision-making. This study used the blood cell count and detection dataset which contains images of monocytes, lymphocytes, neutrophils, and eosinophils for WBC classification. In this paper, we propose a CNN model for WBC classification called WBC-KICNet (knowledge-infused convolutional neural network). The present work uses two CNN models: the first model generates the knowledge vector from input images and the domain expert (hematologist); the second model extracts deep features from the input image. A feature fusion mechanism is then used to combine these two features to classify the WBCs. Several metrics have been used to evaluate the performance of the WBC-KICNet model. These measures yielded impressive results. Accuracy, precision, recall, specificity, and F1-score were rated 99.22%, 99.25%, 99%, 99.77%, and 99.25%, respectively. In each of the WBC classes, accuracy rates are: 98.7% for eosinophils, 99.83% for lymphocytes, 100% for monocytes, and 98.32% for neutrophils. As a result, the proposed WBC-KICNet classifies WBCs accurately and without much misclassification, and the results have been confirmed by a statistical hypothesis test (t -test).

1. Introduction

White blood cells (WBCs), also known as leukocytes play a fundamental role in the human immune system, protecting the body against several infections and diseases. The WBCs are instrumental in detecting microbes in the body, providing information about specific health conditions. These cells are versatile and come in various types, each with a unique functionality. Their versatility lies in their ability to adapt and respond to a wide range of pathogens and immune challenges. The different types of WBCs are neutrophils, lymphocytes, monocytes, eosinophils, and basophils [1]. Each WBC has a unique function and characteristic. The five types of WBCs in the blood are each designed to attack a specific type of microorganism, so their composition and nature are different. The classification and quantification of WBCs from blood samples obtained through blood tests provide crucial insights into a patient's health status and immune response. Consequently, physicians emphasize analyzing WBCs to understand a person's health. Manual examination under a microscope in the clinical laboratory is traditionally used to identify and classify WBCs. In this context, the development of automated systems, particularly those based on advanced technologies like image processing and deep learning, has become a necessity for accurate and efficient WBC

classification, contributing to improved patient care and early disease detection. The use of deep learning models to classify WBCs has been widely discussed which focuses on convolutional neural network (CNN), pre-trained models, hybrid models, feature extraction, and attention mechanisms. Hybrid models have been used for WBC classification but they require large datasets and have high complexity potentially leading to the overfitting problem (Tiwari *et al* in 2018 [1], Cheuque *et al* [2], Cinar and Tuncer [3], Chen *et al* [4], Firat *et al* in 2024 [5]). Feature Extraction and Feature Fusion Methods help in extracting morphological features and deep features and produce more useful features for learning still these methods pose risks of dimensionality incompatibility, data loss, and integration challenges (Togacar *et al* [6], Banik *et al* [7], Dong *et al* [8]). Furthermore, Attention Mechanisms that help in considering the features from the Region of Interest show enhanced prediction but rely heavily on the availability of large datasets for optimal performance, yet medical data often suffer from limited availability (Wang Zee *et al* in 2022 [9], Li *et al* in 2021 [10]).

Object recognition algorithms have been used for WBC classification since blood smear images are simple and contain few objects. However, these algorithms require large datasets to generalize well and are computationally intensive (Ha *et al* in 2022 [11]). WBC classification using CNN is a widely explored one since CNNs were specially developed for image classification. WBC classification using CNN and customized CNN have less complexity but performance can be enhanced through hyperparameter optimization part (Chen *et al* in 2012 [12], Lei and Chen in 2012 [13], Wang *et al* in 2022 [14], Girdhar *et al* in 2022 [15]). In 2022, Ramya *et al* [16] proposed a CNN with hyperparameter optimization for WBC classification, yet the study did not adequately address the requirement for high data volume.

Pre-trained models perform classification more easily since they are already trained. Sharma *et al* proposed an optimized pre-trained model based on DenseNet 121 for automatic WBC classification (Sharma *et al* [17]). However, customizing the fixed architecture of the pre-trained model for WBC classification still poses a challenge. Additionally, pre-trained models tend to learn a lot of irrelevant data, resulting in degraded performance compared to models trained from scratch.

Deep learning models are chosen greatly for their self-learning capability but require huge data for learning but in the medical domain obtaining huge data is a challenge. Hence, medical applications suffer from unavailability or less availability of datasets. This leads to overfitting issues in deep learning models (Shin *et al* in 2016 [18]). Feature extraction and feature fusion in deep learning help in creating effective deep learning models but the morphological characteristics of the WBC types are quite similar, potentially leading to misclassification. The major challenges in WBC classification can be overcome through techniques such as: reducing the model's complexity [17], regularization [18], and data enhancement [20]; however, these techniques do not add additional data rather, only concentrate on enhancing the already available data.

Whereas, the problem can be effectively addressed by adding knowledge to the deep learning model (Xie *et al* in 2021 [21], Tian *et al* in 2024 [22]). Xie *et al* [21] discuss the various methods for incorporating domain knowledge into deep learning models for medical image analysis where the author says that most domain knowledge incorporation methods use feature extraction through transfer learning and feature fusion through CNNs or SVMs (Xie *et al* in 2019 [23], Liu *et al* in 2019 [24]). The transfer learning process though easily executable, the model becomes more complex when multiple feature extraction techniques are included. So, domain knowledge incorporation into a CNN through feature fusion will give a light and effective model for WBC classification. Hence, to the best of our knowledge, a CNN is proposed which incorporates domain knowledge about WBC based on vector generation and vector concatenation for WBC classification. The aim is to present an improved model for WBC classification based on domain knowledge infusion through feature fusion techniques.

The contribution of the proposed work is multi-fold:

1. The hematologist designs four domain expert knowledge vectors for each class of WBC based on its significant features from the input blood smear images.
2. Input images can be processed using CNN to generate expertise vectors similar to those generated by domain experts for each image of the class, eliminating the need to use domain experts for every image.
3. Fusion of the knowledge vector with CNN-extracted features leads to better and more accurate classification of WBC.
4. The model has been extensively tested on the blood cell count and detection (BCCD) dataset for WBC classification and compared with earlier works in the literature.
5. The proposed model achieved higher accuracy in WBC classification (99.22%) compared to the state-of-the-art methods reported in the literature.

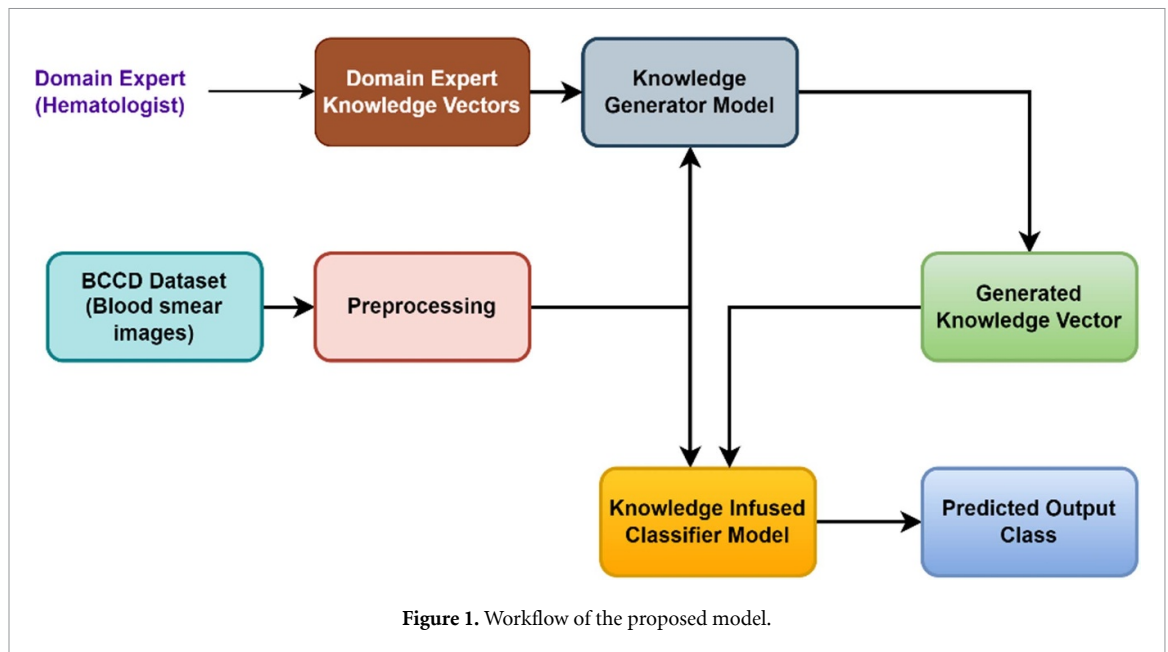


Table 1. Distribution of images between train and test datasets for each WBC type.

Dataset	Description	Classes	Split up	Class name	No. of images
BCCD	12,444 images	4	Train set (9957)	Monocytes	2478
				Lymphocytes	2483
				Eosinophils	2497
				Neutrophils	2499
			Test set (2487)	Monocytes	620
				Lymphocytes	620
				Eosinophils	623
				Neutrophils	624

2. Materials and methods

The overview of the proposed WBC-KICNet (knowledge-infused convolutional neural network) is shown in figure 1.

