Research Article

High-Efficiency Classification of White Blood Cells Based on Object Detection

Jiangfan Yao, Xiwei Huang ⓘ, Maoyu Wei, Wentao Han, Xuefeng Xu, Renjie Wang, Jin Chen, and Lingling Sun

Key Laboratory of RF Circuits and Systems, Ministry of Education, Hangzhou Dianzi University, Hangzhou 310018, China

Correspondence should be addressed to Xiwei Huang; huangxiwei@hdu.edu.cn

White blood cells (WBCs) play a significant role in the human immune system, and the content of various subtypes of WBCs is usually maintained within a certain range in the human body, while deviant levels are important warning signs for diseases. Hence, the detection and classification of WBCs is an essential diagnostic technique. However, traditional WBC classification technologies based on image processing usually need to segment the collected target cell images from the background. This preprocessing operation not only increases the workload but also heavily affects the classification quality and efficiency. Therefore, we proposed one high-efficiency object detection technology that combines the segmentation and recognition of targets into one step to realize the detection and classification of WBCs in an image at the same time. Two state-of-the-art object detection models, Faster RCNN and Yolov4, were employed and comparatively studied to classify neutrophils, eosinophils, monocytes, and lymphocytes on a balanced and enhanced Blood Cell Count Dataset (BCCD). Our experimental results showed that the Faster RCNN and Yolov4 based deep transfer learning models achieved classification accuracy rates of 96.25% and 95.75%, respectively. For the one-stage model, Yolov4, while ensuring more than 95% accuracy, its detection speed could reach 60 FPS, which showed better performance compared with the two-stage model, Faster RCNN. The high-efficiency object detection network that does not require cell presegmentation can remove the difficulty of image preprocessing and greatly improve the efficiency of the entire classification task, which provides a potential solution for future real-time point-of-care diagnostic systems.

1. Introduction

White blood cells (WBCs) are important parts of human blood and indispensable security guards in the immune system, which mainly include granulocytes, monocytes, and lymphocytes [1]. Granulocytes are differentiated from hematopoietic stem cells in the bone marrow. They are a type of leukocytes containing granules in the cytoplasm and can be divided into three subtypes: neutrophils, eosinophils, and basophils, according to their morphological characteristics under Wright's staining (see Table 1). Neutrophils are composed of multisplit nuclei containing between two and five lobes. They are the most common phagocytes, accounting for 50%–60% of the total number of WBCs. The content of eosinophils is between 1% and 6%, and there are generally two split nuclei. Basophils are one of the least

common cells in bone marrow and blood, and their content is less than 2%. Monocytes are produced in the bone marrow and are a kind of WBCs with nongranular cytoplasm, accounting for 2%–10% of whole WBCs. Lymphocytes are the smallest type of WBCs, which can be divided into T cells and B cells. They have almost no cytoplasm, accounting for about 20%–30%. These WBC populations have characteristic concentration ranges in healthy people. Many diseases are accompanied by their concentration deviations [2], such as inflammation and bacterial infection. Therefore, the classification and statistics of WBCs have important medical diagnostic significance.

Traditional WBC classification is achieved by experienced medical personnel, who directly differentiated the WBCs from blood smear images according to their morphologies under the microscope [3]. Manual classification

TABLE 1: 3D models, normal concentration ranges, and morphological characteristics for various subtypes of WBCs.

Subtype of WBC		3D model [1]	Content (%)	Morphological characteristics
Granulocytes	Neutrophil		50~60	Consist of a multilobed nucleus; the number of lobes can be 2–5; stained in natural pink color.
	Eosinophil			Have a two-lobed nucleus; stained in brick-red in acidic stains
	Basophil			Contain large cytoplasmic granules which obscure the cell nucleus under the microscope when stained.
Monocytes			2~10	Amoeboid in appearance; have nongranulated cytoplasm.
Lymphocytes	B-cell		20~30	Have a large, dark-staining nucleus with little cytoplasm; a coarse and dense nucleus approximately the size of a red blood cell (RBC).
	T-cell			

has many shortcomings and difficulties. For example, human observation cannot guarantee unbiased estimation, which may lead to unsatisfactory accuracy. Moreover, manual classification is time-consuming, complicated, and requires strict professional skills of the inspectors, which cannot meet the requirements of high-efficiency classification tasks on large scales nowadays. Therefore, automatic WBC classification technologies have been extensively developed [4, 5].

Existing automatic classification technologies are mostly based on cell image analysis because they are relatively easy to obtain, and different WBC types show distinctive morphological characteristics. In addition, image processing technologies have matured in recent years with the help of advancing computing power and intelligent algorithms. The two most representative technologies are feature engineering based on machine learning [6, 7] and automatic feature extraction based on deep learning [8, 9].

The feature engineering classification system based on machine learning mainly follows three steps: (I) target segmentation from the background, (II) manual extraction of effective or unique features, and (III) classifier design (see Figure 1(a)). This method uses a shallow machine learning model that relies on input features and attempts to quantify the relevant features extracted from digital images with an analysis method similar to morphologists; then, it uses them as the input of the prediction algorithm. Shallow machine learning methods commonly used in classification generally include support vector machine (SVM), naive Bayes classifier, linear discriminant analysis (LDA), and multilayer perceptron (MLP).

For example, Lippeveld et al. studied the WBC classification performance of classical machine learning with manually selected feature values as input and obtained 0.776 mean average precision (mAP) on unstained WBCs captured by the imaging flow cytometer (IFC) [10]. Nassar et al. used a gradient boosting algorithm to achieve the label-free classification of four types of WBCs (neutrophils, eosinophils, monocytes, and lymphocytes) and two types of lymphocytes (B/T lymphocytes) based on IFC, and the $F1$

scores reached 97% and 78%, respectively [11]. However, the focus of these studies is mostly on image preprocessing (target segmentation) and feature selection, which are the prerequisites for the good performance of the classification system.

Deep learning is a technology widely used in research fields such as computer vision, speech analysis, and natural language processing (NLP) [12]. As a data-based representation learning algorithm, deep learning uses supervised or semisupervised feature learning and hierarchical extraction algorithms to replace manual feature acquisition. The automatic feature extraction system based on deep learning can be divided into two steps: (I) image preprocessing (cell segmentation, data enhancement) and (II) neural network design (see Figure 1(b)).

