IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON TRENDS, PERSPECTIVES AND PROSPECTS OF MACHINE LEARNING APPLIED TO BIOMEDICAL SYSTEMS IN INTERNET OF MEDICAL THINGS

Received May 8, 2018, accepted June 8, 2018, date of publication July 3, 2018, date of current version July 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2846685

Combining Convolutional Neural Network With Recursive Neural Network for Blood Cell Image Classification

GAOBO LIANG, HUICHAO HONG, WEIFANG XIE, AND LIXIN ZHENG

Fujian Provincial Academic Engineering Research Centre in Industrial Intellectual Techniques and Systems, College of Engineering, Huaqiao University, Quanzhou 362000, China

Corresponding author: Lixin Zheng (zlx@hqu.edu.cn)

This work was supported in part by the Science and Technology Bureau of Xiamen under Grant 3502Z20173045, in part by the Technology Bureau of Quanzhou under Grant 2017G036, and in part by the Subsidized Project for Postgraduates' Innovative Fund in Scientific Research of Huaqiao University.

ABSTRACT The diagnosis of blood-related diseases involves the identification and characterization of a patient's blood sample. As such, automated methods for detecting and classifying the types of blood cells have important medical applications in this field. Although deep convolutional neural network (CNN) and the traditional machine learning methods have shown good results in the classification of blood cell images, they are unable to fully exploit the long-term dependence relationship between certain key features of images and image labels. To resolve this problem, we have introduced the recurrent neural networks (RNNs). Specifically, we combined the CNN and RNN in order to propose the CNN–RNN framework that can deepen the understanding of image content and learn the structured features of images and to begin end-to-end training of big data in medical image analysis. In particular, we apply the transfer learning method to transfer the weight parameters that were pre-trained on the ImageNet dataset to the CNN section and adopted a custom loss function to allow our network to train and converge faster and with more accurate weight parameters. Experimental results show that compared with the other CNN models such as ResNet and Inception V3, our proposed network model is more accurate and efficient in classifying blood cell images.

INDEX TERMS Artificial intelligence, convolutional neural network, recurrent neural network, transfer learning.

I. INTRODUCTION

It is well known that blood cells mainly include red blood cells, white blood cells and platelets. In blood, leucocyte plays an important role in the human immune function, so it is also called the immune cell. Usually, hematologists use granulated information and shape information in leukocytes to divide white blood cells into granular cells: neutrophil, eosinophil, basophil and non-granular cells: monocyte and lymphocyte. The proportion in the blood of these five types of cells is different for the diseased and non-diseased bloods. Doctors often use these basic data as criteria for determining the type and severity of this disease. Therefore, the study of white blood cell classification has important significance and value for medical diagnosis.

Based on the significance of blood cell classification in the diagnosis, researchers have proposed many algorithms to classify blood cells. In 2003, Sinha and Ramakrishnan [1] classified cells using SVM with a recognition rate of 94.1%. In 2006, Yampri *et al.* [2] used 100 images to perform the same experiments. They implemented the automatic threshold and adaptive contour to segment cells, and used the smallest error method to classify them, and the recognition rate was 96% [2]. Yampri *et al.* [2] utilized the KNN algorithm. However, the KNN algorithm does not handle unbalanced samples well. If the sample capacity of a class is large, while the sample capacity of other classes is small, some issues arise. For example, when a new sample is input into the diagnostic system, it may result in a class with a large capacity of being dominant in the K nearest neighbors of this sample. In addition, the algorithm is computationally expensive because each sample needs to be categorized in order to calculate its distance from all known samples so as to obtain its K nearest neighbors. Therefore, this algorithm is more suitable for larger data samples. And for those data

with fewer samples, it is more likely to produce incorrect classifications. The above experiments on white blood cells have a common characteristic. In particular, regardless of the final result being good or bad, they only use a small amount of white blood cell images, or use their own database to test their identification, and the results become biased. To the best of our knowledge, there are no large and public white blood cell detection and classification databases. Therefore, in order to describe the effectiveness of the proposed algorithm fairly and objectively, we have collected some currently known dataset. We used a BCCD dataset (small-scale dataset for blood cell detection) and pre-processed the dataset, which then turn it into 12,444 blood cell-enhanced images (comprising 9,957 training data and 2,487 test data). In this dataset, we divide the blood cells into 4 different types, namely, eosinophil, lymphocyte, monocyte, and neutrophil.

With the improvement of computer performance, convolutional neural network has become more popular lately. In ImageNet on 2012, Krizhevsky *et al.* [3] refreshed the records of the image classification. Here, they used a structure known as AlexNet [3]. At this point of time, deep learning has become the most sought after research topic, and excellent neural network algorithms such as VGG, Xception and ResNet have emerged. They also perform well in image classification tasks [4]–[7].

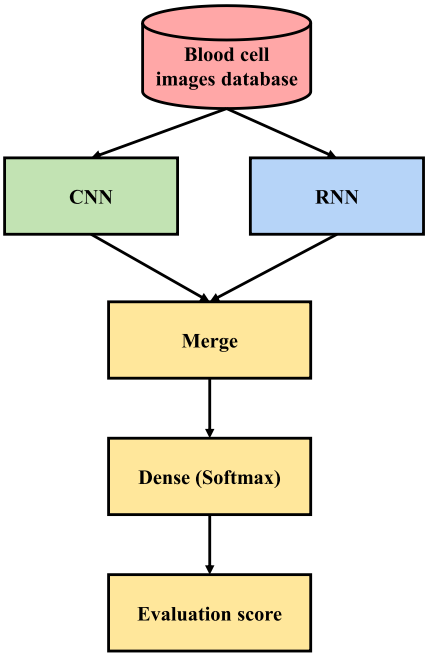


FIGURE 1. Overview of the proposed method using CNN-RNN framework and transfer learning for classifying blood cell images.

