BCNet: A Novel Network for Blood Cell Classification

Ziquan Zhu¹, Siyuan Lu¹, Shui-Hua Wang¹*, Juan Manuel Górriz²* and Yu-Dong Zhang¹,³*

¹School of Computing and Mathematical Sciences, University of Leicester, Leicester, United Kingdom, ²Department of Signal Theory, Networking and Communications, University of Granada, Granada, Spain, ³Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin, China

Aims: Most blood diseases, such as chronic anemia, leukemia (commonly known as blood cancer), and hematopoietic dysfunction, are caused by environmental pollution, substandard decoration materials, radiation exposure, and long-term use certain drugs. Thus, it is imperative to classify the blood cell images. Most cell classification is based on the manual feature, machine learning classifier or the deep convolution network neural model. However, manual feature extraction is a very tedious process, and the results are usually unsatisfactory. On the other hand, the deep convolution neural network is usually composed of massive layers, and each layer has many parameters. Therefore, each deep convolution neural network needs a lot of time to get the results. Another problem is that medical data sets are relatively small, which may lead to overfitting problems.

Methods: To address these problems, we propose seven models for the automatic classification of blood cells: BCARENet, BCR5RENet, BCMV2RENet, BCRRNet, BCRENet, BCRSNet, and BCNet. The BCNet model is the best model among the seven proposed models. The backbone model in our method is selected as the ResNet-18, which is pre-trained on the ImageNet set. To improve the performance of the proposed model, we replace the last four layers of the trained transferred ResNet-18 model with the three randomized neural networks (RNNs), which are RVFL, ELM, and SNN. The final outputs of our BCNet are generated by the ensemble of the predictions from the three randomized neural networks by the majority voting. We use four multi-classification indexes for the evaluation of our model.

Results: The accuracy, average precision, average F1-score, and average recall are 96.78, 97.07, 96.78, and 96.77%, respectively.

Conclusion: We offer the comparison of our model with state-of-the-art methods. The results of the proposed BCNet model are much better than other state-of-the-art methods.

Keywords: blood cells, convolutional neural network, randomized neural network, ResNet-18, transfer learning

INTRODUCTION

Blood cells can spread throughout the body by the blood. There are three types of blood cells in mammals: 1) Red blood cells: transporting oxygen is the main function. 2) Leukocytes: mainly play the role of immunity when the body is invaded by bacteria. At that time, leukocytes can focus on the invasion site of bacteria, surround the bacteria, and swallow them. 3) Platelets: which are very important in hemostasis.
Most blood diseases, such as chronic anemia, leukemia (commonly known as blood cancer), and hematopoietic dysfunction, are caused by environmental pollution, substandard decoration materials, radiation exposure, and long-term use of certain drugs. They are all insidious, and the early symptoms are very mild and easy to ignore. Therefore, it is necessary to detect the number of various blood cells regularly. After increasing menstruation and skin purpura, detecting blood cells in time is necessary. By detecting the various blood cells in the blood, it can help doctors diagnose these diseases. Thus, it is vital to classify the blood cell images. In this paper, we automatically classify three cell types that affect neonatal blood cells. This blood cell image data are from the blood cell data set published on the Kaggle website (Mooney, 2017). However, Some diseases and their complications have no significant effect on neutrophil dynamics (Manroe et al., 1979). Therefore, it will not be tested in this paper. In a blood examination of the patient, medical professionals create a slide coated with blood, fix the slide, stain with chemical reagents such as Wright Gimsa and hematoxylin-eosin, and then carefully observe the blood cell changes (Ryu et al., 2020). It takes a long time for doctors to complete a blood cell test. However, the test results are also easily affected by the dyeing quality.