For example, in our previous work, we used a fine-tuned ResNet50 network to realize the label-free classification of WBCs and neutrophils of different activation states, reaching an accuracy of more than 90% [13]. Chen et al. proposed a high-throughput quantitative imaging system using photon time stretching and achieved high-precision classification of label-free cells (T lymphocyte and colon cancer cells) using deep learning algorithms [14]. Shu et al. used quantitative phase microscopy to image unstained leukocytes and used the CNN model to extract cell morphological features to achieve the classification of four types of WBCs [15] with 90% accuracy obtained in both training set and test set.

The above two classification systems are designed from the perspective of image classification. They must first segment the objects from the backgrounds to ensure that there are as few irrelevant backgrounds as possible in the image input to the network. The idea of object segmentation is to obtain the pixel block containing the target from the original image by setting a threshold, and the intercepted area only contains at most one target. When the image background is not much different from the cells, or the image has excessive noise, the difficulty of cell segmentation will be greatly increased. Moreover, the performance of the classification system depends heavily on the preprocessing of
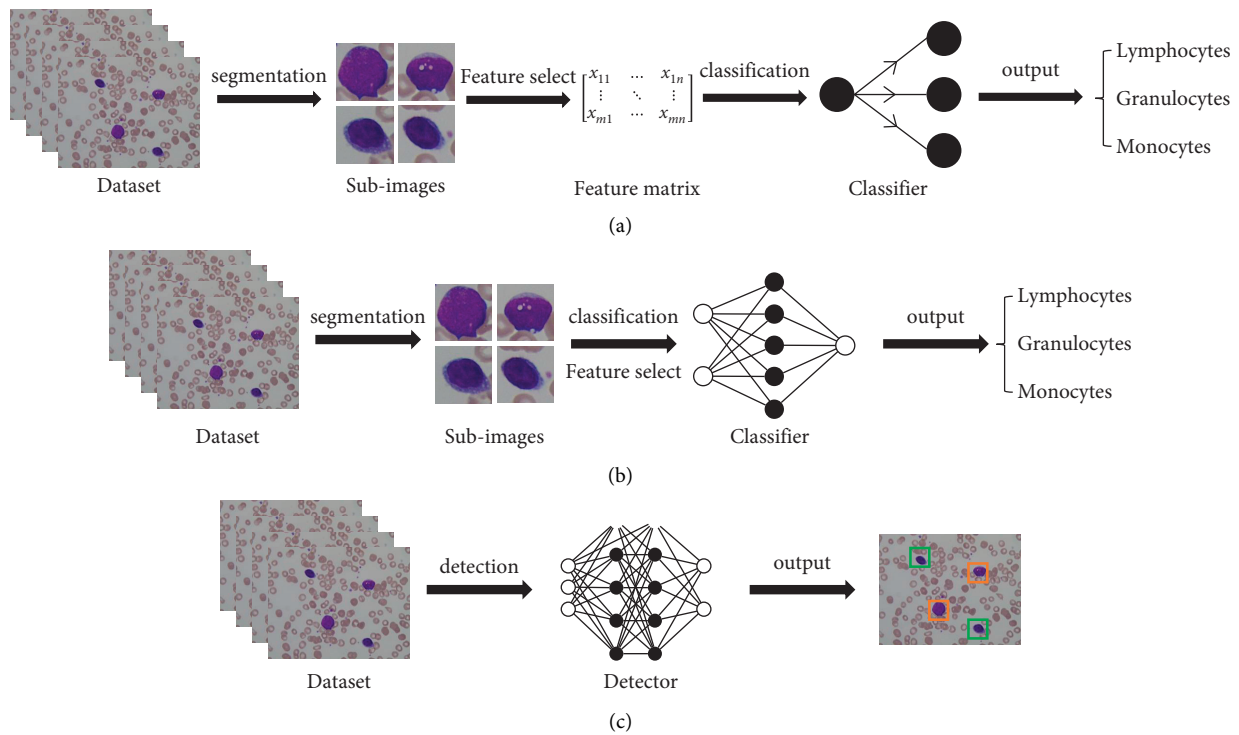
FIGURE 1: Three WBC classification systems: (a) feature engineering classification system based on machine learning; (b) automatic feature extraction classification system based on deep learning; and (c) high-efficiency object detection classification system based on deep learning.

cell segmentation, which makes it difficult to achieve end-to-end optimization of the entire system. By contrast, object detection is an algorithm based on target geometric and statistical features, which combines the tasks of target segmentation and recognition into one process. While finding the precise location of a specific object in a given image, a corresponding class label is also assigned to each object instance (see Figure 1(c)). This algorithm combining classification and localization does not need to perform object segmentation in advance; hence, it effectively solves the problems of the above two systems.

Current object detection algorithms can be divided into two-stage models and one-stage models. The two-stage models first use a latent region generator to generate a sparse proposal region and extract features from each proposal and then cascade a region classifier to predict the candidate region labels. The most representative two-stage detectors are regional convolution neural networks (RCNN) [16], Fast RCNN [17], Faster RCNN [18], and Mask RCNN [19]. By contrast, the one-stage models directly classify and predict the objects at each location of the feature map, so that cascaded region classification is not required. The representative models of one-stage detectors include YOLO [20] and SSD [21]. In theory, a two-stage model can achieve a high accuracy rate but has more parameters and a slower detection speed. A one-stage model can achieve a speed increase but usually at the expense of accuracy.

For example, Kutlu et al. [22] tested several RCNN models with different backbone networks on randomly mixed Blood Cell Count Dataset (BCCD) and Leukocyte

Images for Segmentation and Classification (LISC) dataset. The best performance is achieved on ResNet50, and the highest accuracy is obtained in the detection of lymphocytes (99.52%). However, the detection speed of this series RCNN is low, only 1.6 FPS. Wang et al. [23] collected high-resolution blood cell images with a Fourier engraving microscope and successfully detected WBCs using the Yolov3 model. The accuracy and recall rate reached 100%, achieving better performance than other algorithms. However, their work is based on high-resolution images collected by a unique imaging system, and they can only identify WBCs from the background but cannot complete the classification task. Wang et al. [24] used the Yolov3 fine-tuning model to achieve the detection of multiple types of WBCs at a speed of 53 FPS, but this model could be further optimized. Jiang et al. [25] proposed an Attention-YOLO for the detection and counting of WBC, red blood cells, and platelets, but its dataset is imbalanced and the processing efficiency in terms of frame rate is not analyzed and compared.