In this paper, we propose an architecture that combines CNN and RNN [9], [10]. As shown in Figure 1, it integrates the local features extracted from the CNN and the features obtained from the RNN [12] to perform the blood cell classification. We use the CNN [11] model pre-trained on the ImageNet dataset and retains its weight parameters [19]. Note that we perceive the CNN and RNN as two separate branches. First, we freeze the pre-trained CNN model, use the pre-processed training data as the input of the RNN model, extract and save the obtained features, and merge the features from the RNN and CNN. The RNN weight parameters are constantly updated during training. Finally, we thaw all network layers and use the training data as input to the CNN model and the RNN model. In the CNN model, we apply different size and weight matrix windows in order to generate multiple feature maps [3], [13]. The features extracted from the RNN model and the features extracted from the CNN model are combined according to the corresponding element multiplication methods. And then, the classification results are displayed using Softmax [21]. In addition, we also use a fine-tuning strategy to retrain the CNN-RNN framework, and finally obtain the classification results.

The proposed blood cell classification method has four major contributions compared to the existing methods as follows.

First Contribution: To the best of our knowledge, this is the first application of the combined model of CNN and RNN in the classification of blood cells. This combination model can effectively use the temporal and spatial characteristics of information to achieve better classification results.

Second Contribution: Using the pre-trained method of the CNN model, its chosen weights are close to a good local optimum, so that they are kept within a high gradient range and they can be effectively fine-tuned.

Third Contribution: RNN is a neural network that captures dynamic information in serialized data by hiding the periodic connections of nodes in layers, and can classify serialized data [14], [15]. Different from other forward neural networks, RNN can save the context state, and even can store in any arbitrarily long context window, learn and express relevant information, and is no longer limited to the spatial boundaries of traditional neural networks. Based on this theory, we introduce a RNN with memory function to generate a continuous time output state for features extracted from CNN. In addition, because the LSTM (a special type of RNN network, Long Short-Term memory networks) [17], [18] can solve the problem of gradient disappearance during traditional RNN network training, our network model can have more layers.

Forth Contribution: The proposed method adjusts the loss function and activation function of RNN, and introduces the RMSProp [23] and Adam optimizers to train the network. At the same time, we have also introduced a fine-tuning strategy to train the CNN-RNN framework. The experimental results show that this method has achieved good results.

The rest of the arrangements are as follows. The section 2 reviews related work. The section 3 describes an improved algorithm based on CNN-RNN model (Xception-LSTM). The section 4 describes the experimental results. The section 5 analyzes and discusses our approach. Finally, the section 6 summarizes the novelty and utility of our

proposed approach and suggests implementations for future work.

II. RELATED WORK

Previously related blood cell classification algorithms mainly include the KNN algorithm, Bayesian classifier, SVM classifier, etc. We briefly review and discuss in this section.

The core idea of the KNN algorithm is that if most of the k most adjacent samples in a feature space belong to a certain category. Note that the sample also has the characteristics of all the other samples in this category. This method determines the class in which the sample is to be classified based on the category of the nearest samples in determining the classification decision. The KNN method is only relevant to a very small number of neighboring samples in the category decision. Based on this theory, Young (1972) experimented with 199 cell images. He first used histogram thresholds to segment white blood cells and classified them using a distance classifier. The recognition rate was 92.46% [24]. Bikhet *et al.* [25] used entropy based and iterative thresholding methods to divide cells and classify them with a distance classifier, with a recognition rate of 90.14%.

Bayesian classification is based on statistical classification and uses its knowledge of probability statistics to classify data. In many classifications, naive Bayes algorithm can be compared with decision tree and neural network algorithm. Sinha and Ramakrishnan [1] used Bayesian classifiers to classify cells and the recognition rate was 82.3%. Theera-Umpon and Dhompongsa (2007) used a Bayesian classifier to classify the bone marrow images of the Ellis Fisher Cancer Center at the center of Missouri (only one cell per picture), and the recognition rate was 77% [26], [27]. Ghosh *et al.* [28] used a watershed algorithm to segment 150 cell images and classify them using a Bayesian classifier, and the recognition rate was 83.2%.

The classification idea of SVM is essentially similar to the linear regression LR classification method. It is to obtain a set of weight coefficients that can be classified after linear representation. SVM first trains a separation hyper-plane, and then the plane is the decision boundary of the classification. Classical SVM algorithm is only suitable for two types of classification problems. After improvement, SVM can also be applied to multiple classification problems. In the actual application of white blood cell classification, it is generally necessary to solve the problem of multiple classifications. For example, the five-classification problem of leukocytes we studied can be solved by combining multiple binary SVM. Rezatofighi and Soltanian-Zadeh [29] used the Gram-Schmidt Orthogonal and Snake algorithm to segment 400 blood smears and classified them using SVM. Their recognition rate was 90% [29].

Recently, convolutional neural networks have been widely implemented in various image classification fields. In particular, convolutional neural networks (ConvNets) [11] achieved unprecedented results in the 2012 ImageNet large-scale visual recognition challenge, which included classifying natural images in the ImageNet dataset into 1000 fine-grained categories [3]. They also significantly improve the performance of various medical imaging applications [30], [31], such as classification of lung diseases and lymph nodes in CT images [32], [33], segmentation (pixel classification) of brain tissues in MRI [34], vessel segmentation based on fundus images [37], and detecting cervical intraepithelial neoplasia (CIN, particularly CIN2+) at patient level based on Cervigram images or Multimodal data [36]. In addition, ConvNets showed superior performance in cell image classification such as pleural cancer [38] and human epithelial cell images [39].

Although these methods can be used to generate good classification engines, they still have some drawbacks. Traditional machine learning methods (such as SVM) need to extract features manually. The acquisition of features mainly depends on the designer's prior knowledge. This feature extraction method is difficult to make full use of the information contained in the image, and will increase the designer's workload. The deep learning algorithm effectively solves this problem. It can automatically learn the effective features of the image. Deep learning algorithms such as deep residual network also have good performance in image classification tasks. However, these neural network classification algorithms cannot fully utilize some features of the image that have a long-term dependency relationship with image labels, and thus these classification methods cannot classify cell images like people with memory. For this purpose, we introduce a recurrent neural network and fuse it with a convolutional neural network to perform the task of blood cell image classification.

