Signal, Image and Video Processing (2025) 19:712
https://doi.org/10.1007/s11760-025-04321-2

ORIGINAL PAPER

BC-YOLO: MBConv-ECA based YOLO framework for blood cell detection

Mustafa Yurdakul[1] · **Şakir Taşdemir**[2]

Received: 5 February 2025 / Revised: 8 May 2025 / Accepted: 31 May 2025 / Published online: 9 June 2025

Abstract

The detection and classification of blood cells from microscopic images plays a vital role in medical diagnosis. However, it is challenging because of the small size, varying shape and dense clustering of cells. Conventional methods rely on manual inspection, which is labor-intensive and depends on expert knowledge. In contrast, Deep Learning (DL) based approaches significantly speed up the process and provide more reliable and consistent results. However, there should be a trade-off between computational costs, model size and accuracy. In this work, we propose the Blood Cell You Only Look Once (BC-YOLO) model built on the YOLOv11 architecture. The backbone of the model is enhanced by replacing traditional convolution structures with MBConv-ECA blocks, resulting in lighter and more efficient feature extraction. This modification reduces computational complexity while increasing accuracy, resulting in a more robust object detection model. The proposed model accurately detects red blood cells (RBC), white blood cells (WBC) and platelets, automating the analysis of microscopic images. Experimental results show that the BC-YOLO Medium model achieves 95.89% mAP@0.5, 92.2% precision and 96.3% recall with 18.5 million parameters and 57.4 GFLOP. The model outperforms not only YOLOv11 but also other YOLO variants and existing studies in the literature. Furthermore, a user-friendly interface has been developed so that end-users can easily upload and analyze microscopic blood cell images as well as inspect the detected cells. In addition, Grad-CAM-based explainability visualizations are also presented to better understand the model's decision-making processes and increase transparency for medical professionals.

Keywords YOLO · Object detection · Blood cell · ECA · MBConv

1 Introduction

Cells are the building blocks of all living organisms and play a critical role in maintaining the body's healthy functioning [1]. Blood cells, which are transported through the circulatory system, perform tasks such as transporting oxygen to tissues, regulating the immune system, and enabling blood clotting [2]. There are three main types of blood blood cells: red blood cells (RBCs), white blood cells (WBCs), and platelets. RBCs transport oxygen thanks to the hemoglobin protein and send carbon dioxide to the lungs to be eliminated [3]. WBCs defend against infections and regulate immunity [4]. Platelets prevent bleeding in case of injury [5]. The numerical and morphological analysis of blood cells is crucial for the early diagnosis of diseases such as leukemia [6], anemia [7], bleeding disorders [8], and thrombosis risk [9]. In traditional methods, these analyses are performed manually under a microscope, which is time-consuming and prone to observer error. These challenges have led to an increased need for computer-assisted systems. In recent years, DL-based models have demonstrated significant success in disease diagnosis due to their ability to recognize complex patterns [10–18]. Among these are cancer classification, analysis of multiple data sets for lung diseases, and interpretable brain tumor detection [19–22]. These approaches not only improve diagnostic accuracy but also enhance interpretability and clinical reliability [23, 24]. DL models are also effectively used in the detection and classification of blood cells.

✉ Mustafa Yurdakul
mustafayurdakul@kku.edu.tr

Şakir Taşdemir
stasdemir@selcuk.edu.tr

[1] Computer Engineering Department, Kirikkale University, Kirikkale, Turkey

[2] Computer Engineering Department, Selcuk University, Konya, Turkey

Zou et al. [25] proposed ICAFF-MobileNetV2 model for WBC classification using multiscale feature extraction module coordinate-based attention mechanism and label smoothing for class imbalance, achieving 98.54% accuracy, 98.21% recall, 98.56% precision and 98.38% f1-score.

Saidani et al. [26] achieved 99.86% accuracy and 99% precision, recall, f1-score with multi-stage preprocessing and a custom CNN model optimized for classification of eosinophils, lymphocytes, monocytes and neutrophils.

Tarimo et al. [27] developed a two-stage model based on YOLOv5 and Vision Transformer (ViT) for WBC detection and classification. In the first stage, cell and nucleus regions were detected with YOLOv5, and in the second stage, classification was performed with ViT_L16 and 96.449% accuracy, 95.663% precision, 95.890% sensitivity and 95.694% f1-score were obtained.

Parab et al. [28] classified RBCs into 10 classes with a CNN-based model including grayscale, Canny edge detection and area-based filtering, achieving 96.7% precision, 96.6% recall, 99.6% specificity and 97.3% f1-score.

Kang et al. [29] proposed the CST-YOLO model based on YOLOv7 and Swin Transformer and used W-ELAN, MCS and CatConv components to improve small object detection The model achieved 92.7% mAP@0.5 on the Blood Cell Count and Detection Dataset(BCCD) dataset, 3.1% better than YOLOv7 and 0.4% better than YOLOv5x.

Fang et al. [30] compared K-means and U-Net+ResNet models for WBC segmentation and classification on BCCD dataset; ResNet18 achieved 97.82% accuracy in classification and U-Net+ResNet18 achieved 98.79% accuracy in segmentation.

Ren et al. [31] proposed the UKSSL model, which consists of MedCLR (Medical Comparative Learning Representation) and UKMLP (Underlying Knowledge Based Multilayer Perceptron) components. The model achieved 94.3% accuracy, 94.3% f1-score, 94.5% recall and 94.1% precision on BCCD with 50% labeled data.