To achieve leukocyte classification with optimized speed and accuracy, we employed the most state-of-the-art one-stage model Yolov4 to achieve detection of neutrophils, eosinophils, monocytes, and lymphocytes on an enhanced and balanced BCCD for training. We adopted the transfer learning mechanism and adjusted the training parameters such as batch size and learning rate so that the models could achieve classification accuracy rates of 95.75% and detection speed of 60 FPS. As a comparison with Yolov4, we also employed the Faster RCNN on the same dataset. By applying different backbone networks (ResNet101, VGG16, Inception

v2), we found that VGG16 converged faster (similar to Yolov4, about 100 epoch), and finally obtained 96.25% accuracy and 15 FPS detection speed. It can be seen that Yolov4 is more efficient, with comparable accuracy in terms of classifying such WBC images. Therefore, the proposed high-efficiency one-stage detection network provides a potential solution for future real-time point-of-care (POC) diagnostic systems [26, 27].

2. Materials and Methods

2.1. Data Enhancement and Annotation. The original dataset used in our experiment is BCCD (https://www.kaggle.com/ paultimothymooney/blood-cells), which contains 364 microscope images stained by the Reiter-Giemsa staining method. It is a small-scale dataset for blood cell detection, which includes WBCs, RBCs, and platelets in each image. The nuclear characteristics of the stained WBCs are relatively obvious, and four subtypes of WBCs (neutrophils, eosinophils, monocytes, and lymphocytes) can be identified as our classification task. From the perspective of cell morphology, T and B lymphocytes are indistinguishable [28]. Both of their cells have a large nucleus with dense heterochromatin. Therefore, we do not distinguish these two subtypes of lymphocytes. In addition, the proportion of basophils in the human body is very small (<2%), which has little reference value in pathology, so we also ignore it.

To a large extent, the performance of a deep learning algorithm requires a large enough dataset in the training phase to avoid overfitting. However, in the field of medical image analysis, it is difficult to obtain adequate raw data for CNN training [29]. Data augmentation is an image processing technology that solves the limited data space, which can improve the size and quality of training data so that it can be used to build better deep learning models. Image enhancement algorithms include geometric transformation, color space enhancement, mixed images, and neural style transfer [30]. In this paper, we mainly use rotation and mirror geometric transformation algorithms to expand the BCCD and improve the performance of our model:

(1) Central rotation: in reality, the origin of the image is in the upper left corner, so we need to move the origin of the upper left corner to the center of the image. Suppose a point on the image (X_0, Y_0), the image width is W, the height is H, the image rotation angle is θ (to avoid the loss of WBC information in the image, here we choose a smaller value of θ), the transformed point is (X_1, Y_1), the width and height of the target image are W^* and H^*, respectively; then, we can get the following transformation formula:

$$\begin{bmatrix} X_1 \\ Y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ -0.5W & 0.5H & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ -0.5W^* & 0.5H^* & 1 \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ 1 \end{bmatrix}. \tag{1}$$

(2) Mirror flip: we used two mirroring methods here, horizontal mirroring and vertical mirroring. Horizontal mirroring takes the vertical centerline of the image as the axis, swapping the pixels of the image, while vertical mirroring takes the horizontal centerline of the image as the axis and reverses the upper half and the bottom half of the image. Therefore, the transformation formula for horizontal mirroring is

$$X_1 = W - X_0 - 1,$$
$$Y_1 = Y_0. \tag{2}$$

The vertical mirror transformation formula is

$$Y_1 = H - Y_0 - 1,$$
$$X_1 = X_0. \tag{3}$$

We randomly selected 225 images of four types of WBCs from the enhanced images to construct our dataset, and the image resolution is 320 × 240. Then, according to the 7 : 2 ratio, it is divided into a training set (175 images of each of the four types of WBCs) and a test set (50 images of each of the four types of WBCs). The image after the rotation and mirroring process has black borders, but this will not affect

our classification task, because the object detection only pays attention to the information of the marked target (localization box and class label), which will not be affected by the background. We used *labelImg* (https://github.com-/ tzutalin/labelImg) to generate the corresponding annotation files for the training and testing later. Examples of image samples and annotations are shown in Figure 2.

