Research Article/Araştırma Makalesi

GAZİ
JOURNAL OF ENGINEERING SCIENCES

Mask R-CNN Based Segmentation and Classification of Blood Smear Images

Hilal Atıcı[*a], **Hasan Erdinç Koçer**[b]

Submitted: 27.06.2022 Revised: 14.02.2023 Accepted: 11.03.2023 doi:10.30855/gmbd.0705058

ABSTRACT

Analysis of microscopic images is a reliable laboratory method that provides useful information for disease diagnosis in medical field. Although advanced technological devices can help with blood disease diagnosis, a microscopic blood smear examination is required for a definitive diagnosis. Nowadays, technicians in many laboratories use microscope to detect anomalies in cells (defects in the cell, parasites, low or excess cell count, etc.). Experts' detection of anomalies provides critical information for disease diagnosis. For the expert, analysing microscopic images is a time-consuming and error-prone procedure. As a result, in this study, a method for accelerating expert examination and automatically detecting cells is presented. The importance of basic blood cell segmentation and classification is emphasized. PBC data set blood smear images were used as the data set. The system was built using the Mask R-CNN architecture, which is a region-based convolutional neural network. Mask R-CNN was tested using various spine structures. The sample segmentation feature in the Mask R-CNN algorithm was used to determine the segmentation of blood cells obtained from images, and the error rates were minimized as a result of the tests. The study focused on detecting eight classes but could be improved by enriching it with further classes and using blood cell images from different angles and better segmentation.

Keywords: Deep Learning, Segmentation, Classification, Blood Cell

[a,*] Selcuk University, Technology Faculty, Computer Engineering 42130 - Konya, Türkiye
Orcid: 0000-0002-1859-8085
e mail: hilalatici@eskisehir.edu.tr

[b] Selcuk University, Technology Faculty, Electrical and Electronics Engineering 42130 - Konya, Türkiye
Orcid: 0000-0002-0799-2140
e mail : ekocer@selcuk.edu.tr

*Corresponding author: hilalatici@eskisehir.edu.tr

Kan Yayma Görüntülerinin Mask R-CNN Tabanlı Segmentasyonu ve Sınıflandırılması

ÖZ

Mikroskobik görüntülerin analizi, sağlık alanında hastalık teşhisinde yararlı bilgiler veren güvenilir bir labaratuvar yöntemidir. Kan hastalıklarının teşhisinde, ileri teknolojili cihazlar önemli bilgiler verseler de kesin tanı için mikroskobik kan yayma incelemesine ihtiyaç duyulmaktadır. Günümüzde, mikroskop birçok laboratuvarda teknisyenler tarafından kullanılmakta ve hücrelerdeki anomaliler (hücredeki bozukluklar, parazitler, düşük veya fazla hücre sayısı vb.) tespit edilmektedir. Uzmanların tespit ettiği anomaliler hastalıkların teşhisinde önemli bilgiler sunmaktadır. Mikroskobik görüntülerin analizi uzman için zaman alan ve hataya açık bir prosedürdür. Bu nedenle, bu çalışmada uzman tarafından uygulanan incelemeyi hızlandıran ve otomatik hücre tespit edebilen bir yöntem önerilmiştir. Temel kan hücrelerinin segmentasyonu ve sınıflandırılması üzerinde durulmuştur. Veri seti olarak PBC veri seti kan yayması görüntüleri kullanılmıştır. Sistemin geliştirilmesinde bölge-tabanlı evrişimsel sinir ağı olan Mask R-CNN mimarisi kullanılmıştır. Mask R-CNN için farklı omurga yapıları kullanılmış ve değerlendirilmiştir. Görüntülerden elde edilen kan hücrelerinin segmentasyonu, Mask R-CNN algoritmasında bulunan örnek bölütleme özelliği sayesinde farklı renklendirmeler yolu ile tespit edilmiş, hata oranları yapılan testler sonucunda en aza indirgenmiştir. Çalışmada sekiz sınıf tespitinde odaklanılmıştır ancak çalışma daha fazla sınıflar ile zenginleştirilerek ve farklı açılardan elde edilen kan hücresi görüntüleri kullanılarak geliştirilebilir ve daha iyi bölütleme yapılabilir.

Anahtar Kelimeler: Derin Öğrenme, Segmentasyon, Sınıflandırma, Kan Hücresi

To cite this article: H. Atici, and H. E. Koçer, "Masl R-CNN Based Segmentation and Classification of Blood Smear Images," *Gazi Journal of Engineering Sciences,* vol. 9, no. 1, pp. 128-143, 2023. doi:10.30855/gmbd.0705058

1. Introduction

The microscope has obviously had an indispensable place in the field of medical diagnosis since its invention. Detection of microorganisms, not found in normal and healthy bodies yet in sick bodies, eliminating the microorganisms and dismissing the symptoms of the disease is a well-known treatment method. Moreover, conditions such as the lack of the required number of cells in the blood are important in the diagnosis and treatment of diseases. In addition, the distortion of the shape of the cells whose shape and properties are known (Amorphous Cell) is a mattering symptom for diseases. As can be conceded from here, the analysis of microscopic images has a vital place in medicine.

The analysis of microscopic images is utilized in many fields of technology and medicine. Today, technicians in many laboratories use the microscope to detect anomalies (defects in the cell, parasites, low/over cell count, and other abnormal conditions) Important information such as the status and number of cells or parasites distinguished by technicians are important in the diagnosis of the disease. There is a need for a more agile, reproducible method than human examination and classification of cells. For this reason, many image processing software has been developed to diagnose the disease and to obtain useful information from medical images.

The lack of specific signs and symptoms of some diseases causes misdiagnosis. Therefore, manual classification of blood cells is time-consuming and error-prone, making it difficult for experts to detect and classify blood cells. Accordingly; different cells need to be identified quickly, accurately and automatically.

The main blood cells examined in the peripheral smear are red blood cells (RBC), white blood cells (WBC) and platelets [1]. In contrast to the similarity in shape seen in red blood cells and platelets, leukocytes differ in cell type. Therefore, the analysis and classification of white blood cells in microscopy have gained prominence.

Many studies on the analysis of microscopic blood images have been published in the literature. Blood and urine analyses were performed in these studies, classification was made, and the diagnosis of some related diseases was emphasized.

The three known red blood cells, Elliptocyte, Discocyte, and Echinocyte, were classified using morphological methods in the 2010 study. As a result, a simple statistical analysis of each edge pixel's distance from the centre of mass was used. This method was 95.36% successful in recognizing Elliptocytes, 96.7% in recognizing normal Discocytes, and 98.63% in recognizing Echinocytes [2].

In a study, which was presented in 2013, the automated Otsu method and blood cell segmentation method were proposed for WBC (white blood cell) segmentation, along with image enhancement and arithmetic. The kNN classifier was used to classify blast cells from normal lymphocyte cells. The system was applied to 108 images in the public image dataset for the study of leukaemia. This method provided 93% accuracy [3].