2.1. Dataset description

Kaggle's BCCD (blood cell count and detection) dataset is used in this work [25]. The dataset consists of about 12,000 blood smear images of size 320×240 , with WBC in each image. There are images for four types of WBCs in the dataset, namely monocytes (3098 images), lymphocytes (3103 images), eosinophils (3120 images), and neutrophils (3123 images). There are no images of the basophil type in the dataset due to its low presence in human blood. Therefore, it is not considered in the work. To ensure robust model evaluation, an 80/20 train-test split was adopted on the dataset. This yielded 9957 images for training and 2487 images for testing, guaranteeing a representative distribution of the four WBC classes across both subsets in table 1.

2.2. Domain knowledge base

As a result of an explicit discussion with a domain expert (hematologist), knowledge was gained about the identification and classification of WBCs. A conventional method of classifying WBCs involves staining, smearing, and microscopic examination. The blood cells become visible after smearing and staining blood onto a glass slide. In this work, a microscope is used to examine stained blood under a microscope to determine the color, size, shape, and type of WBCs. Therefore, WBCs are characterized by their morphological characteristics.

In this way, ten morphological features (characteristics) were identified and listed by the hematologist who has significant expertise in this domain. Those features include: 1—granules, 2—dense pink granules, 3—dense blue granules, 4—thin cytoplasm rim, 5—abundant cytoplasm, 6—multilobed nucleus (3–5), 7—bilobed nucleus, 8—kidney bean nucleus, 9—large and eccentric Nucleus, 10—dense, round, small nucleus. To develop the domain knowledge vectors, these ten features were framed as Boolean feature vectors

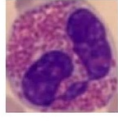
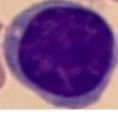
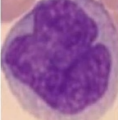
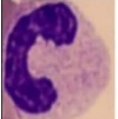
Image	Class & Features	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	Domain Knowledge Vector for the class
	<u>EOSINOPHIL</u> Feature 1 Feature 2 Feature 7	1	1	0	0	0	0	1	0	0	0	[1, 1, 0, 0, 0, 0, 1, 0, 0, 0]
	<u>LYMPHOCYTE</u> Feature 4 Feature 10	0	0	0	1	0	0	0	0	0	1	[0, 0, 0, 1, 0, 0, 0, 0, 0, 1]
	<u>MONOCYTE</u> Feature 5 Feature 8 Feature 9	0	0	0	0	1	0	0	1	1	0	[1, 1, 0, 0, 0, 0, 1, 0, 0, 0]
	<u>NEUTROPHIL</u> Feature 1 Feature 6	1	0	0	0	0	1	0	0	0	0	[1, 0, 0, 0, 0, 1, 0, 0, 0, 0]

Figure 2. Visual features and domain knowledge vectors for the four types of WBCs.

for each class of WBC (eosinophils, neutrophils, lymphocytes, and monocytes). There will be 10 Boolean values in the domain knowledge vector for each class related to the 10 features. If it is a characteristic of that WBC class, the value will be 1, otherwise it will be 0. Figure 2 shows the visual characteristics, feature vector attributes, and domain knowledge vectors.

2.3. Data preprocessing

In contrast to other networks, CNNs can detect visual patterns in unprocessed pixel data without much preprocessing [26]. In the proposed work, image segmentation is performed on the image before it is fed to CNN. Image segmentation removes unnecessary parts of an image and provides the necessary regions where the required object can be found. Therefore, it increases the chance of identifying the required object for classification more quickly.

WBCs are included in the dataset images along with other blood elements, so segmenting the WBCs would provide well-preprocessed images to the deep learning model. Segmentation can be performed using deep neural networks [26], but it is time-consuming. In traditional segmentation models, significant parameters are used to reduce time [27]; however, Image Contouring has significant parameters and is faster [28]. A copy of the original input image is created and the colour channel arrangement is changed from BGR to RGB. In the output image, pixel borders are drawn by choosing pixel values that contribute significantly to creating edges. A light grey border is formed with pixels that fall within a range of BGR values. In the following step, thresholding is applied to change all pixels that will contribute to WBC segmentation to a value of 1 (white) and the background pixels to a value of 0 (black). Using the OpenCV library, image contouring [29] is performed on the input image. Image contouring involves two processes: edge detection and contour extraction [30]. First, Gaussian blurring is applied to the image to reduce noise and enhance edge visibility. Next, the Canny edge detection algorithm is used to identify edges based on pixel values that fall within specified threshold ranges.

To exclude edges created by noise, threshold values have been set such that edges caused by noise are not considered. By applying morphological erosion and dilation, the detected edges are refined and enhanced. The contours are extracted from the edged binary image by considering joining edges, and then they are sorted based on their area values ascending. In the sorted contours list, a minimum area rectangle with the actual corner points of the rectangle is identified and converted into a NumPy array. To draw bounding boxes, the Euclidean distance between the top-left corner and the bottom-left corner is calculated. To remove invalid boxes, all other non-zero areas in the bounding box will be filtered out using Euclidean distance. Afterward, the bounding box's center point is determined. Lastly, to draw the contours, a mask image in the black color is created and placed on the input image to draw the contours with the maximum and minimum

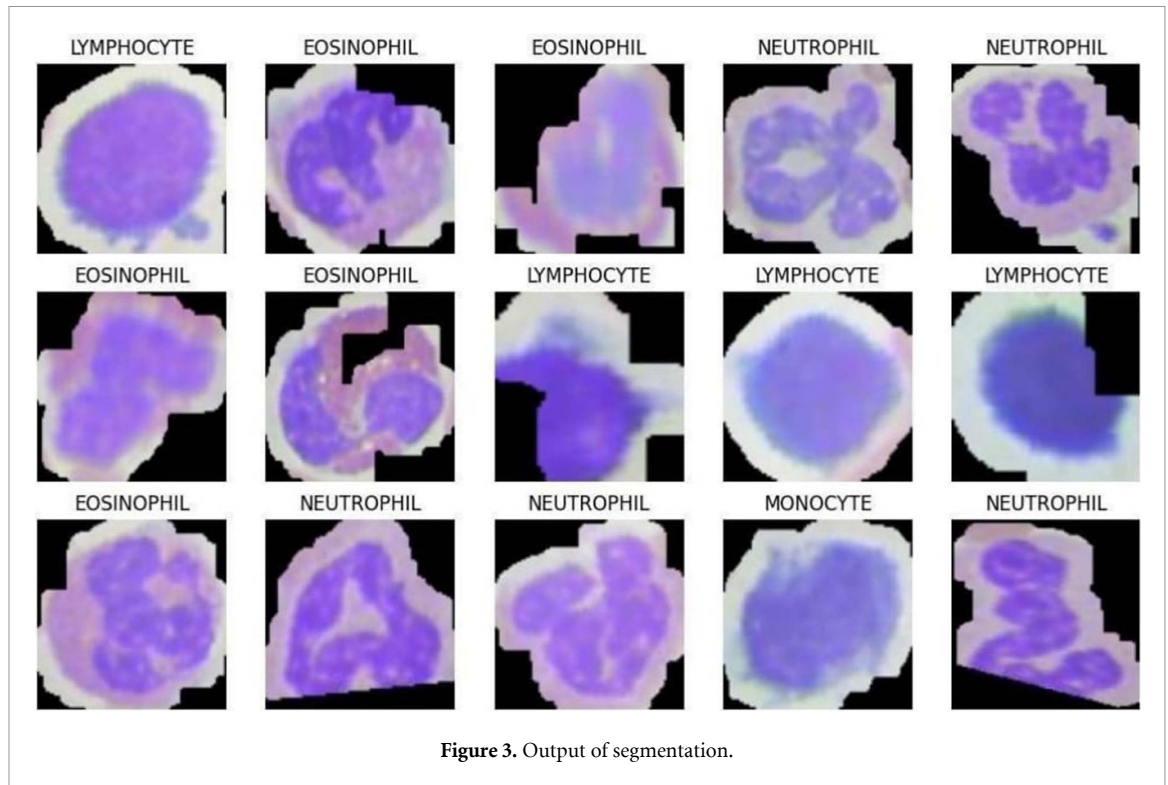


Figure 3. Output of segmentation.