3. Detectors

3.1. Faster RCNN. Faster RCNN [18] is the most outstanding product of the RCNN series algorithms so far, and it is also the most classic object detection algorithm in the two-stage models. It abandons the traditional Sliding Window and Selective Search algorithms, which directly uses the Region Proposal Network (RPN) to generate the detection box, achieving end-to-end training and greatly speeding up. The first stage of Faster RCNN is to find the anchor boxes of the object to be detected in the images (classification of the background and the object), and the second stage is to classify and regress these anchor boxes. The employed Faster RCNN structure based on the VGG16 [31] backbone is shown in Figure 3, which is composed of three parts: backbone, regional proposal network (RPN), and classifier.
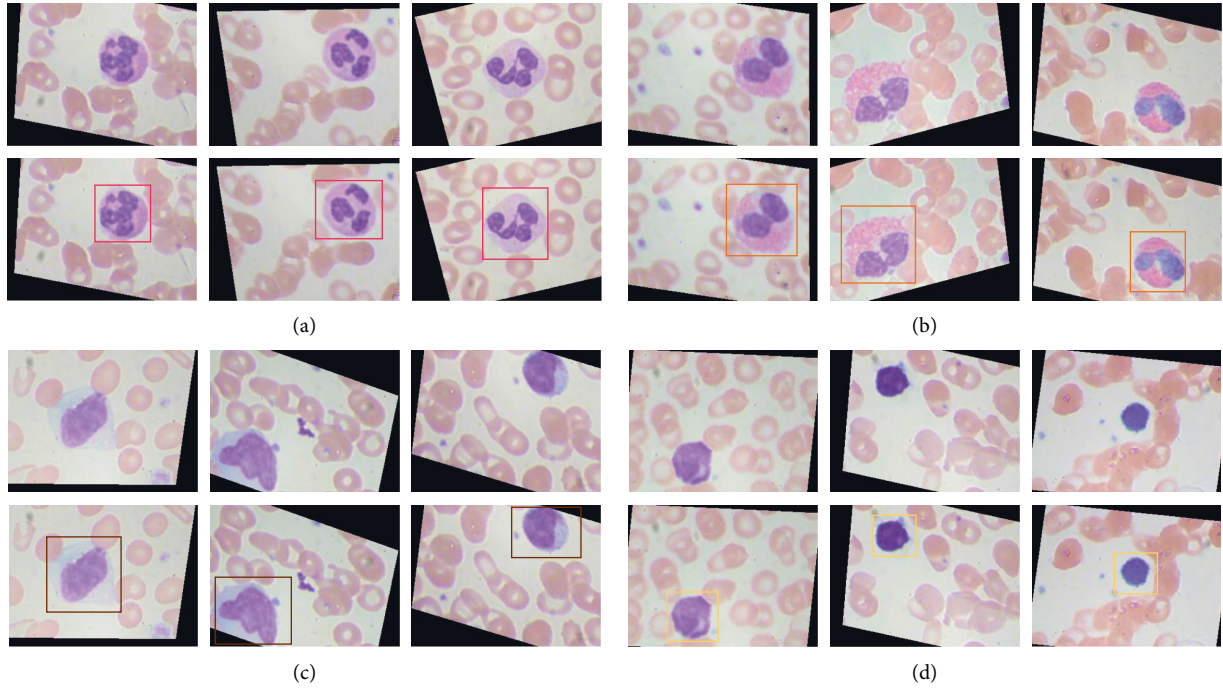
Figure 2: Samples (top) and their corresponding annotations (bottom) of the four WBC subtypes in our dataset. (a) Neutrophil, (b) eosinophil, (c) monocyte, and (d) lymphocyte.
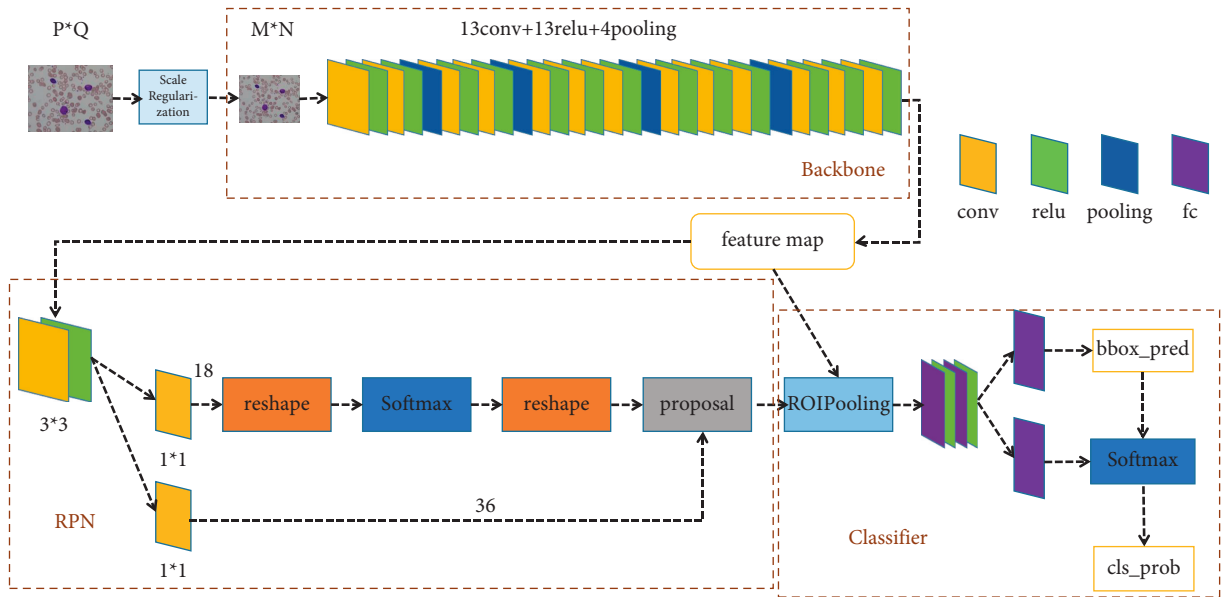

Figure 3: Faster RCNN architecture. The backbone is VGG16 without a fully connected layer used to extract the features of entire images to generate the feature map, RPN is used to generate the detection box of region proposal, and classifier network is used to classify the candidate detection box and output the detection result.

The output of RPN and classification regression network is coordinate regression and classification values, so the Loss function of these two networks can be expressed as follows:

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \frac{\lambda}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*), \quad (4)$$

where i represents the anchor index, p_i represents the probability of positive class that Softmax outputs, and p_i^* represents the corresponding ground truth prediction probability. When the IoU between the ith anchor and ground truth is greater than 0.7, this anchor is considered to be positive ($p_i = 1$). Otherwise, when IoU < 0.3, the anchor is considered to be negative ($p_i = 0$). t_i^* represents the prediction probability of the bounding box. t_i^* represents the prediction probability of the ground true corresponding to the positive anchor. L_{cls} represents the classification loss function (used the anchor's classification in training), which can be used for different loss functions. L_{reg} represents the regression loss function (used the bounding box regression in training), which uses $smooth_{L1}$ here. The calculation is as follows:

$$L_{reg}\left(t_i, t_i^*\right) = \sum_{i\varepsilon(x,y,w,h)} smooth_{L1}\left(t_i - t_i^*\right),$$

$$smooth_{L1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1, \\ |x| - 0.5, & \text{otherwise.} \end{cases} \tag{5}$$

In the actual process, the gap between N_{cls} and N_{reg} is too large, so the parameter λ is introduced to balance them (e.g., when $N_{cls} = 512$, $N_{reg} = 2400$, set $\lambda = N_{reg}/N_{cls} \approx 5$). Therefore, the total Loss can be balanced between the two kinds of Loss.

3.2. Yolov4.
Yolo (you only look once) was proposed by Redmon et al. [20] in 2016. As the pioneering work of the one-stage detector, it combines the region proposal generation and candidate box regression into one network and directly predicts the bounding box and class probabilities from the complete image to achieve end-to-end optimization of the entire detection performance. Redmon continued to optimize the algorithm in the next few years and finally developed it into three versions: Yolo, Yolov2 [32], Yolov3 [33]. Yolov3 is an end-to-end one-stage detector with excellent performance, which is comparable to the most advanced object detection system in terms of accuracy and greatly improves the detection speed. Inspired by Yolov3, Bochkovskiy et al. [34] proposed an improved version of Yolov3 in 2020 and named it Yolov4. A large number of experiments have proved that compared with Yolov3, Yolov4 improves the mAP by 10 points on the MS COCO dataset and increases the speed by 12%. In the previous preexperiment, we used Yolov3 and Yolov4 to train the same dataset and found that the Yolov4 fits faster, but the loss is only 1/10 of Yolov3's loss. The network structure of Yolov4 follows the classic one-stage structure, which is mainly divided into three parts: Backbone, Neck, and Prediction. The overall architecture is shown in Figure 4.