In another study, which was examined in 2013, proposed a method for recognizing and classifying blood cells in microscopic images based on invariant moments and multi-class support vector machines (SVM). The research is divided into four stages: pre-processing, feature extraction, classification, and testing. Grayscale, median filtering, contrast, thresholding, and morphological-logical processes are all part of the pre-processing stage. The constant moment values of the blood cells are calculated during the feature extraction stage. In the classification phase, a multi-class support vector machine (SVM) was used to classify the features extracted in the previous step. In the test phase, the success rates of the proposed system were examined. As a result of the test, a total success rate of 98.4% was determined [4].

In 2014, L.Putzo et al. were presented a complete and fully automated method for WBC identification and classification using microscopic images. Unlike other methods for defining the nucleus, the proposed method isolates the entire leukocyte before separating the nucleus and cytoplasm. This approach is required to thoroughly examine each cell component. Using a novel approach to remove the background pixel, different features such as shape, colour and texture are extracted from each cell component. This feature set was used to train various classification models to determine which one is

best suited for leukaemia detection. Out of 33 images taken with the same camera and under the same lighting conditions, 245 (92% accuracy) of a total of 267 leukocytes were correctly detected using this method. Using different classification models, it was found out that the support vector machine with a Gaussian radial-based core was the best model for leukaemia identification, with 93% and 98% accuracy [5].

In 2016, a study was conducted on the classification and counting of white blood cells. The study consists of image acquisition, image segmentation, feature extraction, classification and counting stages. Segmentation consists of two stages: cell and nucleus. The image dataset consists of 70 microscopic images of a blood smear. There are 90 samples of leukocytes including Lymphocyte, Neutrophil and Eosinophil. The classification was performed using the Neural Network Pattern Recognition tool in MATLAB. The overall accuracy using this method is 98.9%. Eosinophils were detected with 100% accuracy, Lymphocytes 96.7% and Neutrophils with 100% accuracy [6].

A software that can detect, count and classify blood cells (red and white) using various image processing techniques on blood sample images acquired digitally in a microscopic environment was developed in a 2018 study. The software employs various image processing techniques such as pre-processing, adaptive thresholding, Watershed and Hough transform. Blood cells were detected with an accuracy of 87-96% using the developed software [7].

In 2019, an effective image processing algorithm was designed to increase the probability of a diagnosis of acute lymphocyte leukaemia (ALL) cells, which are rather common in children and can result in death if left untreated. In the first step, the image is called up and converted to a grey-level image. Then, a discrete-time wavelet transform is applied to the image and a grey-level co-occurrence matrix is created for feature extraction and all features are written into the matrix. Features are extracted for training and test data. In the classification phase, the SVM method was used. While the success rate was 95.700% for cancer data, it was calculated as 96.466% for non-cancer data [8].

In a study conducted in 2020, an automatic PBC (Peripheral Blood Cell) image classifier compatible with mobile devices was designed. With its easy-to-implement lightweight structure and computationally efficient architecture, the proposed ShuffleNet based model has achieved a good solution for PBC classification. By eliminating manual feature extraction, the proposed system requires the expert to only feed the model with raw input images. The proposed deep learning model has reached an accuracy value of 97.94% while leaving behind the benchmark study with the same data set [9]. In 2021, a study was conducted on peripheral blood smear classification by tuning to pre-trained CNN architectures on the ImageNet dataset. Transfer learning is used by deep learning models to use previously learned features and to learn new features. All 27 models achieved greater than 98% accuracy in experiments, and 14 of them achieved greater than 99% accuracy. Thus, the study achieved success compared to previous studies. The study will be useful for future studies with a basic approach for blood cell classification in peripheral blood smear [10]. In a 2021 study, the Blood-Caps architecture was used to classify cells in a peripheral blood smear. PBC dataset consisting of images of 8 blood cell types was used. BloodCaps performed well for classification on limited datasets when tested. BloodCaps architecture outperforms AlexNet, VGG16, ResNet18 and InceptionV3. This study yielded a good result by classifying cells with 99.3% accuracy in blood smear microscopic examination [11]. Classification of blood cells is important for disease diagnosis and treatment. White blood cells also play a key role in diagnosing critical illnesses. There are different types of these cells. When diagnosing diseases related to white blood cells, experts need to distinguish the total number and types of the cells. In one study, a pre-trained deep learning model was used to distinguish between different types of WBCs. PBC dataset was used in this study. The best model on the data set was the Inception ResNetV2 model with Adam optimizer with 98.4% classification accuracy [12].

2. Material and Method

In this study, a segmentation study was performed on blood sample images obtained digitally in the microscopic environment in the PBC data set. Mask R-CNN architecture, a region-based convolutional neural network, was used for the segmentation and classification of cells on the PBC Dataset. Different deep learning models can be used as backbones for feature mapping in Mask R-CNN architecture. The backbone structure is a standard convolutional neural network and is used as a feature extractor. These

are the architectures ResNet50, ResNet101, ResNet152, ResNeXt50, VGG19, DenseNet121 and DenseNet169.

2.1. Deep Learning

Deep learning is a technology that is among the machine learning techniques. With deep learning, the aim is to transfer the learning ability of human beings to computers in our daily life and to perform the desired operations on the data. Deep learning algorithms are more advanced versions of artificial neural networks (ANNs). Structures, defined as neurons in ANN systems, are modelled to be interconnected in the same way that neurons in the biological nervous system interact with one another. The distinguishing aspect of Deep Learning algorithms from existing algorithms in machine learning is that they require a very high amount of data and hardware with very high computing power that can process this high data with its complex structure [13].

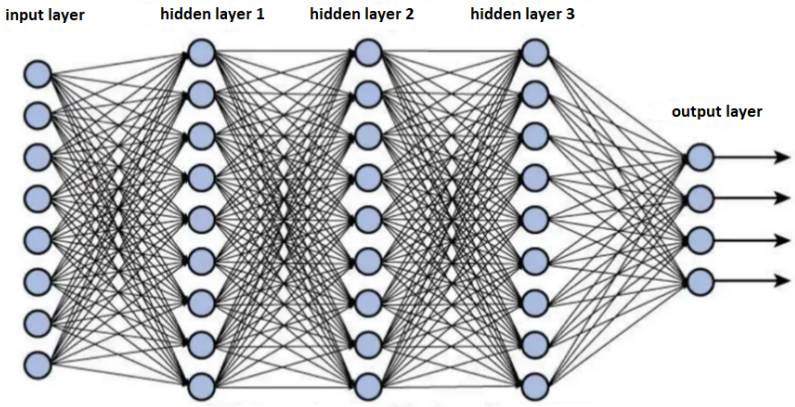


Figure 1. Deep network architecture with multiple layers[14]

Deep Learning methods learn distinguishing features on their own using a large number of inputs. The system must be adequately trained for this feature learning process to be successful. The feature learning phase is divided into layers. Lower-level features are less distinctive, whereas higher-level features, which are formed by merging lower-level layers, are more distinctive. Lower-level features serve as the foundation for creating more meaningful features. This method of learning differs from traditional Machine Learning algorithms. Because the features determined by human must be calculated prior to the training phase in traditional Machine Learning algorithms. The learning process is guided by these calculated properties [13].