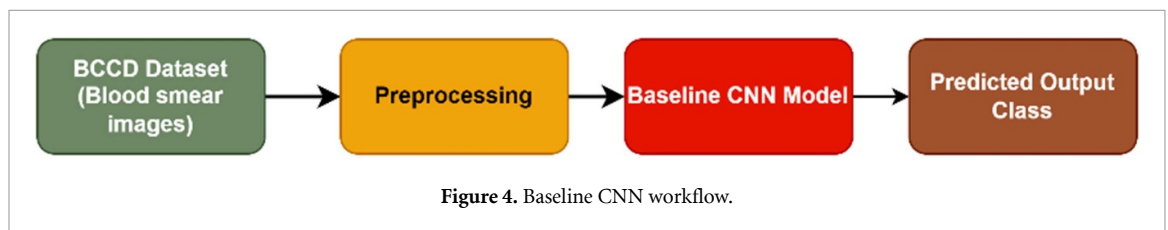


Figure 4. Baseline CNN workflow.

x , and y coordinates and the bounding box's center coordinates. In this case, the contour that is the region of interest will be filled in white and the rest in black. Afterward, the segmented WBCs are obtained by cropping the masked area of interest. By contouring, the cell can be clearly defined from the background, and its edges are clearly defined. Figure 3 shows the output of contouring. Upon segmenting the WBC images, they are resized to $[64 \times 64]$ to fit the model, and their pixels are normalized from 0–255 to 0–1. In the model, this segmentation technique provides a properly delineated WBC image for the CNN.

2.4. Baseline CNN model

This paper aims to classify WBCs from blood smear images with the 'Baseline CNN', which is used as the first step in the work. Figures 4 and 5 show the workflow and architecture of the Baseline CNN. The baseline CNN requires input images of 64×64 pixels and in RGB color channel. The algorithm includes three convolutional layers with varying numbers of filters (64, 128, 256), followed by maximum pooling and dropouts. Each convolutional layer is followed by an activation function called a rectified linear unit (ReLU). A max pooling layer with a pool size of (2, 2) is used to downsample feature maps. Dropout regularization with a dropout rate of 0.25 is implemented after each max pooling layer to prevent overfitting. A flattened layer converts the output feature maps from the final convolutional layer into a one-dimensional (1D) array. Flattened features are passed through a dense layer with 1024 units and a ReLU activation function. We apply a dropout layer after this dense layer with a 0.25 dropout rate.

The proposed CNN model includes two additional dense layers with 64 units and ReLU activation functions. The final dense layer consists of 4 units and applies the SoftMax activation function. This layer produces the output probabilities for each class in the classification problem. Since the problem involves multi-class classification, SoftMax activation is implemented to ensure that the output values are normalized to represent probabilities. The loss function used is sparse categorical cross-entropy, which works well for multi-class classification jobs where the labels are integers. Compilation of the model is performed using Adam optimizer. The metrics used for evaluation are accuracy, precision, recall, F1-score, and specificity.

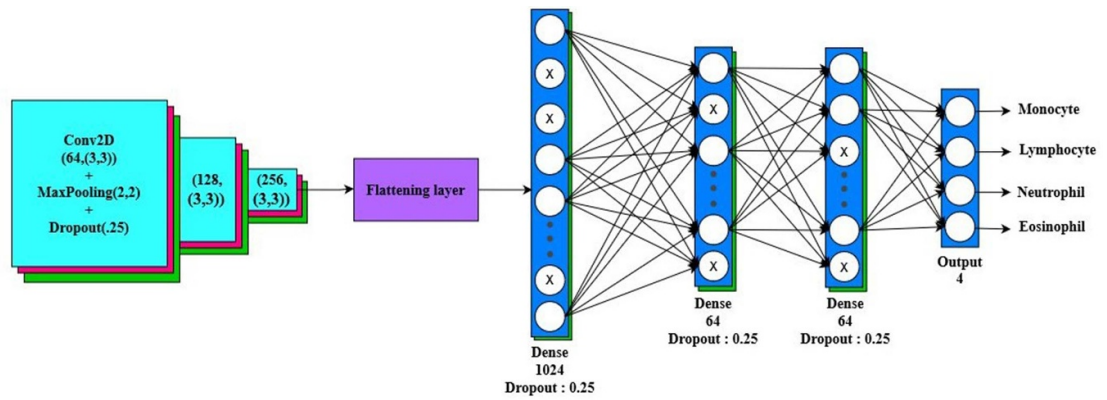


Figure 5. Baseline CNN architecture.

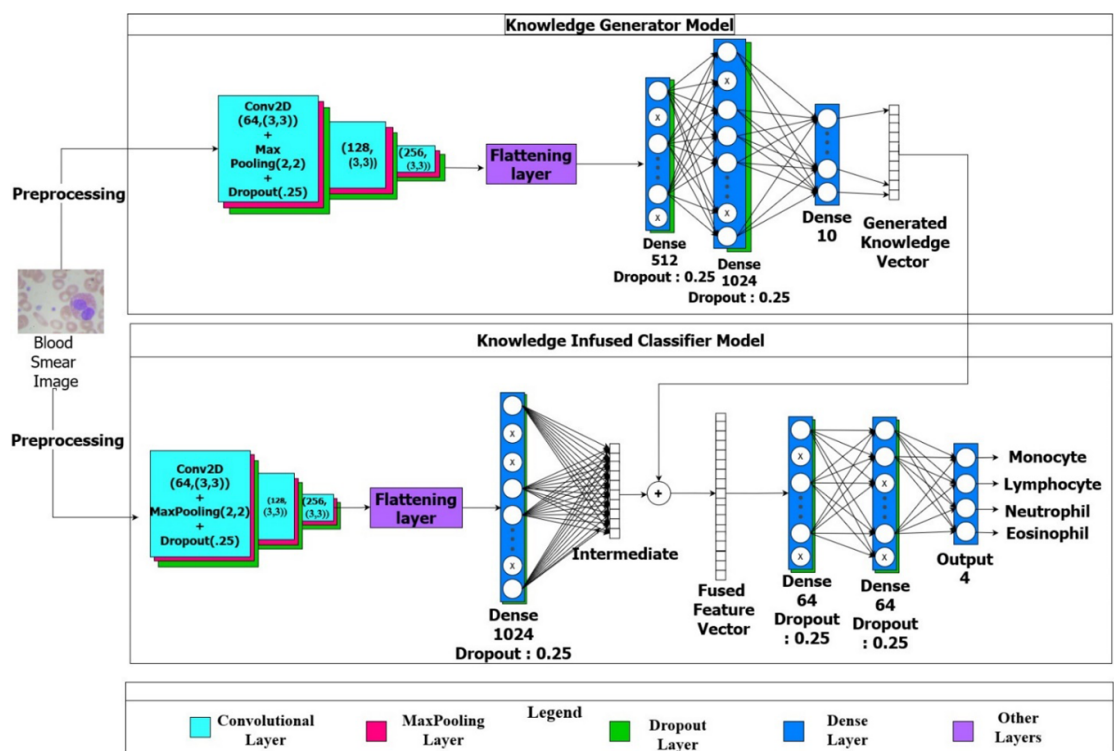


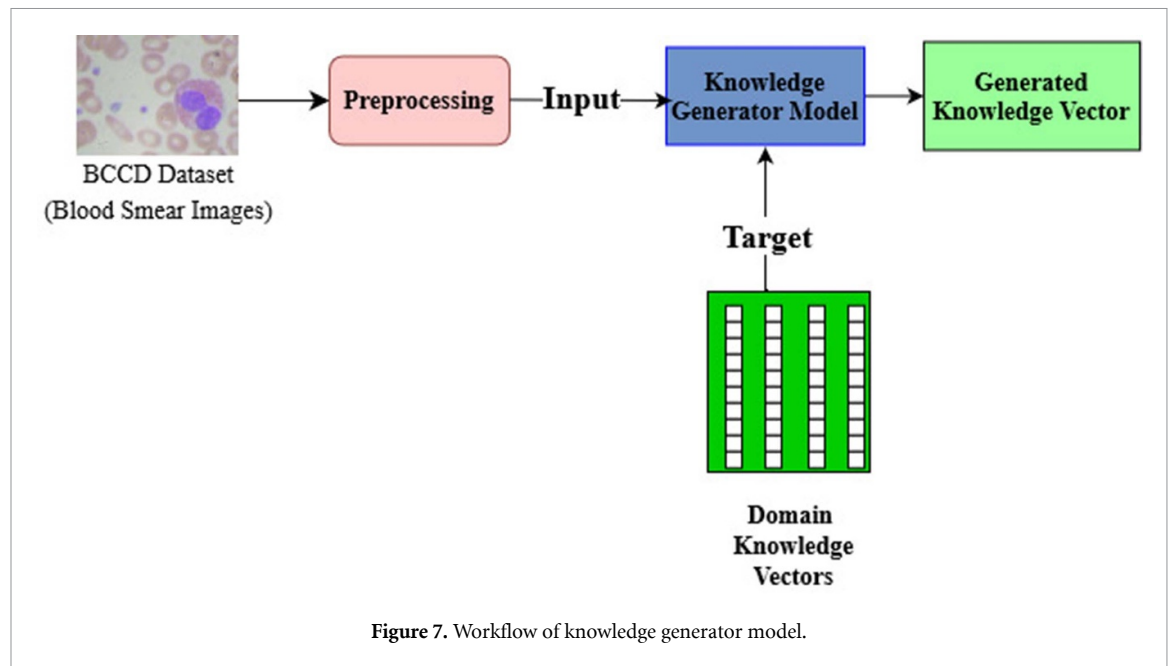
Figure 6. Architecture of the proposed WBC-KICNet.

