IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 2, 2020, accepted July 5, 2020, date of publication July 10, 2020, date of current version July 23, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3008401*

# High Accuracy Thyroid Tumor Image Recognition Based on Hybrid Multiple Models Optimization

**WANRONG GU[1], YIJUN MAO[1], YICHEN HE[1], ZAOQING LIANG[1], XIANFEN XIE[2], ZIYE ZHANG[3], AND WEIJIANG FAN[1]**

[1]College of Mathematics and Informatics, South China Agricultural University, Guangzhou 520642, China
[2]School of Economy, Jinan University, Guangzhou 520632, China
[3]School of Mathematical, South China University of Technology, Guangzhou 520641, China

Corresponding author: Yijun Mao (yijunmao@163.com)

This work was supported in part by the Natural Science Foundation Project of Guangdong Province under Grant 2018A030313437, in part by the Ministry of Education Humanities and Social Sciences Project under Grant 18YJCZH037, in part by the Philosophy and Social Science of Guangdong Province under Grant GD18CXW01 and Grant GD19CGL34, in part by the National Key Research and Development Program under Grant 2017YFC1601701, and in part by the National Statistical Science Research Key Project under Grant 2019LZ37.

**ABSTRACT** With the development of computer vision recognition technology, more and more researchers apply this technology to the recognition of tumor images. But for cost reasons, many hospitals still use low-cost ultrasound and other cheap equipment, resulting in ambiguity, artifacts and many similar tumor noise areas images. Recent related studies have high precision in clear image recognition, such as face and number recognition in color images. However, they showed low accuracy and unstable results in ultrasonic image due to its fuzziness. The straight reason may be that many existing algorithms are not suitable for the fuzzy and noise image, and easily to misjudge the real objects and noise areas. In this paper, we proposed an approach based on R-CNN and RPN to distinguish the real objects from noise areas after data enhancement and morphological filtering with the optimization of a series of hyper-parameters and the combination of Color Doppler Flow Imaging (CDFI) blood flow signal model. We found that the key features of high-noise ultrasound images can be obtained quickly and accurately. From the experimental results, the accuracy and stability of our proposed method outperforms the state-of-the-art approaches on the real world thyroid tumor image data set.

**INDEX TERMS** Deep learning, tumor image recognition, hybrid multiple models.

## I. INTRODUCTION

Doppler ultrasound is a common technique for the medical diagnosis. Compared with other imaging techniques, it has the advantages of real-time imaging, low cost, no need for biopsy and no damage for patient. Despite these advantages, there are some problems in thyroid neoplasm ultrasound images, such as severe spots, high noise level, low resolution and contrast, and various tumor shapes. The above disadvantage factors lead to low accuracy and instability in the automatic tumor identification process. In order to overcome the problem of tumor recognition in multi-noise images, many studies have been proposed recently [1]. Most of them focus on noise reduction and image sharpening rather than integrating them and analyzing to fit a specific symptom, physiological feature, and reach an appropriate level. Besides, most of the recent studies still used the traditional feature selection methods, such as mutual information quantity and Principal Components Analysis (PCA), and the effective information is lost greatly. In some literatures, deep learning method has been proposed for feature mining, and some results have been achieved, but some feature extraction algorithms do not work well in medical image processing. Especially, they have not been adapted to the tumor ultrasound image. The accuracy and stability are still insufficient. In the real medical treatment common, the high misjudgment rate will delay the medical treatment.

Consider with the above background, there are at least two main problems on the tumor image recognition research: 1) The input data is an ultrasonic image with low quality. In other words, the image is fuzzy and contains much noise or with blurring edges of the shape. 2) Mainstream feature

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

selection techniques, such as deep learning and machine learning, are less adaptive to specific real world thyroid ultrasound images. To solve these two problems, this paper proposes to use data enhancement, augmentation and morphological filtering to ensure the data quality that can be used efficiently for feature extraction, and then use Regions with CNN features (R-CNN) and Region Proposal Network (RPN) multi-layer depth network to fully exploit the potential features of the mass of the image. In order to make the thyroid ultrasound image can be better and more accurate to identify the location of the tumor; we presented High Accuracy Thyroid Tumor Image Recognition Based on Hybrid Multiple Models Optimization. This model mainly uses the mainstream deep learning network to scan and identify the tumor location, and around how to improve the recognition accuracy, a series of super parameter optimization such as image enhancement, basic network selection, iteration times and learning rate are carried out to achieve a higher recognition standard. In addition, the recognition of CDFI blood flow signal is added to judge the benign and malignant tumor. In this way, we can mine differences between the tumor region and the noise region, and realize multi model fusion optimization. And then, our algorithm can lock the tumor region more accurately and reduce the false rate. Due to the high accuracy and efficiency of the proposed method, the tumor region can be quickly locked, and it can also be applied to dynamic video thyroid neoplasm recognition in the future.