Researchers and practitioners try to classify blood cells based on computer technology. Salau and Jain (2021) proposed a method to predict Akt protein cells based on ML technology of multilayer perceptron (MLP) and radial basis function (RBF). Finally, the accuracy of this paper was 99.93%. Dudaie et al. (2020) put forward a model, based on a digital holographic microscope and machine learning, to detect and classify untouched cancer cells. In the experiment, the throughput was 15 cells per second. According to the experimental process’s cell flow morphology and quantitative phase characteristics, the accuracy can reach 92.56%. Marostica et al. (2021) suggested using a deep convolutional neural network for the detection and diagnosis of renal cancer. This method linked the quantitative pathological model with the patient’s genome map and prognosis. Finally, the AUC value of malignant tumors in the detection and validation cohort was 0.964–0.985. Wagner and Yanai (2018) developed a hierarchical machine learning framework called Moana. Moreover, the framework could construct a classifier in heterogeneous scRNA-Seq data sets. Begambre et al. (2021) proposed a low-cost classification approach by using artificial intelligence and computer vision to detect leukocytes. The final accuracy was 96.4%. Liang et al. (2018) introduced a recurrent neural network framework (CNN − RNN) that combined CNN and RNN. The framework can help to understand the image content and learn its characteristics. They tested and compared with other CNN models, and finally, they concluded that their proposed network model can better complete blood cell classification. Bur et al. (2019) presented an automatical system for detecting occult lymph node metastasis in clinical lymph node-negative oral squamous cell carcinoma (OCSCC). They finally got an AUC value of 0.840. Kocak et al. (2018) proposed to use texture analysis for the classification of renal cell computed tomography (CT) texture analysis. Zhang et al. (2017) employed a CNN to classify cervical cells. Finally, the accuracy of the method is 98.3%, and the area under the curve is 0.99. Şengür et al. (2019) presented a system that was a combination of machine learning (ML) with graphics processing (IP) to measure and classify leukocytes. The accuracy of the depth feature was 82.9%, the shape feature was 80.0%, and the accuracy of combining the two was 85.7%. Özel Duygan et al. (2020) expanded and accelerated microbiota analysis by a supervised algorithm. Imran Razak and Naz (2017) proposed an effective contour-aware segmentation method. The method was based on a fully traditional network structure. In the classification process, they used extreme machine learning to extract CNN features from each unit. Habibzadeh et al. (2018) proposed a preprocessing algorithm for color distortion, bounding box distortion, and image flip mirror. Then, they used Inception and ResNet architecture to extract and recognize the characteristics of leukocytes. They obtained an accuracy of 98.33% and an AUC value of 0.9833. Lei et al. (2018) presented a classification framework for cell images to handle the challenges of intragroup changes caused by uneven illumination. The framework was based on the deeply supervised residual network. Kim et al. (2018) presented a CNN based on machine learning to automatically classify the morphology of red blood cells in blood flow. Khamparia et al. (2020) presented an artificial intelligence system that was driven by the internet of healthy things for the detection of cervical cancer. Experiments show that the ResNet50 training model got the highest accuracy of 97.89%. Varghese (2020) classified four types of blood cells by using machine learning. Kan (2017) modeled the segmentation and tracking of individual cells and the reconstruction of phylogenetic trees using ML in optical microscope experimental image analysis. Wedin and Bengtsson (2021) used three classifiers to classify the cell types of mouse digital reconstruction images. The three classifiers were CNN, random forest classifier, and support vector classifier. Feng et al. (2018) proposed a method combining supervised machine learning and diffraction images to detect and classify different stages of apoptosis. Finally, the accuracy of this method was more than 90%. Su et al. (2020) suggested generating lighting patterns in the cells by single-mode fiber cytometry. Iliyasu and Fatichah (2017) proposed a new method (Qfuzzy) to extract and classify cervical smear cells’ characteristics based on particle swarm optimization and k-nearest neighbors. Alom et al. (2018) presented a new system to classify and detect colon cancer. This method combined the densely connected convolutional network (DCRN) and the recurrent residual u Network (R2U-Net).

It can be concluded from the above latest research analysis that most of the cell classification is based on the manual feature, machine learning classifier or the deep convolution neural network model (Jiao et al., 2019a). However, manual feature extraction is a very tedious process, and the results are usually unsatisfactory. Manually labeling the complete set to only consider the real information in that image is an unaffordable task. On the other hand, the deep convolution neural network is usually composed of massive layers, and each layer has many parameters (Jiao et al., 2019b). Therefore, each deep convolution neural network needs a lot of time to get the results. Another problem is that medical data sets are relatively small, which may lead to overfitting problems.
To deal with these problems described above, we propose seven models for the automatic classification of blood cells. The contributions of this paper are summarized as below:

- Seven models are proposed to automatically classify blood cells: BCARENet, BCR5RENet, BCMV2RENet, BCRRNet, BCRENet, BCRSNet, and BCNet.
- The BCNet model is the best model among the four proposed ensemble models after statistical experiments.
- After comparison, the proposed BCNet is better than the other three individual models.
- Three RNNs are selected to substitute the last four layers of the trained transferred ResNet-18 to shorten training time.
- The outputs of the BCNet are generated based on predictions from three RNNs using the majority voting.

