

Received May 30, 2020, accepted June 7, 2020, date of publication June 11, 2020, date of current version June 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3001571

Segmentation of Cell Images Based on Improved Deep Learning Approach

CHUANBO HUANG^{1,2}, HUALI DING¹, AND CHUANLING LIU³

¹Department of Computer Science, Jining University, Qufu 273155, China

²School of National Defense Science and Technology, Southwest University of Science and Technology, Mianyang 621010, China

³School of Information Technology, Shangqiu Normal University, Shangqiu 476000, China

Corresponding author: Chuanbo Huang (huangchuanboqc@126.com)


This work was supported in part by the National Natural Science Foundation of China under Grant U1504610, Grant 61971339, Grant 61471161, and Grant 61972057.

ABSTRACT The improved U_net algorithm based on mixed convolution blocks (McbUnet), which combines the advantages of U-Net and residual learning, is proposed for cell image segmentation in this paper. The network is mainly composed of two kinds of mixed convolution blocks. There are three main benefits to this algorithm. First, the convolution block can utilize different size kernels to overcome the limitation of a single size convolution kernel in traditional deep convolution. Second, in the mixed convolution blocks, two hyperparameters (Width multiplier and Resolution multiplier) are used to quickly adjust the model to fit a specific environment. Third, the residual paths are improved. We test the proposed network and compare it with other recent segmentation methods based on deep learning. The proposed method is superior to comparison methods, which shows its effectiveness.

INDEX TERMS Microscopy images, deep learning, cell segmentation, convolutional neural network.

I. INTRODUCTION

Quantitative analysis of microscopic images has been widely used in medical research fields such as pathological management, pathological diagnosis system, estimates for cancer grade, cancerous classification and so on. These applications are closely related to the development of new technologies in computer vision and machine learning for image segmentation and classification [1], [2]. Cell segmentation and extraction is an important technology, which is directly related to the reliability of the diagnosis and is also a difficult problem in medical image processing. So far, there is no universal and efficient segmentation method that can be widely applied to cell segmentation. There are several reasons for its implementation. First, the size of the cells varies greatly, and the shape of the nucleus is various. Second, cell images are complex, not only white blood cells, red blood cells and platelets but also other things, and are divided into many different categories depending on the degree of maturation of white blood cells. Third, cell images are often affected by uneven staining and inconsistent illumination, resulting in changes in grayscale values. Fourth, cell images often overlap, with noise and artifacts, no obvious boundaries [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Hossein Rahmani .

All these decrease contrast between cells and background and seriously affects the effective segmentation of cell image [4]. Therefore, cell segmentation is a difficult and challenging task. Research on cell image segmentation methods is absolutely necessary.

Traditional segmentation methods, including histogram-based method, contour-based method, watershed algorithm, minimum-error-threshold method [3]–[6], can't be directly used to correctly segment cell images. Since the 1990s, supervised learning methods using training data have become increasingly popular in the aspect to medical image analysis. The most reasonable idea is to let the computer learn the data characteristics that best represent the current problem [7]. Deep learning is a computational model similar to human cognitive systems and can be effectively used in different applications [8]. In practice, the original deep learning architecture was an artificial neural network with many hidden layers. Convolutional neural networks (CNNs) are the form of deep learning that suit medical image processing very well [9]–[11]. CNNs composed of many layers, in the process of converting input data to output data, learns more and more advanced data features. The CNN proposed by Krizhevsky and Sutskever [12], namely AlexNet, has attained favorable results. In the following years, the research on deeper convolutional network architecture has gained

significant progress [13]. Research fields in computer vision, deep convolutional networks have become the first technology considered [14]. In medical image analysis, the application research on deep learning methods was first presented at conference, since then, it has developed rapidly. Among some new CNN architectures related to medical image analysis, the most well-known is the U-net, proposed by Ronneberger *et al.* [15]. The main innovation of the U-net structure is the integration by an equal number of downsampling and upsampling layers. This way of connecting the features in the reduction and expansion paths can process the entire image forward and directly generate a segmentation map. Some researchers have proposed new methods depend on the U-net structure. For instance, Drozdal *et al.* proposed a method that added a short-hop connection structure similar to ResNet on the basis of the U-net architecture [16]. According to the U-net structure, Milletari *et al.* performed a 3D segmentation method utilizing a 3D convolution layer with Dice coefficient as the objective function, which is called V-net [17]. Chen *et al.* combined a bidirectional LSTM-RNN with a 2D U-net structure to construct a segmentation structure to segment anisotropic 3D electron microscope images [18]. Deep learning-based methods work by automatically learning increasingly complex functional layers directly from the data. Ideally, the deeper the CNN's architecture design, the better the results. Therefore, the research focuses on designing the architecture and improving the hierarchy.

Inspired by the pioneering work on deep CNNs by [15], [19], [20], we propose an improved U-net algorithm for medical cell image segmentation. One of the latest trends in the design of convolutional neural networks is increasing accuracy and efficiency. Medical image processing must be more accurate. The fashion of a deep convolution kernel applied to each individual channel can reduce the calculation cost. When designing deep convolutional neural networks, an important factor, kernel size, is often overlooked. Although the 3×3 convolution kernel stack is often simply used to replace large convolution kernels, recent research results show that larger kernel sizes may improve the accuracy and efficiency of the method [19]. We utilize convolution kernels of different sizes to perform parallel transformation to form convolution blocks, in this paper. The original U_Net layer convolution is replaced with a block of convolutions of different sizes stacked in parallel. We research the application of the batch normalization method proposed by Ioffe and Szegedy [21] in a pre-processing step that aims to improve the model's convergence speed, and alleviate the "gradient dispersion" problem in deep networks to a certain extent, so that the training of deep network models is easier and more stable. We also investigate the width multiplier and resolution multiplier proposed by Howard *et al.* [22]. The width multiplier is used to control the size of the input and output channels, the resolution multiplier is employed controlling the resolution of the input. The accuracy of the model is improved without significantly increasing the magnitude of the parameters. This article has made some contributions,

which are mainly reflected in the following three aspects:

1) The operation of mixing multiple convolution kernels of different sizes in one convolution block can utilize different kernel sizes to overcome the limitation of a single convolution kernel size in traditional deep convolution. In medical image segmentation, not only large convolution kernels are required to capture high-resolution features, but also small convolution kernels are required to capture low-resolution features to obtain better model accuracy and efficiency.

2) In the mixed convolution blocks, two hyper-parameters (Width multiplier and Resolution multiplier) are used to quickly adjust the model to fit a specific environment. Width multiplier is used to control the number of input and output channels, so that the size of the model can be controlled. Resolution multiplier is used to change the internal structure of the convolutional layer. The performance of the mixed convolutional layer is optimized by changing the proportion of convolution kernels with different resolutions.

3) A residual path can improve the efficiency of network training. The connections within the residual unit and between the high and low layers of the network will help the information spread, which makes it possible to design neural networks with fewer parameters. Thus, better performance can be achieved in medical image segmentation.

The rest of this article is composed as follows: We review related work on medical image segmentation methods in Section II. The proposed approach is introduced in Section III. Detailed description of the database used in the experiment and the evaluation methods in Section IV. We carry out analyzing and discussing for the experimental results in Section V. At last, we summarize the full text in Section VI.

II. PREVIOUS RELATED WORK

Due to the large differences in cell image data (different morphologies, cell densities, cell types), automatic cell segmentation is still considered a very challenging problem [23]. Currently, many algorithms have been proposed to solve the problem of cell segmentation. Traditional cell segmentation methods generally use basic image processing techniques, such as intensity thresholding [24], morphological filtering [25], watershed [26], active contour [27], and graph cut [28]. These methods generally require manual adjustment of parameters to optimize their performance.