Liu [32] conducted a comparative study between YOLOv5 and Faster R-CNN models on the BCCD dataset; YOLOv5 initially achieved 93.2% mAP@0.5 and 62.7% mAP@0.5–0.95, while these values increased to 93.8% and 6.32% with additional configurations. Faster R-CNN achieved 89% mAP@0.5 and 64.3% mAP@0.5–0.95. Sazak and Kotan [33] trained the YOLOv11 Large model on the BCCD dataset with data augmentation and optimization strategies and obtained 93.8% mAP@0.5, 66.3% mAP@0.5–0.95, 87.3% precision and 89.9% recall; YOLOv10 Large yielded lower results in terms of these metrics.

These studies show that DL-based approaches have made significant progress in the detection and classification of blood cells. However, the majority of models focus only on specific cell types, cannot handle RBCs, WBCs and platelets at the same time, and are inadequate for real-time use due to high resource requirements. Furthermore, object detection has been secondary in most studies and the lack of explainable artificial intelligence (XAI) limits clinical reliability. Therefore, there is a need for new lightweight, fast and explainable DL models that can detect multiple cell types simultaneously.

The main contributions of this study can be summarized as follows;

- A comprehensive literature analysis was conducted and the limitations of existing methods were identified.
- A total of 23 object detection models, including various versions of YOLO (v8–v11) as well as other state-of-the-art architectures such as EfficientDet and DETR, were evaluated on the BCCD dataset through a comprehensive performance analysis.
- The BC-YOLO model is proposed by replacing the convolution blocks in the backbone of YOLOv11 with a more efficient and lightweight MBConv-ECA, which reduces the computational load and performs more powerful feature extraction.
- The BC-YOLO Medium model outperformed all the tested YOLO versions and other studies in the literature, becoming the most successful model.

YOLOv11 was chosen as the backbone of the BC-YOLO model for blood cell detection due to its C3K2, SPPF and C2PSA components that efficiently detect small and dense objects. Although YOLOv8 offers similar accuracy, YOLOv11 provides higher efficiency [34, 35]. Although Faster R-CNN, RetinaNet and transformer-based models provide high accuracy, they are not suitable for real-time clinical applications due to high computational demand. YOLOv11 is balanced in terms of speed, accuracy and resource usage, enabling BC-YOLO to provide an effective solution for blood cell detection.

The rest of this paper is structured as follows: The Materials and Methods section details the dataset, model architecture, and methods used in the study. The Experimental Setup section describes the experimental environment, the hardware and software tools used, and the evaluation criteria of the model. The Discussion section discusses the strengths and weaknesses of the model by comparing the results with other studies in the literature. Finally, the Conclusion section provides an overview of the study and discusses future research directions.

2 Materials and methods

2.1 Dataset

BCCD is a publicly available benchmark dataset for detecting and classifying blood cells [32]– [33]. The BCCD is composed of 640 × 480 microscopic images and contains labeled data including RBCs, WBCs, and Platelets. Sample images from the dataset are shown in Fig. S1 in online supplementary information. A single image can contain more than one cell type. The dataset contains 364 microscopic cell images in total. Table S1 in online supplementary information shows the number of labels per cell.

In order to fairly compare the performance of the proposed model with other works in the literature, the dataset is split according to the train/test split scheme in recent studies [29, 32, 33]. In this way, the model evaluation directly compared with methods in the literature.

2.2 MBConv

MBConv is a fundemental block used in MobileNetV2 and EfficientNet models, enabling the development of lightweight and robust DL models for mobile and embedded systems by improving computational efficiency [37, 38]. The schematic structure is shown in Fig. S2. In MBConv, the input feature map is first expanded with Conv 1 × 1, then each channel is individually processed using DWConv 3 × 3. Important channels are then emphasized via the Squeeze and Excitation (SE) [39] block, which includes a channel attention mechanism. At the output, the feature map is compressed again using Conv 1 × 1, and the learning process is facilitated by a skip connection between the input and output. Unlike traditional convolution layers, MBConv performs, significantly reducing FLOPs and parameters while maintaining performance.

2.3 Efficient channel attention

ECA [40] is an advanced module that makes the channel-based attention mechanism more efficient. It learns channel relationships in the input feature map, assigning appropriate weights to each channel and emphasizing important information. It has a lower computational burden compared to SE blocks; it uses adaptive dimensional 1D convolution instead of fully connected (FC) layers. While SE builds global dependencies between all channels, ECA only learns local relationships, eliminating redundant processing. The schematic structure is given in Fig. S3.

ECA transforms the input tensor (H×W×C) into a vector summarizing each channel using GAP (Global Average Pooling) (Eq. S1). Then, an adaptive kernel-sized 1D convolution is applied to model the relationship between channels (Eq. S2). The resulting values are normalized by sigmoid activation to calculate channel weights (Eq. S3) and the input channels are scaled by multiplying them by these weights (Eq. S4).

2.4 MBConv-ECA

The SE and ECA mechanisms are used as channel-based attention mechanisms in CNN. The SE learns channel weights and rescales the input feature map using two FC layers after GAP. Although SE enhances the attention mechanism, it requires additional parameter, especially due to the FC layers. On the other hand, ECA significantly reduces the number of parameters by removing the FC layers in SE and modeling the channel relationships with only small-sized 1D convolutions. In this way, ECA provides a much lighter and computationally efficient structure compared to SE, while trying to maintain good performance. In this context, replacing the SE block in the MBConv block with ECA will make the block lighter and more efficient. The MBConv block with the SE block replaced with ECA is shown in Fig. S4. The proposed MBConv-ECA block combines the low computational cost and high efficiency of ECA with the robust feature extraction capability of MBConv to provide a faster and more accurate DL block.

2.5 BC-YOLO

One-stage object detection approaches are advancing rapidly and the YOLO series stands out among the models that achieve a successful balance between speed and accuracy [41–43]. The YOLOv11, as the most advanced version of the series, enables more efficient and accurate object detection compared to previous versions with components such as C3K2, SPPF, and C2PSA [35, 44].