In the proposed BCNet model, there are three randomized neural networks (RNNs), and all of them are feedforward neural networks with a single hidden layer. They are random vector functional link (RVFL), Schmidt neural network (SNN), and extreme learning machine (ELM). The number of parameters required in randomized neural networks (RNNs) is far less than that required by the deep convolution neural network model. Therefore, the proposed BCNet structure can avoid the problems of overfitting and greatly shorten training time and cycle.

The framework of this paper is as follows. Material section introduces the used materials in this paper. Methodology section presents the details and explanation of the proposed BCNet. The experiment settings and results of the BCNet are included in Experiment Settings and Results section. Conclusion section is about the conclusion.

MATERIAL

This blood cell image data can be downloaded on the Kaggle website (Mooney, 2017). However, Some diseases and their complications have no significant effect on neutrophil dynamics (Manroe et al., 1979). And therefore, it will not be tested in this paper. We selected three cell types for training and testing. The three cell types are eosinophils, lymphocytes, and monocytes, as shown in Supplementary Figure S1A.

Eosinophils are a component of leukocytes. It is a hematopoietic stem cell-derived from bone marrow. Parasites and bacteria are eliminated by the Eosinophils. The lymphocyte is the smallest white blood cell which is produced by the Lymphatic organs. It is one of the most important components of the body’s immune. Monocytes are the largest blood cells in the blood, which are the irreplaceable component of the human defense system.

There are about 3,000 images Table 1 for each of three different cell types: Eosinophil, Lymphocyte, and Monocyte, respectively. The information of the data set is given in Supplementary Figure S1A. For training our models in this study, we encode the labels as: [100]T, [010]T, and [001]T, which represents Eosinophil, Lymphocyte, and Monocyte, respectively.

METHODOLOGY

At present, many diagnostic systems based on artificial intelligence are used to analyze and classify images (Sheng et al., 2021; Jiao et al., 2021a). It is inevitable to extract the features of images for analyzing and classifying images (Jiao et al., 2021b). However, because each image contains a lot of information, extracting the discrimination rate features is difficult. With the continuous progress of computer vision technology and machine learning, many models have achieved great success, such as convolution neural networks (CNN) (Jiao et al., 2020a). The convolution layer in the CNN model can significantly reduce the parameters to reduce the computation, as shown in Figure 1A. The pooling layer further reduces the dimension of features while maintaining the dominant information (Jiao et al., 2018), as shown in Figure 1B.

Proposed BCNet

With more and more research on CNN, there are many CNN models (Jiao et al., 2019c), such as DenseNet (Huang et al., 2017), VGG (Simonyan and Zisserman, 2014), MobileNet (Sandler et al., 2018), EfficientNet (Tan and Eficientnet, 2019), AlexNet (Krizhevsky et al., 2017), ResNet (He et al., 2016), SqueezeNet (Iandola et al., 2016), and so on. This paper proposes seven models for the automatic classification of blood cells: BCARENet, BCR5RENet, BCMV2RENet, BCRRNet, BCRENet, BCRSNet, and BCNet. The BCNet model is the best model among the seven proposed models. The flowchart of our model is given in Figure 2A. The pseudocode of our model is shown in Table 2. The backbone model in our method is selected as the ResNet-18, which is pre-trained on the ImageNet set. We transfer the ResNet-18 model. After that, the transferred ResNet-18 is trained on the processed training set. Compared with ResNet-18, the training time of randomized neural networks is much shorter. To improve the performance of the proposed model, the end four layers of the trained transferred ResNet-18 model are replaced by three randomized neural networks (RNNs) which are RVFL, ELM, and SNN. The features $F$ extracted from FC256 layer are used to train RVFL, ELM, and SNN. Because the randomized neural network parameters from the input layer to the hidden layer are random, we use the ensemble of the predictions of the three RNNs to improve the robustness of the network. The final outputs of our BCNet are generated by the ensemble of the predictions from the three randomized neural networks by the majority voting. To better verify the performance of our proposed BCNet model, we use multi-classification indexes to evaluate it.