III. METHODS

Convolutional neural networks are models proposed by the development of biotechnology. Neurons are like local filtering of the entire input space, and they are well-organized together to achieve an understanding of the image in the entire field of view. Convolutional neural networks (CNN) may extract the local and deep features of the input image.

Recurrent neural networks are a neural network used to process sequence data. In the traditional neural network model, from the input layer to the hidden layer to the output layer, the layers are connected with each other, and the nodes between each layer are connected. Such network model cannot handle this kind of sequential data. Socher et al. (2012a) presented a convolutional-recursive deep model for 3D object classification that combined the convolutional and recurrent neural networks (CNN and RNN) together. The CNN layer learns low-level translation invariant features and then uses it as input to multiple fixed-tree RNNs to form higher order features. RNNs can be seen as combining convolution and pooling into one efficient, hierarchical operation [16]. Kim *et al.* [8] described a model that employed a convolutional neural network (CNN) and a highway network over characters, whose output is given to a long short-term memory (LSTM) recurrent neural network language

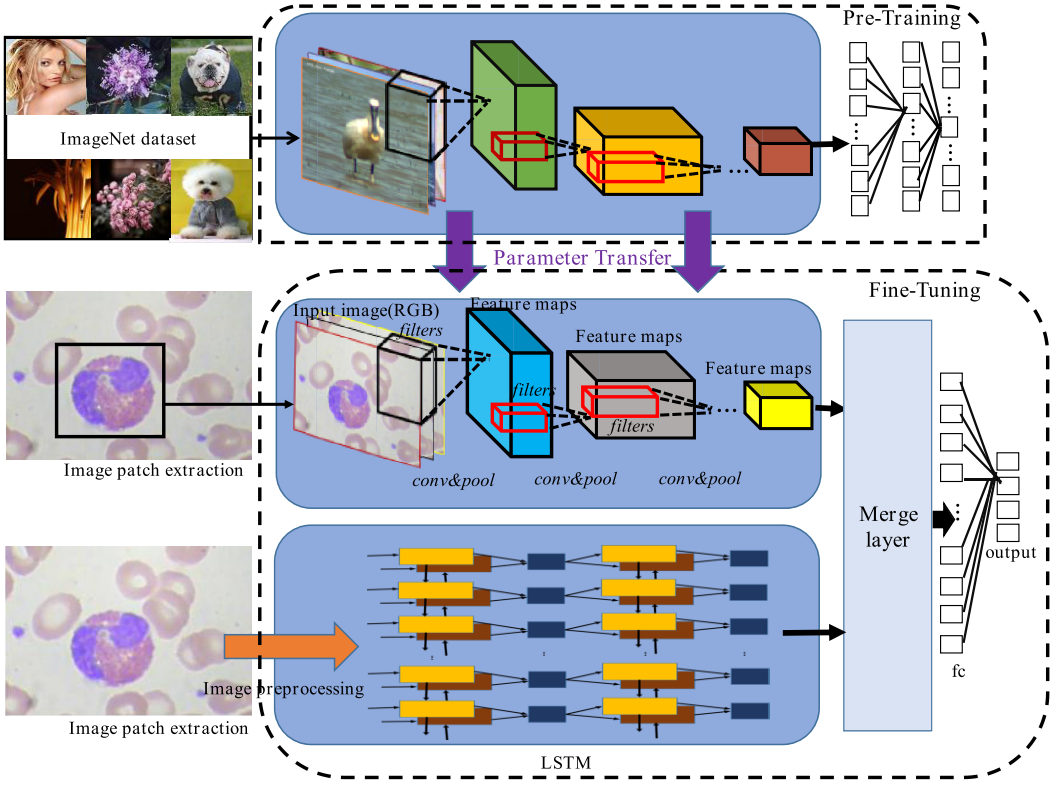


FIGURE 2. Combinatorial model that is fine-tuned based on human blood cell dataset with parameters transferred from another CNN pre-trained on the ImageNet dataset. The weight parameters of the blue area in the pre-trained model are migrated to the same position of another CNN model for fine-tuning on blood cell dataset.

model (RNN-LM). These two models both obtain better results compared to a-priori methods.

Inspired by these works, we combined the use of both CNN and RNN frameworks for the classification of blood cell images. We termed the model proposed in this paper as CNN-RNN. The model architecture is shown in Figure 2. The proposed method includes a training phase and a testing phase. During the training phase, our CNN model was first pre-trained on the ImageNet dataset and data pre-processed techniques were applied to the blood cell dataset. Next, a transmission learning technique is applied so that pre-trained network parameters are used to initialize a new CNN. Then all CNN layers are frozen and the RNN model is trained. After the training is completed, the CNN model is thawed and the entire CNN-RNN model is trained. At the same time, neural network attention mechanisms are used to merge both features arising from the CNN and RNN. In the testing phase, the pre-processed test images are input into the fine-tuned CNN-RNN model, and the classification results are obtained through the Softmax layer. Further details are described below.

A. DATASET

Our raw data is retrieved from the BCCD dataset (https://github.com/Shenggan/BCCD_Dataset) and publicly available dataset (https://www.kaggle.c-om/paultimothymooney/blood-cells/data). We obtained a new dataset comprising 12,444 augmented images of blood cells (JPEG), with 9,957 and 2,487 training and test images, respectively. We divided these images into four different types according to the type of blood cells. The cell types are eosinophil, lymphocyte, monocyte, and neutrophil. In the training dataset, there are 2,497 eosinophil, 2,483 lymphocyte, 2,478 monocyte, and 2,499 neutrophil images; in the test dataset, there are 623 eosinophil, 623 lymphocyte, 620 monocyte, and 624 neutrophil images. Initially, the entire network used RGB images of size $320 \times 240 \times 3$ pixels. Figure 3 shows the different types of cell images.