Convolutional neural networks have recently been used for semantic-based image segmentation tasks, which assign a class label to each pixel. The early application of CNN was a patch-based method in biomedical image processing, classifying pixels with the help of their neighborhood attributes to solve segmentation problems such as neural cells [29], [30]. Patch-based methods can produce highly localized output. Such methods are less efficient due to redundant calculations of the patch overlap. In addition, positioning accuracy and availability of contextual information are greatly affected by the choice of patch size. In the current, CNN-based

image segmentation methods have become increasingly popular. Long *et al.* [31] proposes a groundbreaking full convolutional network (FCN) that enables pixel-level and end-to-end semantic segmentation. Ronneberger *et al.* [15] combined upsampling and downsampling layers of the network by the corresponding skip connection and proposed the U-net algorithm, which greatly improved the segmentation effect. From a training perspective, the entire image can be processed forward through U-net to directly generate a segmentation map. This allows U-net to make full use of the entire contextual information of the image, which has a clear advantage over patch-based CNNs. Therefore, many extended papers based on U-net have appeared. For example, Milletari *et al.* [17] proposed a U-net-based 3D structure called V-net, which performs 3D image segmentation using 3D convolution based on the objective function of Dice coefficients. Drozdal *et al.* [16] used the short-hop connection similar to ResNet in addition to long-hop connections in the U-net. Gu *et al.* [32] use pretrained ResNet block as the fixed feature extractor and propose a DAC block and RMP block to capture more high-level features and preserve more spatial information. Fu *et al.* [33] proposed the M-Net that mainly consists of multi-scale input layer, U-shape convolutional network, side-output layer, and multi-label loss function. Yi *et al.* [34] proposed an attentive cell instance segmentation method that builds on a joint network that combines a U-net and a single shot multi-box detector. Ramesh and Tasdizen [35] uses multi-task learning in combination with the similarity interface to detect and segment cells in microscopy images. He *et al.* [36], proposed Mask R-CNN algorithm that is very flexible and can be used to complete a variety of tasks, including target classification, target detection, semantic segmentation, instance segmentation, human pose recognition, and other tasks. It has very good scalability and ease of use. Compared with traditional methods, these CNN-based segmentation methods show superior performance and have been widely used in medical and biological image processing. For example, abdominal aortic thrombosis [37], histological cell segmentation [38] and multimodal biomedical image segmentation [20]. In this paper, we use an improved U-net-based algorithm for accurate cell segmentation.

III. METHOD

When designing deep learning network architecture with a deep convolution kernel, an important factor is the kernel size, but it is often overlooked. Although many approaches replace larger convolution kernels with continuous 3×3 convolution kernels, recent research results show that larger kernel sizes, such as 5×5 convolution kernels and 7×7 convolution kernels, may increase model accuracy and efficiency. To overcome the limitation of the single kernel size, we need both to capture low-resolution modes by a small convolution kernel and capture high-resolution modes by a large convolution kernel in order to obtain better model accuracy and efficiency [19]. Based on the above analysis,

we propose an improved method called the U_net algorithm based on mixed convolution blocks (McbUnet). Our model is an improved multi-scale deep network based on U_net. It uses different sizes of convolution kernels to obtain different resolution modes to complete cell image segmentation. This scheme is shown in Fig.1. Our model was originally based on the architecture proposed by [15], [20], but it made some structural improvements. First, we increased the depth and width of the original convolutional network model (more convolutional layers and parallel convolution kernels of different sizes). Second, referring to the paper [22], we introduced a width multiplier α and a resolution multiplier ρ in two types of convolution blocks to adjust the size of the convolution blocks. Third, in the convolution block, according to the analysis of the paper [19], we used convolution kernels at four different scales, 3×3 , 5×5 , 7×7 and 1×1 in parallel. Through this modification, the network scale is controlled to a certain extent and performance is improved.

The following concepts are the important aspects involved in this cell segmentation model:

A. ARCHITECTURE OF McbUnet METHOD

To understand our McbUnet method, Fig.1 shows the entire structure of the network. Our method is a deep network of convolution blocks based on two multi-class convolution kernels. The two convolution blocks are illustrated in Fig.2 and Fig.3 respectively. Small kernels can save computing costs, while large kernels can improve accuracy. Larger mixed convolutional deep networks tend to use larger kernels and more layers in pursuit of higher accuracy, but require more parameters. In addition, we should also notice that the deep convolution of oversized kernels will seriously reduce the accuracy.

The network structure of the McbUnet method is shown in the Fig.1, and is mainly composed of the encoding path on the left, the decoding path on the right and residual paths. In addition, we use 1×1 simple convolution to do linear mapping transformation to ensure that the dimensions of the “add” layer are consistent at the dotted line. The McbUnet method combines the advantages of U-Net and residual neural networks. Because the features in the encoder are relatively lower level feature, while the corresponding features in the decoder are relatively higher level, there is a semantic gap between them [20]. It is not advisable to splice the two parts directly. In this paper, inspired by the deep residual learning [39], [40], we utilize residual paths corresponding to different numbers of mixed convolution block II. The encoding path consists of the composite convolution block I (The structure is shown in Fig.2.) and a 2×2 max pooling layer according to a stride of 2 for completing the downsampling. During downsampling process, the number of channels is actually doubled. The decoding path is composed of an upsampling to halve the number of channels, a combination with the relevant feature map of the residual path, and the convolution block I. The residual paths corresponding to different layers consist respectively of corresponding number

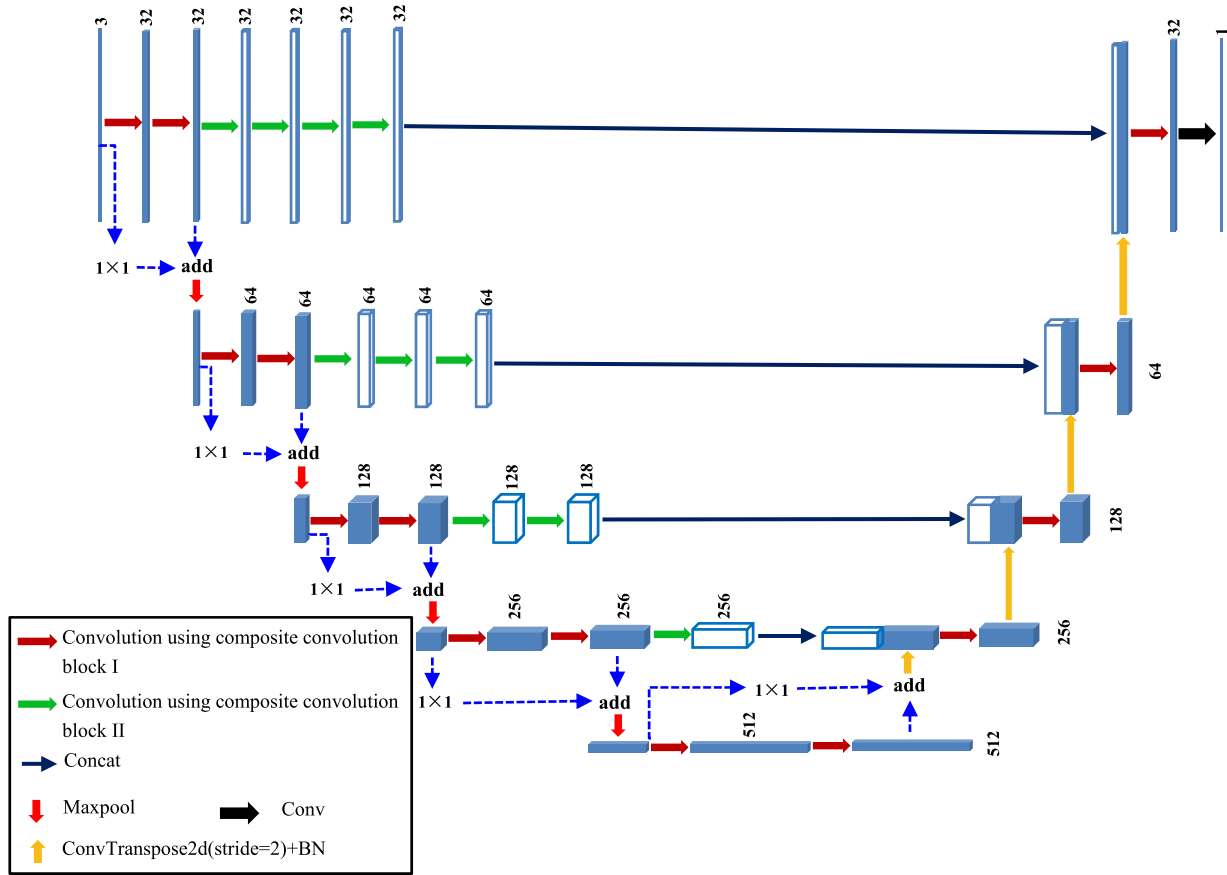


FIGURE 1. Structure of the proposed method.

of composite convolution block II (The structure is displayed in Fig.3.). In the end, each 32-dimensional feature vector is projected to the required class number using a 1×1 convolution.