Backbone of Proposed BCNet

The activation function (Jiao et al., 2020b) is added to activate some neurons in the CNN model, and the activated neurons are transmitted to the next layer, as shown in Figure 1C. If the activation function is not added, the neurons of each layer of CNN are linear.

However, when the neurons of each layer are nonlinear, it is challenging to implement identity mapping (He et al., 2016) in the training iteration. If the number of layers is deepened for a trained network structure, it is not simply stacking more layers
but stacking one layer to make the output after stacking the same as before stacking and then continuing training. In this case, it is reasonable that the training results should not be worse because the level before adding layers has been taken as the initial before the training starts. However, the experimental results show that the results will be worse after the network layers reach a certain depth, which is the problem of degradation. This shows that the traditional nonlinear expression of multilayer network structure is difficult to represent identity mapping, as shown in Figure 2B. In this paper, we use the residual mechanism to deal with this problem. A stacked-layer structure is shown in Figure 2C.

When the input is $X$, the original learned feature is recorded as $T(X)$, and $L(X)$ is obtained through the residual network formula, which is as follows:

$$L(X) = T(X) - X$$  \hspace{1cm} (1)

Through the above formula conversion, the original learned feature is:

$$T(X) = L(X) + X$$  \hspace{1cm} (2)

Compared with direct learning, residual learning is a better method for original features. Because when the residual is 0, the residual learning can at least carry out the identity mapping. Thus, it would not have any bad influences on the system performance. When the residual is not 0, it can learn new features from other layers.

The backbone model in our method is selected as the ResNet-18, which is pre-trained on the ImageNet set. We transfer the ResNet-18 model. The transfer learning in the ResNet-18 is shown in Figure 2D. We replace FC1000 with FC3 because there are three types of images of blood cells in this paper and add FC256 to reduce dimensional differences. In addition, we delete the last four layers of the trained transferred ResNet-18 model and add three RNNs. Therefore, the trained transferred ResNet-18 model is the proposed BCNet feature extractor in this paper. FC256 is the feature layer.

**RNNs Ensemble in BCNet**

With the continuous research of the CNN models, the CNN models are becoming more and more excellent (Ji et al., 2021). Especially for training and testing on large data sets, the CNN models achieve better and better results. In the proposed BCNet model, the end four layers of the trained transferred ResNet-18 are replaced by three randomized neural networks (RNNs). The three RNNs are RVFL (Pao et al., 1994), ELM (Huang et al., 2006), and SNN (Schmidt et al., 1992).

As shown in Figure 3, these three diagrams are 1) RVFL, 2) ELM, and 3) SNN, respectively. The blue box represents the input, the hidden nodes in the hidden layer are shown by the orange circle, and the pink box is the output. It can be seen from Figure 3 that the main difference between SNN and ELM is that there are biases in the SNN. The main difference between RVFL and the other two RNNs (SNN and ELM) is that they can be connected directly from the input layer to the output layer in RVFL.

Although the structures of the three RNNs are more and less different, the calculation steps of the three RNNs are similar. First, for $N$ arbitrary distinct samples, set a data set with the $i$-th sample as $(x_i, y_i)$:

$$x_i = (x_{i1}, \ldots, x_{in})^T \in \mathbb{R}^n, \quad i = 1, \ldots, N, \hspace{1cm} (3)$$

![Figure 1](image1.png)  | Explanations of CNN. (A) Convolution layer flow chart. (B) An example of average and max pooling. (C) Activation function.
\[ y_i = (Q_{i1}, \ldots, Q_{im})^T \in \mathbb{R}^m, \quad i = 1, \ldots, N, \]  

(4)

where \( n \) represents the input dimension, \( m \) represents the output dimension.

The calculation of three RNNs: \( A_j \) is the weight vector connecting the \( j \)-th hidden node and the input nodes, \( K_j \) is the bias of the \( j \)-th hidden node. Thus, the output matrix of the hidden layer containing \( Z \) hidden nodes can be calculated:

\[ U_{RVFL(i)} = \text{concat}(X, V), \]  

(5)

where \( X = (x_1, \ldots, x_N)^T \) is the input characteristic matrix. \( V \) is the random hidden mappings. The formula is as follows:

\[ V_{RVFL(i)} = \sum_{j=1}^{Z} g(A_j x_i + K_j), \quad i = 1, \ldots, N, \]  

(6)

where \( g() \) represents the sigmoid function.