The C3K2 blocks used in YOLOv11 were developed to provide more efficient feature extraction than traditional convolution structures. The C3K2 block has a structure containing two C3K (3 × 3 convolution) layers. It reduces the computational cost of the model by reducing the number of parameters during deep feature extraction, while at the same time increasing the detection sensitivity of the model by capturing details over a larger area. Fig. S5 details the internal structure of the C3K2 block. The SPPF module is a component that combines feature maps from multiple scales to improve the overall performance of the model in detecting objects of different sizes. The SPPF module captures information at different spatial scales with MaxPool2D processes running in parallel. Additionally, SPPF is particularly advantageous for the detection of objects with varying dimensions, for example, blood cells. Fig. S5 shows how

the SPPF structure works. Attention mechanisms play a critical role in improving the accuracy of object detection models by identifying important regions. The C2PSA (Parallel Spatial Attention) mechanism used in YOLOv11 provides more effective object detection by combining the spatial attention mechanism with parallel convolution layers. The C2PSA structure helps to better discriminate between small and overlapping objects in particular. Fig. S5 shows the layout of C2PSA in the model.

The proposed BC-YOLO model introduces a lighter and more robust structure with innovations in the backbone structure of YOLOv11. As shown in Fig. 1, the computational load of the model is further reduced by using MBConv-ECA blocks instead of traditional convolution blocks. The MBConv blocks reduce the number of parameters by using a depthwise separable convolution structure, while the ECA mechanism allows the model to highlight important features more effectively. These modifications make the BC-YOLO model more effective in applications such as microscopic image analysis by maintaining accuracy despite having a smaller model size. In the neck and output layers of the BC-YOLO model, the original structure of YOLOv11 is preserved.

2.6 Experimental setup

2.6.1 Experiment environment

All models were trained on a high-performance computer using the up-to-date DL libraries. Hardware and software details are given in Table S2. For fair comparison, the YOLO models were trained with a batch size of 20, 100 epochs, a learning rate of 0.001 (cosine annealing) and a weight decay of 0.0005.