B. DATA PRE-PROCESSING

In order to improve the accuracy of the model and reduce over-fitting, we need to enhance the dataset. We use matrix transformations so as to increase the number of image samples. Figure 4 shows a blood cell image that is obtained by application of rotation matrix. However, rotating cell images may slightly reduce their high-frequency content (which can be considered poor image quality) but should not change the abnormality/normality of most cells. I In fact, the augmentation step based on image rotation is crucial to the success of our method, and it has been proved to be important for improving the accuracy of cell image classification based on Xception-LSTM model, given the limited number of images. Other data augmentation approaches, such as scale and color transformations are not used, because the size and intensity of

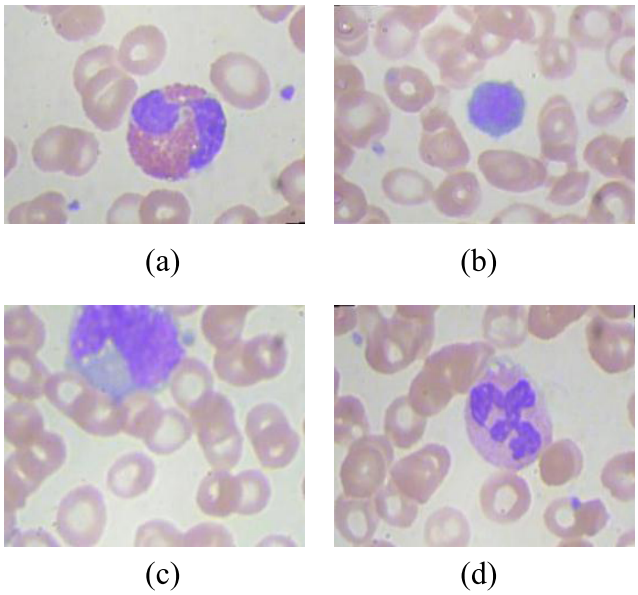


FIGURE 3. Database comprising 12,444 cell images of (a) eosinophil, (b) lymphocyte, (c) monocyte, and (d) neutrophil.

the nucleus are fundamental features that distinguish blood cell types. Classifiers tend to show prejudice against most categories. Although from a medical perspective, high sensitivity is ideal, from a practical point of view; high false alarm rates are unpredictable. A common solution to this dilemma is to achieve sample equalization, which also improves the training model's accuracy and convergence speed. Therefore, we have balanced the number of different types of blood cell images in the dataset [42].

C. MODEL

Our model consists of the following parts: Pre-trained convolutional neural network layer, RNN layer, Merge layer, and fully connected layer with Softmax output. The overall framework of the model is shown in Figure 2.

1) PRE-TRAINED CONVOLUTIONAL NEURAL NETWORK LAYER

We use the weight parameters obtained by pre-training on the ImageNet dataset as the initialization weights of our CNN model. Convolutional neural networks include convolutional layer and pooling layer.

2) CONVOLUTIONAL LAYER

As the most important part of the convolutional neural network, the main way to calculate this layer is to use convolution windows with different sizes in order to perform convolution operations with the feature maps of the previous layer. Convolution windows of different sizes slide in sequence onto the feature map of the previous layer. The window size is usually 3×3 or 5×5, and the number of weight parameters of the convolutional layer also changes accordingly. The values of the neurons on each feature map in the convolutional layer are convoluted through corresponding windows, and then the final result is obtained based on the excitation function used in the layer.

3) POOL LAYER

The calculation process of this layer is similar to the operation of the convolutional layer. The difference is that the sliding window of the lower sampling layer is usually 2×2, and the sliding step is 2. Therefore, this process will usually halved the feature map of the size of the previous layer, which to a large extent can greatly reduce the convolution weights of neural network parameters, the number of there are very good for the overall speed of the network training process to promote. At the same time, it also enables the network to become more adaptive to the scale of the image changes.

In this article, our pre-trained model is Xception [40], and we use the ReLU (Linear Rectification Function) as our activation function. Xception is another improved model of Google's Inception v3 [7]. It mainly implements depthwise separable convolution to replace the original convolution operation in Inception v3. Xception, as an improved model for Inception v3, mainly introduces depthwise separable convolution on the basis of Inception v3, which improves the model's effect without increasing network complexity. In addition, we found that Xception has a better effect on cell classification in experiments.

4) RNN LAYER

Both RNN and CNN include an input layer, a hidden layer, and an output layer. The connection of these hidden layers is the most important feature of RNN. The input layer nodes and the hidden layer nodes are connected with each other, and the hidden layer is output to the output layer. The node output information returns to the hidden layer node again, and can even include the hidden layer adjacent nodes to each other. This is a dynamic network. Biological neural networks are cyclic networks that can understand serial data, so RNNs are represented closer to the biological nervous system. In this paper, we mainly use a special type of recurrent neural network - LSTM (Long Short-Term Memory), which is illustrated in Figure5. This approach can learn long-term dependence information. The difference between LSTM and RNN is that it adds a ''processor'' to determine whether the information is useful or not. The structure of this processor is called the cell. Three doors are placed in a cell and they are called an input gate, a forgotten gate, and an output gate. A message enters the LSTM network and can be judged by rules. Only the information that meets the algorithm's certification will remain, and the inconsistent information will be forgotten through the Oblivion Gate. An LSTM consists of a memory cell and three multiplicative gates, namely the input, output and forget gates. Compared to the standard RNN repeat module, the LSTM has a more complex internal structure.