Building on the CutMix [35] algorithm, Yolov4 introduced Mosaic data enhancement at the input. The main idea is to randomly flip, zoom, gamut the color, and crop the four

pictures and then stitch them in a picture as training data (see Figure 5). The cutting process will keep the complete label frame. Four pictures are calculated at the same time during batch normalization so that the minibatch size does not need to be large, reducing computing resources. In our dataset, a training sample contains only two WBCs to be detected at most, so the mosaic algorithm can greatly enrich the background of the images and make the network more robust.

Like Faster RCNN, the loss function of Yolov4 is generally composed of two parts: a classification loss function and a regression loss function. Starting from Faster RCNN applying $smooth_{L1}$ Loss to bounding box regression, the development of the loss function has gone through IoU Loss (2016), GIoU Loss (2019), DIoU Loss (2020), and now CIoU Loss (2020) applied in Yolov4. IoU Loss mainly considers the overlapping area of the detection box and ground truth. GIoU Loss solves the problem when the bounding boxes do not coincide based on IoU. DIoU Loss considers the information of the center point distance of the bounding box based on IoU and GIoU. CIoU Loss adds an impact factor αv to the DIoU's penalty item, which takes into account the aspect ratio of the predicted box to fit the ground truth:

$$R_{CIoU} = \frac{\rho^2\left(b, b^{gt}\right)}{c^2} + \alpha v, \tag{6}$$

where b and b^{gt} represent the center points of the penalty terms of the prediction box B and the ground truth B^{gt}, respectively; $\rho(\cdot)$ represents the Euclidean distance; c represents the diagonal distance of the smallest outer rectangle of B and B^{gt}; and α is the parameter for trade-off, which is defined as

$$\alpha = \frac{v}{(1 - IoU) + v}, \tag{7}$$

where IoU represents the intersection ratio between the prediction box and the ground truth and v is a parameter used to measure the consistency of the aspect ratio, which is defined as

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w^2}{h}\right). \tag{8}$$

Therefore, the CIoU Loss function is defined as

$$L_{CIoU} = 1 - IoU + R_{CIoU}$$

$$= 1 - IoU + \frac{\rho^2\left(b, b^{gt}\right)}{c^2} + \alpha v. \tag{9}$$

In this way, CIoU Loss can consider three important geometric factors for the box regression function, overlap area, center point distance, and aspect ratio, which improves the speed and accuracy of prediction box regression.
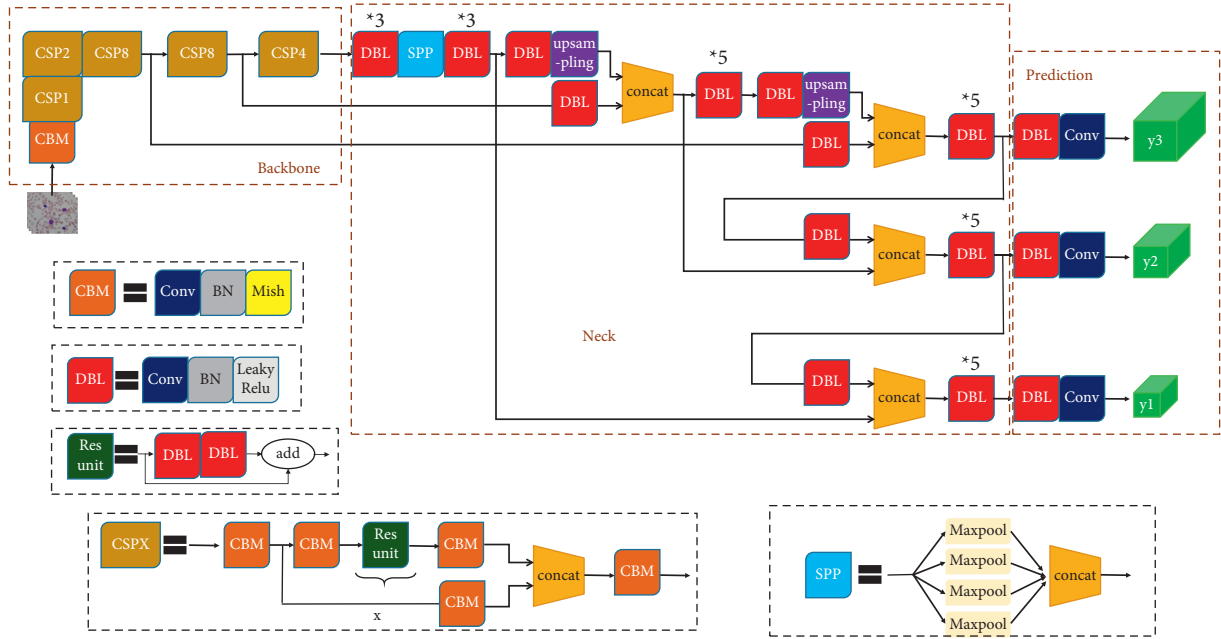
FIGURE 4: Yolov4 architecture. The backbone is CSPDarknet53 used to feature extraction, the Neck layer (usually between backbone and output) is used to collect feature maps in different stages to better extract fusion features, and the prediction module is used for the output detection result.
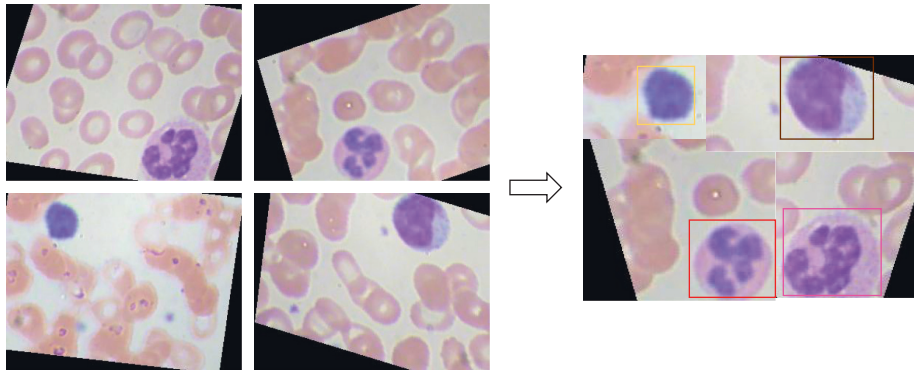

FIGURE 5: Mosaic data enhancement.