For ELM, the equation is as follows:

\[ U_{ELM(i)} = \sum_{j=1}^{Z} g(A_j x_i + K_j), \quad i = 1, \ldots, N. \]  

(7)

For SNN, the formula is as follows:

\[ U_{SNN(i)} = \sum_{j=1}^{Z} g(A_j x_i + K_j), \quad i = 1, \ldots, N. \]  

(8)

The final output weights \( (W) \) are obtained by pseudo-inverse:

\[ W = U_{\text{net}}^+ Y, \quad \text{net} = \text{ELM or RVFL}, \]  

(9)

where \( U_{\text{net}}^+ \) denotes the pseudo-inverse matrix of \( U_{\text{net}} \) and \( Y = (y_1, \ldots, y_N)^T \) is the ground-truth label matrix of the dataset.
TABLE 2 | Pseudocode of the proposed BCNet.

Step 1: Load the pre-trained ResNet-18.
Step 2: Divide the blood cell data set into training and testing sets.
Step 3: Preprocessing
   Resize samples in the training and testing set based on the input size of ResNet-18.
Step 4: Generate the transferred ResNet-18.
   Step 4.1: Remove FC1000, softmax, and classification layer from the pre-trained ResNet-18.
   Step 4.2: Add FC256, ReLU, FC3, softmax, and classification layer.
Step 5: Train the transferred ResNet-18.
   Step 5.1: Input is the processed training set.
   Step 5.2: Target is the corresponding labels.
Step 6: Replace the last 4 layers of the trained transferred ResNet-18 with three neural networks by the majority voting. Suppose model are generated by the ensemble of the predictions from the three RNNs: RVFL, ELM, and SNN, respectively, and three models are obtained: BCRRNNet, BCRENNet, and BCRRSNet. The details of the proposed three individual models are given in Table 3.
Step 7: Extract features \( F \) as the output of the FC256 layer.
Step 8: Train the three RNNs on the extracted features \( F \) and the labels.
   Step 8.1: Input is the extracted features \( F \).
   Step 8.2: Target is the labels of the processed training set.
Step 9: Add the majority voting layer.
   Step 9.1: Ensemble the predictions of the three RNNs.
   Step 9.2: Majority voting of the ensemble of the predictions from the three RNNs.
Step 10: Test the trained BCNet on the processed testing set.
   Step 10.1: Input is the extracted features \( F \).
   Step 10.2: Target is the labels of the processed testing set.
Step 11: Report the classification performance of the trained BCNet.

Because SNN adds biases \( E \) on the output layer, its formula is:

\[
(W, E) = U_{\text{net}}^+ Q,
\]

where \( U_{\text{net}}^+ \) denotes the pseudo-inverse matrix of \( \begin{pmatrix} U_{\text{net}} \\ 1 \end{pmatrix} \).

For improving the robustness of BCNet, the final outputs of our model are generated by the ensemble of the predictions from the three randomized neural networks by the majority voting. Suppose given an image \( s_{\text{sam}} \) and \( L(s_{\text{sam}}) \) is the function of the final output, \( Q_{\alpha}, Q_{\beta}, \) and \( Q_{\gamma} \) mean three predictions from three RNNs for image \( s_{\text{sam}} \), respectively.

\[
L(s_{\text{sam}}) = \begin{cases} Q_{\alpha}, & \text{if } \exists Q_i = Q_{\alpha}, \quad t, b \in \{\alpha, \beta, \gamma\} \\ 100^T, & \text{otherwise} \end{cases}
\]

where \([100]^T\) denotes the Eosinophil.

Other Proposed Models

Compared with ResNet-18, the training time of randomized neural networks is much shorter. We replace the end four layers of the trained transferred ResNet-18 model with three RNNs: RVFL, ELM, and SNN, respectively, and three models are obtained: BCRRNNet, BCRENNet, and BCRRSNet. The details of the proposed three individual models are given in Table 3.

Because the randomized neural network parameters from the input layer to the hidden layer are random, we use the ensemble of the outputs of the three RNNs to improve the robustness of the network. This paper’s other three proposed ensemble models are BCARENet, BCRR5ENet, and BCMV2RENet, respectively, as shown in Table 3. These three proposed ensemble models select the pre-trained AlexNet, the pre-trained ResNet-50, and the pre-trained MobileNet-V2 as their backbones. Three backbones are trained as the operations to get the trained transferred ResNet-18.