2.5. Proposed WBC-KICNet

WBC-KICNet uses a CNN model that is less complex than other deep-learning models for WBC classification. In this model, a framework is called a ‘knowledge generator (KG)—knowledge infused classifier (KC)’. Based on CNN, the KG model and the KC models are developed. In the KG model, domain knowledge features are extracted from the pre-processed image and converted to a 1×10 knowledge feature vector. A KC model combines knowledge features and features extracted from an input image to produce fused features. The fused features are then used to classify the image. Infusing domain knowledge into the classifier model is the function of the KG model, which provides the knowledge vector that is needed by the KC model. Figure 6 illustrates the detailed WBC-KICNet architecture.

2.5.1. KG model

Knowledge generator (KG) is a custom CNN model used for feature extraction. A CNN is trained to generate domain knowledge feature vectors from pre-processed BCCD dataset images. The four domain knowledge vectors provided by the hematologist have significant features that help identify the type of WBC which is subjective of the WBC image. The elements in the vector represent the features characteristic of the WBC



class with a '1' and '0' otherwise. The feature number and its characteristic is mentioned in section 2.2 of the manuscript. All features characteristic of the particular class need not be present in every WBC image, but they are important for classifying the WBC. The fixed domain knowledge vectors encapsulate the essential characteristics of each WBC class. Hence, we have trained a CNN model to extract those features by providing them as target vectors. The CNN's multi-layer design extracts hierarchical features that capture both low-level and high-level details, attention mechanisms within the CNN dynamically focus on the most relevant parts of the images, and multi-scale feature extraction ensures that features at different levels of granularity are recognized. These strategies collectively ensure that the domain knowledge vectors contribute effectively to generating reliable knowledge vector predictions that accurately reflect the specific content of each WBC image, accounting for variability and incompleteness in the image, thereby improving the KG model's overall performance and robustness. The sigmoid activation function in the final layer squashes feature values between 0 and 1, generating knowledge vectors that represent each image within a class. By allowing for continuous values rather than strict binary outputs, these vectors capture varying degrees of feature presence, effectively accounting for variations within the class. Figure 7 illustrates the workflow of the KG model. The model uses pre-processed images as input and gives domain knowledge vectors as output. In this model, there are three convolutional layers with increasing numbers of filters (64, 128, 256) and a ReLU activation function. A max pooling layer with a pool size of 2 and a dropout layer with a dropout rate of 0.25 are added after each convolutional layer.

Using a flattened layer, the last convolutional layer's feature map is converted into a 1D array. A dense layer with 1024 units and ReLU activation is used. Before passing the feature map to the final layer with 10 output units and a sigmoid activation function, a dropout layer with a dropout rate of 0.25 is also added. In CNN, ReLU is a non-linear activation function that helps resolve the vanishing gradient issue. ReLU will round up all x values less than and equal to 0 to 0. For values greater than 0, it will return the value as such. It is given by the formula in equation (1),

$$F(x) = 0, \text{ for } x \leq 0 \text{ and } x \text{ for } x > 0. \quad (1)$$

KG is a standalone model that can be integrated with any neural network model to generate knowledge feature vectors. The final dense layer has 10 units, which equals the Domain Knowledge Vector length. As a result, the model output is a vector with the same dimension as the target vectors. The output is considered a 'Generated Knowledge vector' in this case. In this model, sigmoid activation is used, which is compatible with Boolean outputs. The sigmoid function will normalize all the values of x between 0 and 1. It is given by the formula in equation (2). Typically, the Adam optimizer and mean squared error (MSE) loss function are combined in model compilation. It is typical for regression tasks to minimize the disparity between predicted continuous values and actual targets,

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \quad (2)$$

2.5.2. KC model

The knowledge-infused classifier (KC) model uses CNN with knowledge incorporated in it. The architecture diagram of the KC model is shown in figure 6. Using the baseline CNN model architecture, the KG model and feature fusion parts are merged. In the KC model, the preprocessed image is used to extract features from the image. The main advantage of CNN is its ability to automatically extract feature information from an image using deep learning methods. Using the flattened layer, the extracted features are converted to a 1D array. This is then given to a dense layer of 1024 neurons activated with ReLu. Following this, the knowledge vectors produced by the KG model are concatenated and fed into two additional dense layers before applying the SoftMax function. Each dense layer consists of 64 nodes activated by ReLU and includes dropout layers with a 25% dropout rate. This setup helps reduce the dimensionality of the combined features and enhances the model's learning capability. There are four nodes in the output layer, one for each of the four classes of WBCs present in the dataset. As part of the output layer, SoftMax activation is used to normalize the output values to represent probability values. Softmax converts a vector of values into a probability distribution. In the output vector, the elements will range from 0 to 1 and sum to 1. The mathematical representation of the SoftMax activation function is given in equation (3). A sparse categorical cross-entropy loss function is selected in the Adam optimizer, which is suitable for multiclass classification with integer labels. Z_i

$$\text{Softmax}(z) \text{ or } \sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^k e^{z_j}} \quad (3)$$

where,

i —set of real numbers

z_i — i th element of the input vector z .

z_j — j th element of the output vector z .

e —base of the natural logarithm (approximately 2.718).

2.6. Performance evaluation

To ensure reproducibility, the hardware and software environment is detailed. The model was developed and tested on Google Colab using NVIDIA Tesla G4 GPU. Version control was maintained with Git. The system comprised an AMD Ryzen 5 4600H processor, 12 GB RAM, and 512 GB storage. The Ubuntu 22.04 Operating System served as the base, while Python along with Pandas (2.1.1), Numpy (1.26.0), Tensorflow (2.14.0), and OpenCV-Python (4.8.1) provided the necessary frameworks and libraries. The model was evaluated using the BCCD dataset which is balanced and has four classes namely, eosinophil, lymphocyte, monocyte, and neutrophil. To enhance clarity and reliability in the evaluation methodology, a unified approach was adopted where the train-test split of 80–20 followed by ten-fold cross validation within the training set. The best-performing model was chosen to ensure consistency in the performance metrics. The metrics used to evaluate the model are accuracy, precision, recall, specificity, F1-score, area under the receiver operating characteristic (ROC) curve, and confidence scores [9]. Training was conducted using the Adam optimizer with a learning rate of 0.001, over 30 epochs, with a batch size of 32.

2.6.1. Description of evaluation metrics

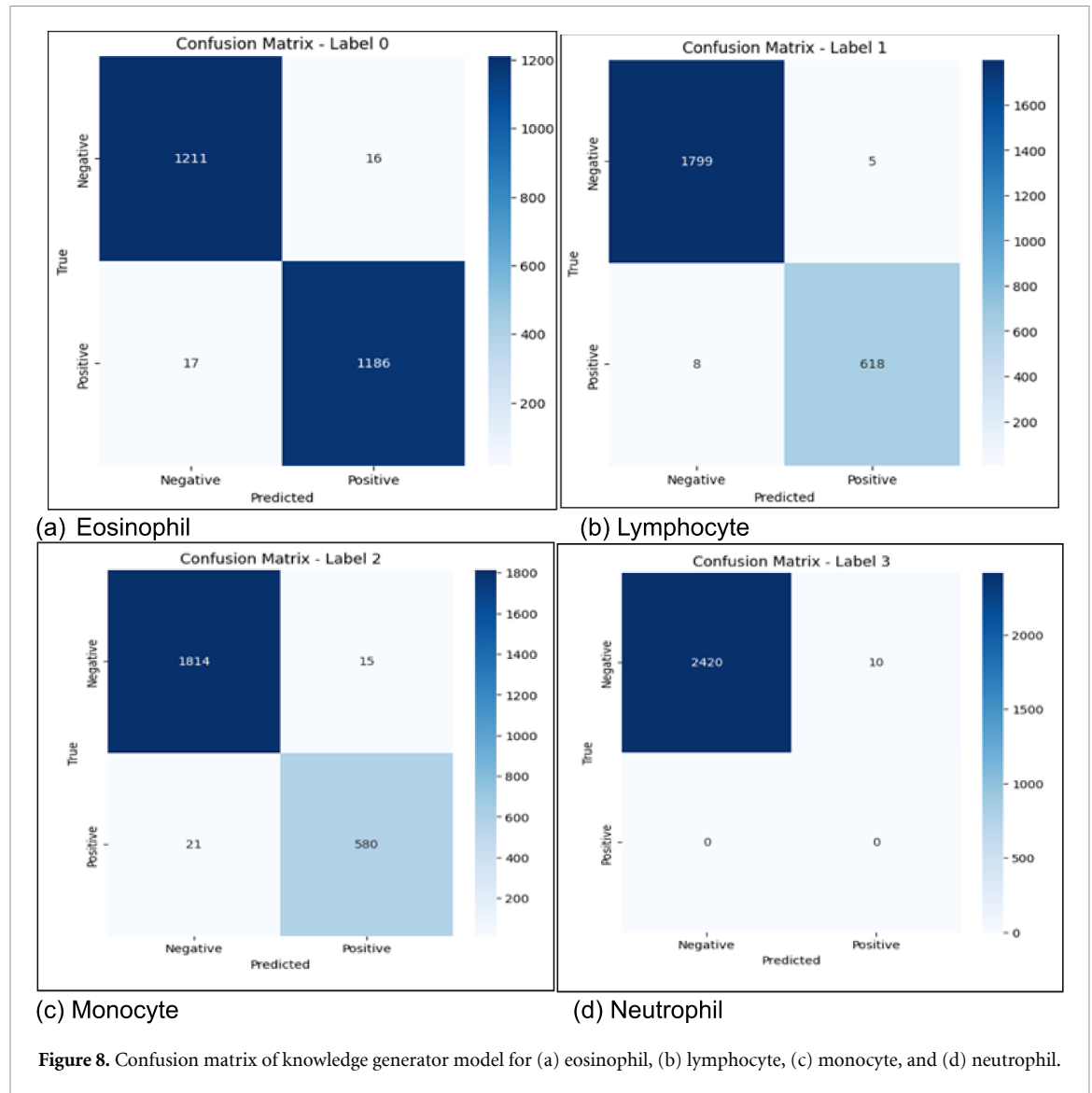
To quantify evaluation measures analytically, it is essential to define true positives (TPs), false positives (FPs), true negatives (TNs), and false negatives (FNs). Understanding the terms is essential when evaluating the performance of a classification model. The WBC-KICNet model is assessed using these values to determine its accuracy, precision, recall, and specificity. These metrics are expressed mathematically by the formulas in equations (4)–(8). In WBC classification, TP, TN, FP, and FN describe different results of the model, which indicate how well it distinguishes between different types of WBCs: a TP occurs when the model correctly identifies a cell belonging to that specific type. An FP occurs when the model incorrectly identifies a cell as belonging to a specific type. A TN occurs when the model correctly identifies a cell that does not belong to a specific type as something else. FN occurs when a cell belonging to a specific type is incorrectly identified as something else,