B. THE IMPORTANT ASPECTS OF THE MCBUNET METHOD

1) PARAMETER INITIALIZATION

a: WIDTH MULTIPLIER α [19]

Different sizes of convolution kernels have different effects on the segmentation results. Through experimental verification, it can be known that a smaller width multiplier is allocated to a larger convolution kernel, and a larger width multiplier corresponds to a smaller convolution kernel. This not only improves the segmentation accuracy, but also effectively reduces the amount of calculations. Therefore, in the convolution block I, the width multiplier assigned to the 3×3 , 5×5 , 7×7 convolution kernels are 0.5, 0.3 and 0.2, respectively. In the convolution block II, the width multiplier assigned to the 3×3 , 5×5 convolution kernels are 0.6 and 0.4, respectively. They have been marked in Fig.2 and Fig.3.

b: RESOLUTION MULTIPLIER ρ [19]

Considering the size of the learning network, ρ is specified as 1 in convolution block I and convolution block II.

c: BATCH NORMALIZATION

Deep learning can be considered as a feature learning process performed layer by layer. The output of each layer is equivalent to the extracted data features of this layer. The batch normalization means [21], [41] is practically implementing data normalization for each network layer. However, in each layer, the calculation overhead to normalize all data is really very great. Therefore, it is possible to refer to the same way using the minimum batch gradient descent, a small batch data is sampled, for this batch data, the output of each network layer is normalized. Let Y_{ij}^n denote the output value of the *i*-th neuron model in the *j*-th layer for the *n*-th data in a certain batch data. μ_{ij} denotes the average of the output value for this data batch at the *i*-th neuron in the *j*-th layer. σ_{ij} indicates the standard deviation of the output value for this data batch at the *i*-th neuron in the *j*-th layer. After batch normalization, the output value is displayed as follow:

$$Y'_{ij} = \frac{Y_{ij}^n - \mu_{ij}}{\sigma_{ij}} \tag{1}$$

where the average output of the neuron is:

$$\mu_{ij} = \frac{1}{m} \sum_{n=1}^m Y_{ij}^n \tag{2}$$

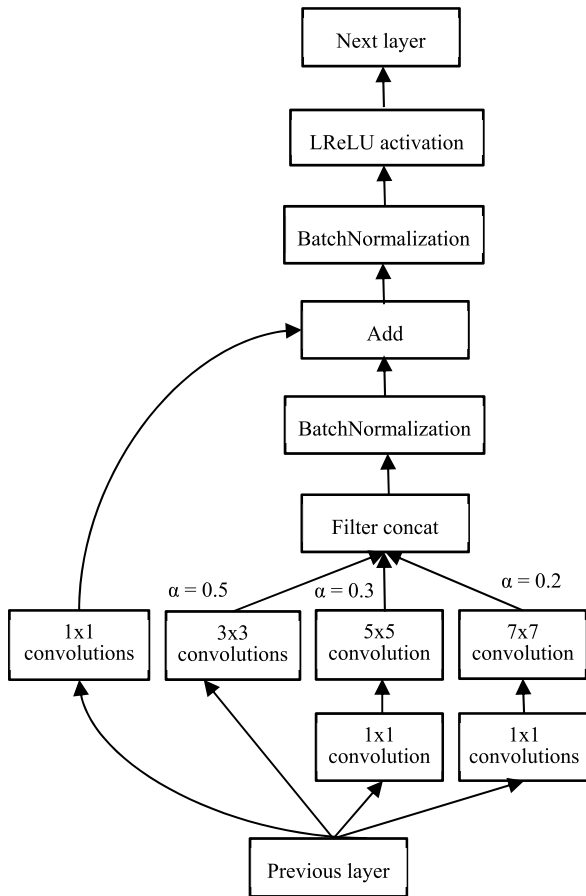


FIGURE 2. Schematic diagram of composite convolution block I of the proposed improved algorithm.

Here, m is the number of data in a batch data. The standard deviation of the output value at the neuron is:

$$\sigma_{ij} = \sqrt{\varepsilon + \frac{1}{m} \sum_{n=1}^m (Y_{ij}^n - \mu_{ij})^2} \quad (3)$$

Here, ε represents a very small number. The intent is to avoid the denominator being zero.

Batch normalization is actually adjusting the input data of each layer in the neural network so that its average value is zero and its variance is 1. In this way, the activations and gradients will be kept at a controlled level, otherwise the gradient of backpropagation may disappear.

2) ACTIVATION FUNCTION

The essence of deep learning is to characterize the internal complex structural characteristics of the problem and perform arbitrary nonlinear transformations between neuron input and output. The formula for rectifier linear units (ReLU) is as follows:

$$h(x) = \max(0, x) \quad (4)$$

Krizhevsky *et al.* [12] found that the convergence speed of SGD using ReLU is much faster than the hyperbolic tangent, or classical sigmoid functions, and better results is

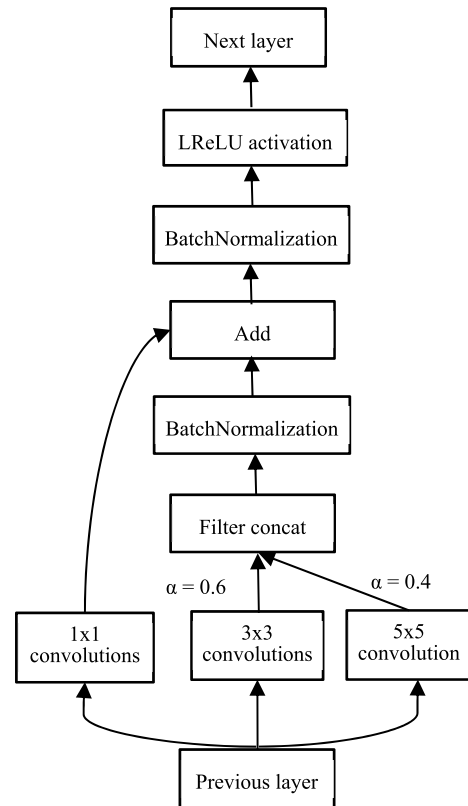


FIGURE 3. Schematic diagram of composite convolution block II of the proposed improved algorithm.

achieved. However, setting the constant to 0 may damage the gradient change and affect the adjustment of the weights [42]. Here, to address these limitations, we refer to the method in [42], [43] and introduce a small slope in the negative part of the function. The specific method is as follows:

$$h(x) = \max(0, x) + \beta \min(0, x) \quad (5)$$

where β is the leakage parameter. Equation (5) is called leaky rectifier linear unit (LReLU).

3) MAXPOOL

The pooling operation combines spatially adjacent features in the feature map. This combination makes the representation more compact and constant for small image changes [44]. Therefore, applying the pooling layer can decrease the model size, increase the calculating speed, and enhance the feature robustness. The effect of the maximum pooling operation is that as long as a feature is extracted in any quadrant, it will remain in the maximized pooling output.

4) LOSS FUNCTION

The main purpose of the loss function is to measure the degree of inconsistency between the true value Y and the predicted value $f(x, W)$ and to use it to optimize the model parameters. It can be expressed as $L(Y, f(x, W))$. Here, W represents the parameters. Generally, the smaller the value of the loss function, the better the robustness of the model. Given a set

of samples $(x_i, y_i), i = 1, 2, \dots, N$. Where x_i denotes the i -th pixel and y_i represents the label corresponding to the i -th pixel. Then the loss function of this set of samples can be defined as follows:

$$L(W) = \frac{1}{N} \sum_{i=1}^N L_i(y_i, f(x_i, W)) + \gamma \varphi(W) \quad (6)$$

where, the previous mean is a data loss function, which is used to match the model's predicted value with the actual value. The latter part is the regularization term to avoid overfitting. r is the parameter that weights the second loss component relative to the first component, and $L(W)$ represents the loss function.