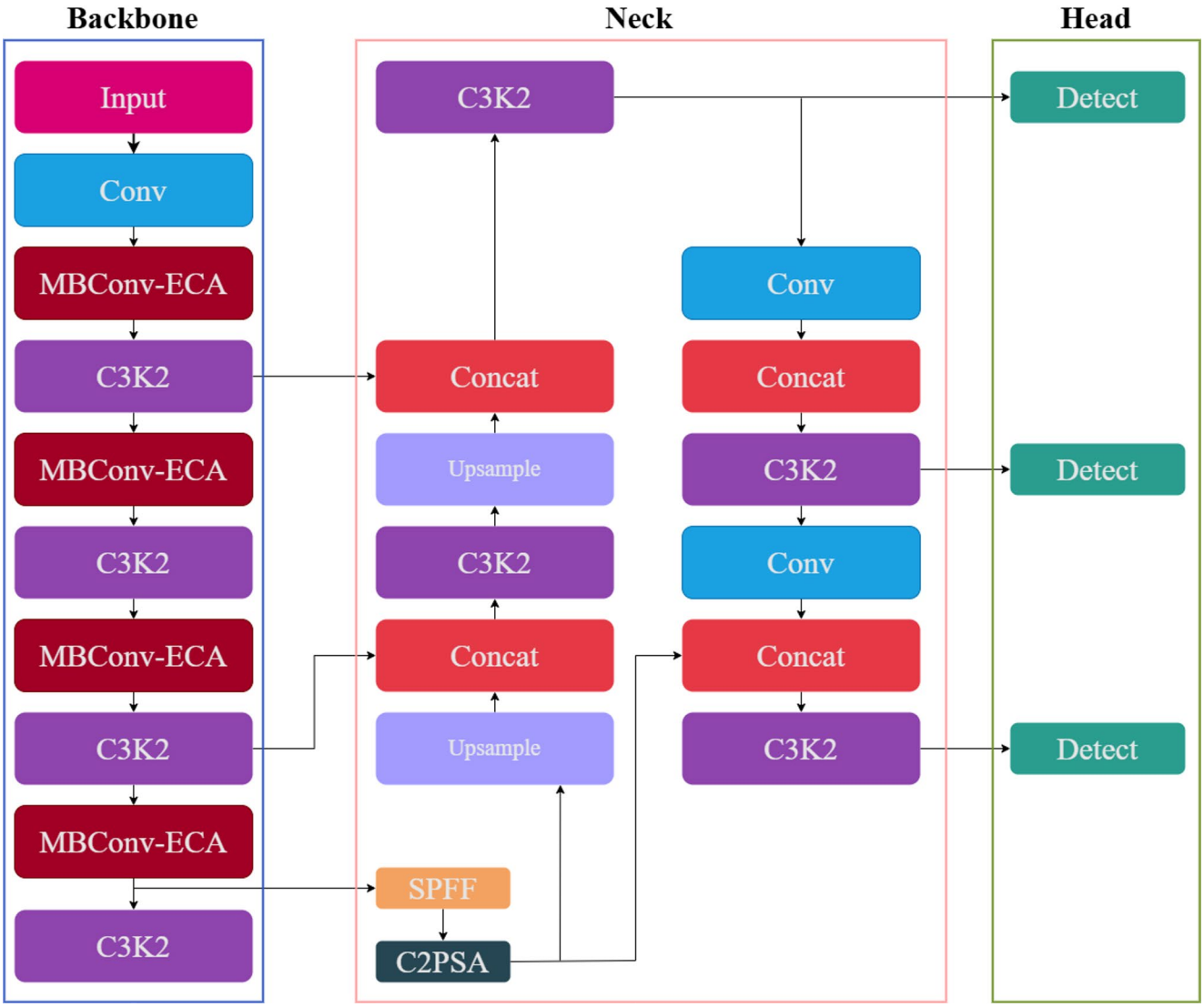

Fig. 1 Overall architecture of the proposed BC-YOLO model

2.6.2 Performance measurement metrics

To evaluate the detection performance of the models tested in this study, various metrics were used. Precision is the ratio of correctly detected objects to total detected objects (Eq. S5), while recall is the ratio of correctly detected objects to total true objects (Eq. S6). In the equations, True Positive (TP) refers to correct detections, False Positive (FP) to false detections and False Negative(FN) to missed objects. The Average Precision(AP) metric is used to measure the overall model success (Eq. S7). The mean Average Precision (mAP) is calculated by calculating the average precision of all classes (Eq. S8), where N is the number of classes. The Giga Floating Point Operations Per Second (GFLOP) metric is used to evaluate the computational load (Eq. S9). Where H and W are the height and width of the input image, K_h and K_w is the kernel size, C_{in} is the number of input channels and C_{out} is the number of output channels. These metrics enable a comprehensive analysis of the accuracy and speed performance of the model.

3 Results

3.1 YOLO variants and baseline detectors results

In this section, the performance of different versions of YOLO (YOLOv8-v11) and commonly used object detectors such as EfficientDet and DETR are comprehensively evaluated on the BCCD dataset. The performance results of YOLO (YOLOv8-v11) and commonly used object detectors are shown in Table 1.

Among the YOLOv8 versions, the Nano model achieved the best result with 93.1% mAP@0.5. It has only 2.5 M parameters and 7.2 GFLOPs and is very fast with an inference time of 2.1 ms and 476 FPS. Precision and recall values are 86.5% and 91.1% respectively. In contrast, the largest model, YOLOv8 X-Large, provides lower accuracy with 92.7% mAP, while the inference time drops to 71.7 ms and FPS to 14.

In the YOLOv9 family, the highest accuracy was achieved in the Extended model with 93.1% mAP@0.5. However, this model consumed high resources with 68.1 M parameters and 164 GFLOPs. In contrast, the Tiny model achieved 92.3% mAP with only 3 M parameters and 8.2 GFLOPs, and offered high speed with 2.4 ms inference time and 417 FPS. The Medium and Compact versions gave balanced results in the range 92.6–92.9% mAP.

Table 1 Performance analysis of YOLO versions on test data(N: Nano, S:Small, M:Medium, L: Large, X: X-Large, C: Compact, E: Extended)

Model	Performance Measurement Metrics								
	Param/M	FLOPs/G	Size/ Mb	*P*/%	*R*/%	mAP@0.5/%	Inference Time/ Ms	FPS	Memory Usage/ Mb
YOLOv8_N	2.5	7.2	5.1	86.5	91.1	93.1	2.1	476	21.9
YOLOv8_S	9.1	24	18.1	84.3	92.2	92.5	7.0	143	55.3
YOLOv8_M	25.0	64.4	49.3	85.7	88.2	93.0	18.7	53	111.5
YOLOv8_L	53.1	135.3	104.2	87.3	90.1	92.4	39.3	25	215.4
YOLOv8_X	97.2	246.9	190.3	83.6	90.4	92.7	71.7	14	360.2
YOLOv9_T	3.0	8.2	6.0	86.1	89.4	92.3	2.4	417	25.3
YOLOv9_S	11.1	28.7	18	85.1	88.6	91.3	8.3	121	59.5
YOLOv9_M	25.8	79.1	50.8	85.7	91.1	92.9	22.8	44	117.4
YOLOv9_C	43.6	133.5	104	87.1	89.5	92.6	38.4	26	199.7
YOLOv9_E	68.1	164	133.5	85	93.2	93.1	47.2	21	280.1
YOLOv10_N	2.0	7.9	4.6	86.8	90.1	92.2	2.3	435	20.2
YOLOv10_S	7.2	27.4	15.2	85.7	92.3	93.1	7.6	131	49.2
YOLOv10_M	20.1	76.5	40.8	83.9	91	92	22.0	45	98.2
YOLOv10_L	25.3	102.3	51.6	87.5	89.9	92.5	29.4	34	124.6
YOLOv10_X	57.3	189.1	117.3	87.5	90.7	92.8	54.4	18	242.0
YOLOv11_N	2.5	6.3	5.5	82.4	93.5	92.4	1.8	556	21.0
YOLOv11_S	9.2	21.6	19.2	**89.4**	**93.5**	**93.5**	6.3	159	56.1
YOLOv11_M	20.0	67.7	40.5	84.3	91.2	92.6	18.0	55	112
YOLOv11_L	25.2	86.6	51.2	86.1	90.3	93.2	31.5	32	158
YOLOv11_X	56.8	194.4	114.4	84.9	86.8	93.1	58.2	17	275
EfficientDet-D2	8.1	36.0	35	77.0	82.0	82.9	25.0	40	65.0
EfficientDet-D3	12.0	77.0	47	79.0	83.0	85.1	30.0	33	80.0
DETR	26.8	60.0	65	85.0	90.0	93.0	23.0	43	120.0

Table 2 Performance analysis of BC-YOLO versions on test data(N: Nano, S:Small, M:Medium, L: Large)

Model	Performance Measurement Metrics								
	Param/M	FLOPs/G	Size/Mb	*P*/%	*R*/%	mAP@0.5/%	Inference Time / Ms	FPS	Memory Usage / Mb
BC-YOLO_N	2.0	5.4	4.8	92.7	95.2	94.3	1.7	588	18.5
BC-YOLO_S	7.8	18.3	17.2	90.1	94.1	95.1	5.8	172	44.1
BC-YOLO_M	18.5	57.4	38.5	92.2	96.3	95.89	16.5	61	85.2
BC-YOLO_L	23.1	74.9	49.9	88.6	93.8	95.7	21.8	46	108.1
BC-YOLO_X	49.7	165.2	109.3	86.3	91.0	95.2	42.0	24	200.9

In the YOLOv10 series, the best performance was obtained by the Small model with 93.1% mAP@0.5. The model achieved 85.7% precision and 92.3% recall with 7.2 M parameters and 27.4 GFLOP, while running fast with 7.6 ms inference time and 131 FPS. The Nano model also achieved similar accuracy (92.2% mAP) and low resource utilization. Larger models (Large and X-Large) achieved limited increases in accuracy, while the computational load and latency increased significantly.