2.2. Mask R-CNN

Mask R-CNN is a deep neural network that aims to solve the instance segmentation problem in computer vision. Mask R-CNN is a convolutional neural network that can segment different objects in an image or video. It has been developed for object recognition, location finding and segmentation on the image [15].

The first layer of the Mask R-CNN architecture is a convolutional neural network (CNN). In Faster R-CNN, by extracting the feature map from the input images, the existing objects on the image are classified and their locations are found. A regional recommendation network called the Region Proposal Network (RPN) is used to identify regions of an image that potentially contain an object. Then, the regional proposals are pooled and an estimate is made among the predetermined class or classes, which is finally presented as input data to the fully connected layer for object estimation [16]. While these processes are carried out more slowly in Fast R-CNN, faster results are obtained in Faster R-CNN [17]. The Mask R-CNN algorithm colors the surface of the predicted object with high accuracy and performs masking. In this segmentation process, instance segmentation feature is used. In example segmentation, if there is more than one object belonging to the same type of object in the image, these objects are separated from each other with different types of colours.

Mask R-CNN (regional convolutional neural network) consists of two stages. The image is scanned in the first step, and suggestions are made for areas of the image that may contain objects. In the second

stage, these suggestions are evaluated and classified, the bounding box detection is improved, and segmentation masks are generated. Each stage depends on the backbone structure. The Mask R-CNN architecture structure used is given in Figure 2.

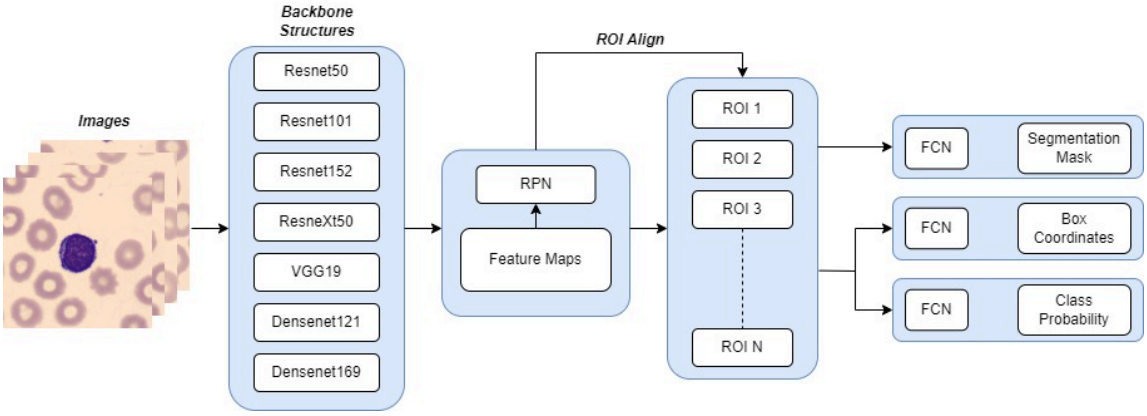


Figure 2. Mask R-CNN architectural structure

The backbone is a convolutional neural network and is used for feature extraction. The feature map is created by passing the image through the backbone structure. This feature map is given as the next layers input value.

2.3. Backbone architectures

ResNet50, ResNet101 and ResNet152 Model

ResNet architecture is a CNN (Convolutional Neural Network) architecture that has been proposed to solve deep and complex classification problems [18]. ResNet differs from traditional sequential network architectures in that it has a microarchitecture module structure [19]. This model [20], created by Microsoft team, was developed to eliminate the problem of gradient values converging to zero in multilayer deep networks. In the ResNet architecture, a bridge is established between the lower and upper layers by sequentially transferring the attribute information obtained from the lower layers to the upper layers [18]. ResNet architecture has evolved into different architectures such as ResNet50, ResNet101 and ResNet152 according to the number of layers used. Jumping between layers in ResNet is called ResBlock. Thanks to ResBlock, even if nothing is learned in the previous layer, the information from the old layer is applied to the new layer, making the model stronger.

The x layer is carried by the shortcut link to collect the input. Links that skip one or more layers are known as shortcut links. Thus, blocks and inputs now propagate faster over the connections between the layers.

Before using the shortcut link, it is multiplied by the weights of the input layer and added with a bias term. Then it goes through the activation function and the output(H(x)) is obtained.

$$H(x) = f(wx + b) \text{ or } H(x) = f(x) \quad (1)$$

With the shortcut link;

$$H(x) = f(x) + x \quad (2)$$

the output changes as (2).

In the second approach, additional parameter is used. The new output we created by adding additional parameters is;

$$H(x) = f(x) + w1.x \quad (3)$$

the output be as (3).

VGG19 Model

This is a deep-learning model developed in 2014, which showed a very successful performance in the ImageNet 2014 competition with an error rate of 7.3%. This architecture, designed by Simonyan and Zisserman at Oxford University, revealed 6 different architectures. It consists of 11, 13, 16, 19 convolution layers in 6 different models [21]. Unlike the convolutions in previous deep learning architectures, 2x2 and 3x3 filters are applied here. This architecture includes a 3-step fully connected (FC FullConnected) layer. The last fully connected layer has 1000 neurons. In the last part of the network, there is the classification layer, the SoftMax layer. Among these models, VGG16 and VGG19 models are more common in the literature [22].

ResNeXt50 Model

ResNeXt was created by Xie et al. (2016). This architecture was developed based on ResNet architecture, which also uses the idea of residual blocks for maintaining information from previous layers. The main difference between ResNeXt and ResNet is instead of having continual blocks one after the other, 'cardinality', which is the size of transformations, was considered and implemented in the architecture, inspiring from Inception/GoogLenet. Compared to ResNet, ResNeXt has fewer parameters but better performance in ImageNet Challenge [23].

DenseNet121 and DenseNet169 Model

During the training of neural networks, since convolution and down sampling are performed, the image feature decreases in feature maps and during transitions between layers. The DenseNet (Dense Convolutional Network) system is [24] developed for more effective use of image feature information. Thus, each layer provides information flow to all subsequent layers, and each layer can access the property information of the previous layers. The biggest advantage of DenseNet architecture is that it provides feature propagation and enables the reuse of the obtained features. This also reduces the number of parameters in the network. There are versions with different layers such as DenseNet-121, DenseNet-169 and DenseNet-201.