U-Net optimizes the model by using pixel-wise cross-entropy as a loss function. Let $h_\theta(x)$ be the predicted value. $g(z)$ is the sigmoid function. Then, we can get the following expression:

$$g(z) = \frac{1}{1 + e^{-z}} \quad (7)$$

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}} \quad (8)$$

In deep learning networks, the sigmoid function is the activation function in the final output layer during binary classification.

Standard form of log loss function:

$$L(Y, P(Y|X)) = -\log P(Y|X) \quad (9)$$

The objective of the loss function $L(Y, P(Y|X))$ is to maximize the probability $P(Y|X)$ when the sample X is classified Y . Because the log function is monotonically increasing, $\log P(Y|X)$ also reaches its maximum value. Therefore, with a negative sign in front, maximizing $P(Y|X)$ is equivalent to minimizing $L(Y, P(Y|X))$. To unify the category labels y to 1 and 0, the $P(Y = y|x)$ expression for logistic regression is as follows:

$$P(Y = y|x) = \begin{cases} h_\theta(x) & \text{if } y = 1 \\ 1 - h_\theta(x) & \text{if } y = 0 \end{cases} \quad (10)$$

Taken together, it can be denoted as follows:

$$P(y|x; \theta) = (h_\theta(x))^y (1 - h_\theta(x))^{1-y} \quad (11)$$

Take the likelihood function as:

$$L(\theta) = \prod_{i=1}^N P(y_i|x_i; \theta) = \prod_{i=1}^N (h_\theta(x_i))^{y_i} (1 - h_\theta(x_i))^{1-y_i} \quad (12)$$

The log-likelihood function can be written as:

$$l(\theta) = \log L(\theta) = \sum_{i=1}^N (y_i \log h_\theta(x_i) + (1 - y_i) \log(1 - h_\theta(x_i))) \quad (13)$$

Maximum likelihood estimation is to find the value of θ when $l(\theta)$ is maximized. To use gradient descent to find the optimal solution, take $J(\theta)$ as:

$$J(\theta) = -\frac{1}{N} l(\theta) = -\frac{1}{N} \left[\sum_{i=1}^N (y_i \log h_\theta(x_i) + (1 - y_i) \log(1 - h_\theta(x_i))) \right] \quad (14)$$

When $y = 1$, this sample is assumed to be positive. If the predicted value $h_\theta(x) = 1$ at this time, the loss for this sample alone is 0. This means that the predictions for this sample are completely accurate. Then if all the samples are accurate, the total loss = 0. But if the predicted value $h_\theta(x) = 0$ at this time, in this case, since the sample is a positive sample at this time, but the predicted result $P(y = 1|x; \theta) = 0$, that is, the probability of predicting $y = 1$ is 0. Then a great punishment term is attached to the loss function. In the same way, when $y = 0$, the same derivation is also used, which is not repeated here. The logic loss function is to convert the predicted value into a probability value through sigmoid function, and then calculate it through the likelihood loss function.

5) FINE-TUNING

In addition, for the convolution framework, we added 1×1 convolutions in block I to decrease the number of parameters, thereby reducing the amount of calculation. Since the batch normalization and activation function LReLU are added after each convolution, the nonlinearity is also added. We use the Adam optimization algorithm here [45], which can iteratively renew the weights of the neural network according to the training data. The learning rate of the traditional stochastic gradient descent (SGD) is manually set and cannot be changed during the entire training process. Only a single learning rate can be used to update the weight. Adam's algorithm is distinct from the SGD. Adam is an adaptive learning rate that is designed to use the first and second moment estimators for calculating gradients and can vary according to different parameters.

C. TIME COMPLEXITY ANALYSIS OF THE MCBUNET METHOD

Here, we simply analyze the complexity of the model proposed in this paper. The time complexity of a single convolutional layer is:

$$Time \sim O(M^2 \cdot K^2 \cdot C_{in} \cdot C_{out}) \quad (15)$$

where K is the side length of each convolution kernel, C_{in} is the number of channels of each convolution kernel, that is, the number of input channels. C_{out} is the number of convolution kernels in this convolution layer, that is, the number of output channels. M is the side length of the output feature map of each convolution kernel. The length M of the output feature map is determined by the four parameters of the input image

size X , the convolution kernel size K , Padding and Stride, which can be expressed as follows:

$$M = \frac{X - K + 2 * \text{Padding}}{\text{Stride}} + 1 \quad (16)$$

It can be seen that the time complexity of each convolutional layer is determined by the output feature map area M^2 , convolution kernel area K^2 , input C_{in} and output channel number C_{out} . The time complexity of the entire model is mainly determined by the cumulative results of the time complexity of all convolutional layers.

In the composite convolution block I of the proposed improved algorithm, we constructed four parallel convolutional blocks of different sizes (as shown in Figure 2), effectively increasing the width of the network. But doing so also increases the time complexity of the network. The countermeasure is to add 1×1 convolution to reduce the number of input channels to a lower value, and then perform real convolution.

On the 1×1 convolution branch, the time complexity is:

$$(M^2 \cdot 1^2 \cdot C_{in} \cdot C_{out})$$

On the 3×3 convolution branch, the time complexity is:

$$(M^2 \cdot 3^2 \cdot C_{in} \cdot C_{out}) * 0.5$$

On the 5×5 convolution branch, the time complexity is:

$$(M^2 \cdot 1^2 \cdot C_{in} \cdot C_{out}^{-1}) + (M^2 \cdot 5^2 \cdot C_{out}^{-1} \cdot C_{out}) * 0.3$$

Here, C_{out}^{-1} are the number of output channels of the 1×1 convolution of the added down channel.

On the 7×7 convolution branch, the time complexity is:

$$(M^2 \cdot 1^2 \cdot C_{in} \cdot C_{out}^{-1}) + (M^2 \cdot 7^2 \cdot C_{out}^{-1} \cdot C_{out}) * 0.2$$

We add up the time complexity of these four branches and multiply it by resolution multiplier 1.1 to get the time complexity of mixed convolution block I. We use a similar method to obtain the time complexity of the composite convolution block II of the proposed improved algorithm. Furthermore, the time complexity of the entire model can be obtained. When the shape of the input image in cell segmentation is (256, 256, 3), the time complexity of the entire model is about 1.16×10^7 .

IV. EXPERIMENTAL SETUP

A. DATABASE

The proposed method was validated on the 2018 Data Science Bowl. The data set is composed of many segmented images. The images are obtained under various conditions and differ in imaging methods (fluorescence and brightfield), cell type, and magnification. This data set requires higher abstraction ability of the algorithm to generalize these changes. For each image, there is a corresponding ImageId. The training set contains the original images and the segmented image of each nucleus in the image. For the test set, there are only the original images. Among them, the test set_1 has 65 images

and contains 11 types of resolution images, the training set has 670 images and contains 9 types of resolution images. The original image is divided into a grayscale image and a color image. (Although they are all 3 or 4 channels, some of the images have the same value for multiple channels, which is actually a grayscale image.). Each image in the training set corresponds to multiple masks, that is, there will be multiple cell nuclei in one image.

B. PARAMETER SELECTION

Some basic parameters in the proposed deep learning network architecture are shown Table. 1. The width multiplier and the resolution multiplier are specified with the help of experiments in Table. 2.

TABLE 1. Hyperparameters of the proposed method.

Stage	Parameter	Value
Train	epochs	40
	Batch	10
	Learning rate (lr)	0.001
	early stopping	10
LReLU	β	0.33

C. EVALUATION

To evaluate the segmentation effect of the algorithm, we use four evaluation indicators. The Sensitivity, Positive Predictive Value (PPV), Accuracy, and Intersection Over Union (IOU) were calculated. Their calculation formula is shown below:

Sensitivity represents the ratio of the number of pixels that are correctly judged positive to the number of all positive pixels. The definition is as follows:

$$\text{Sensitivity (SEN)} = \frac{TP}{TP + FN} \quad (17)$$

PPV represents the ratio of true positives in positive pixels. It is defined as:

$$PPV = \frac{TP}{TP + FP} \quad (18)$$

Accuracy indicates the percentage of the pixels that are correctly evaluated against the total number of pixels, defined as:

$$\text{Accuracy (ACC)} = \frac{TP + TN}{TP + FP + TN + FN} \quad (19)$$

where, TP, TN, FP and FN are respectively the numbers of true positive, true negative, false positive and false negative.