4. Results and Discussion

The methods based on Faster RCNN and Yolov4 are implemented using TensorFlow and PyTorch frameworks, respectively, which configure the CUDA8.0 toolkit with cuDNN6.0 library on the 64-bit Ubuntu 16.04 operating system. All experiments are performed with the configuration of CPUi7-6700 (3.40 GHz), RAM16 GB, GPU NVIDIA GTX1080Ti (11 GB), and Python 2.7. In the training phase, we adopted a transfer learning mechanism: initialize the CNN framework with a pretrained model based on the ImageNet (http:/www.image-net.org) and then use our dataset to refine all layers of the network.

We used precision and recall to evaluate the performance of our object detection model, which are defined as follows:

$$
\begin{aligned}
\text{precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}} \\
&= \frac{\text{TP}}{\text{all detections}}, \\
\text{recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \\
&= \frac{\text{TP}}{\text{all groundtruth}}.
\end{aligned}
\tag{10}
$$

where TP (true positive) means correct detection of the positive bounding box; FP (false positive) means false detection of nonexistent objects or misplacement detection of existing objects; and FN (false negative) means missing detection of the positive bounding box. The precision
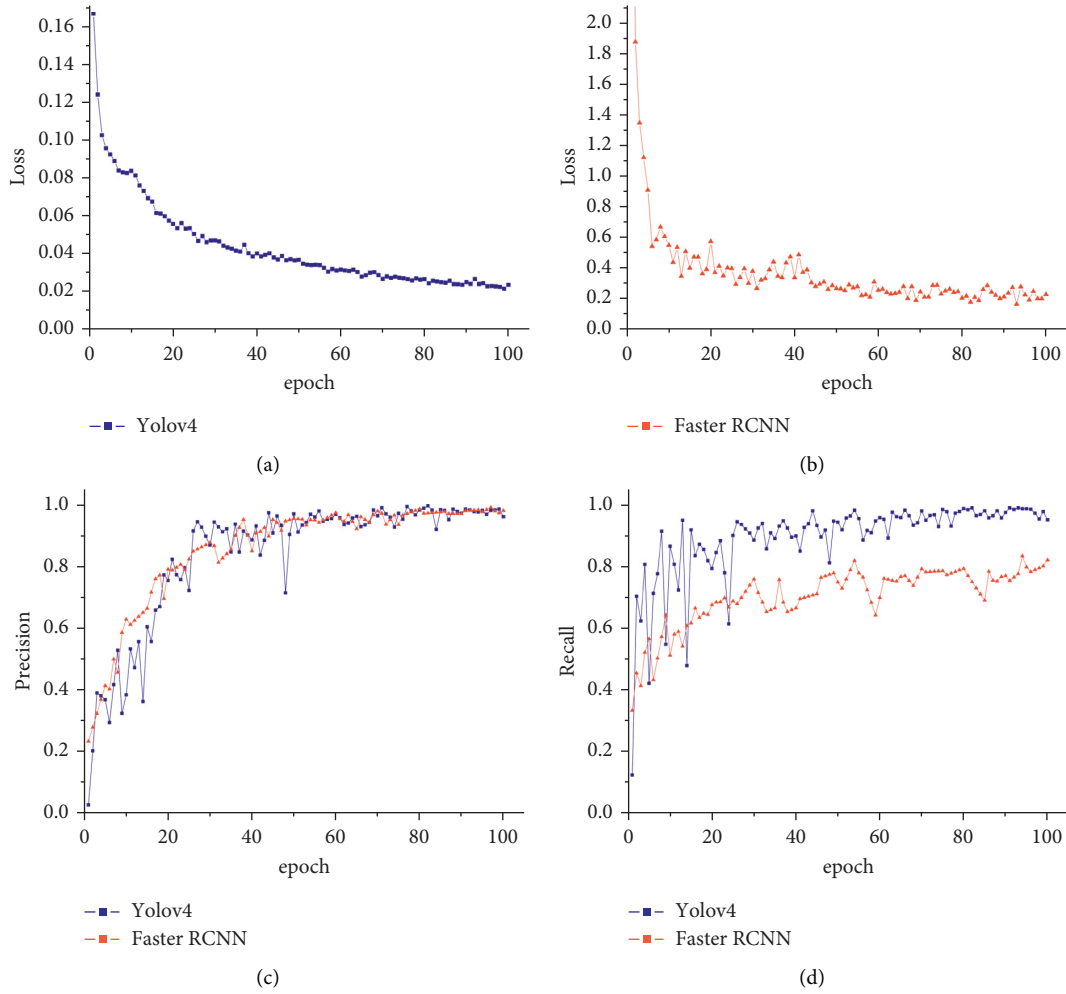
(a)

(b)

(c)

(d)

Figure 6: Training output of Yolov4 and Faster RCNN. After 100 epochs, Faster RCNN's loss value curve (a) drops to 0.2, while Yolov4's loss value (b) drops to 0.02. Yolov4's precision and recall rate curve fluctuate widely, and Faster RCNN's precision curve is smoother, but the recall rate is about 10 percentage points lower than Yolov4's (c, d).
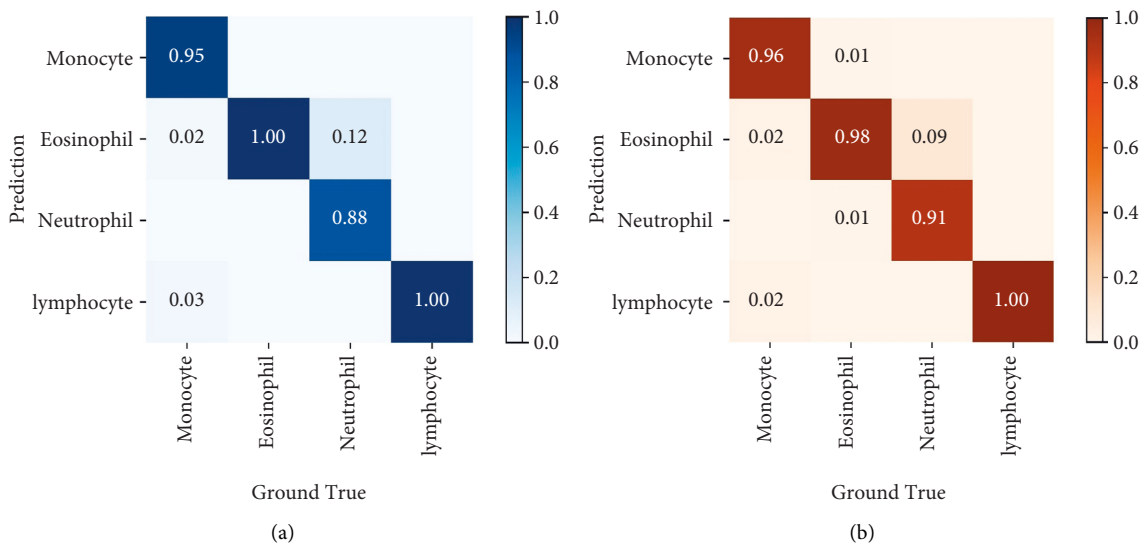