Evaluation

We use multi-classification indexes to evaluate the proposed BCNet. In this paper, there are three categories \((c = 1, \ldots, 3)\). When the label of one category is set to positive, the labels of the other two categories are set to negative. When the label of the B category is positive, the definitions of true positive (TP), true negative (TN), false negative (FN), and false positive (FP) are shown in Supplementary Figure S1B.

Four multi-classification indexes are applied to this paper. They are accuracy, average recall, average F1-score, and average precision. The formulas of the three indexes (precision, recall, and F1) are as follows:

\[
\begin{align*}
\text{precision} (c) &= \frac{\text{TP} (c)}{\text{TP} (c) + \text{FP} (c)}, \\
\text{recall} (c) &= \frac{\text{TP} (c)}{\text{TP} (c) + \text{FN} (c)}, \\
F1 (c) &= \frac{2 \times \text{precision} (c) \times \text{recall} (c)}{\text{precision} (c) + \text{recall} (c)}, \quad c = 1, \ldots, 3.
\end{align*}
\]

In this paper, we use the macro average for multi-classification indexes calculation. The formulas for calculating the three multi-

![FIGURE 3](Image 123x101 to 227x211)

FIGURE 3 | Structure of three RNNs. (A) RVFL. (B) ELM. (C) SNN.
The accuracy in multi-classification is the proportion of correctly classified samples in total samples.

### EXPERIMENT SETTINGS AND RESULTS

#### Experiment Settings

We adjust the parameters of the proposed BCNet model in this paper. Our mini-batch size for each time is 50. To avoid the overfitting problems, we set the max-epoch to 2. Based on experience, we set the learning rate to 1e-4. We set a super parameter in the three RNNs. The number of hidden nodes Z is 400, which is determined based on the input dimension of RNNs. The hyper-parameter settings of BCNet are demonstrated in Supplementary Figure S1C.

#### The Performance of BCNet

The blood cell data set has been divided into the experiment’s training and testing set. Table 4 shows the test confusion matrix.

The calculation principle of the four multi-classification evaluation indexes in this paper is shown in Three Proposed Ensemble Models section. There are three categories in this paper, and the results of each category are shown in Table 5.

The specific calculations of four multi-classification indexes are:

\[
\text{accuracy} = \frac{(623 + 620 + 560)}{(623 + 620 + 560 + 60)} = 96.78% \\
\text{average - precision} = \frac{91.22\% + 100\% + 100\%}{3} = 97.07% \\
\text{average - recall} = \frac{100\% + 100\% + 90.32\%}{3} = 96.77% \\
\text{average - F1} = \frac{95.41\% + 100\% + 94.92\%}{3} = 96.78% 
\]

### Table 4 | The test confusion matrix of BCNet.

| Actual class | Eosinophil | Lymphocyte | Monocyte |
|--------------|------------|------------|----------|
| Eosinophil   | 623        | 0          | 0        |
| Lymphocyte   | 0          | 620        | 0        |
| Monocyte     | 60         | 0          | 560      |

The comparison of the proposed BCNet with other proposed models is shown in Table 6. The comparison of the proposed BCNet with the other proposed models is shown in Table 7. For a more intuitive view, the comparison of the proposed BCNet with the other proposed
models is shown in Figure 4. As shown in the table and figure, we can conclude that the proposed BCNet has better performance than the other proposed models.

**Explainability of the Proposed BCNet**

In this section, we explain the proposed BCNet model. Usually, it’s hard to understand how deep models make predictions. However, with the help of Gradient-weighted class activation mapping (Grad-CAM) (Selvaraju et al., 2017), we can observe where the deep model pays attention. The Gradient-weighted class activation mapping is shown in Figure 5. In the raw image, eosinophilic nuclei are mostly C-type, S-type, or irregular type. Lymphocyte nuclei are quasi round or round, often on one side. The nuclei of monocytes are irregular, distorted, and overlapped. As can be seen from the heatmap, there are probably three red, blue, and orange colors on the Grad-CAM figure. The red area represents the place with the closest attention, the orange area is close attention, and the blue area has the lowest attention.

**Comparison With Other State-Of-The-Art Methods**

To better show the superiority of the proposed BCNet in this paper, we compare it with other state-of-the-art methods. These state-of-the-art methods are CNN+RNN (Liang et al., 2018), ML (Kihm et al., 2018), Q-fuzzy (Iliyasu and Fatichah, 2019).