TABLE 2. Hyperparameters selection experiment of the proposed method.

Experiment	Stage	ρ	α	Val_accuracy (%)
1	convolution block I	1	3x3:0.5; 5x5:0.3; 7x7:0.2	97.49
	convolution block II	1	3x3:0.6; 5x5:0.4;	
2	convolution block I	1	3x3:0.2; 5x5:0.3; 7x7:0.5	97.43
	convolution block II	1	3x3:0.4; 5x5:0.6;	
3	convolution block I	0.9	3x3:0.5; 5x5:0.3; 7x7:0.2	97.44
	convolution block II	0.9	3x3:0.6; 5x5:0.4;	
4	convolution block I	1.1	3x3:0.5; 5x5:0.3; 7x7:0.2	97.57
	convolution block II	1.1	3x3:0.6; 5x5:0.4;	
5	convolution block I	1.3	3x3:0.5; 5x5:0.3; 7x7:0.2	97.51
	convolution block II	1.3	3x3:0.6; 5x5:0.4;	

The IOU indicator is the ratio of intersection over union, which has been used as a standard measure in semantic segmentation. The calculation formula is:

$$IOU = \frac{target \cap prediction}{target \cup prediction} \quad (20)$$

V. EXPERIMENTS AND RESULT ANALYSIS

We will analyze the capability of the proposed method in this section. In addition, we compared the method of this article with the latest medical image segmentation technology, namely CE-Net [32] and MultiResUNet [20] on the same database.

A. SELECTION OF WIDTH MULTIPLIER AND RESOLUTION MULTIPLIER

Here, we choose the values of the width multiplier and the resolution multiplier parameters through experiments on the training set. For this reason, we consider the combined value of these two parameters and choose the combination with the best performance. The specific results are shown in Table. 2.

According to the results of the possible combinations of different parameters in Table.2, the parameter combination that produces the highest val_accuracy is selected. Large convolution kernels capture high-resolution modes, and small convolution kernels capture low-resolution modes. It can be known from experiments that, for specific databases,

constructing mixed convolution blocks with appropriate proportions of convolution kernels of different sizes can obtain better model accuracy and efficiency. As the resolution factor decreases, the number of parameters reduces significantly, so the amount of calculation also goes down quickly. Considering the segmentation precision and calculation amount, we choose the parameter value of Experiment 4 in Table.2. Fig.4 shows the variation of the accuracy value and loss function for the five methods that are used to select model parameters in the training and validation set. Experiment 1 stopped training after the 30th epoch because EarlyStopping reached the patience value.

B. ALGORITHM VERIFICATION

We confirm the usefulness of the new method by studying the improvement of the segmentation performance of the McbUnet method through 10-fold cross-validation. We use the metrics proposed earlier, namely the Sensitivity, Positive Predictive Value (PPV), IOU indicator and Accuracy, to evaluate the average gain of performance. The comparison of our approach with state-of-the-art methods on the 2018 Data Science Bowl is shown in Table. 3. and Fig.5.

As shown in Table. 3, the compound convolution kernel is composed of a large convolution kernel and a small kernel, which can produce better results, due to the performance of the larger neighborhood in the cell image.

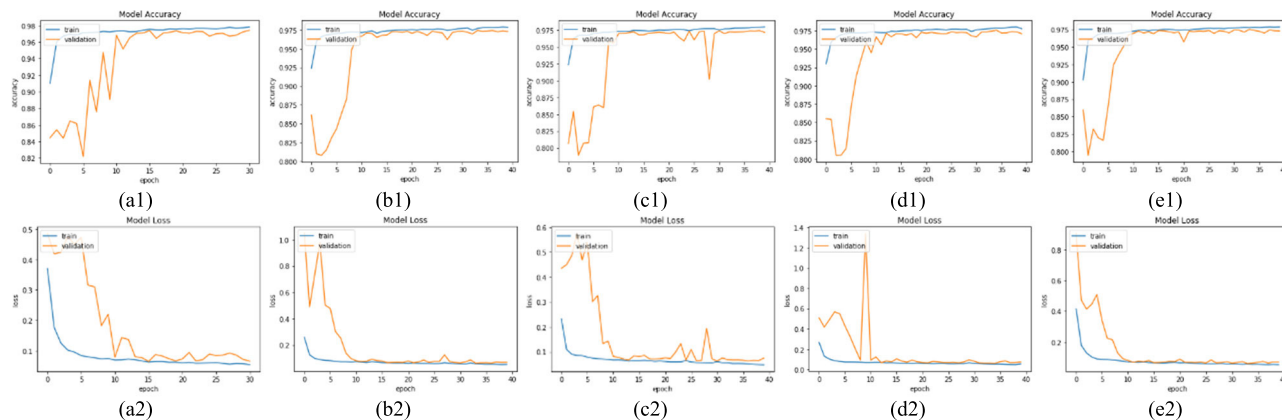


FIGURE 4. Diagram for training and verifying the accuracy and learning curve of nucleus segmentation. The five columns a1 and a2, b1 and b2, c1 and c2, d1 and d2, e1 and e2 in the figure are the results of experiments 1, 2, 3, 4 and 5 in Table 2, respectively.

TABLE 3. The experimental results of verifying the standard U_net, multiresunet, CE-NET and the McbUnet method respectively for 40 epochs.

Model	ACC (%)	PPV (%)	SEN (%)	IOU (%)
standard U_net	97.04	89.48	91.26	82.27
MultiResUNet(Nabil Ibtehaz et al., 2020)	97.42	89.69	92.71	83.85
CE-Net (Zaiwang Gu et al., 2019)	97.44	90.14	92.62	83.96
Ours-the McbUnet method	97.57	90.23	92.66	84.79

Fig.5 showed nucleus image segmentation results. For illustrative purposes, we have shown the original images, ground truth images, and 10-fold cross-validation segmentation images of five nucleus images for the fourth methods.

C. QUALITATIVE ANALYSIS EXPERIMENT IN TEST SET

We assess the property of the proposed architecture through the test set_1 of the 2018 Data Science Bowl. Since there are only the original images in the test set and no segmented contrast image. Therefore, we can only show the qualitative segmentation results as shown in Figure 6.

Figure 6 qualitatively analyzes the effect of nucleus segmentation in the test set_1. The result shows our method yields best performance over other competitors in general, especially for the nuclei in smaller size.

The above experimental results show that the proposed algorithm has good performance as a whole. The MultiResUNet [20] and CE-Net [32] methods showed slightly worse results. One possible reason is that they use continuous 3 × 3 convolution kernels to replace larger-scale convolution kernels, which can reduce the number of parameters to a certain extent, but ignore the limitation of the size of a single kernel. In the segmentation algorithm, both large

TABLE 4. Quantitative results of the proposed method for image segmentation in ISIC-2017 dataset.

Model	ACC (%)	IOU (%)
standard U_net	95.23	79.57
MultiResUNet(Nabil Ibtehaz et al., 2020)	95.51	80.71
CE-Net (Zaiwang Gu et al., 2019)	95.62	81.34
Ours-the McbUnet method	95.69	81.62

convolution kernels are needed to capture high-resolution modes, and small convolution kernels are required to capture low-resolution modes to obtain better model accuracy and efficiency. The main reason why our proposed method has better performance is that mixing different convolution kernels can improve efficiency and accuracy. Convolution kernels of different scales are mixed together, and larger size kernels can obtain more stable accuracy.