(a)

(b)

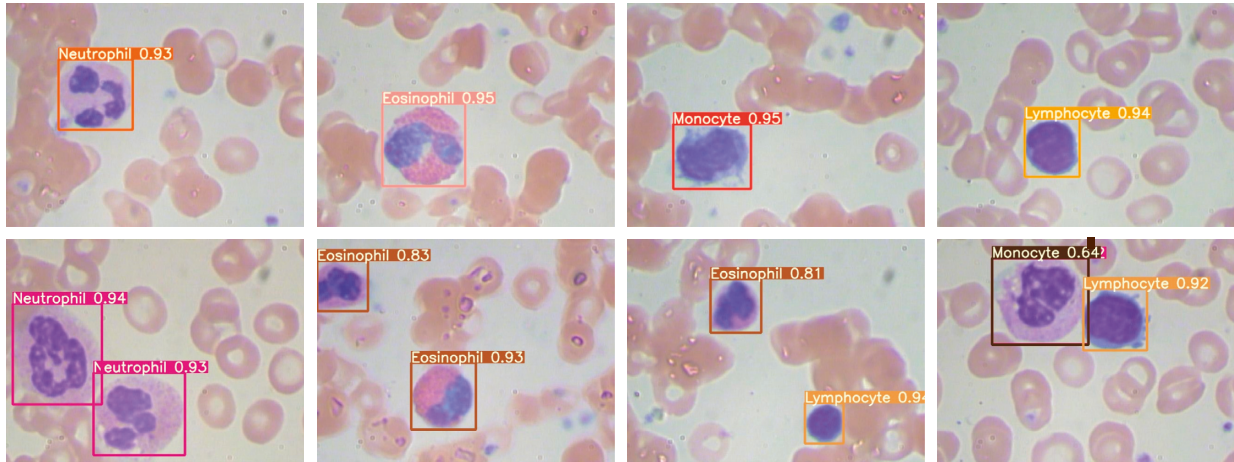Figure 7: Confusion matrix of (a) Yolov4 and (b) Faster RCNN.

FIGURE 8: The inference performance of the Yolov4 model on unseen images.

evaluates the ability of models to identify related objects, and the recall rate evaluates the ability of models to find all relevant positive bounding boxes.

We trained the two models for 100 epochs and evaluated their training loss, precision, and recall rate, as shown in Figure 6. We found that the training loss of both models will decay as the number of training iterations increases. The training loss of the Yolov4 model fluctuates less and the convergence is small (Figure 6(a)), while the training precision of Faster RCNN fluctuates slightly but its recall rate is more than 10% lower than that of Yolov4 (Figures 6(c) and 6(d)). It took 0.65 hours of training time for the Yolov4 to fit the model, which was 0.215 times that of Faster RCNN. It can be seen that Yolov4 is lighter and requires less computing power, so easier to integrate into miniaturized equipment.

We verified both models on the test set and obtained the confusion matrix of the four types of WBCs (see Figure 7). In the confusion matrix, we can see that both models have achieved great performance. The classification accuracy rates reached 95.75% and 96.25%, respectively. In general, the classification effects of lymphocytes and eosinophils were the best, and the classification effect of neutrophils was the worst. Both models tend to recognize some neutrophils as eosinophils. Both neutrophils and eosinophils are granulocytes, and the difference in nucleus morphology was not significant compared with other types of WBCs. The identification of such subtle differences has always been the driving force for the continuous development of object detection. The detection of the specific dataset (such as raw staining-free WBCs image by self-build imaging system) may require special modifications to the network, such as increasing the depth of feature extraction backbone and optimizing the recognition of small targets, which is our main direction in the future research.

Finally, we used the weights obtained from Yolov4 training to infer the original untrained image in the BCCD. As a result, the model took about 16 ms to detect an image, which is about 60 FPS. It was proved that Yolov4 had a great classification and recognition effect for the four WBC subtypes (see Figure 8). It could also achieve excellent detection for some cells that were closely spaced or located at the edges. The confidence of all inference results is mostly above 0.8.

5. Conclusions and Future Perspectives

WBC classification is essential in medical diagnostics as hematologists can estimate a patient's immune condition using accurate WBC classification technology. The traditional manual classification method is limited by many factors, and the automatic WBC classification technology based on computer vision has gradually developed. Feature engineering classification systems based on machine learning and automatic feature extraction classification system based on deep learning are the two most representative technologies. However, both of them need to pre-segment abundant cells from the background, and only one single cell can be classified at a time; this greatly reduces processing efficiency. As a result, the complexity of the entire system is increased and the effect of classification strictly depends on image preprocessing.

In this paper, we described the superiority of using high-efficiency object detection for WBC classification compared to classic classification methods with preprocessing of single-cell segmentation. We comparatively applied two object detection models to classify the four subtypes of WBCs on an enhanced BCCD with more than 95% classification accuracy. Through the comparative experiments, we have verified that Yolov4 achieves excellent detection speed (up to 60 FPS) while ensuring high accuracy compared to Faster RCNN (about 15 FPS), which provides a favorable reference for real-time WBC classification at the POC. Therefore, it is potentially feasible to use one-stage object detection models in a microfluidic imaging system, that is, no preprocessing work; for example, cell segmentation is required to directly classify WBCs on an image containing multiple cells.