In the YOLOv11 family, the highest accuracy was achieved in the Small model with 93.5% mAP@0.5. The Small model has 9.2 M parameters and 21.6 GFLOPs with 89.4% precision and 93.5% recall, and is well balanced with 6.3 ms inference time and 159 FPS. The Nano model achieved 92.4% mAP and high speed (1.8 ms, 556 FPS) despite lower parameter and FLOP values. In the large-scale models, accuracy increased slightly, while resource consumption increased significantly.

Although EfficientDet and DETR, which are among the alternative models, produce reasonable results in terms of accuracy, they have some disadvantages compared to YOLO-based models. EfficientDet-D2 and D3 models provide 82.9% and 85.1% mAP@0.5 accuracy respectively, but are limited by low precision, recall and FPS values. The DETR model provided high accuracy of 93.0% mAP@0.5 but was inefficient for real-time applications with 23 ms extraction time, 43 FPS and high memory usage. These findings show that more compact and balanced models such as YOLOv11 Small offer significant advantages, especially in terms of speed and resource efficiency.

When the YOLO series and alternative models are evaluated in general, the most balanced and successful performance in terms of accuracy, speed and resource utilization is exhibited by the YOLOv11 Small model. Achieving high accuracy values such as 93.5% mAP@0.5, 89.4% precision and 93.5% recall with only 9.2 M parameters and 21.6 GFLOP computational cost; the model excels in real-time applications with its low latency (6.3 ms) and high FPS (159) values. Although alternative models such as DETR and EfficientDet provide competitive results in terms of accuracy, they lag behind YOLOv11 Small in terms of inference time and resource efficiency. In this context, YOLOv11 Small has the best overall performance by combining high accuracy with hardware efficiency.

3.2 BC-YOLO results

Table 2 presents a detailed performance analysis of the different versions of the BC-YOLO model. The findings reveal that all BC-YOLO versions provide higher accuracy and efficiency using fewer parameters and FLOPs compared to YOLOv11 models. In particular, the BC-YOLO Medium model achieved 92.2% precision, 96.3% recall and 95.89% mAP@0.5 with 18.5 M parameters, 57.4 GFLOPs, and 38.5 MB model size. It is also suitable for real-time applications with an inference time of 16.5 ms and 61 FPS. Nano, one of the lightweight models, has a performance of 94.3% mAP@0.5, 92.7% precision, 95.2% recall, 1.7 ms inference time and 588 FPS, working with only 2.0 M parameters and 5.4 GFLOPs, making it an ideal solution for embedded systems. The Small version combines high accuracy with low resource consumption with 7.8 M parameters, 18.3 GFLOPs and 95.1% mAP. Although the larger models, Large and X-Large, have higher resource requirements of 23.1 M/49.7 M parameters and 74.9/165.2 GFLOPs respectively, they only achieved limited increases in mAP values of 95.7% and 95.2%. This shows that when the model parameters exceed a certain threshold, the increase in accuracy stagnates, while costs such as inference time (21.8–42.0 ms) and memory usage increase.

Overall, the BC-YOLO model combines the powerful structure of the YOLOv11 with the more efficient MBConv-ECA blocks to provide a powerful alternative in areas requiring high sensitivity, such as blood cell detection. The Medium model is the most successful version in terms of accuracy-cost balance, while the Nano and Small versions stand out as lightweight but effective solutions optimized for low-power devices.

In the larger model versions, Large and X-Large offered more balanced performance rather than higher accuracy, despite the higher number of parameters and computational costs. This suggests that when the model goes beyond a certain parameter boundary, the increase in accuracy is limited relative to the computational cost. Finally, the BC-YOLO model retained the strengths of YOLOv11, made

more efficient with the MBConv-ECA blocks in the backbone part, and showed superior performance, especially in areas that require precise and fast analysis, such as blood cell detection. The Medium model offers the most balanced performance in terms of accuracy and computational cost, while the Nano and Small models stand out as ideal solutions for embedded systems and low-power devices. Fig. S6 shows the changes in the accuracy, recall, and mAP values of the most successful model, BC-YOLO Medium, along with the decrease in both training and validation losses throughout the training process. As can be seen in the graph, while the model's losses decrease steadily, the accuracy and recall rates increase rapidly and stabilize after a certain epoch. It indicates that the model performs balanced learning during the training process and minimizes the overfitting problem. Fig. S7 shows the detection performance of the BC-YOLO Medium model for RBCs, WBCs, and Platelets in a batch. The model effectively detects and classifies multiple cells across various microscopic images, demonstrating its ability to handle densely packed cells with high precision and recall.

3.3 Attention results

In this section, the effects of three different attention mechanisms (ECA, SE, CBAM) integrated into the MBConv structure used in the backbone of the BC-YOLO Medium model are analyzed and comparative results are given in Table S3. Considering all performance metrics, the analysis shows that the ECA mechanism provides the most successful results in terms of both accuracy and efficiency.