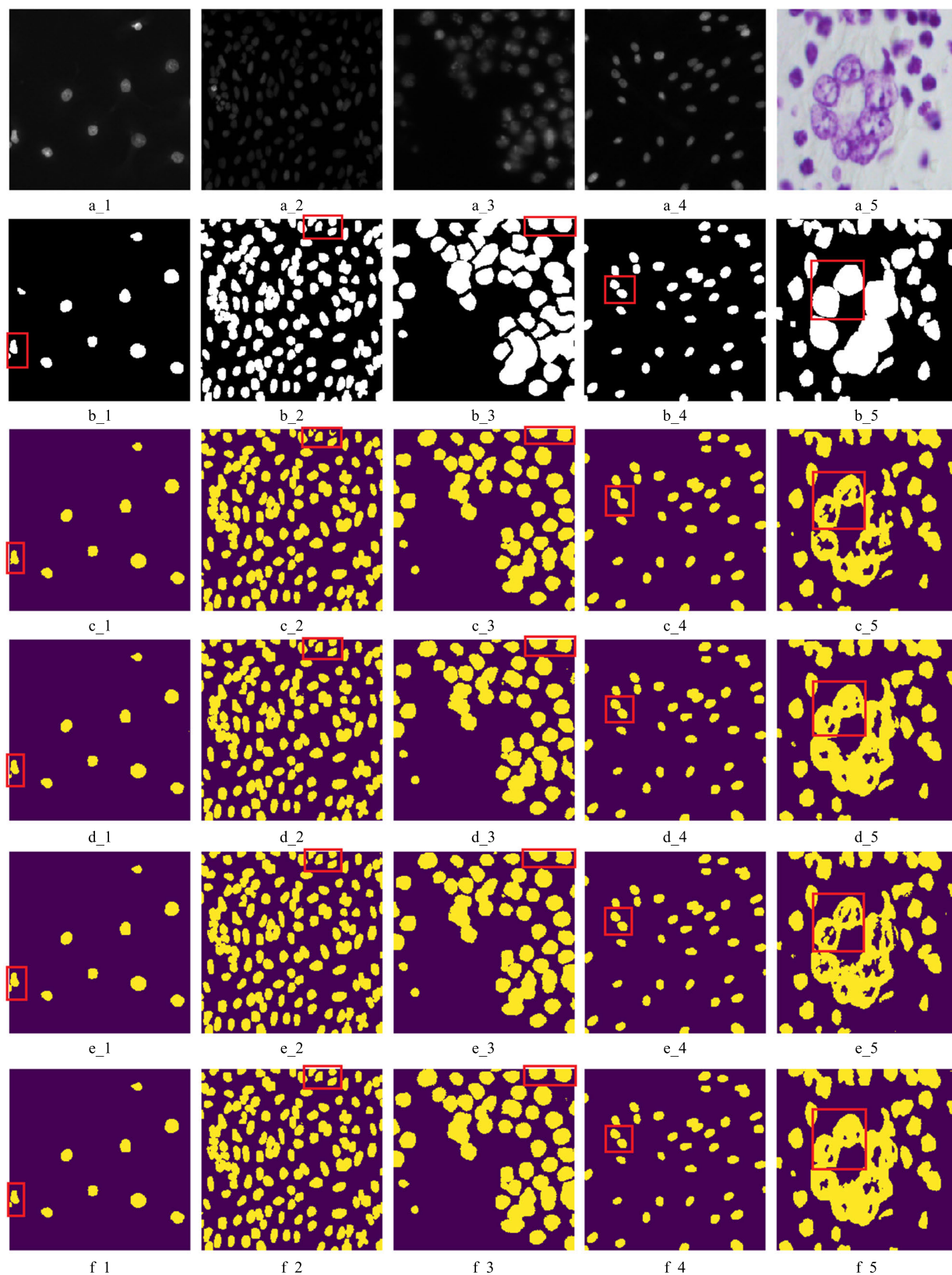


FIGURE 5. Examples of segmentations show the effect of the proposed method. The first line images (a_1-a_5) are different original images. The second line images (b_1-b_5) represent the ground-truth cell nucleus obtained by pathology and verified by image processing methods. The third line images (c_1-c_5) show the segmentation result images using the U_net method. The fourth line images (d_1-d_5) show the segmentation result images using the MultiResUNet [20] method. The fifth line images (e_1-e_5) show the segmentation result images using the CE-Net [32] method. they look like there are some errors in their segmentation. In contrast, the method proposed in this paper can better identify and reject those outliers, as shown in the sixth line images (f_1-f_5).

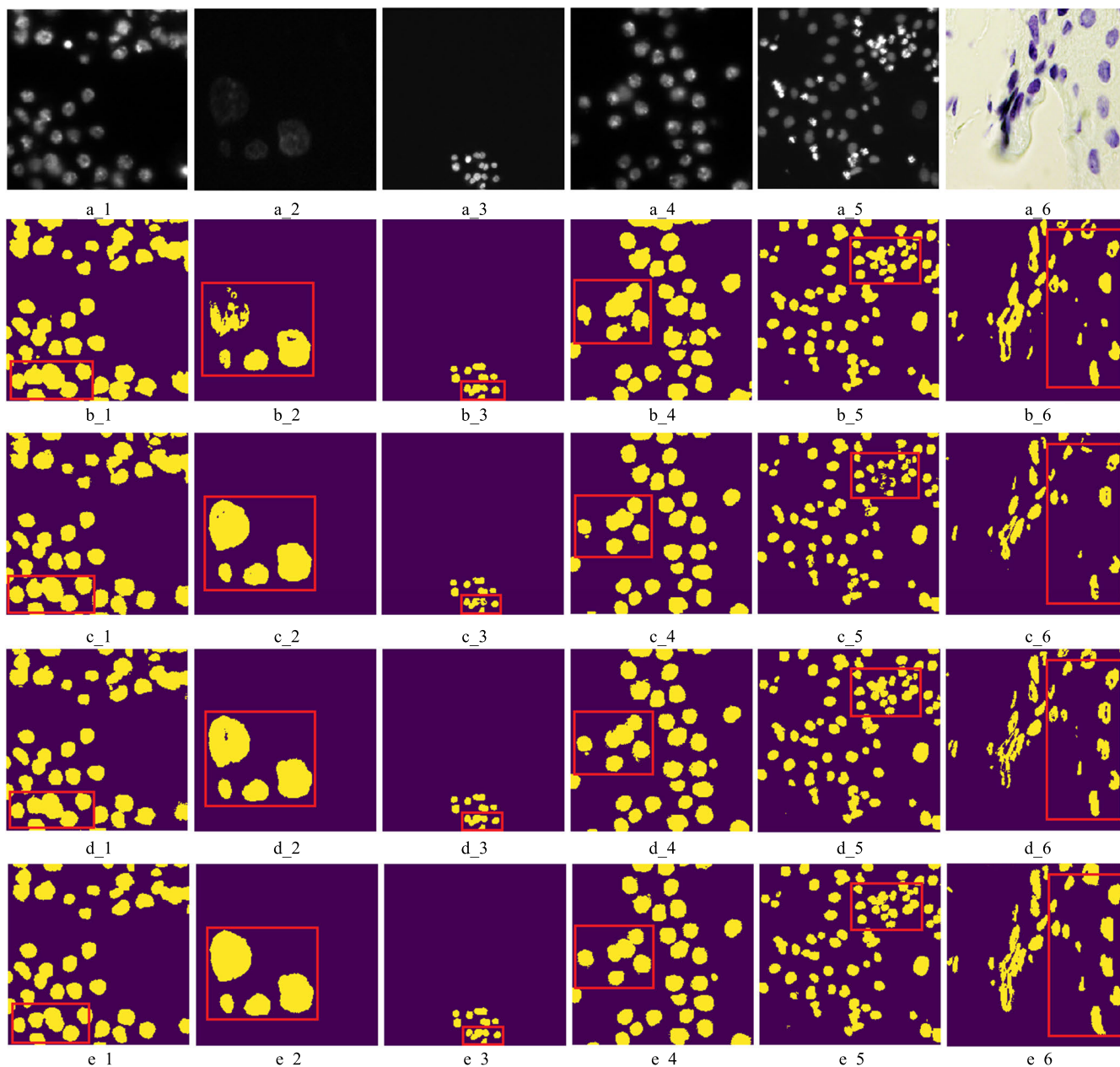


FIGURE 6. Qualitative examples of segmentations show the effect of the proposed method in the test set_1. The first line images (a_1-a_6) are different original images. The second line images (b_1-b_6) show the segmentation result images using the U_net method. The third line images (c_1-c_6) show the segmentation result images using the MultiResUNet [20] method. The fourth line images (d_1-d_6) show the segmentation result images using the CE-Net [32] method. they look like there are some errors in their segmentation. In contrast, the method proposed in this paper can better identify and reject those outliers, as shown in the fifth line images (e_1-e_6).