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} * 100 \quad (4)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} * 100 \quad (5)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} * 100 \quad (6)$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} * 100 \quad (7)$$



$$F1 - \text{Score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (8)$$

3. Numerical results and discussion

3.1. Performance of KG model

The KG CNN was trained and tested on the BCCD dataset. The KG model automates the process of generating domain knowledge vectors as manually visualizing images and generating knowledge vectors is time-consuming for hematologists. Therefore, CNN has been used to automate the process. The KG model is a key component in the framework as it can impact the classifier's performance. The quality of the domain knowledge given to the KC model depends on the correctness of the vector generated by the KG model. The KG model performs with an accuracy of 98.1%. In addition, the precision value is 97.4%, the recall is 97.79%, the specificity is 99.8%, and the F1 value is 97.64%. Figure 8 shows the confusion matrix for each class of WBC. The confusion matrix for neutrophils has a 0 for both TP and TN, which means it does not have any incorrect vectors for this class. The figure 8 shows that there are 8 incorrect vectors for the Eosinophil class, 17 for the Lymphocyte class, and 21 for the Monocyte class. Since the model is a vector generator and not a classifier, the MSE is used which can quantify the average squared error between the ground truth and predicted values.

3.2. Performance comparison between baseline CNN and WBC-KICNet

To better understand the work of incorporating domain knowledge into a regular CNN, the performance of the proposed WBC-KICNet was compared to that of a baseline CNN by training and testing them on the

Table 2. Performance evaluation between the baseline CNN and WBC-KICNet.

WBC class	Accuracy (%)		Precision (%)		Recall (%)		Specificity (%)		F1-score (%)	
	Baseline CNN	WBC-KICNet	Baseline CNN	WBC-KICNet	Baseline CNN	WBC-KICNet	Baseline CNN	WBC-KICNet	Baseline CNN	WBC-KICNet
Eosinophil	81.7	98.7	89	99	82	99	96	99.6	85	99
Lymphocyte	96.6	99.83	99	100	97	100	99	100	98	100
Monocyte	96.7	100	87	100	97	99	95	99.9	91	100
Neutrophil	87.3	98.32	87	98	87	98	95	99.6	87	98
Average	90.5	99.22	90.5	99.25	90.75	99	96.25	99.77	90.25	99.25

BCCD dataset. We have tabulated the average scores of the five metrics for both models in table 2. In the baseline CNN, accuracy, precision, recall, specificity, and F1-scores are 90.5%, 90.5%, 90.75%, 96.5%, and 90.25%, respectively. Other than specificity, the rest of the values are around 90%. As a result of the WBC-KICNet's high accuracy of 99.2%, it predicts 99.22% of all samples correctly, indicating a strong overall performance. The model has an impressively high TP rate of 99.25%, so it accurately classifies true samples as positive. Based on the recall score of 99%, it appears that the model correctly identifies relevant samples in a large majority of cases. It is particularly important to have a high specificity in medical diagnostics to avoid misdiagnosing healthy people as ill. With an exact specificity score of 99.77%, the model has a very low FP rate, meaning it rarely misclassifies positive samples as negative. Despite a high F1-score of 99.25%, the WBC-KICNet achieves a high balance between precision and recall. In a medical diagnosis application such as WBC classification, where precision and specificity are crucial metrics, the proposed WBC-KICNet outperforms the Baseline with challenging scores of 99.25% and 99.77%. The effort of knowledge infusion has helped CNN learn the dataset better and produce reliable results.

3.3. Class-wise performance evaluation

Table 2 shows the class-wise accuracy, precision, recall, F1-score, and specificity values for the baseline CNN model and the proposed WBC-KICNet model. 1. Considering the results of both the proposed and baseline models, the accuracy for monocytes and lymphocytes is higher than that for eosinophils and neutrophils due to their agranular cytoplasm. Furthermore, the proposed WBC-KICNet model has been able to achieve higher accuracy scores for all four classes. The accuracy scores for the four classes are 98.7%, 99.83%, 100%, and 98.32% for WBC-KICNet, compared to 81%, 87%, 96.6%, and 96.7% for the baseline CNN. Based on our best judgment, both models have been trained and tested equally, and the results prove that the CNN prediction in the proposed model across all four WBC classes is at a higher level, thanks to the knowledge given to it. This suggests that the model does not overfit any WBC class and could be used in a routine clinical setting.

3.4. Visualization results of baseline CNN and WBC-KICNet

The confusion matrices in figure 9 show the classification results of the baseline CNN and the WBC-KICNet. A major improvement in the WBC-KICNet is the tapered misclassification of eosinophils and neutrophils. According to the Baseline CNN, 52/622 Eosinophil images were misclassified as neutrophils, whereas the WBC-KICNet misclassified 7/622 actual Eosinophil images as neutrophils. Based on the baseline CNN, 69/599 neutrophils are misclassified as eosinophils, while 8/599 are misclassified as eosinophils based on the WBC-KICNet.

By identifying the confidence level with which it classified those images that were misclassified by the baseline CNN, we have visualized the classification power of the WBC-KICNet. In a Deep Learning model, a confidence score is a value between 0 and 1 that indicates the model's prediction strength. The model's prediction is highly desirable if the confidence score is between 0.7 and 9, but less desirable when it is below 0.7. The confidence scores of baseline CNN and WBC-KICNet on the misclassified images are shown in figure 10. The domain knowledge has helped the model overcome the confusion of misclassifying neutrophils and eosinophils. A second improvement is the improved classification of monocytes. Monocytes and neutrophils can be misclassified because they share immune receptors, including T cell receptors, with T cells. According to the baseline, 47, 22, and 16 monocyte images were misclassified as neutrophils, eosinophils, and lymphocytes, respectively. Similarly, the baseline model shows poor classification performance for monocytes. As a result of the addition of feature-based domain knowledge, the WBC-KICNet has increased the monocyte classification by a bare minimum of 2/597 images misclassified as neutrophils. Third, classification performance of lymphocytes have shown an appreciable improvement. With the proposed model, 621/622 images of the lymphocyte class are correctly identified, while the baseline

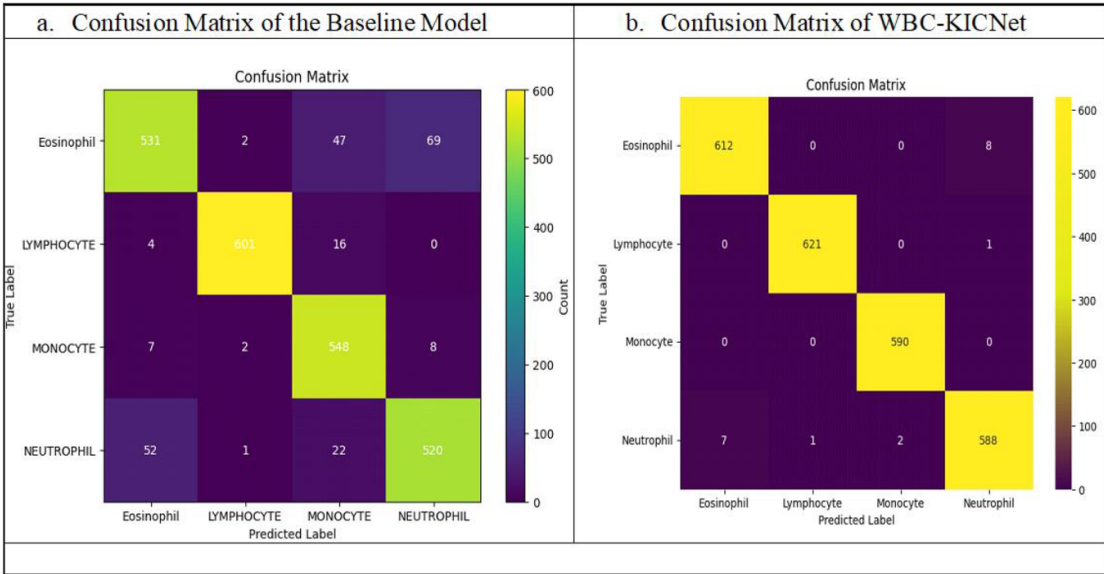


Figure 9. Confusion matrix of baseline CNN (a) and WBC-KICNet (b).