2.4. PBC (Peripheral Blood Cell) Dataset

The PBC dataset consists of a total of 17,092 normal cell images obtained using the CellaVision DM96 analyzer in the Core Laboratory at the Barcelona Hospital Clinic. The format of the images is jpg and the size is 360x363 pixels. All images were obtained in the color space RGB. Dataset were labeled by clinical pathologists at the Hospital Clinic [25]. Figure 3 shows examples of the eight types of normal peripheral blood cells (Basophil, Eosinophil, Erythroblast, Lymphocyte, Mmy, Monocyte, Neutrophil, Platelet) taken from the PBC dataset.

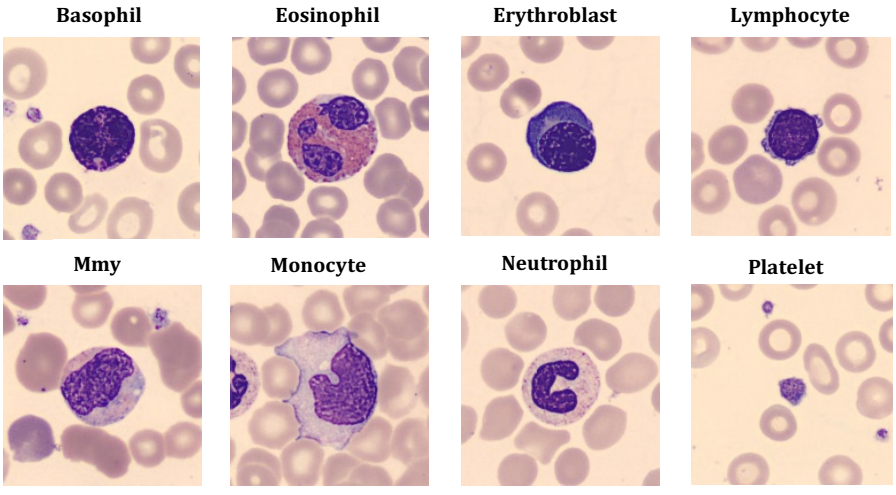


Figure 3. Sample cell images from the PBC dataset

As we told before, the PBC dataset consist of eight classes: neutrophils, eosinophils, basophils, lymphocytes, monocytes, immature granulocytes (promyelocytes, myelocytes, and metamyelocytes - Mmy in short), erythroblasts, and platelets. The cell informations (total number of images and percentages) of PBC dataset is given in Table 1. Images were obtained from individuals who did not have infection, hematological or oncological disease and did not receive any pharmacological treatment.

Table 1. Types and number of cells in each group

Cell Type	Total of Images by Type	%
Neutrophils	3329	19.48
Eosinophils	3117	18.24
Immature Granulocytes (Metamyelocytes, Myelocytes and Promyelocytes)	2895	16.94
Platelets (Thrombocytes)	2348	13.74
Erythroblasts	1551	9.07
Monocytes	1420	8.31
Basophils	1218	7.13
Lymphocytes	1214	7.10
Total	17,092	100

In order to use the dataset in the deep learning model, firstly, the labeling process was performed. The data labeling process was done with the MakeSense tool [26], which can be labeled online. The labeling process is important in terms of giving deep learning algorithms to the training set where it can distinguish the desired objects and train itself. After the dataset was separated as the training and validation set, the necessary labels were made for the images in the train and validation set (Figure 4). Eight classes in the dataset were added as labels. The cells in each image are marked as polygons and the class label they belong to is selected.

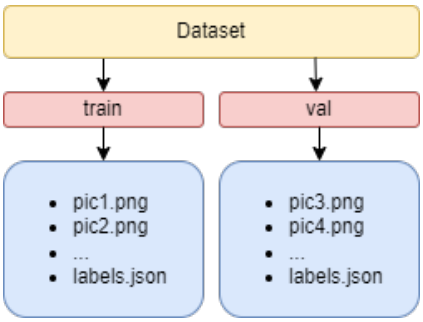


Figure 4. Dataset file structure

2.5. Performance Metrics

The segmentation performance of an object detection model is evaluated using measurement criteria such as IoU (Intersection over Union), Precision, Recall, Average Precision (AP) and mean Average Precision (mAP).

IoU: IoU is an evaluation metric used to measure similarity between Ground Truth and model prediction. The metric is calculated as given in equation (4).

$$IoU = \frac{Area\ of\ Union}{Area\ of\ intersection} \tag{4}$$

Average Precision (AP): AP is measurement metric that includes precision and recall metrics used to evaluate object detection performance. It is a number metric that summarizes the Precision-Recall curve by averaging the recall values from 0 to 1. The metric is calculated as given in equation (5).

$$AP = \frac{1}{11}\sum_{r\in(0,0.1,0.2,\ldots,1)} P_{interp(r)} \tag{5}$$

mAP: The mAP value is obtained by summing the APs of each class and dividing by the number of classes. The metric is calculated as given in equation (6).

$$mAP = \frac{1}{M}\sum_{j=1}^{M} AP_j \quad (6)$$

Measurement metrics such as sensitivity, specificity, and accuracy are used to evaluate the performance of classification models. Data from the confusion matrix are expressed as TP, TN, FP ve FN to calculate evaluation metrics. TP is the number of correctly detected instances of the class. FN is the number of instances belonging to the class but detected as a different class. TN is the number of correctly identified instances that do not belong to the class. FP is the number of samples that do not belong to the class but are determined to belong.

Accuracy: Accuracy is calculated by dividing the correct predictions detected against all detected predictions. The metric is calculated as given in equation (7).

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \quad (7)$$

Precision: Precision is calculated by dividing the correctly predicted positive samples against all predicted positive samples. The metric is calculated as given in equation (8).

$$Precision = \frac{TP}{(TP+FP)} \quad (8)$$

Recall: Recall is obtained by dividing the correctly predicted positive samples to all samples in the real class. The metric is calculated as given in equation (9).

$$Recall = \frac{TP}{(TP+FN)} \quad (9)$$

F1-Score: F1-Score is calculated by taking the weighted average of Precision and Sensitivity. The metric is calculated as given in equation (10).

$$F1 - Score = \frac{Precision*Recall}{Precision+Recall} \quad (10)$$

Specificity: Specificity is a evaluation metric used to measure the proportion of true negatives predicted as negative. The metric is calculated as given in equation (11).

$$Specificity = \frac{TN}{(TN+FP)} \quad (11)$$

Total loss in the training process; change of general error in cell detection, loss of validation; reduction of the error in determining the zones where the objects are located, frame loss; variation of the error of the boxes delineating the zones in which the objects are located, loss of classification; classification error and masking loss on the training set; shows the change of masking error on the training set.

3. Experimental Results

Mask R-CNN is a state-of-the-art algorithm capable of accurately detecting and segmenting the target object. In this study, an improved Mask R-CNN structure was proposed, and this structure was used to segment and classify the blood cell data set. The Mask R-CNN architecture's results in cell detection in the PBC dataset with eight different classes were evaluated, and the results- findings were analysed.

To train the deep network, a large number of labelled images is required. As a result, each class in the PBC dataset is labelled in the image's cells. A total of 4000 images were used for the training set, including 300 images from each class, 100 for validation, and 100 for testing.