D. APPLICATION TO OTHER DATASETS

In order to verify the generalization ability of the method proposed in the paper, we apply this method to the public ISIC-2017 dataset. Here, it is used in our extended experiments to show that the method proposed in this paper has a certain generalization ability and is suitable for different types of image data. In the experiment, we divided the training datasets (2000 images) into two parts: the training

set (1800 images) and the validation set (200 images). We use the same architecture settings as cell segmentation and select two evaluation indicators, IOU and Accuracy. The quantitative and qualitative results of the experiments on the validation set are respectively shown in Table. 4 and Fig.7. It can be seen that the method proposed in the paper can be applied to other biomedical image segmentation tasks.

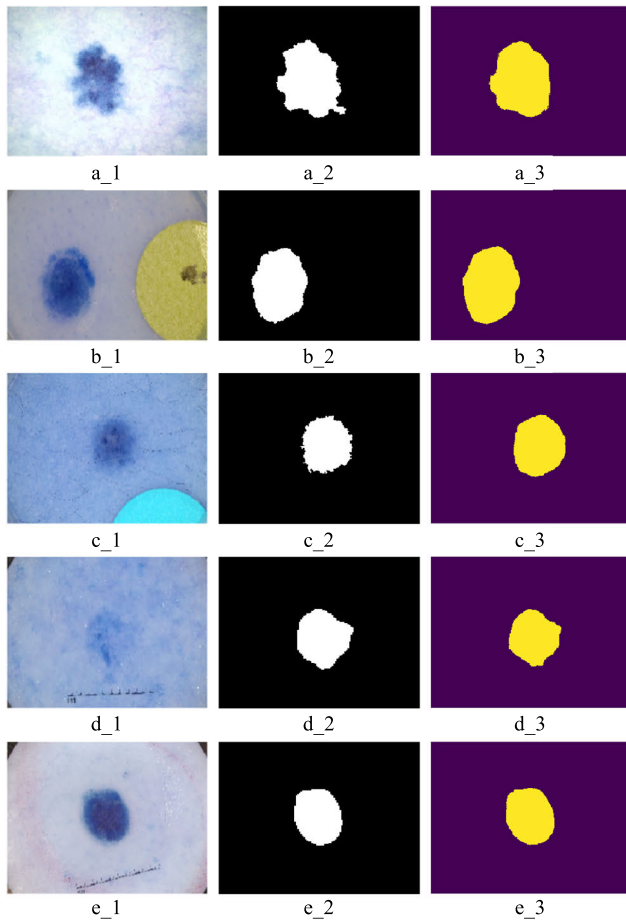


FIGURE 7. Qualitative results of image segmentation by the method proposed in this paper. The first column images (a_1-e_1) are different original images. The second column images (a_2-e_2) represent the ground-truth in the dataset ISIC2017. The method proposed in this paper shows good segmentation effect, as shown in the third column images (a_3-e_3).

VI. CONCLUSION

This letter has proposed a McbUnet method for cell nuclei images segmentation. First, our proposed method uses convolution kernels of different sizes in the same layer, and adds a 1×1 convolution before the large convolution kernel convolution. In the convolution block, we introduced two hyperparameters to quickly adjust the model to adapt to a specific environment, and selected the more appropriate parameters by measuring the impact of the parameters on the model performance and scale. Resolution multiplier is used to control the resolution of the input, which can quickly reduce the number of parameters and the amount of calculation, and can easily control the model size. Width multiplier is used to control the number of input and output channels. In this paper, it is used to adjust the proportion of different kernel convolutions, thereby improving the segmentation accuracy of the model. When designing a convolutional neural network with a deep convolution kernel, an important factor that is often overlooked is the kernel size. Although the common practice is to simply use 3×3 convolution kernels, research

results show that larger kernel sizes, such as 5×5 convolution kernels and 7×7 convolution kernels, will improve the accuracy and efficiency of the model. We need both large convolution kernels to capture high-resolution modes and small convolution kernels to capture low-resolution modes to obtain a better image segmentation model. Here, we use mixed depth convolution, fuse convolution kernels of different sizes in a convolution block, and employ Width multiplier to adjust the proportion of different convolution kernels, so that it can easily capture different resolution modes. This improves the adaptability of this method to different resolution features. Second, we use the activation function LReLU to replace ReLU for more efficient performance. In the neural network, some nonlinear factors need to be introduced to better solve complex problems, and the activation function can just help the neural network to do this. The ReLU activation function forces the output result of the input value x less than 0 to become 0, so that the network has sparse expression ability, which can effectively alleviate the occurrence of overfitting. However, sparseness will reduce the effective capacity of the model and may cause “necrosis”. The LReLU function is a modification of the ReLU function. When the input value x is less than 0, the output is no longer forced to 0, but the output ax (a is a small constant), so that negative activation can be propagated in CNNs, thereby improving the efficiency of feature learning. Third, our learning model introduces residuals to solve the problem of deep networks that are difficult to train and have poor effects. After the introduction of residuals, the mapping becomes more sensitive to the output changes, and the output changes have a greater effect on the adjustment of the weights, so the segmentation effect of the model is better. The idea of the residual is to remove the same main part, so as to highlight the small changes, and the residual structure can be easily adjusted to a better effect. Our model structure has multiple bypass branches to connect the input to the later layers, so that the latter layers can learn the residuals well. This solves the problems of information loss in the traditional convolutional layer or fully connected layer when information is transmitted. At the same time, the integrity of information is protected, and the entire network only needs to learn the part of the difference between input and output, simplifying the learning objectives and difficulty. In addition, we use 1×1 simple convolution to do linear mapping transformation to ensure that the dimensions of the “add” layer are consistent. Again, our model has been fine-tuned for better performance, for example, our model employs Batch Normalization to make the model more robust and prevent overfitting. In summary, our proposed method combines the advantages of the residual learning and the U-Net. The path between the corresponding layers of the encoding and decoding of the network is connected by block II, so that useful information is propagated forward and backward in the calculation. We assessed the usefulness of the McbUnet method in the 2018 Data Science Bowl and the public ISIC-2017 dataset. The algorithm in this paper has high segmentation accuracy.

Deep convolution is becoming more and more popular in modern efficient convolutional neural networks, but its kernel size is often ignored. In this paper, by increasing the size of the kernel, such as 5×5 convolution kernel and 7×7 convolution kernel, on the one hand, the accuracy and efficiency of the model are improved, but on the other hand, the parameters are also increased. To this end, width multiplier and resolution multiplier are introduced to adjust the complexity of the model. The shortcoming of this paper is that the large convolution kernel acquires high-resolution features with more details at the cost of adding more parameters and computation. The comparison of the number of training parameters among the four methods is that the method in this article is the most, followed by the MultiResUNet [20] method, and again the CE-Net [32] methods, and the least is the standard Unet. Therefore, the algorithm proposed in this paper is the most complex, which is also the weakness of this method. This is also the direction of our next research. In the future, we will strengthen research and exploration in the following aspects: 1) We start by improving the structure of the convolutional layer and conduct a study to reduce the number of convolutional layer parameters. 2) Starting from the shape of the convolution kernel, we explore whether the deformed convolution kernel can be used to analyze only the image area of interest, so that the recognized features will be more conducive to improving the segmentation performance. 3) We study the loss function and find a loss function that is more suitable for image segmentation using deep learning.

CONFLICTS OF INTEREST

The authors declare that there are no conflicts of interest regarding the publication of this paper.