The architecture using ECA has only 18.5 M parameters, 57.4 GFLOPs, and 38.5 MB model size. Despite this, it achieved the highest accuracy values of 92.2% precision, 96.3% recall, and 95.89% mAP@0.5. It also performed fast and resource-efficient with 16.5 ms inference time, 61 FPS, and 85.2 MB memory usage.

The SE mechanism models channel relationships through fully connected (FC) layers. With this structure, it has reached 20.2 M parameters, 65.5 GFLOPs, and 40.2 MB model size. The accuracy of SE was lower than ECA with 91.2% precision, 94.2% recall, and 94.9% mAP@0.5. Furthermore, the inference time is 18.2 ms, FPS is 55, and memory usage is 91.0 MB. The increased computational load did not contribute enough to the performance.

CBAM combines both channel and spatial attention mechanisms, resulting in the highest computational cost:

22.1 M parameters, 72.8 GFLOPs, and 44.1 MB model size. Accuracies are the lowest with 89.5% precision, 91.9% recall, and 92.12% mAP@0.5. It is also slow and resource consuming with 20.1 ms extraction time, 50 FPS, and 99.0 MB memory usage. Especially the low recall value shows that CBAM is inadequate for sensitive tasks such as the detection of small and dense objects.

In conclusion, taking all metrics into account, the BC-YOLO Medium model equipped with the ECA module offered the highest accuracy at the lowest computational cost and proved to be the most suitable choice for sensitive areas such as medical imaging.

3.4 Comparison with literature

In this section, the performance of the proposed BC-YOLO model is compared with previous studies using the BCCD dataset. Table S4 lists the related studies. When previous studies are analyzed, the CST-YOLO model developed by Kang et al. [29] obtained 92.7% mAP@0.5 despite having 47.5 million parameters and 235.6 GFLOPs computational cost. In the study by Liu [32], the mAP@0.5 value of the model was calculated as 93.8%. Sazak and Kotan [33] achieved a mAP@0.5 value of 93.8% using the YOLOv11 model.

The proposed BC-YOLO model achieved the highest accuracy with both fewer parameters and lower FLOPs compared to previous studies. With only 18.5 million parameters and 57.4 GFLOPs, the BC-YOLO Medium model achieved the best results with 95.89% mAP@0.5, 92.2% precision, and 96.3% recall. As a result, BC-YOLO not only outperformed different YOLO versions but also outperformed other studies in the literature and became the most successful model.

3.5 User interface

A user interface was designed to enable end-users to easily use the BC-YOLO model proposed in this study. The interface enables users to upload microscopic blood cell images, perform automatic analysis, and visualize and examine the detected cells (RBC, WBC, Platelet). The interface supports medical diagnostic processes by presenting the outputs of the model in a user-friendly way. The interface visualization shown in Fig. S8. shows an image upload area, a dashboard displaying the analysis results, and detailed cell classification information. The interface displays high-accuracy detection results with colored bounding boxes and labels.

4 Discussion

In this study, firstly, different configurations of YOLOv8, YOLOv9, YOLOv10 and YOLOv11 series (Nano, Small, Medium, Large, X-Large) as well as commonly used baseline detectors such as EfficientDet-D2/D3 and DETR were extensively trained and evaluated on the BCCD dataset.

According to the results in Table 1, the YOLOv11 Small model is the most efficient classical model in terms of both accuracy and speed with 93.5% mAP@0.5, 89.4% precision, 93.5% recall, 6.3 ms extraction time and 159 FPS. Among the baseline detectors, EfficientDet-D2 and D3 achieved lower accuracy with mAP values of 82.9% and 85.1%, respectively, while the DETR model achieved a remarkable accuracy of 93.0% mAP but suffered from a 23 ms inference time, 43 FPS and high memory usage in real-time applications. This indicates that the YOLOv11 Small model is the most successful structure among the classical models in terms of both accuracy and resource efficiency. In the second stage, based on the robust structure of YOLOv11, the BC-YOLO model is proposed in which lightweight and efficient MBConv-ECA blocks are integrated in the backbone layer instead of classical convolutions. As shown in Table 2, the BC-YOLO Medium model achieved highly successful results such as 95.89% mAP@0.5, 92.2% precision, 96.3% recall, 16.5 ms inference time and 61 FPS with only 18.5 M parameters and 57.4 GFLOPs. This model was the most successful build, beating both previous YOLO versions and models such as EfficientDet and DETR with lower resource consumption. The lightweight BC-YOLO Nano (94.3% mAP, 2 M parameters, 1.7 ms inference, 588 FPS) and Small (95.1% mAP, 7.8 M parameters) models provide fast and accurate solutions for embedded systems.

In the third stage, the effect of the attention mechanisms used in the model was analyzed. ECA, SE and CBAM blocks were compared on the same MBConv architecture and the results are given in Table S3. The structure equipped with the ECA module provided the best result with 95.89% mAP, 92.2% precision, 96.3% recall, 16.5 ms inference time and only 85.2 MB memory usage. The SE and CBAM mechanisms produced lower accuracy despite higher parameter and FLOPs values. In particular, CBAM had the highest computational cost with 22.1 M parameters and 72.8 GFLOPs, but the lowest performance with 92.12% mAP and 91.9% recall. These findings show that ECA is ideal for sensitive areas such as medical imaging due to its light weight and focusing success. Fig. S9 shows the regions of focus of the model using the Grad-CAM method to assess the explainability of the model. In the images, it was observed that in WBCs, the model focused especially on regions containing nuclei and granules, whereas in RBCs, the model focused on cell margins. For small structures such as platelets, accurate classification was achieved by focusing on clear and limited areas. The absence of activation in background regions suggests that the model was able to focus its attention only on meaningful cellular structures and filter out distractions. This reveals that the model performs meaningful learning not only numerically but also biologically.