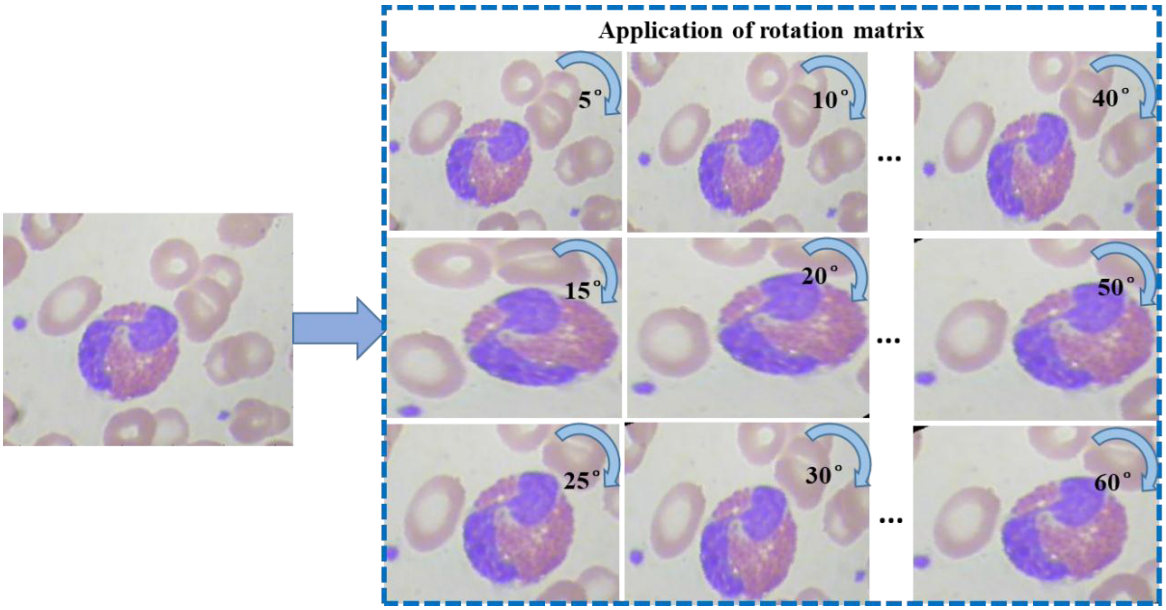


FIGURE 4. Image patches are generated from the blood cell image by application of rotation matrix so that the acquired images can be cropped to further augment the dataset.

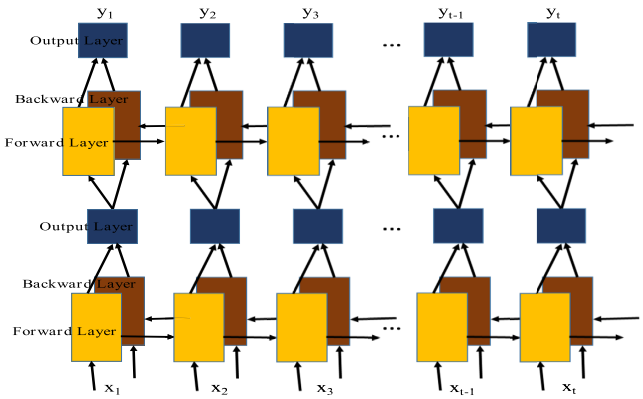


FIGURE 5. Structure of using bidirectional LSTM in the above figure (X refers to data input, Y refers to the output), which combine forward (left to right) and backward (right to left) to form a Bi-directional LSTM.

5) MERGE LAYER

The merge layer function is to use a specific method to merge the features extracted from the CNN and the features obtained from the RNN. The merge layer function is to use a specific method to merge the features extracted from the CNN and the features obtained from the RNN. In our combinatorial model, we introduce attentional mechanisms of neural networks. The neural network attention mechanism is a neural network that can focus on its input (or features), and it can select specific input. We use the corresponding element-wise multiplication operations for feature merging.

6) FULLY CONNECTED LAYER WITH SOFTMAX OUTPUT

After the features generated by the RNN are merged with the features generated by the CNN, they are passed to the fully connected Softmax layer, the output of which is the probability distribution of all classes. In addition, we use the cross-entropy as a loss function to measure the difference between the actual output and the target output.

D. NETWORK TRAINING

Our model is divided into two separate branches. The weights in the CNN branch use parameters that were pre-trained on the ImageNet dataset, and the RNN branch randomly initialize the parameters. In the training process, these weights are iteratively updated through the gradient of the cross-entropy loss function. The CNN layer is first frozen, the training samples are calculated by the RMSProp optimizer, and requiring 100 epochs for training. And then the CNN layer is thawed, the entire network uses the Adam optimizer to calculate training samples, the learning rate is 0.0001, and 70 epochs are required for training at this time. The training process terminates after a predetermined number of periods. The model with the lowest verification loss value was chosen as the final network.

E. TRANSFER LEARNING

Transfer learning refers to the migration of trained model parameters to a new model to help train the new model. In this study, the conv and pool layers of the first few pre-trained ImageNet classification dataset (the upper blue area in Figure 2) of the Xception model are used as the base of our network, on top of which several task-specific fc layers with random initialized weights are attached. In order to facilitate the transmission of features, the same network layer (conv and pool) as Xception is transferred to the same position in our model (blue area in the middle of Fig. 2).

In addition, Xception uses the RGB channel as an input, and RNN takes a single channel map as input. All of these layers are jointly trained (tweaked) on our blood cell dataset and use the original learning rate to train the fc layer from scratch.

IV. EXPERIMENTS AND RESULTS

The proposed model consists of two different sections: the CNN section, which uses the Xception model, and uses the transfer learning method; the other section is the RNN, which uses the Bi-directional LSTM model.

Our network is trained in two sections. In the first section, all layers of CNN are frozen and only the last classification layer and RNN network are trained. This is done using the RMSProp optimizer. In the second section, all layers of the entire network were thawed and adjusted using the Adam optimizer [22], and the learning rate was 0.0001. In addition, we introduce a cross-entropy loss function in the model and optimize it so that the loss function does not only fit the one-hot distribution, but also consider fitting the uniform distribution. The purpose of our proposed model is to divide our blood cell images into 4 categories. Xception-LSTM runs on the tensorflow framework using an NVIDIA Tesla K40c GPU with 12GB of memory.