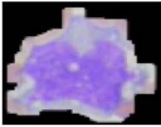



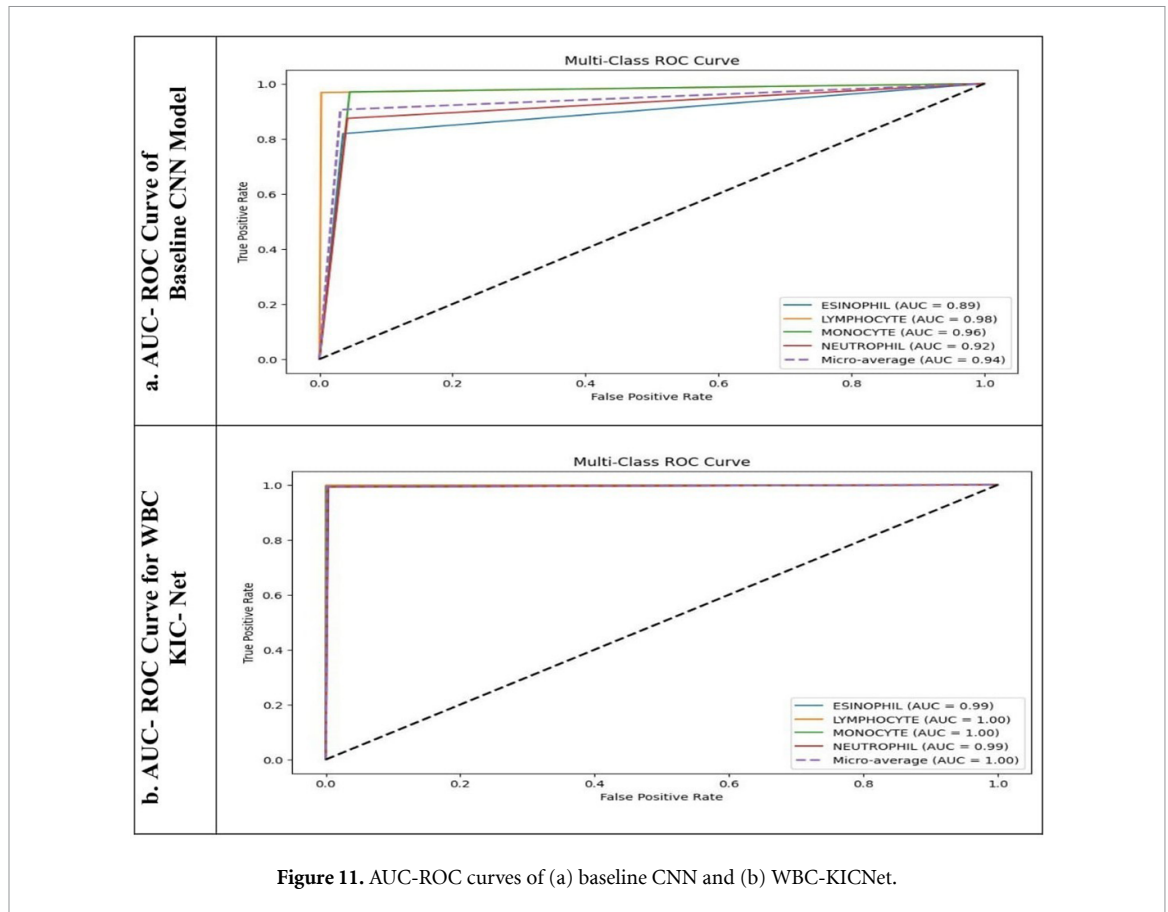
Baseline CNN	 <div>Actual Label: Eosinophil Predicted Label: Neutrophil Confidence Score: 0.526</div>	 <div>Actual Label: Monocyte Predicted Label: Lymphocyte Confidence Score: 0.9461</div>	 <div>Actual Label: Eosinophil Predicted Label: Neutrophil Confidence Score: 0.7422</div>	 <div>Actual Label: Eosinophil Predicted Label: Monocyte Confidence Score: 0.9663</div>	 <div>Actual Label: Neutrophil Predicted Label: Monocyte Confidence Score: 0.3803</div>
Proposed WBC-KICNet	<div>Actual Label: Eosinophil Predicted Label: Eosinophil Confidence Score: 0.9993</div>	<div>Actual Label: Monocyte Predicted Label: Monocyte Confidence Score: 0.9999</div>	<div>Actual Label: Eosinophil Predicted Label: Eosinophil Confidence Score: 0.9997</div>	<div>Actual Label: Eosinophil Predicted Label: Eosinophil Confidence Score: 0.9990</div>	<div>Actual Label: Neutrophil Predicted Label: Neutrophil Confidence Score: 0.9990</div>

Figure 10. Confidence scores of baseline CNN and WBC-KICNet.

CNN could only classify 601/606 test samples correctly. The fourth improvement is the aggregated high classification performance across all four categories. According to the classification results the correct classifications are 590/592 for monocytes, 612/622 for eosinophils, 621/622 for lymphocytes, and 588/597 for neutrophils. As a result, to the best of our knowledge and genuineness, the sincere effort of adding domain knowledge to improve WBC classification performance has been successful.

Figure 11’s ROC curves demonstrate the diagnostic capabilities of the Baseline CNN and WBC-KICNet models as the discrimination thresholds are adjusted. The X-axis represents the FP rate or specificity, which represents the number of real positive classes misclassified as negatives, and the Y-axis represents the TP rate or sensitivity, formula in (6). In figure 11, the ROC curves for the WBC-KICNet model reach the top-left corner of the plot for all four classes, indicating an ideal scenario with a high TP rate and a low FP rate across various threshold values. In contrast, only the ROC curve for the lymphocyte class in the baseline CNN model touches the top-left corner, while the other classes do not. Additionally, in figure 11 the AUC values



for both the WBC-KICNet and the Baseline CNN models are depicted in order to quantitatively evaluate the performance of the classifiers. For all four classes, the AUC values are above 0.5, indicating good overall performance for both models. The Baseline CNN model demonstrates strong performance, particularly for Lymphocyte and Monocyte classifications, with AUC values of 0.98 and 0.96, respectively. However, the Eosinophil classification shows a lower AUC of 0.89. On the other hand, the WBC KIC-Net model exhibits markedly superior performance across all classes, with AUC values approaching or reaching 1.00. This suggests near-perfect classification performance for all WBC types. As a result, the WBC-KICNet faces reduced overfitting problems.

3.5. Accuracy comparison

The graphs in figure 12 demonstrate the ability of both the Baseline CNN and the proposed WBC-KICNet models to generalize effectively across different folds and epochs. The first graph on the left depicts the training and validation accuracies over 30 epochs. Initially, the training accuracy starts around 0.5 and rapidly increases, while the validation accuracy starts higher, around 0.7, and follows a similar upward trend. Both accuracies converge and stabilize after approximately 10 epochs, with the training accuracy slightly surpassing the validation accuracy. The graph on the right shows the learning ability of the WBC-KICNet model where, the training accuracy begins around 0.8 and quickly reaches close to 1.0 by the second epoch, after which both training and validation accuracies stabilize at nearly perfect levels. This graph illustrates the model's strong performance showing minimal deviation between the training and validation accuracies, suggesting excellent generalization and minimal overfitting.