In this study, 4000 images from the PBC dataset were trained using the Mask R-CNN algorithm with 100 iterations in each step for a total of 240 iterations to determine the cells in the dataset image. To begin, the images in the data set were divided into three datasets: training, validation and test, and the cells in each image were labelled using the Makesense.ai image labelling tool. The coordinates of the cells in the training and validation sets were determined by drawing them polygonally. Furthermore,

the error rates can be observed according to iterations using the Tensorboard graphical interface in the Tensorflow library to determine the error rates at the end of the training. The study was trained using a 12 GB RAM GPU in the Google Colaboratory cloud service. In addition, the study was written in Python programming language, and at the same time Keras [27] and Tensorflow [28] libraries were used.

The primary goal of this study is to investigate the effect of using different backbone network architectures during training for cell segmentation in the dataset and how the measurement values change as a result. The backbones used in this study consist of ResNet50, ResNet101, ResNet152, ResNeXt50, VGG19, DenseNet121 and DenseNet169. Many studies in the literature have used ResNet, ResNeXt (a different version of ResNet network), VGG19 and DenseNet networks with positive results. At the same time, these deep networks were used in the ImageNet competition as they did well. ResNet and DenseNet architectures have been tried and evaluated for networks of varying depths to achieve the best results.

Table 2. Settings specified for the Mask R-CNN model

Hardware	Feature	Description
Backbone	ResNet50 vb.	Backbone network model
Gpu Count	2	Number of GPUs
Image min dim	512	Image minimum edge length
Image_max dim	512	Image maximum edge length
Train rois per image	100	Estimated number of regions to be extracted from each image
Max gt instances	50	Maximum number of regions that can be found in an image
Images per gpu	2	Number of images per GPU
Num classes	9	Number of classes (8 cell classes + 1 background)
Steps Per Epoch	100	Number of repetitions per round
Epoch Count	240	Number of tour
Threshold	0.75	Threshold of object detection considered successful

Table 2 shows the special settings, including the parameters used for the Mask R-CNN model in this study. These settings have been tested and determined to produce the best results in experimental studies. The success of the tests has confirmed that the optimal iteration number is 100 and the number of rounds is 240 for trials with dual GPUs. Simultaneously, various improvements for test reviews and training were made using the model developed on the open-source framework.

Table 3. Loss rates

Backbone	Total Loss	Validation Loss	Frame Loss	Classification Loss	Masking Loss
ResNet50	0.09887	0.1062	0.01586	0.01282	0.05166
ResNet101	**0.08486**	**0.07841**	**0.01382**	**0.01157**	**0.04641**
ResNet152	0.1480	0.2893	0.0273	0.0365	0.0491
ResNeXt50	0.2387	0.1041	0.0499	0.04812	0.08295
VGG19	0.2493	0.2989	0.02605	0.1076	0.06895
DenseNet169	0.3681	0.1986	0.08537	0.04326	0.1055
DenseNet121	0.4103	0.1994	0.0667	0.0167	0.1356

Within the scope of this study, the error variation of the loss functions that occur during the training phase with the use of different backbones in the Mask R-CNN based model proposed on the dataset, depending on the tour number, is presented in Appendix 1. The loss rates obtained according to different backbone structures are given in Table 3. Segmentation performance results obtained with the validation set are given in Table 4.

Table 4. Consequences of backbone architectures

Backbone	mAP (0.5)	mAP (0.7)	mAP (0.9)	mAP (0.5-0.95)	mAR	F1-Score (%)
ResNet50	0.8902	0.8877	0.8443	0.8340	0.932	0.9105
ResNet101	**0.9139**	**0.9089**	**0.8645**	**0.8505**	**0.953**	**0.9329**
ResNet152	0.7572	0.7535	0.6707	0.6848	0.871	0.8102
ResNeXt50	0.6151	0.6107	0.4824	0.5429	0.784	0.6894
VGG19	0.2481	0.2393	0.1518	0.2060	0.313	0.2768
DenseNet169	0.4784	0.4596	0.3236	0.4057	0.624	0.5416
DenseNet121	0.7688	0.7563	0.5741	0.6802	0.907	0.8323

The dataset is trained using different backbones. The classification accuracy rates were evaluated using a test set of 100 images from each cell and a total of 800 images. The confusion matrix obtained for each spine is given in Appendix 2. Mask R-CNN structure achieved classification accuracy of 0.9898 with ResNet50 backbone, 0.9931 with ResNet101, 0.9688 with ResNet152, 0.9590 with ResNeXt50, 0.9580 with VGG19, 0.9638 with DenseNet121 and 0.9291 with DenseNet169. The performance metric results are given in the Table 5. The classification results according to the backbones used are given in Figure 5. The segmentation results of some cells on blood cell images are shown in the Appendix 3.

Table 5. Classification performance metrics

Backbone	Accuracy	Presicion	Recall	F1-Score	Specificity
ResNet50	0.9898	0.9575	0.9600	0.9575	0.9942
ResNet101	**0.9931**	**0.9725**	**0.9737**	**0.9687**	**0.9961**
ResNet152	0.9688	0.8250	0.8937	0.8275	0.9823
ResNeXt50	0.9590	0.7500	0.7425	0.6562	0.9778
VGG19	0.9580	0.4600	0.4037	0.4300	0.5687
DenseNet169	0.9638	0.8500	0.8637	0.8475	0.9707
DenseNet121	0.9291	0.6155	0.7537	0.5968	0.9265

Table 6 draws bigger picture of the overall performances of the whole scenario in the study. As it is clear from the overall performance measurements, the proposed method achieved an accuracy of 99.31%.

Table 6. Overall performances of the whole scenario

Studies	Accuracy	Presicion	Recall	Specificity
Proposed Study	**0.9931**	**0.9725**	**0.9737**	**0.9961**
[9]	0.9794	0.9794	0.9794	0.9971
[29]	0.9620	0.9700	0.9600	0.9680
[30]	0.9519	0.9232	0.9039	0.9525
[31]	0.9560	0.9840	0.9500	0.5687