In order to prove that our model has better blood cell classification effect, we used the following models to conduct experiments on our dataset and compare the experimental results: ResNet50, Xception, InceptionV3, ResNet50-LSTM, Xception-LSTM, and InceptionV3-LSTM, Xception-ResNet50-LSTM.

A. MODEL LEARNING RESULTS

Figure 6 illustrates the fine-tuning of our model (Xception-LSTM) during 70 training epochs on the dataset. As shown in the figure, after 59 epochs, the validation loss reached its minimum value (0.637), and the corresponding validation accuracy was 0.9079. Figure 7 shows a visualization of the

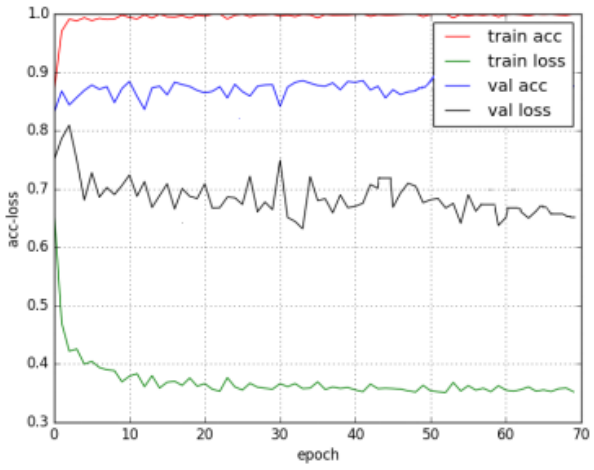


FIGURE 6. Training and validation loss, training and validation accuracy versus number of training epochs.

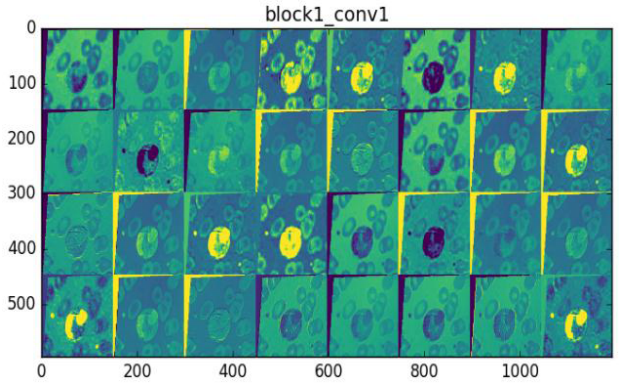


FIGURE 7. Visualization of the result of the activation (feature maps) of in the first convolutional layer of the Xception-LSTM fine-tuned on the dataset (block1_conv1 is the first convolutional layer of our model).

output features of the first convolutional layer (32 filters of size $3 \times 3 \times 3$ pixels) of the model after training on the dataset. The first layer convolution layer is similar to the edge detection function. At this stage, the convolution kernel basically retains all the information of the image. Along with these convolved feature maps, the activation of an example cell on different pool layers (block2_pool, block3_pool, and block4_pool) is provided in Figure 8.

B. QUANTITATIVE RESULTS ON DATASET

Table 1 shows that our combined model (Xception-LSTM) has the highest classification accuracy of 90.79% for blood cell images compared to other models.

C. COMPUTATIONAL SPEED

The average training time of an Xception-LSTM running over up to 70 epochs is 14 hours approximately, the testing time for one blood cell image is 3.8 seconds on average.

V. DISCUSSION

Compared with the existing methods and based on their experimental results, we found that the combined structure of CNN and RNN models is superior to CNN models and traditional machine learning algorithms in blood cell image classification. We use Xception model and RNN model, so as to obtain higher classification accuracy compared to the existing models.

A. COMPARISON WITH PREVIOUS METHODS

The methods in [1], [25]–[29] followed a traditional cell classification process that is characterized by manually segmented cytoplasm/nuclei. In contrast, our method can adaptively learn the feature of the input image and is therefore not limited by the cell segmentation or feature design [43]. The traditional method of cell classification is greatly affected by the balance of the sample, which causes the classifier to predict more cell abnormalities, and our method can effectively solve this problem. In addition, compared with some classical neural network models (ResNet, GoogleNet, etc.),

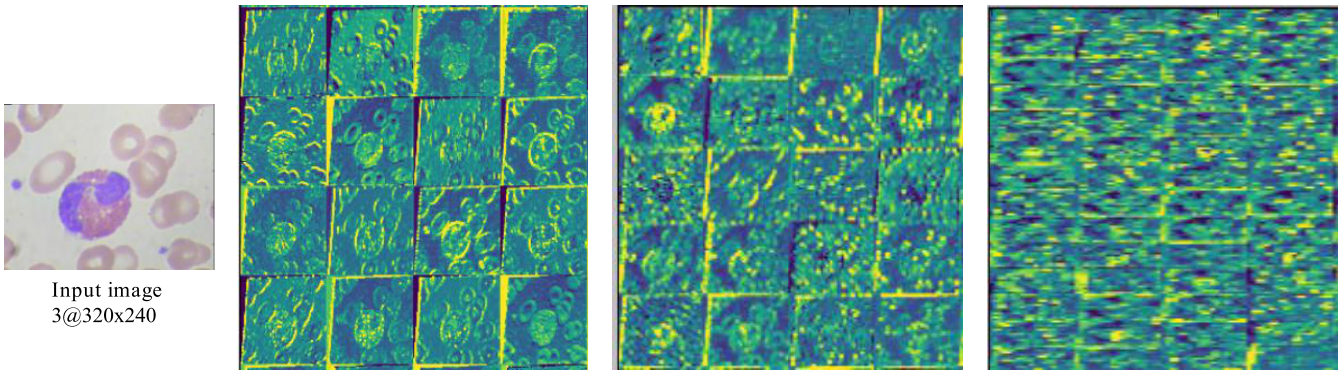


FIGURE 8. Visualization of the activation (feature maps) of three pooling layers, block2_pool, block3_pool, and block4_pool for an input blood cell image.

TABLE 1. Comparison of classification results by the various models.