5 Conclusion

In this work, we propose BC-YOLO, a lightweight and highly accurate object detection model for the detection of microscopic blood cells. The model is built on the powerful structure of YOLOv11, but is made more efficient by integrating MBConv-ECA blocks instead of traditional convolutions in the backbone stage. In this way, both the number of parameters and the computational load are reduced, while the accuracy is increased. Experiments have shown that the BC-YOLO Medium model achieves impressive results of 95.89% mAP@0.5, 92.2% precision, 96.3% recall, 16.5 ms inference time and 61 FPS with only 18.5 M parameters, 57.4 GFLOPs and 38.5 MB model size. In addition, the Nano (2.0 M parameters, 5.4 GFLOPs) and Small (7.8 M parameters, 18.3 GFLOPs) models also provided high accuracy despite low resource consumption, making them particularly suitable solutions for embedded systems and portable devices. In a comparative analysis of the attentional mechanisms used in the model, the ECA attentional module provided higher accuracy with lower parameter and FLOP values compared to both SE and CBAM. Explainability analyses with Grad-CAM also showed that the model focused on cellular structures in a biologically meaningful way and successfully filtered out distractions. However, this study has some limitations. The model was only tested on the BCCD dataset and extended cases such as different image sources or multi-class cell types were not evaluated. Furthermore, the study focuses only on the task of object detection; detailed analysis such as segmentation or subclassification is excluded. In future studies, it is aimed to test the model with different and larger datasets, to integrate segmentation with multi-tasking structures and to develop prototypes that can run in real-time on low-power mobile/edge devices. In this respect, the BC-YOLO model can be considered as a scalable, efficient and explainable solution for many biomedical imaging problems, especially blood cell detection.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11760-025-04321-2.

Author contributions Mustafa Yurdakul conducted data analysis, model development, and experimental analysis. Şakir Taşdemir provided contributions to the methodological design of the study and validation of the results. Mustafa Yurdakul wrote and edited the manuscript. Şakir Taşdemir reviewed and revised the manuscript. All authors read the manuscript and approved the final version.

Data availability No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate This article does not contain any studies with animal or human subjects.

Consent for publication All authors agree to the publication.

Competing interests The authors declare no competing interests.

Conflict of interest The authors declare no conflict of interest.