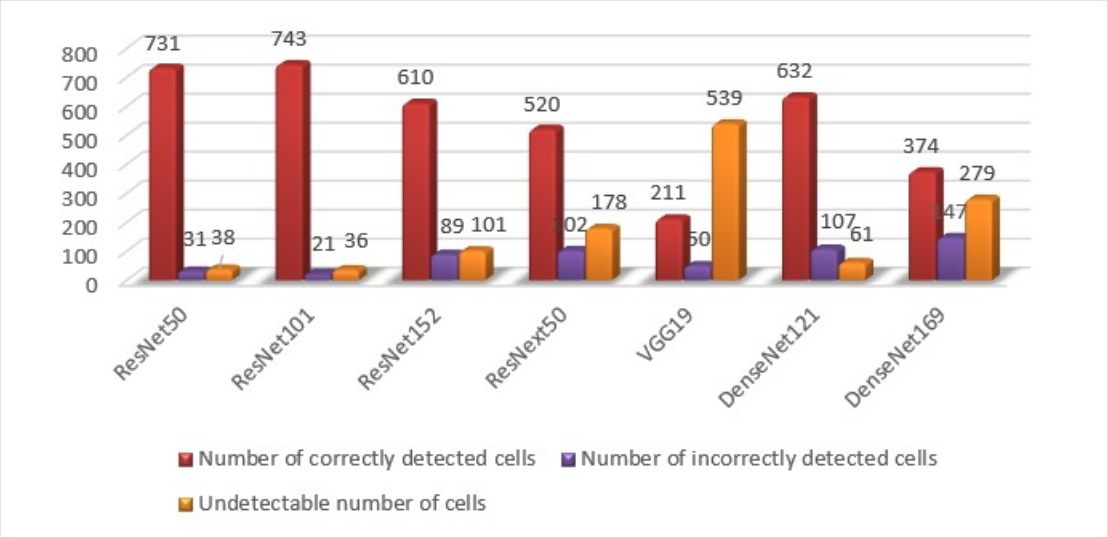


Figure 5. Backbone architectures classification results

4. Conclusion and Discussion

The Mask R-CNN algorithm was used in this study to detect blood cells using the PBC data set. The sample segmentation feature in the Mask R-CNN algorithm was used to determine the segmentation of blood cells obtained from images, and the error rates were minimized as a result of the tests. The study focused on the identification of eight classes, but it could be expanded to include more classes and used for a variety of purposes. Furthermore, better segmentation can be achieved by passing images

containing more blood cells or different cell structures obtained from various angles during the training process.

Many studies on blood cells using various methods have been proposed in the literature on the PBC open data set used in this study. However, comparing previous studies on computer-aided cell segmentation on blood cell images is rather hard as the data sets are not the same, the algorithms are diverse and dissimilar similarity measures are used. This study presented a structured deep-learning method for assisting experts in the interpretation of PBC dataset images. Simultaneously, the Mask R-CNN segmentation method was trained and evaluated using various backbone architectures. To extract feature maps from images, backbone architectures are used. As the backbone architecture, deep learning architectures ResNet50, ResNet101, ResNet152, ResneXt50, VGG19, Densenet121 and Densenet169 were used. When the results of experimental studies using different backbone architectures are compared, it is clear that the segmentation results obtained using the ResNet101 backbone architecture of the Mask R-CNN method are relatively more successful. Although the metric results confirmed that very high segmentation performances were obtained on the data set, the segmentation of some cells on blood cell images is not at the expected level in tests performed with the proposed Mask R-CNN model.

Although the image segmentation results are reasonable, this and similar issues can have a variety of causes. Many factors can be considered, including the presence of noise in the related images, the intertwining of the textural colours of the non-segmented cell and other cells, the similarity of their structural features, and an insufficient number of rounds in the Mask R-CNN network's training to detect the cells in these images. When the segmented images are not successful, it can be seen that the cell overlaps with the cells around it, the characteristics of the cells such as structure, shape, texture and intracellular saturation are similar to other cells, the colour tones of the cell types are close to each other, the edges of the cells are not sharply defined, the cell nuclei are similar. In such cases, it was understood that the images could not be fully segmented. In some cases, cells were segmented but resulted in classification failure due to similarities in cell types.

The contribution of computer-assisted secondary tools to assist physicians in segmenting cells on digital images of blood cells has been growing in recent years. It is observed that the segmentations made manually by physicians contain too many time-consuming procedures that are prone to human error. Therefore, automatic segmentation of cells on blood cell images has noteworthy advantages in terms of both time and cost. In this study, a method including Mask R-CNN-based current technologies that can help physicians and specialists for blood cell segmentation is proposed. In the experimental studies on the PBC dataset, the proposed method was found to be quite successful in the segmentation of cells. In addition, the results of Mask R-CNN supported by the flexible parameter selection proposed for the region of interest, and the effectiveness and validity of the proposed method have been verified. The contributions of this study can be summarized as follows. (1) In this study, which is proposed for the segmentation of cells on blood cell images in the PBC data set, performance evaluation was made using different backbones and supported by the flexible parameter selection recommended for the region of interest, for ResNet101 in experimental studies conducted with Mask R-CNN model; segmentation performance metrics of mAP(0.5), mAP(0.7), mAP(0.9), mAP(0.5-0.95), mAR and F1-Score (%) were calculated as 0.91, 0.90, 0.86, 0.85, 0.95 and 0.93, respectively. (2) As a secondary tool, a computer-aided support system to assist physicians in segmenting blood cells is presented. In the future, it is planned to create a new dataset with microscopic images obtained from blood and urine at various resolutions, sizes, and scanners, and to achieve high detection performance in the studies to be performed on this dataset.

Thus, it is predicted that the studies can be used by experts as a tool in the recognition of blood cells. As a result, it has been demonstrated that appropriate deep-learning models can be used for blood cell segmentation and classification, as well as anomaly detection. In addition to the easy classification of blood cells using the model created with deep learning libraries, if a data set with a sufficient number of images is available, the recognition of cells in microscopic images of many blood cells other than eight blood cells can also be achieved. This allows the cells in the blood to be identified in less time. A data set can be created from microscopic images of blood and urine sediment for the detection of specific diseases and applied to deep learning networks for this purpose.

Conflict of Interest Statement

The authors declare that there is no conflict of interest.

References

[1] A. Purcell, *Basic Biology, Blood cells*, Basic Biology Limited, 2018.

[2] R. Soltanzadeh and H. Rabbani, "Classification of three types of red blood cells in peripheral blood smear based on morphology," *IEEE 10th International conference on signal processing proceedings*, Beijing, China, pp. 707-710, 2010. doi:10.1109/ICOSP.2010.5655754

[3] M. D. Joshi A. H. Karode and S. R. Suralkar, "White blood cells segmentation and classification to detect acute leukemia," *in International Journal of Emerging Trends & Technology in Computer Science (IJETTCS),* vol. 2, no. 3, 2013, pp. 147-148.

[4] M. Türkoğlu, "Otomatik Kan Hücrelerinin Tanınması ve Sınıflandırılmasında Değişmez Momentlere Dayalı Görüntü İşleme Yöntemlerinin Kullanılması," MS thesis, Fırat Üniversitesi, Elazığ, 2013.

[5] L. Putzu, G. Caocci and C. D. Ruberto, "Leucocyte classification for leukaemia detection using image processing techniques," *Artificial intelligence in medicine,* vol. 62, no. 3, pp. 179-91, 2014. doi:10.1016/j.artmed.2014.09.002

[6] S. Manik, L. M. Saini and N. Vadera, "Counting and classification of white blood cell using Artificial Neural Network (ANN)," *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, Delhi, India, pp. 1-5, 2016. doi:10.1109/ICPEICES.2016.7853644

[7] A. Karakuzulu, "Kan Hücrelerinin Görüntü İşleme Teknikleriyle Tespiti, Sayılması ve Sınıflandırılması," Master Thesis, Mersin Üniversitesi, Mersin, January 24, 2019.