Model	Sub-Models	Four-classification accuracy
CNN	Inception V3	84.08
	ResNet50	87.62
	Xception	88.70
CNN-RNN	Inception V3-LSTM	87.45
	ResNet50-LSTM	89.38
	Xception-LSTM	**90.79**
	Xception- ResNet50-LSTM	88.58

our combined model can make full use of the spatiotemporal information of image features. Therefore, our new method has great application potential in the field of blood cell image classification.

B. ADVANTAGES OF THE PROPOSED METHOD

The proposed model can adaptively learn the characteristics of the blood cell images; thereby avoiding manual feature extraction (no cytoplasmic/nuclear segmentation is needed). Moreover, we used a combination of CNN and RNN models to classify blood cells. In particular, we use the transfer learning method to migrate the weight parameters pre-trained on the ImageNet dataset to the CNN branch, thereby enhancing the robustness of the combined model and accelerating the convergence of the model.

Combination model uses the characteristics of the recurrent neural network to use Bi-directional LSTM combined with CNN to classify cells. This method enables our model to take into account the spatiotemporal characteristics of the information contained in the image, and can effectively learn the structural characteristics of blood cell images, which will result in higher classification accuracy.

Our method when combined with Xception and LSTM achieved a good classification effect in the experiment. As is well-known, the Xception structure is a linear stack of depthwise separable convolution layers with residual connections. Unlike Inception, Xception does not divide the input data into several compressed data blocks, but instead maps the spatial correlation for each output channel and then performs a 1×1 depth convolution to obtain a cross-channel correlation. Xception introduces a deeply separable convolution, which basically does not increase the complexity of the network under the premise of improving the model. Therefore, our combined model also has the same features as Xception. In addition, the neural network attention mechanism we introduce incorporates features from the RNN branch and the CNN branch, allowing the model to be more focused on finding useful information in the input data that is relevant to the current output, and thereby improving the quality of

the output. The experiment confirmed the validity of our proposal.

C. LIMITATIONS

Despite its high performance, our method still has some shortcomings, and this may affect its clinical application. The classification of a single cell image takes about 3.8 seconds, which is clinically too slow. It can be solved by improving computer hardware performance or further optimizing the network structure and reducing the data dimension. Despite the high classification accuracy based on the dataset, our experiments are currently performed on most single cell images. In future research, we must consider the effects of overlapping nuclei and cell clusters on classification accuracy more extensively, because clinically, our method will face situations where multiple cells overlap, and we may need to design a task-specific classifier to handle such problems [41].

VI. CONCLUSION

In this work, we propose a depth neural network architecture that combines the features of convolutional neural networks (Xception) and recursive neural networks (LSTM). We then implement the combined Xception-LSTM framework for blood cell image classification. Our model preserves the temporal and spatial information of image features and can learn structured information of image features. Unlike previous manual feature extraction methods, which rely on cytoplasmic/nuclear segmentation, our method can automatically extract and classify the deep features embedded in cell image patches. Compared with the previous existing methods, our proposed technique achieved the highest performance in terms of classification based on the blood cell dataset. We hope that this segmentation-free, highly accurate blood cell classification method can be used to develop medical-aided diagnostic systems for blood-related diseases in the future.

ACKNOWLEDGMENT

The authors would like to thank the Innovation Center of Huaqiao University of Engineering College for providing high performance computing platforms and servers in order to run our code.

REFERENCES

[1] N. Sinha and A. G. Ramakrishnan, "Automation of differential blood count," in *Proc. Conf. Convergent Technol. Asia–Pacific Region (TENCON)*, Bengaluru, India, vol. 2, 2003, pp. 547–551.

[2] P. Yampri, C. Pintavirooj, S. Daochai, and S. Teartulakarn, "White blood cell classification based on the combination of eigen cell and parametric feature detection," in *Proc. 1st IEEE Conf. Ind. Electron. Appl.*, Singapore, May 2006, pp. 1–4.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, vol. 60, 2012, pp. 1097–1105.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.

[6] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.* , Jun. 2015, pp. 1–9.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2818–2826.

[8] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models," *Comput. Sci.*, Aug. 2015.

[9] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.

[10] B. Su and S. Lu, "Accurate scene text recognition based on recurrent neural network," in *Proc. ACCV*, vol. 9003, 2014, pp. 35–48.

[11] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[12] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Vancouver, BC, Canada, vol. 38, May 2013, pp. 6645–6649.

[13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[14] J. Almazán, A. Gordo, A. Fornés, and E. Valveny, "Word spotting and recognition with embedded attributes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 12, pp. 2552–2566, Dec. 2014.

[15] J. A. Rodriguez-Serrano, A. Gordo, and F. Perronnin, "Label embedding: A frugal baseline for text recognition," *Int. J. Comput. Vis.*, vol. 113, no. 3, pp. 193–207, 2015.

[16] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3D object classification," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 656–664.

[17] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[18] X. Zhu, P. Sobhani, and H. Guo, "Long short-term memory over recursive structures," in *Proc. Int. Conf. Int. Conf. Mach. Learn.*, 2015, pp. 1604–1612.

[19] G. Carneiro, J. Nascimento, and A. P. Bradley, "Unregistered multiview mammogram analysis with pre-trained deep learning models," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, Springer, 2015, pp. 652–660.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] Y. Sun, Y. Chen, X. Wang, and X. Tang , "Deep learning face representation by joint identification-verification," in *Proc. Conf. Workshop Neural Inf. Process. Syst.*, vol. 2, 2014, pp. 1988–1996.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–15.

[23] L. M. Soria-Morillo, Á. Arcos-García, and J. A. Álvarez-García, "Deep neural network for traffic sign recognition systems: An analysis of spatial transformers and stochastic optimisation methods," *Neural Netw.*, vol. 99, no. 12, pp. 158–165, 2018.

[24] I. T. Young, "The classification of white blood cells," *IEEE Trans. Biomed. Eng.*, vol. BME-19, no. 4, pp. 291–298, Jul. 1972.