REFERENCES

- [1] B. S. Deshmukh and V. H. Mankar, "Segmentation of microscopic images: A survey," in *Proc. Int. Conf. Electron. Syst., Signal Process. Comput. Technol.*, Jan. 2014, pp. 4–362.
- [2] Z. Liu, J. Wang, G. Liu, and L. Zhang, "Discriminative low-rank preserving projection for dimensionality reduction," *Appl. Soft Comput.*, vol. 85, Dec. 2019, Art. no. 105768.
- [3] Y. Chen, K. Biddell, A. Sun, P. A. Relue, and J. D. Johnson, "An automatic cell counting method for optical image," in *Proc. BMES/EMBS*, Atlanta, GA, USA, 1999, p. 819.
- [4] R. M. Haralick and L. G. Shapiro, "Survey: Image segmentation techniques," *Comput. Vis., Graphic, Image Process.*, vol. 29, no. 1, pp. 100–132, 1985.
- [5] K. Wu, D. Gauthier, and M. D. Levine, "Live cell image segmentation," *IEEE Trans. Biomed. Eng.*, vol. 42, no. 1, pp. 1–12, Jan. 1995.
- [6] F. Ambriz-Colin, M. Torres-Cisneros, J. G. Avina-Cervantes, J. E. Saavedra-Martinez, O. Debeir, and J. J. Sanchez-Mondragon, "Detection of biological cells in phase-contrast microscopy images," in *Proc. 5th Mex. Int. Conf. Artif. Intell.*, Nov. 2006, pp. 68–77.
- [7] Z. Liu, Z. Lai, W. Ou, K. Zhang, and R. Zheng, "Structured optimal graph based sparse feature extraction for semi-supervised learning," *Signal Process.*, vol. 170, May 2020, Art. no. 107456.
- [8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
- [9] P. Kainz, M. Pfeiffer, and M. Urschler, "Segmentation and classification of colon glands with deep convolutional neural networks and total variation regularization," *PeerJ*, vol. 5, p. e3874, Oct. 2017.
- [10] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P.-M. Jodoin, and H. Larochelle, "Brain tumor segmentation with deep neural networks," *Med. Image Anal.*, vol. 35, pp. 18–31, Jan. 2017.
- [11] Y. Xu, Y. Li, M. Liu, Y. Wang, M. Lai, I. Eric, and C. Chang, "Gland instance segmentation by deep multichannel side supervision," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Athens, Greece: Springer, 2016, pp. 496–504.
- [12] A. Krizhevsky and I. H. G. Sutskever, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 1–42, 2014.
- [14] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.
- [15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput.-Assist. Intervent.*, in Lecture Notes in Computer Science, vol. 9351, pp. 234–241, doi: 10.1007/978-3-319-24574-4_28.
- [16] M. Drozdal, E. Vorontsov, G. Chartrand, S. Kadoury, and C. Pal, "The importance of skip connections in biomedical image segmentation," in *Proc. Deep Learn. Med. Image Anal. (DLMIA)*, in Lecture Notes in Computer Science, vol. 10008, 2016, pp. 179–187.
- [17] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," 2016, *arXiv:1606.04797*. [Online]. Available: <http://arxiv.org/abs/1606.04797>
- [18] J. Chen, L. Yang, Y. Zhang, M. Alber, and D. Z. Chen, "Combining fully convolutional and recurrent neural networks for 3D biomedical image segmentation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3036–3044.
- [19] M. Tan and V. Q. Le, "MixConv: Mixed depthwise convolutional kernels," in *Proc. BMVC*, 2019, pp. 1–13.
- [20] N. Ibtchaz and M. S. Rahman, "MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural Netw.*, vol. 121, pp. 74–87, Jan. 2020.
- [21] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: <http://arxiv.org/abs/1502.03167>
- [22] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: <http://arxiv.org/abs/1704.04861>
- [23] Q. Wu, F. A. Merchant, and K. R. Castleman, *Microscope Image Processing*. Burlington, MA, USA: Academic, 2008.
- [24] P. Sankaran and V. K. Asari, "Adaptive thresholding based cell segmentation for cell-destruction activity verification," in *Proc. 35th IEEE Appl. Imag. Pattern Recognit. Workshop (AIPR)*, Oct. 2006, p. 14.
- [25] C. Di Rubeto, A. Dempster, S. Khan, and B. Jarra, "Segmentation of blood images using morphological operators," in *Proc. 15th Int. Conf. Pattern Recognit.*, vol. 3, 2000, pp. 397–400.
- [26] C. F. Koyuncu, S. Arslan, I. Durmaz, R. Cetin-Atalay, and C. Gunduz-Demir, "Smart markers for watershed-based cell segmentation," *PLoS ONE*, vol. 7, no. 11, Nov. 2012, Art. no. e48664.
- [27] P. Wu, J. Yi, G. Zhao, Z. Huang, B. Qiu, and D. Gao, "Active contour-based cell segmentation during freezing and its application in cryopreservation," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 1, pp. 284–295, Jan. 2015.
- [28] R. Bensch and O. Ronneberger, "Cell segmentation and tracking in phase contrast images using graph cut with asymmetric boundary costs," in *Proc. IEEE 12th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2015, pp. 1220–1223.
- [29] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," in *Proc. Conf. Neural Inf. Process. Syst.*, 2012, pp. 2843–2851.
- [30] D. Ciresan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2013, pp. 411–418.
- [31] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
- [32] Z. Gu, J. Cheng, H. Fu, K. Zhou, H. Hao, Y. Zhao, T. Zhang, S. Gao, and J. Liu, "CE-Net: Context encoder network for 2D medical image segmentation," *IEEE Trans. Med. Imag.*, vol. 38, no. 10, pp. 2281–2292, Oct. 2019.

- [33] H. Fu, J. Cheng, Y. Xu, D. W. K. Wong, J. Liu, and X. Cao, "Joint optic disc and cup segmentation based on multi-label deep network and polar transformation," *IEEE Trans. Med. Imag.*, vol. 37, no. 7, pp. 1597–1605, Jul. 2018.
- [34] J. Yi, P. Wu, M. Jiang, Q. Huang, D. J. Hoepfner, and D. N. Metaxas, "Attentive neural cell instance segmentation," *Med. Image Anal.*, vol. 55, pp. 228–240, Jul. 2019.
- [35] N. Ramesh and T. Tasdizen, "Cell segmentation using a similarity interface with a multi-task convolutional neural network," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 4, pp. 1457–1468, Jul. 2019.
- [36] K. He, G. Gkioxari, P. Dollár, R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2980–2988.
- [37] K. López-Linares, N. Aranjuelo, L. Kabongo, G. Maclair, N. Lete, M. Ceresa, A. García-Familiar, I. Macía, and M. A. González Ballester, "Fully automatic detection and segmentation of abdominal aortic thrombus in post-operative CTA images using deep convolutional neural networks," *Med. Image Anal.*, vol. 46, pp. 202–214, May 2018.
- [38] H. Chen, X. Qi, L. Yu, Q. Dou, J. Qin, and P.-A. Heng, "DCAN: Deep contour-aware networks for object instance segmentation from histology images," *Med. Image Anal.*, vol. 36, pp. 135–146, Feb. 2017.
- [39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Jun. 2016, pp. 770–778.
- [40] Z. Zhang, Q. Liu, and Y. Wang, "Road extraction by deep residual U-Net," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 749–753, May 2018.
- [41] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. Int. Conf. Artif. Intell. Stat.*, 2010, pp. 249–256.
- [42] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, vol. 30, 2013, pp. 1152–1160.
- [43] S. Pereira, A. Pinto, V. Alves, and C. A. Silva, "Brain tumor segmentation using convolutional neural networks in MRI images," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1240–1251, May 2016.
- [44] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.
- [45] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, San Diego, CA, USA, 2015, pp. 1–15.



CHUANBO HUANG received the B.S. degree in computer engineering from the PLA the Artillery Academy, Hefei, China, in 1995, the M.S. degree in computer software and theory from the PLA Information Engineering University, Zhengzhou, China, in 2005, and the Ph.D. degree in control science and engineering from the Nanjing University of Science and Technology, Nanjing, China, in 2011. He is currently an Associate Professor with the Department of Computer Science, Jining University, Qufu, China. His current research interests include image processing, image modeling, segmentation, and tissue characterization, with a particular interest in applications to medical image processing.



HUALI DING received the B.S. degree in computer engineering from Qufu Normal University, in 1998, and the M.S. degree in computer software and theory from Nankai University, in 2010. He is currently a Lecturer with the Department of Computer Science, Jining University. His research interests include image processing and computer vision.



CHUANLING LIU received the Ph.D. degree in computer applied technology from the Nanjing University of Science and Technology, Nanjing, China, in 2012. He is currently an Associate Professor of computer applied technology with Shangqiu Normal University, Shangqiu, China. His current research interests include pattern recognition, face recognition, image processing, and scene matching.

• • •