---

**TABLE 5 | The results of each category.**

| Category     | Precision (%) | Recall (%) | F1 (%)  |
|--------------|---------------|-----------|---------|
| Eosinophil   | 91.22         | 100       | 95.41   |
| Lymphocyte   | 100           | 100       | 100     |
| Monocyte     | 100           | 90.32     | 94.92   |

---

**TABLE 6 | Confusion matrices of other proposed models.**

| Predicted       | Eosinophil | Lymphocyte | Monocyte |
|-----------------|------------|------------|----------|
| **BCRRNet (Individual)** |            |            |          |
| Actual Eosinophil | 621        | 2          | 0        |
|                 | Lymphocyte | 0          | 619      | 1        |
|                 | Monocyte   | 72         | 0        | 548      |

| **BCRENet (Individual)** |            |            |          |
| Actual Eosinophil | 623        | 0          | 0        |
| Lymphocyte        | 0          | 619        | 1        |
| Monocyte          | 68         | 0          | 552      |

| **BCRSNet (Individual)** |            |            |          |
| Actual Eosinophil | 623        | 0          | 0        |
| Lymphocyte        | 0          | 619        | 1        |
| Monocyte          | 65         | 0          | 555      |

| **BCARENet (Ensemble)** |            |            |          |
| Actual Eosinophil | 432        | 62         | 129      |
| Lymphocyte        | 7          | 600        | 13       |
| Monocyte          | 176        | 0          | 444      |

| **BCRSRENet (Ensemble)** |            |            |          |
| Actual Eosinophil | 550        | 73         | 0        |
| Lymphocyte        | 0          | 620        | 0        |
| Monocyte          | 90         | 0          | 530      |

| **BCMV2RENet (Ensemble)** |            |            |          |
| Actual Eosinophil | 582        | 35         | 6        |
| Lymphocyte        | 8          | 601        | 11       |
| Monocyte          | 153        | 7          | 460      |
The comparison results are shown in Table 8. For a more intuitive view, the comparison chart is shown in Figure 6. It is not difficult to conclude from the figure and table that the experimental results of the proposed BCNet model are far better than these of other state-of-the-art methods.

CONCLUSION

The paper proposes seven models for the automatic classification of blood cells: BCARENet, BCR5RENet, BCMV2RENet, BCRNet, BCRENet, BCRSNet, and BCNet. The BCNet model is the best model among the seven proposed models. The backbone model in our method is selected as the ResNet-18, which is pre-trained on the ImageNet set. To improve the performance of the proposed model, we replace the last four layers of the trained transferred ResNet-18 model with the three RNNs: RVFL, ELM, and SNN. The final outputs of our BCNet are generated by the ensemble of the predictions from the three randomized neural networks by the majority voting. We use four multi-classification indexes for the evaluation of our model. The accuracy, average precision, average F1-
score, and average recall are 96.78, 97.07, 96.78, and 96.77%, respectively. We offer the comparison of our model with state-of-the-art methods. The results of the proposed BCNet model are much better than other state-of-the-art methods.

Although the proposed BCNet model in this paper has achieved excellent outputs, there are still some deficiencies. This paper mainly tests single cells but does not include overlapping cells and cell clusters tests. We do not detect all types of images in this data set. This paper uses only three types of this data set.

In future research, we will do more research about single cells, overlapping cells, and cell clusters classification and more methods to classify blood cells, such as CAD systems. At the same time, we will also use this method on other data sets.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://www.kaggle.com/paultimothymooney/blood-cells.

REFERENCES

Alom, M. Z., Yakopcic, C., Taha, T. M., and Asari, V. K. (2018). Microscopic Nuclei Classification, Segmentation and Detection with Improved Deep Convolutional Neural Network (DCNN) Approaches. arXiv preprint arXiv:1811.03447.