[25] S. F. Bikhet, A. M. Darwish, H. A. Tolba, and S. I. Shaheen, "Segmentation and classification of white blood cells," in *Proc. IEEE Int. Conf. Acoust.*, vol. 6, Jun. 2000, pp. 2259–2261.

[26] N. Theera-Umpon and P. D. Gader, "System-level training of neural networks for counting white blood cells," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 32, no. 1, pp. 48–53, Feb. 2002.

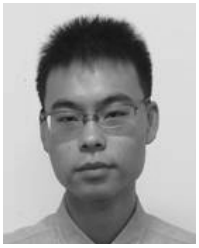
[27] N. Theera-Umpon, "Automatic white blood cell classification using biased-output neural networks with morphological features," *Sci. Technol. Asia*, vol. 8, no. 1, pp. 64–71, 2003.

[28] M. Ghosh *et al.*, "Statistical pattern analysis of white blood cell nuclei morphometry," in *Proc. Students' Technol. Symp. (TechSym)*, 2010, pp. 59–66.

[29] S. H. Rezatofighi and H. Soltanian-Zadeh, "Automatic recognition of five types of white blood cells in peripheral blood," *Comput. Med. Imag. Graph.*, vol. 35, no. 4, pp. 333–343, 2011.

[30] H. Greenspan, B. V. Ginneken, and R. M. Summers, "Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1153–1159, Mar. 2016.

[31] R. M. Summers, "Progress in fully automated abdominal CT interpretation," *Amer. J. Roentgenol.*, vol. 207, pp. 67–79, Jul. 2016.
[32] H.-C. Shin *et al.*, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1285–1298, May 2016.
[33] H. R. Roth *et al.*, "Improving computer-aided detection using convolutional neural networks and random view aggregation," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1170–1181, May 2016.
[34] T. Xu *et al.*, "Multi-feature based benchmark for cervical dysplasia classification evaluation," *Pattern Recognit.*, vol. 63, pp. 468–475, Mar. 2017.
[35] T. Xu, H. Zhang, X. Huang, S. Zhang, and D. N. Metaxas, "Multimodal deep learning for cervical dysplasia diagnosis," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, Springer, 2016, pp. 115–123.
[36] P. Moeskops, M. A. Viergever, A. M. Mendrik, L. S. de Vries, M. J. N. L. Benders, and I. Išgum, "Automatic segmentation of MR brain images with a convolutional neural network," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1252–1261, May 2016.
[37] P. Liskowski and K. Krawiec, "Segmenting retinal blood vessels with deep neural networks," *IEEE Trans. Med. Imag.*, vol. 35, no. 11, pp. 2369–2380, Nov. 2016.
[38] P. Buyssens, A. Elmoataz, and O. Lézoray, "Multiscale convolutional neural networks for vision–based classification of cells," in *Proc. Asian Conf. Comput. Vis.*, vol. 7725, 2012, pp. 342–352.
[39] Z. Gao, L. Wang, L. Zhou, and J. Zhang, "HEp-2 cell image classification with deep convolutional neural networks," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 2, pp. 416–428, Mar. 2017.
[40] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 1800–1807.
[41] L. Zhang *et al.*, "Automation-assisted cervical cancer screening in manual liquid-based cytology with hematoxylin and eosin staining," *Cytometry A*, vol. 85, no. 3, pp. 214–230, 2014.
[42] J. Li, S. Fong, Y. Sung, K. Cho, R. Wong, and K. K. L. Wong, "Adaptive swarm cluster-based dynamic multi-objective synthetic minority oversampling technique algorithm for tackling binary imbalanced datasets in biomedical data classification," *BioData Mining*, vol. 9, no. 1, p. 37, 2016.
[43] J. Li, S. Fong, R. K. Wong, R. Millham, and K. K. L. Wong, "Elitist binary wolf search algorithm for heuristic feature selection in high-dimensional bioinformatics datasets," *Sci. Rep.*, vol. 7, no. 1, 2017, Art. no. 4354.



GAOBO LIANG was born in Dawu County, China, in 1994. He is currently pursuing the M.S. degree in computer science and technology with the College of Engineering, Huaqiao University, China. Since 2017, he has been a Research Assistant with the College of Engineering, Huaqiao University. His research interests include the application of deep learning in medical image analysis and the application of deep learning in rapid visual identification and positioning of components used by industrial sorting robots.

HUICHAO HONG was born in Hui'an County, China, in 1991. He is currently pursuing the M.S. degree in computer science and technology with the Huaqiao University of Engineering. He has authored high-level academic papers in multiple journals. His research interests include the application of parallel computing in image processing and the application of deep learning in medical image analysis.

WEIFANG XIE was born in China in 1994. She is currently pursuing the M.S. degree with the College of Engineering, Huaqiao University, China. Since 2016, she has been a Research Assistant with the College of Engineering, Huaqiao University. Her research interests include computer vision and the application of deep learning in the classification of bio-ore and Flammulina mushroom.

LIXIN ZHENG was born in 1967. He is currently the Dean and a Professor with the Engineering College, Huaqiao University. His research mainly focuses on image recognition and machine vision technology. He is the Vice President of the Fujian Automation Society, the Vice President of the Xiamen Automation Society, and the Director of the Fujian Power Supply Society.

He received the bachelor's degree from the Department of Electronic Engineering, Huaqiao University, in 1987, the master's degree in the field of testing and automation control from the Department of Mechanical Engineering in 1990, and the Ph.D. degree in predictive control from Tianjin University in 2002. In 1997, he went to Japan for further study. He received the Backbone of the Outstanding Young and Middle-Aged Teachers of Huaqiao University in 2004 and the title of "Outstanding Teacher of Huaqiao University" in 2006. In 2007, he was supported by the Fujian Excellent Talent Support Program.

Dr. Zheng is currently a Professor specialized in the sensor and detection technology, digital image processing, pattern recognition, automatic control system of electric drive, computer control system, control theory, field bus technology, and other undergraduate and graduate courses.