3.6. Statistical analysis

Based on the k -fold cross-validation accuracy, a one-tailed t -test was performed on the models [19]. The k -fold cross-validation was performed on the training set and the results are tabulated in table 3. Despite the above empirical analysis confirming that WBC-KICNet performs better than baseline CNN, it is important to confirm the results statistically. The performance of both models is compared using a one-tailed t -test (hypothesis testing). The null hypothesis H_0 is that there is no significant difference between Baseline CNN

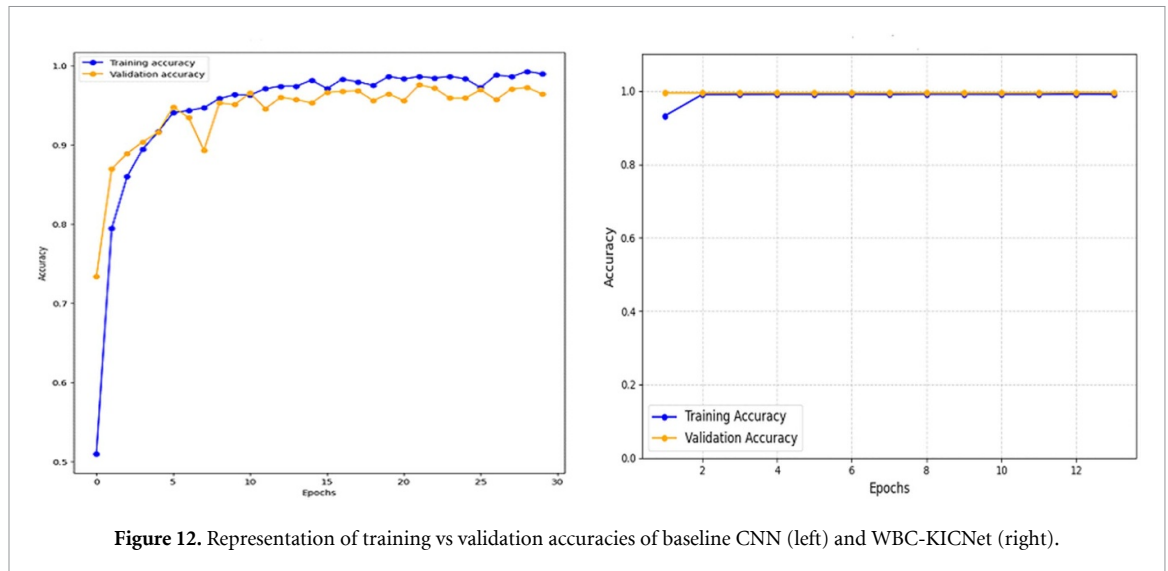


Figure 12. Representation of training vs validation accuracies of baseline CNN (left) and WBC-KICNet (right).

Table 3. Statistical t -test evaluation measures.

Fold	K -fold cross validation accuracies of baseline CNN (T1)	K -fold cross-validation accuracies of proposed WBC-KICNet (T2)	Difference (T2 – T1)	Square of deviation
1	97.11	99.09	1.98	0.29
2	96.21	99.09	2.88	0.13
3	95.63	99.42	3.79	1.61
4	96.13	98.76	2.63	0.01
5	96.95	99.25	2.3	0.05
6	97.20	98.76	1.56	0.92
7	98.10	99.42	1.32	1.44
8	95.79	99.17	3.38	0.74
9	97.11	99.01	1.9	0.38
10	95.71	99.17	3.46	0.88
Average			2.52	6.46

and WBC-KICNet, whereas the alternative hypothesis H_1 is that WBC-KICNet is more efficient than baseline CNN at a 5% significance level. On the K -fold ($K = 10$) cross-validation accuracy of Baseline CNN and WBC-KICNet, a one-tailed t -test with 0.05 as the significance level is applied, and the p -value obtained is 0.000 01. If the p -value is less than α , then the null hypothesis is rejected, otherwise it is not rejected. In this case, the p -value of compared accuracy is less than 0.05, indicating that the null hypothesis H_0 is rejected. With a 95% confidence level, the statistical results confirm that WBC-KICNet is more efficient than baseline CNN.

3.7. Quantitative result analysis with existing deep learning models

A comparison of the proposed WBC-KICNet model and nine existing CNN-based classification models is presented in table 4 and pre-trained models in table 5. In the works, values that are not reported are mentioned as ‘NP’. Models based on BCCD datasets are considered righteous comparisons. WBC classification has been attempted by authors using CNNs and pre-trained CNNs, either in their original form or modified and the results of testing is compared. Tiwari *et al* [1] have proposed a CNN model for the classification of WBCs and achieved an accuracy of 78%. Cheuque *et al* [2] have performed a two-stage classification of WBC using Faster-RCNN in the first stage and two parallel CNNs in the second stage and achieved an accuracy of 98.4%. Chen *et al* [4] developed a framework by coupling ResNet and DenseNet with spatial and channel attention modules and obtained an accuracy of 88.4%. Togacar *et al* [6] proposed a model for WBC classification by using Maximal Information Coefficient and Ridge feature selection for feature extraction and CNN for classification and achieved an accuracy of 97.95%. Banik *et al* [7] have developed a CNN model by fusing the features of the first and the last convolutional layers for WBC classification thereby considering the morphological characteristics of WBC and achieving an accuracy of 96%. Wang *et al* [9] used transfer learning in SE-ResNeXt 50 and utilized attention mechanisms with squeeze

Table 4. Performance of proposed WBC-KICNet vs existing deep learning models on BCCD dataset.

Authors/Models	Accuracy %	Precision %	Recall %	Specificity %	F1-score %
Tiwari <i>et al</i> (Double CNN model) [1]	78	88	83	NP	83
Banik <i>et al</i> (Nucleus-based CNN classification) [7]	96	96.25	96	98.6	96
Togacar <i>et al</i> (CNN & feature selection) [6]	97.95	NP	NP	NP	NP
Yao <i>et al</i> (TWO DCNN) [19]	NP	91.6	91.6	NP	91.6
Cheque <i>et al</i> (ML-CNN) [2]	98.4	98.4	98.4	NP	98.4
Ramya <i>et al</i> (CNN-PSO) [16]	99.1	NP	94.56	98.78	NP
WBC-KICNet (Proposed model)	99.22	99.25	99	99.77	99.25
NP: Not reported					

Table 5. Performance of proposed WBC-KICNet vs pre-trained networks based deep learning models on BCCD dataset.

Authors/Models	Accuracy %	Precision %	Recall %	Specificity %	F1-score %
Chen <i>et al</i> (Pre-trained Rest-Net and DenseNet with SCAM) [4]	88.44	90.84	88.45	NP	88.73
Wang <i>et al</i> (WBC-AMNet) [9]	98.39	90.72	NP	89.22	89.48
Sharma <i>et al</i> (Transfer learning using DenseNet 121) [17]	98.84	99.33	98.85	99.61	NP
WBC-KICNet (Proposed model)	99.22	99.25	99	99.77	99.25
NP: Not reported					

and excitation and gather-excite modules for WBC classification and obtained an accuracy of 98.39%. Ramya *et al* [16] proposed a CNN-based classification of WBC and achieved an accuracy of 99.1% which is in par with the accuracy of the proposed model but it is observed that recall and specificity values are higher at 99% and 99.77% respectively. Yao *et al* [19] proposed a methodology based on two-module weighted optimized deformable CNN for WBC classification. Sharma *et al* [17] used DenseNet 121 model optimized with pre-processing techniques such as normalization and data augmentation and achieved an accuracy of 98.84%. However, while compared with the experimental results of the works in literature, the proposed WBC-KICNet model based on infusing domain knowledge into a CNN using feature fusion stands out with an accuracy of 99.22%. The other evaluation measures displayed in the tables 4 and 5 also prove that the WBC-KICNet model outperforms the existing works in the literature.

The performance of WBC-KICNet is finally compared with five classical CNN models: Inception V3, DenseNet121, MobileNetV2, RestNet 50, and VGGNet16. On the BCCD dataset, the models were trained for WBC classification through transfer learning. Model weights obtained from the ImageNet dataset were retained in the convolutional layers, and only the output softmax activation function parameters were set to 4. According to the comparison between the pre-trained models and the proposed WBC-KICNet model, the proposed WBC-KICNet performs better for the BCCD dataset with the knowledge feature vectors fused with CNN extracted features. Table 6 summarizes the results.

The novel WBC-KICNet model offers competitive accuracy in classifying WBCs while requiring minimal computational resources. Its design allows fast training and analysis, making it ideal for time-sensitive applications like emergency medicine. WBC-KICNet utilizes standard color images, reducing data acquisition costs and enabling real-world use. Compared to existing methods, it achieves superior accuracy and precision, leading to more reliable WBC classification. The model also avoids complexities associated with pre-trained models and simplifies the process compared to ensemble or hybrid approaches. Future research could explore using pre-trained models with WBC-KICNet and investigate the model's applicability to other medical image analysis tasks. Additionally, the model's ability to learn kernel weights autonomously holds promise for improved interpretability and potentially even higher accuracy.

4. Conclusion

A CNN based model called the WBC-KICNet was developed to accurately classify WBCs from blood smear images. A key objective was to improve the model's accuracy by incorporating WBC-specific knowledge. To delineate the WBCs in the dataset, Boolean Domain Knowledge vectors were combined with a contouring preprocessing step. We developed two deep-learning models to achieve our goal. The first model generates the knowledge vector from the images, and the second model uses the knowledge vector and image data to

Table 6. Result comparison of proposed WBC-KICNet with classical CNN models on BCCD dataset.