[8] A. Kh. S. Gihedan, "Görüntü İşleme Yöntemlerini Kullanarak Mikroskobik Görüntülerinden Lösemi hücrelerinin Teşhisi," Ph.D. dissertation, Kastamonu Üniversitesi, Kastamonu, 2019.

[9] F. Ucar, "Deep Learning Approach to Cell Classificatio in Human Peripheral Blood," *2020 5th International Conference on Computer Science and Engineering (UBMK)*, Diyarbakir, Turkey, pp. 383-387, 2020. doi:10.1109/UBMK50275.2020.9219480

[10] E. Gavas and K. Olpadkar, "Deep CNNs for peripheral blood cell classification," *Proceedings of Machine Learning Research*, arXiv preprint arXiv:2110.09508, 2021. doi:10.48550/arXiv.2110.09508

[11] F. Long, J. Peng, W. Song, X. Xia and J. Sang, "BloodCaps: A capsule network based model for the multiclassification of human peripheral blood cells," *Computer methods and programs in biomedicine*, vol. 202, 2021. doi:10.1016/j.cmpb.2021.105972

[12] R. A. Bagido, M. Alzahrani and M. Arif., "White blood cell types classification using deep learning models," *International Journal of Computer Science & Network Security*, vol. 21, no. 9, pp. 223-229, 2021. doi:10.22937/IJCSNS.2021.21.9.30

[13] F. Ş. Rende, G. Bütün and Ş. Karahan, "Derin Öğrenme Algoritmalarında Model Testleri: Derin Testler," *in Bilişim Teknolojileri Enstitüsü*, TÜBİTAK BİLGEM, Kocaeli, Turkey, 55, 2016.

[14] M. Nielsen, *Neural networks and deep learning*, Vol. 25. San Francisco, CA, USA: Determination press, 2015.

[15] W. Abdulla, "Splash of Color: Instance Segmentation with Mask R-CNN and TensorFlow," *engineering.matterport.com*, Matterport Engineering Techblog, Mar 20, 2018. [Online]. Available: https://engineering.matterport.com. [Accessed: 17.02.2023].

[16] K. He, G. Dollar and R. Girshick,, "Mask R-CNN," *in Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2961-2969, 2017. [Online]. Available: ICCV 2017 open access, https://openaccess.thecvf.com/. [Accessed: 17 Feb. 2023].

[17] S. Ren, K. He, R. Girshick and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015. doi:10.48550/arXiv.1506.01497

[18] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 770-778, 2016. doi:10.1109/CVPR.2016.90

[19] A.S. Almryad, "Identification of butterfly species using machine learning and image processing techniques," Ph.D. dissertation, Karabuk University, Karabük, Türkiye, 2020.

[20] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770-778. doi:10.48550/arXiv.1512.03385

[21] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014. doi:10.48550/arXiv.1409.1556

[22] F. Doğan and İ. Türkoğlu, "Derin Öğrenme Modelleri ve Uygulama Alanlarına İlişkin Bir Derleme," *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, vol. 10, no. 2, pp. 409 - 445, 2019. doi:10.24012/dumf.411130

[23] S. Xie, R. Girshick, P. Dollar, Z. Tu and K. He, "Aggregated residual transformations for deep neural networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1492-1500.doi:10.48550/arXiv.1611.05431

[24] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger "Densely connected convolutional networks," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700-4708, 2017. doi:10.48550/arXiv.1608.06993

[25] A. Acevedo, A. Merino, S. Alférez, Á. Molina, L. Boldúa, J. Rodellar, "A dataset of microscopic peripheral blood cell images for development of automatic recognition systems," Data in brief vol. 30 105474. 8 Apr., 2020. doi:10.1016/j.dib.2020.105474

[26] P. Skalski, " Make Sense," 2019. [Online]. Available: https://github.com/SkalskiP/make-sense/. [Accessed: 20.05.2022].

[27] A. Gulli, and S. Pal, *Deep learning with Keras*. Packt Publishing Ltd, 2017.

[28] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu and X. Zheng, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467,* 2016. doi:10.48550/arXiv.1603.04467

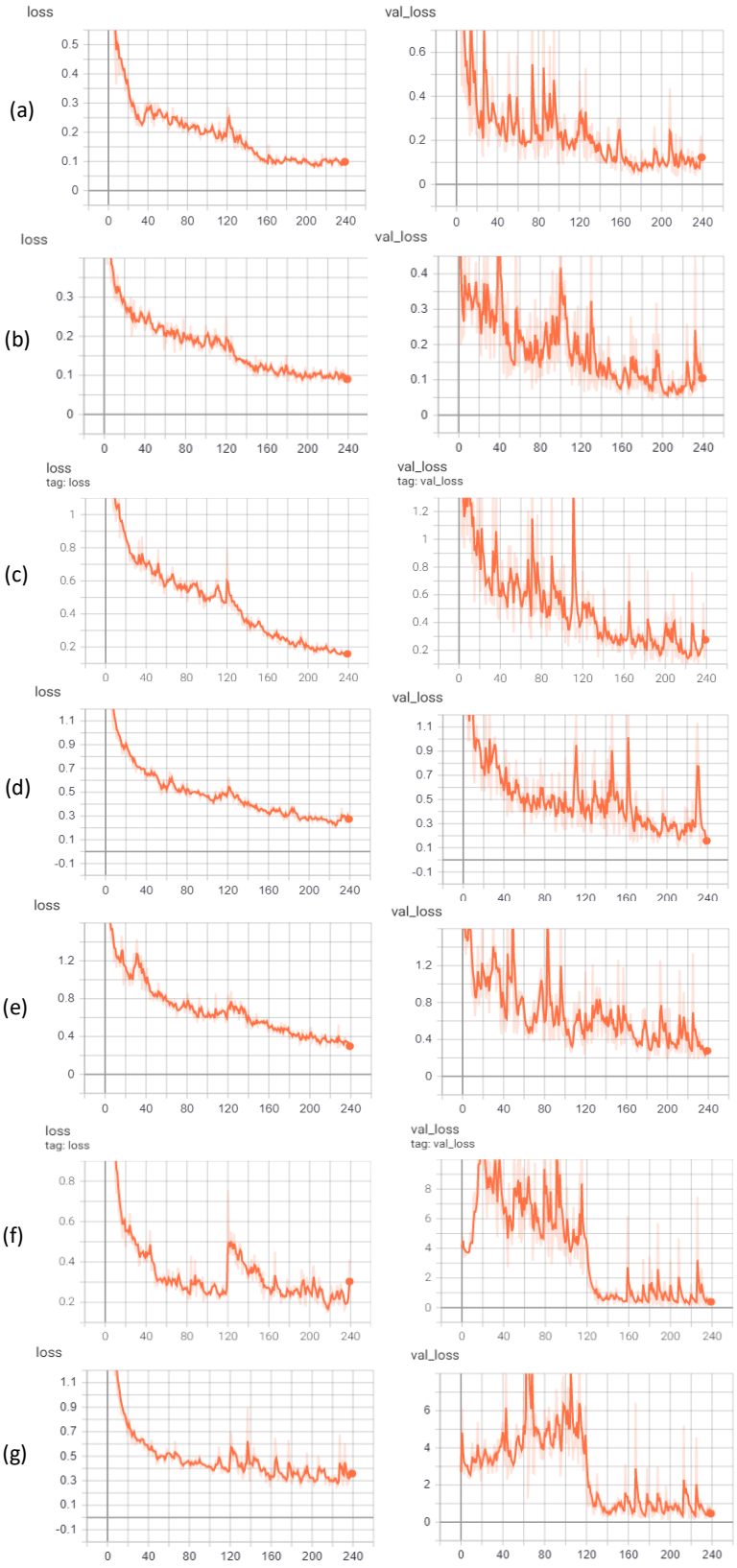
[29] A. Acevedo, S. Alférez, A. Merino, L. Puigví and J. Rodellar, "Recognition of peripheral blood cell images using convolutional neural networks," *Computer methods and programs in biomedicine*, vol. 180, 2019. doi:10.1016/j.cmpb.2019.105020