References

1. Rafii, S., Butler, J.M., Ding, B.-S.: Angiocrine functions of organ-specific endothelial cells. Nature. **529**(7586), 316–325 (2016)
2. Villa, C.H., Anselmo, A.C., Mitragotri, S., Muzykantov, V.: Red blood cells: Supercarriers for drugs, biologicals, and nanoparticles and inspiration for advanced delivery systems. Adv. Drug Deliv. Rev. **106**, 88–103 (2016)
3. Hsu, G., Spilburg, C., Bull, C., Hoffman, B.: Coboglobins: heterotropic linkage and the existence of a quaternary structure change upon oxygenation of cobaltohemoglobin, *Proceedings of the National Academy of Sciences*, vol. 69, no. 8, pp. 2122–2124, (1972)
4. Kutchai, H., Staub, N.C.: Steady-state, hemoglobin-facilitated O2 transport in human erythrocytes. J. Gen. Physiol. **53**(5), 576–589 (1969)
5. Minors, D.S.: Haemostasis, blood platelets and coagulation. Anaesth. Intensive Care Med. **8**(5), 214–216 (2007)
6. Abramson, N., Melton, B.: Leukocytosis: Basics of clinical assessment. Am. Family Phys. **62**(9), 2053–2060 (2000)
7. Dondorp, A.M., et al.: Red blood cell deformability as a predictor of anemia in severe falciparum malaria. Am. J. Trop. Med. Hyg. **60**(5), 733–737 (1999)
8. Krishnegowda, M., Rajashekaraiah, V.: Platelet disorders: An overview. Blood Coagul. Fibrinolysis. **26**(5), 479–491 (2015)
9. Koupenova, M., Kehrel, B.E., Corkrey, H.A., Freedman, J.E.: Thrombosis And platelets: An update. Eur. Heart J. **38**(11), 785–791 (2017)
10. Uyar, K., Yurdakul, M., Taşdemir, Ş.: Abc-based weighted voting deep ensemble learning model for multiple eye disease detection. Biomed. Signal Process. Control. **96**, 106617 (2024)
11. Ozdemir, B., Pacal, I.: An innovative deep learning framework for skin cancer detection employing ConvNeXtV2 and focal self-attention mechanisms. Results Eng. **25**, 103692 (2025)
12. Ávila-Jiménez, J.L., Cantón-Habas, V., Carrera-González, M.P., Rich-Ruiz, M., Ventura, S.: A deep learning model for alzheimer's disease diagnosis based on patient clinical records. Comput. Biol. Med. **169**, 107814 (2024)
13. Wang, M., et al.: A fast interpretable adaptive meta-learning enhanced deep learning framework for diagnosis of diabetic retinopathy. Expert Syst. Appl. **244**, 123074 (2024)
14. Alnowaiser, K., Saber, A., Hassan, E., Awad, W.A.: An optimized model based on adaptive convolutional neural network and grey Wolf algorithm for breast cancer diagnosis. PloS One. **19**(8), e0304868 (2024)
15. Saber, A., Elbedwehy, S., Awad, W.A., Hassan, E.: An optimized ensemble model based on meta-heuristic algorithms for effective detection and classification of breast tumors. Neural Comput. Appl. **37**(6), 4881–4894 (2025)
16. Hassan, E., Saber, A., Elbedwehy, S.: Knowledge distillation model for acute lymphoblastic leukemia detection: Exploring the impact of nesterov-accelerated adaptive moment Estimation optimizer. Biomed. Signal Process. Control. **94**, 106246 (2024)
17. Pacal, I., Ozdemir, B., Zeynalov, J., Gasimov, H., Pacal, N.: A novel CNN-ViT-based deep learning model for early skin cancer diagnosis. Biomed. Signal Process. Control. **104**, 107627 (2025)
18. Pacal, I.: Investigating deep learning approaches for cervical cancer diagnosis: A focus on modern image-based models. Eur. J. Gynaecol. Oncol., **46**, 1, (2025)
19. Ullah, N., De Falco, I., Sannino, G.: A novel deep learning approach for colon and lung cancer classification using histopathological images, in *IEEE 19th International Conference on e-Science (e-Science)*, 2023: IEEE, pp. 1–10. (2023)
20. Ullah, N., Khan, J.A., De Falco, I., Sannino, G.: Bridging clinical gaps: multi-dataset integration for reliable multi-class lung disease classification with DeepCRINet and occlusion sensitivity, in *IEEE Symposium on Computers and Communications (ISCC)*, 2024: IEEE, pp. 1–6. (2024)
21. Ullah, N., et al.: Chestcovidnet: An effective dl-based approach for covid-19, lung opacity, and pneumonia detection using chest radiographs images. Biochemistry Cell. Biology No Ja, (2024)
22. Ullah, N., Marzougui, M., Ahmad, I., Chelloug, S.A.: Deeplungnet: An effective dl-based approach for lung disease classification using cris, *Electronics*, vol. 12, no. 8, p. 1860, (2023)
23. Ullah, N., Hassan, M., Khan, J.A., Anwar, M.S., Aurangzeb, K.: Enhancing explainability in brain tumor detection: A novel DeepEBTDNet model with LIME on MRI images. Int. J. Imaging Syst. Technol. **34**(1), e23012 (2024)
24. Ullah, N., Khan, J.A., De Falco, I., Sannino, G.: Explainable artificial intelligence: Importance, use domains, stages, output shapes, and challenges. ACM Comput. Surveys. **57**(4), 1–36 (2024)
25. Zou, Y., Wu, L., Zuo, C., Chen, L., Zhou, B., Zhang, H.: White blood cell classification network using MobileNetv2 with multiscale feature extraction module and attention mechanism. Biomed. Signal Process. Control. **99**, 106820 (2025)
26. Saidani, O., et al.: White blood cells classification using multi-fold pre-processing and optimized CNN model. Sci. Rep. **14**(1), 3570 (2024)
27. Tarimo, S.A., Jang, M.-A., Ngasa, E.E., Shin, H.B., Shin, H., Woo, J.: WBC YOLO-ViT: 2 Way-2 stage white blood cell detection and classification with a combination of YOLOv5 and vision transformer. Comput. Biol. Med. **169**, 107875 (2024)
28. Parab, M.A., Mehendale, N.D.: Red blood cell classification using image processing and CNN. SN Comput. Sci. **2**(2), 70 (2021)
29. Kang, M., Ting, C.-M., Ting, F.F., Phan, R.C.-W.: CST-YOLO: A novel method for blood cell detection based on improved YOLOv7 and CNN-swin transformer, in *2024 IEEE International Conference on Image Processing (ICIP)*,: IEEE, pp. 3024–3029. (2024)
30. Fang, T., Huang, X., Chen, X., Chen, D., Wang, J., Chen, J.: Segmentation, feature extraction and classification of leukocytes leveraging neural networks, a comparative study. Cytometry Part. A, (2024)
31. Ren, Z., Kong, X., Zhang, Y., Wang, S.: UKSSL: Underlying knowledge based semi-supervised learning for medical image classification. IEEE Open. J. Eng. Med. Biology, (2023)
32. Liu, B.: Blood cell count and detection method based on YOLO. Highlights Sci. Eng. Technol. **27**, 594–599 (2022)
33. Sazak, H., Kotan, M.: Automated Blood Cell Detection and Classification in Microscopic Images Using YOLOv11 and Optimized Weights, *Diagnostics*, vol. 15, no. 1, p. 22, (2024)
34. Ultralytics. [Online]: Available: https://docs.ultralytics.com/models/yolo11/s
35. Khanam, R., Hussain, M.: Yolov11: An overview of the key architectural enhancements, *arXiv preprint arXiv:2410.17725*, (2024)

36. Zhong, J., et al.: PMFSNet: Polarized multi-scale feature self-attention network for lightweight medical image segmentation, *Computer Methods and Programs in Biomedicine*, p. 108611, (2025)
37. Sun, J., et al.: ESC-YOLO: Optimizing Apple fruit recognition with efficient Spatial and channel features in YOLOX. J. Real-Time Image Proc. **21**(5), 162 (2024)
38. Amjad, K., Asif, S., Waheed, Z., Guo, Y.: A novel lightweight deep learning framework with knowledge distillation for efficient diabetic foot ulcer detection. Appl. Soft Comput. **167**, 112296 (2024)
39. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141. (2018)
40. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: ECA-Net: Efficient channel attention for deep convolutional neural networks, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11534–11542. (2020)
41. Ru, H., Zhang, W., Wang, G., Ding, L.: OMAL-YOLOv8: real-time detection algorithm for insulator defects based on optimized feature fusion. J. Real-Time Image Proc. **22**(1), 52 (2025)
42. Zhang, K., et al.: SF-YOLO: Designed based on tiny feature for PCB surface defect detection and deployment in embedded systems. Signal. Image Video Process. **19**(3), 263 (2025)
43. Du, Y., Liu, W., Wang, Y., Li, R., Xie, L.: YOLO-CPC: a breast tumor detection and identification algorithm based on improved YOLOv7, *Signal, Image and Video Processing*, vol. 19, no. 3, p. 260, (2025)
44. Jocher, G., Qiu, J.: Ultralytics yolo11, *GitHub: San Francisco, CA, USA*, (2024)