Begambre, S., Castillo, C., Villamizar, L. H., and Aceros, J. (2021). "Low Cost Classification Method for Differentiated White Blood Cells Using Digital Image Processing and Machine Learning Algorithms," in 2021 IEEE Colombian Conference on Applications of Computational Intelligence (ColCACI) (Cali, Colombia: IEEE), 1–5. doi:10.1109/colcacii.2021.9469040

Bur, A. M., Holcomb, A., Goodwin, S., Woodroof, J., Karadaghy, O., Shnayder, Y., et al. (2019). Machine Learning to Predict Occult Nodal Metastasis in Early Oral Squamous Cell Carcinoma. Oral Oncol. 92, 20–25. doi:10.1016/j.oraloncology.2019.03.011

Dudaie, M., Nissim, N., Barnea, I., Gerling, T., Duschl, C., Kirschbaum, M., et al. (2020). Label-free Discrimination and Selection of Cancer Cells from Blood during Flow Using Holography-Induced Dielectrophoresis. J. Biophotonics 13 (11), e202000151. doi:10.1002/jbio.202000151

Feng, J., Feng, T., Yang, C., Wang, W., Su, Y., and Feng, Y. (2018). Feasibility Study of Stain-free Classification of Cell Apoptosis Based on Diffraction Imaging Flow Cytometry and Supervised Machine Learning Techniques. Apoptosis 23 (5-6), 290–298. doi:10.1007/s10495-018-1454-y

Habibzadeh, M., Jannesari, M., Rezaei, Z., Baharvand, H., and Totonchi, M. (2018). "Automatic white Blood Cell Classification Using Pre-trained Deep Learning Models: ResNet and Inception," in Tenth international conference on machine learning with applications.

AUTHOR CONTRIBUTIONS

ZZ: Conceptualization, Software, Data Curation, Writing - Original Draft, Writing - Review & Editing. SL: Conceptualization, Methodology, Software, Data Curation, Writing - Original Draft, Visualization. S-HW: Software, Validation, Investigation, Data Curation, Writing - Review & Editing. JG: Validation, Formal analysis, Investigation, Resources, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision, Project administration, Funding acquisition.

FUNDING

The paper was partially supported by: Royal Society International Exchanges Cost Share Award, United Kingdom (RP202G0230); Medical Research Council Confidence in Concept Award, United Kingdom (MC_PC_17171); Hope Foundation for Cancer Research, United Kingdom (RM60G0680); British Heart Foundation Accelerator Award, United Kingdom (AA/18/3/34220); Sino-United Kingdom Industrial Fund, United Kingdom (RP202G0289); Global Challenges Research Fund (GCRF), United Kingdom (P202PF11); Guangxi Key Laboratory of Trusted Software (kx201901).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fcell.2021.813996/full#supplementary-material

TABLE 8 | Comparison with other state-of-the-art methods.

| Method              | Accuracy | Average-precision | Average-recall | Average-F1 | Source  | Category |
|---------------------|----------|-------------------|----------------|-------------|---------|----------|
| CNN+RNN Liang et al. (2018) | 90.79%   | —                 | 86.25%         | 86.22%      | Public  | Four     |
| ML Kihm et al. (2018) | 86.70%   | 86.19%            | 85%            | —           | Private | Three    |
| Q-fuzzy Iliyasu and Fatichah (2017) | —     | 85%               | —              | —           | Public  | Seven    |
| DCRN+R2U Alom et al. (2018) | 91.14%   | 81.80%            | —              | 96.78%      | Public  | Four     |
| BCNet (Ours)        | 96.78%   | 97.07%            | 96.77%         | 96.78%      | Public  | Three    |

FIGURE 6 | Comparison with other state-of-the-art methods.
Tan, M., and Efficientnet, Q. Le. (2019). "Rethinking Model Scaling for Convolutional Neural Networks," in International Conference on Machine Learning (Long Beach, USA: PMLR), 6105–6114.

Varghese, N. (2020). Machine Learning Techniques for the Classification of Blood Cells and Prediction of Diseases. Int. J. Comput. Sci. Eng. 9 (1), 66–75.

Wagner, F., and Yanai, I. (2018). Moana: a Robust and Scalable Cell Type Classification Framework for Single-Cell RNA-Seq Data. BioRxiv 2018, 456129.

Wedin, M., and Bengtsson, I. (2021). A Comparative Study on Machine Learning Models for Automatic Classification of Cell Types from Digitally Reconstructed Neurons. Stockholm, Sweden: KTH Royal Institute of Technology, 31.

Zhang, L., Le Lu, L., Nogues, I., Summers, R. M., Liu, S., and Yao, J. (2017). DeepPap: Deep Convolutional Networks for Cervical Cell Classification. IEEE J. Biomed. Health Inform. 21 (6), 1633–1643. doi:10.1109/jbhi.2017.2705583

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Zhu, Lu, Górriz and Zhang. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.