[30] P. P. Banik, R. Saha and K. -D. Kim, "Fused Convolutional Neural Network for White Blood Cell Image Classification," *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, Okinawa, Japan, 2019, pp. 238-240. doi:10.1109/ICAIIC.2019.8669049

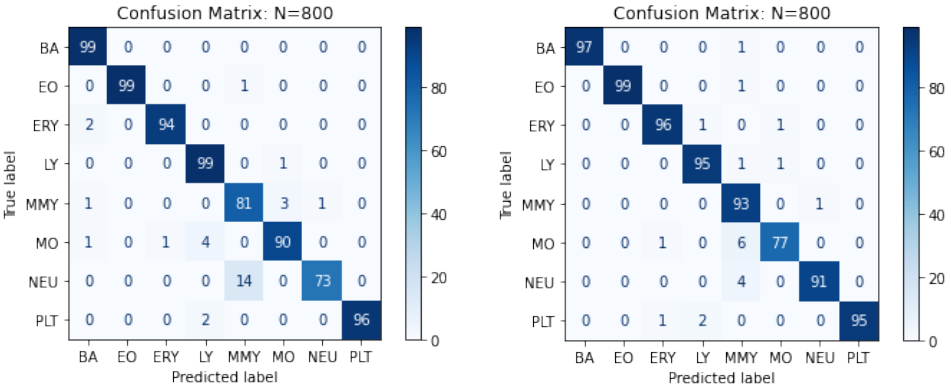
[31] C. D. Ruberto, A. Loddo and Lorenzo Putzu, "Detection of red and white blood cells from microscopic blood images using a region proposal approach." *Computers in biology and medicine*, vol. 116, 2020. doi:10.1016/j.compbiomed.2019.103530

APPENDIX

Appendix 1. Loss and validation loss graphics of ResNet50 (a), ResNet101 (b), ResNet152 (c), ResNeXt50 (d), VGG19 (e), DenseNet121 (f) and DenseNet169 (g)

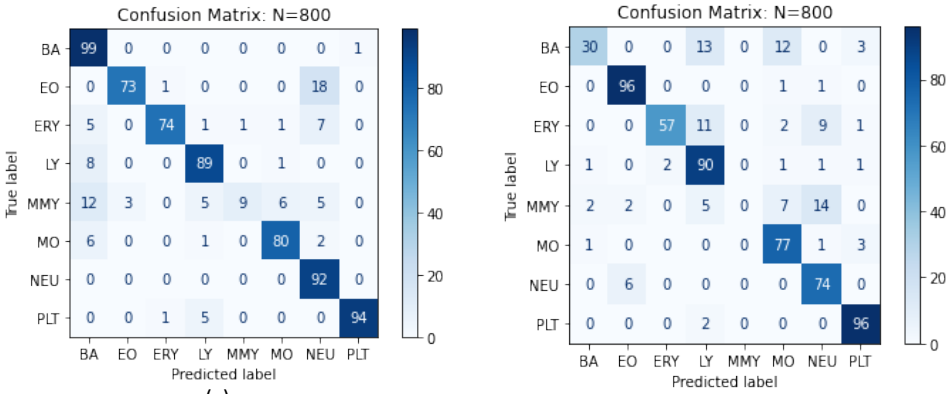


Appendix 2. Confusion Matrix of ResNet50 (a), ResNet101 (b), ResNet152 (c), ResNeXt50 (d), VGG19 (e), DenseNet121 (f) and DenseNet169 (g) Model



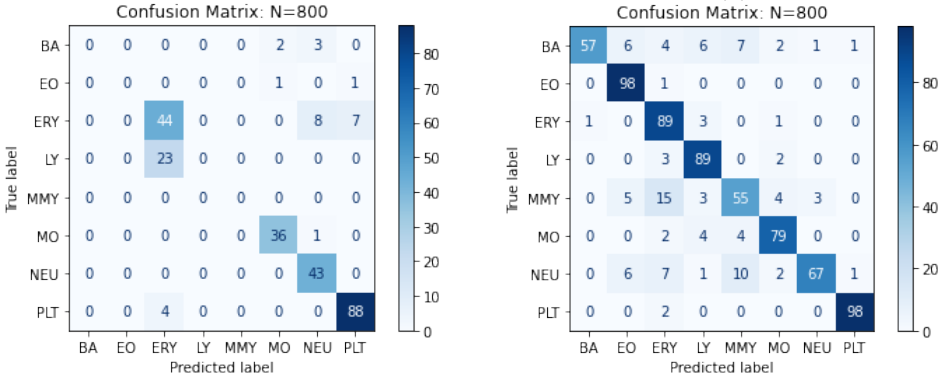
(a)

(b)



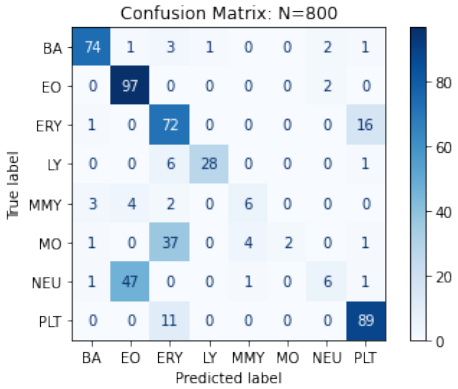
(c)

(d)



(e)

(f)



(g)

Appendix 3. Cell samples detected with (1) ResNet50, (2) ResNet101, (3) ResNet152, (4) ResNeXt50, (5) VGG19, (6) DenseNet121, (7) DenseNet169 models