In future research, we intend to apply the Yolov4 to classify the staining-free WBCs that have blurrier morphological features and no obvious nuclei. Based on this study, our envisioned plan to optimize the Yolov4 based model is to increase the depth of the backbone network to

extract smaller granularity features and strengthen the feature fusion algorithm of the Neck network so that the detection performance of small WBC targets can be improved. Meanwhile, we will also consider the overall efficiency of the network, obtaining the trade-off between accuracy and speed.

Data Availability

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] B. Medical, "Medical gallery of blausen medical 2014," *WikiJournal of Medicine*, vol. 1, no. 2, pp. 1–79, 2014.

[2] Q. Wang, J. Wang, M. Zhou, Q. Li, Y. Wen, and J. Chu, "A 3D attention networks for classification of white blood cells from microscopy hyperspectral images," *Optics & Laser Technology*, vol. 139, Article ID 106931, 2021.

[3] G. Gulati, J. Song, A. D. Florea, and J. Gong, "Purpose and criteria for blood smear scan, blood smear examination, and blood smear review," *Annals of laboratory medicine*, vol. 33, no. 1, pp. 1–7, 2013.

[4] S. Khan, M. Sajjad, T. Hussain, A. Ullah, and A. S. Imran, "A Review on Traditional Machine Learning and Deep Learning Models for WBCs Classification in Blood Smear Images," *IEEE Access*, vol. 9, Article ID 10657, 2020.

[5] K. A. K. Al-Dulaimi, J. Banks, V. Chandran, I. Tomeo-Reyes, and K. Nguyen Thanh, "Classification of white blood cell types from microscope images: Techniques and challenges," *Microscopy Science: Last Approaches on Educational Programs and Applied Research (Microscopy Book Series, 8)*, pp. 17–25, CRC Press, Boca Raton, FL, USA, 2018.

[6] A. Zheng and A. Casari, *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*, O'Reilly Media, Inc., Newton, MA, USA, 2018.

[7] B. J. Erickson, P. Korfiatis, Z. Akkus, and T. L. Kline, "Machine learning for medical imaging," *RadioGraphics*, vol. 37, no. 2, pp. 505–515, 2017.

[8] J. Riordon, D. Sovilj, S. Sanner, D. Sinton, and E. W. K. Young, "Deep Learning with Microfluidics for Biotechnology," *Trends in Biotechnology*, vol. 37, no. 3, pp. 310–324, 2019.

[9] G. Litjens, T. Kooi, B. E. Bejnordi et al., "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[10] M. Lippeveld, C. Knill, E. Ladlow et al., "Classification of Human White Blood Cells Using Machine Learning for Stain-Free Imaging Flow Cytometry," *Cytometry, Part A*, vol. 97, no. 3, pp. 308–319, 2020, (in English).

[11] M. Nassar, M. Doan, A. Filby et al., "Label-Free Identification of White Blood Cells Using Machine Learning," *Cytometry, Part A*, vol. 95, no. 8, pp. 836–842, Aug 2019.

[12] J. Guo, H. He, T. He et al., "GluonCV and GluonNLP: Deep Learning in Computer Vision and Natural Language Processing," *Journal of Machine Learning Research*, vol. 21, no. 23, pp. 1–7, 2020.

[13] X. Huang, J. Liu, J. Yao et al., "Deep-Learning Based Label-Free Classification of Activated and Inactivated Neutrophils for Rapid Immune State Monitoring," *Sensors*, vol. 21, no. 2, p. 512, 2021.

[14] C. Chen, A. Mahjoubfar, L. Tai et al., "Deep Learning in Label-free Cell Classification," *Scientific Reports*, vol. 6, no. 1, p. 21471, 2016 (in English).

[15] X. Shu, S. Sansare, D. Jin, K. Tong, R. Pandey, and R. Zhou, "White blood cell classification using quantitative phase microscopy based deep learning," in *Proceedings of the Bio-Optics: Design and Application*, April 2019.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, 23-28 June 2014.

[17] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, December 2015.

[18] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2016.

[19] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969, Venice, Italy, October 2017.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.

[21] W. Liu, D. Anguelov, D. Erhan et al., "SSD: Single shot multibox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Springer, September 2016.

[22] H. Kutlu, E. Avci, and F. Özyurt, "White blood cells detection and classification based on regional convolutional neural networks," *Medical Hypotheses*, vol. 135, Article ID 109472, 2020.

[23] X. Wang, T. F. Xu, J. Z. Zhang, S. N. Chen, and Y. Z. Zhang, "SO-YOLO Based WBC Detection With Fourier Ptychographic Microscopy," *Ieee Access*, vol. 6, 2018 (in English), Article ID 51566.

[24] Q. Wang, S. Bi, M. Sun, Y. Wang, D. Wang, and S. Yang, "Deep learning approach to peripheral leukocyte recognition," *PloS One*, vol. 14, no. 6, Article ID e0218808, 2019.

[25] Z. Jiang, X. Liu, Z. Yan, W. Gu, and J. Jiang, "Improved detection performance in blood cell count by an attention-guided deep learning method," *OSA Continuum*, vol. 4, no. 2, pp. 323–333, 2021.

[26] X. Huang, D. Xu, J. Chen et al., "Smartphone-based analytical biosensors," *Analyst*, vol. 143, no. 22, pp. 5339–5351, 2018.

[27] D. Xu, X. Huang, J. Guo, and X. Ma, "Automatic smartphone-based microfluidic biosensor system at the point of care," *Biosensors and Bioelectronics*, vol. 110, pp. 78–88, 2018.

[28] R. L. E. Cano and H. D. E. Lopera, "Introduction to T and B lymphocytes," in *Autoimmunity: From Bench to Bedside*

[Internet], El Rosario University Press, Bogota, Colombia, 2013.

[29] J.-G. Lee, S. Jun, Y. W. Cho et al., "Deep Learning in Medical Imaging: General Overview," *Korean Journal of Radiology*, vol. 18, no. 4, pp. 570–584, 2017, (in English).

[30] C. Shorten and T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, 2019.

[31] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.

[32] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, Honolulu, HI, USA, July 2017.

[33] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018, https://arxiv.org/abs/1804.02767.

[34] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," 2020, https://arxiv.org/abs/2004.10934.

[35] S. Yun, S. Yun, D. Han et al., "CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features," in *Proceedings of the International Conference on Computer Vision*, 2019.