Model	WBC class	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Inception V3	Eosinophil	95.66	98.79	95.66	97.20
	Lymphocyte	98.87	99.35	98.87	99.11
	Monocyte	90.35	96.38	90.35	93.27
	Neutrophil	97.56	89.02	97.56	93.09
	Average	95.61	95.88	95.61	95.67
DenseNet121	Eosinophil	98.71	98.07	98.71	98.39
	Lymphocyte	99.68	99.67	99.68	99.68
	Monocyte	99.32	98.81	99.32	99.07
	Neutrophil	97.49	98.64	97.42	98.06
	Average	98.8	98.79	98.78	98.8
MobileNetV2	Eosinophil	96.93	97.56	96.93	97.24
	Lymphocyte	99.67	90.37	99.67	94.80
	Monocyte	92.03	98.90	92.03	95.34
	Neutrophil	92.80	95.85	92.80	94.30
	Average	95.36	95.67	95.36	95.42
ResNet 50	Eosinophil	93.06	94.28	93.06	93.66
	Lymphocyte	98.71	99.51	98.71	99.11
	Monocyte	98.64	96.67	98.64	97.65
	Neutrophil	94.14	93.98	94.14	94.06
	Average	96.13	96.11	96.13	96.12
VGGNet16	Eosinophil	68.38	99.29	68.38	80.99
	Lymphocyte	100.0	96.58	100.0	98.26
	Monocyte	99.15	94.66	99.15	96.85
	Neutrophil	93.47	75.43	93.47	83.49
	Average	90.25	91.49	90.25	89.89
WBC-KICNet (Proposed model)	Eosinophil	98.7	99.00	99.00	99
	Lymphocyte	99.83	100.00	100.00	100
	Monocyte	100	100.00	99.00	100
	Neutrophil	98.32	98.00	98.00	98
	Average	99.22	99.25	99	99.25

classify the images. A key component of the approach was the incorporation of domain information, which significantly improved the performance of the model. In addition, the Baseline CNN model's accuracy without the knowledge vector was 90.5%. However, domain knowledge enabled this accuracy to reach 99.22%, which is a remarkable achievement. By incorporating domain knowledge into the classification process, we can achieve such an enormous improvement in accuracy. The results of the proposed method have been validated using a statistical hypothesis test (t -test). As knowledge and technology advance, the proposed work will open the door for additional investigation and innovation in medical image analysis and classification. In addition, the work leads to the development of more accurate and effective diagnostic tools and the advancement of medical image processing.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments

We would like to thank Dr Viveka Priyadharshni S, M.D- Immuno-hematology & Transfusion Medicine, Consultant- Transfusion Medicine, KG Hospital, Coimbatore for her time and effort in providing domain knowledge about WBCs.

Conflict of interest

The authors strive to maintain objectivity and transparency in their research. To ensure this, we disclose the following:

- The authors have no financial ties to any companies or organizations that may benefit from the publication of this manuscript.
- No author has any personal or professional relationships that could be perceived as influencing the research presented here.

ORCID iDs

Jeneessha P  <https://orcid.org/0000-0001-8522-0480>

Vinoth Kumar Balasubramanian  <https://orcid.org/0000-0001-8564-5120>

M Murugappan  <https://orcid.org/0000-0002-5839-4589>

References

- [1] Tiwari P, Qian J, Li Q, Wang B, Gupta D, Khanna A, Rodrigues J J P C and de Albuquerque V H C 2018 Detection of subtype blood cells using deep learning *Cognit. Syst. Res.* **52** 1036–44
- [2] Cheuque C, Querales M, León R, Salas R and Torres R 2022 An efficient multi-level convolutional neural network approach for white blood cells classification *Diagnostics* **12** 248
- [3] Cinar A and Tuncer S A 2021 Classification of lymphocytes, monocytes, eosinophils, and neutrophils on white blood cells using hybrid Alexnet-GoogleNet-SVM *SN Appl. Sci.* **3** 1–11
- [4] Chen H, Liu J, Hua C, Feng J, Pang B, Cao D and Li C 2022 Accurate classification of white blood cells by coupling pre-trained ResNet and DenseNet with SCAM mechanism *BMC Bioinform.* **23** 282
- [5] Firat H 2024 Classification of microscopic peripheral blood cell images using multi-branch light weight CNN- based model *Neural Comput. Appl.* **36** 1599–620
- [6] Togacar M, Ergen B and Comert Z 2020 Classification of white blood cells using deep features obtained from convolutional neural network models based on the combination of feature selection methods *Appl. Soft Comput.* **97** 106810
- [7] Banik P, Saha R and Kim K 2020 An automatic nucleus segmentation and CNN model-based classification method of white blood cell *Expert Syst. Appl.* **149** 113211
- [8] Dong N, Feng Q, Zhai M, Chang J and Mai X 2023 A novel feature fusion based deep learning framework for white blood cells classification *J. Ambient Intell. Humaniz. Comput.* **14** 9839–51
- [9] Wang Z, Xiao J, Li J, Li H and Wang L 2022 WBC-AMNet: automatic classification of WBC images using deep feature fusion network based on focalized attention mechanism *PLoS One* **17** e0261848
- [10] Wang Q, Wang J, Zhou M, Li Q, Wen Y and Chu J 2021 A 3D attention networks for classification of white blood cells from microscopy hyperspectral images *Opt. Laser Technol.* **139** 106931
- [11] Ha Y, Du Z and Tian J 2022 Fine-grained interactive attention learning for semi-supervised white blood cell classification *Biomed. Signal Process. Control* **75** 103611
- [12] Shah R, Shastri J, Bohara M H, Panchal Y and Goel P 2022 Detection of different types of blood cells: a comparative analysis *IEEE 2022 Int. Conf. on Distributed Computing and Electrical Circuits and Electronics (Ballari, India)* pp 1–5
- [13] Lei X and Chen Y 2012 Multiclass classification of microarray data samples with flexible neural tree *Spring Congress on Engineering and Technology* (<https://doi.org/10.1109/SCET.2012.6341960>)
- [14] Wang H, Wong H, Zhu H and Yip T 2009 A neural network-based biomarker association information extraction approach for cancer classification *J. Biomed. Inf.* **42** 654–66
- [15] Girdhar A, Kapur H and Kumar V 2022 Classification of white blood cell using convolution neural network *Biomed. Signal Process. Control* **71** 103156
- [16] Balasubramanian K, Anantha Moorthy N P and Ramya K 2022 An approach to classify white blood cells using convolutional neural network optimized by particle swarm optimization algorithm *Neural Comput. Appl.* **34** 16089–101
- [17] Sharma S, Gupta S, Gupta D, Juneja S, Gupta P, Dhiman G and Kautish S 2022 Deep learning model for the automatic classification of white blood cells *Comput. Intell. Neurosci.* **2022** 7384131
- [18] Shin H-C, Roth H R, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D and Summers R M 2016 Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning *IEEE Trans. Med. Imaging* **35** 1285–129
- [19] Yao X, Sun K, Bu X, Zhao C and Jin Y 2021 Classification of white blood cells using weighted optimize deformable convolutional neural networks *Artif. Cells, Nanomed. Biotechnol.* **49** 147–55
- [20] Zhu J-Y, Park T, Isola P and Efros A 2017 Unpaired image-to-image translation using cycle-consistent adversarial networks *Proc. IEEE Int. Conf. on Computer Vision* pp 2223–32
- [21] Xie X, Niu J, Liu X, Chen Z, Tang S and Yu S 2021 A survey on incorporating domain knowledge into deep learning for medical image analysis *Med. Image Anal.* **69** 101985
- [22] Tian C, Chen Y, Zhang J and Feng Y 2024 Integrating domain knowledge with deep learning model for automated worker activity classification in mobile work zone *J. Inf. Technol. Constr.* **29** 264–80
- [23] Xie Y, Xia Y, Zhang J, Song Y, Feng D, Fulham M and Cai W 2019 Knowledge based collaborative deep learning for benign—malignant lung nodule classification on chest CT *IEEE Trans. Med. Imaging* **38** 991–1004
- [24] Liu T, Guo Q, Lian C, Ren X, Liang S, Yu J, Niu L, Sun W and Shen D 2019 Automated detection and classification of thyroid nodules in ultrasound images using clinical knowledge-guided convolutional neural networks *Med. Image Anal.* **58** 101555
- [25] Kaggle BCCD Dataset (available at: www.kaggle.com/datasets/paultimothymooney/blood-cells) (Accessed 4 February 2024)
- [26] Joshi S et al 2019 Issues in training a convolutional neural network model for image classification *Adv. Comput. Data Sci.* **10** 282–93
- [27] Chen X et al 2019 Learning active contours for medical image segmentation *IEEE 2019 CVF/ Conf. on Computer Vision and Pattern Recognition, 2019 (Long Beach, CA, USA)* pp 11624–32
- [28] Fan Z, Herrick J E, Saltzman R, Matteis C, Yudina A, Nocella N, Crawford E, Parker R and Van Zee J 2017 Measurement of soil color: a comparison between smartphone camera and the munsell color charts *Soil Sci. Soc. Am. J.* **18** 1139–46
- [29] Yang Y, Hou X and Ren H 2022 Efficient active contour model for medical image segmentation and correction based on edge and region information *Expert Syst. Appl.* **194** 116436
- [30] Gong X-Y et al 2018 An overview of contour detection approaches *Int. J. Autom. Technol.* **15** 2–3