Generally, the contributions of our manuscript are:

- **We proposed a high-precision thyroid tumor image recognition method based on hybrid multiple models optimization.** We build a thyroid tumor feature extraction framework by approximate joint training method and stochastic gradient descent optimization algorithm. The problems of poor quality of thyroid tumor ultrasound image or the existence of artifacts that seriously disturb the training results are solved.
- **We carried out experimental work on regional extraction and regional detection of real-world tumor ultrasound images and CDFI blood flow signal images.** In the candidate region judged by Fast R-CNN, the time of traditional candidate region extraction is reduced, the network convergence time is also reduced by the use of approximate joint training method, and different convolution networks, data set partitioning forms, learning rates and other factors are compared and analyzed.
- **We analyze the influence of different convolution network, data set partition form, learning rate and other factors on the model.** We find that since the bottom layer of CNN focuses on the extraction of image edge information and contour information, and the higher layer is the abstraction of image semantic information and essential information, the deep layer CNN can better represent the semantic features of the image, with better accuracy. The uniform ratio of the training set and the verification set can prevent the model from over-fitting and improve the generalization ability of the model.

The remaining of this paper is organized as follows. Section II covers related technology and works relevant to our study, including image preprocessing, feature extraction and tumor recognition. In Section III, we describe the problem statement, data model and algorithm we proposed in this paper. Besides, we show the detail training processing and parameter adjustment in our technology framework. Section IV shows the experiments and results analysis compared to classical models and the state-of-the-art approaches. Finally, we conclude this manuscript and discuss the future work in Section V.

## II. TECHNOLOGY BACKGROUND

In the field of medical image recognition, it contains three main steps:

### A. IMAGE PREPROCESSING

Generally, the input images need to be processed before image recognition [1], such as denoising, contrast enhancement, sharpening, region segmentation and so on. In our study, the ultrasonic image carries much noise information, the image mold and the pre-processing steps are essential. Yu-Len and Dar-Ren [2] proposed a watershed segmentation method for breast tumor, and Gaetano *et al.* [3] studied the segmentation for the remote sensing images recently. Naimi *et al.* [4] focused on the medical image denoising. Some studies proposed feature-reduction for the image segmentation [5]. More and more researches concentrate on the segmentation for the fuzzy medical images [6]. However, the existing pre-processing methods have a good effect on clear color image processing, and have excellent performance in many application fields [7], [8]. The thyroid neoplasm image has only black and white mold and edge, and has more pulse noise interference. Most of the existing methods are not suitable for the application of this paper.

### B. FEATURE EXTRACTION

Morphological and texture features are extracted from the image as the basis for quantifying the appearance of benign and malignant nodules [9]. The morphological features include the tubercle's direction, shape, echo, edge, etc, and used in many applications [10], [11]. Texture features include ultrasound appearance and echo model. Traditional feature extraction mainly includes shape [12], color [13] and texture [12], [14]. In addition, SIFT and HOG [15] are also commonly used in feature extraction of image recognition. Feature extraction can be used in many other research field, some research used deep learning and feature extraction for intrusion detection [16] in IoMT, which had been achieved good result in their application.

### C. CLASSIFICATION AND OBJECT RECOGNITION

Classifier or cluster is used to identify the lesion area in the segmented image. Before classification, we need to screen out

the features with strong classification performance, one is to improve the generalization ability of the classifier, the other is to reduce the computational dimension to avoid the disaster of dimensionality. Because the features of benign and malignant tumors are autocorrelation and interactivity, nonlinear mapping method can be used as classifier or cluster. The commonly used methods include BP, decision tree, FLD, SVM, Bayesian Classifier, SOM etc. However, the traditional machine learning feature extraction method is too experience-dependent. In many new application scenarios, there is not much experience to learn from. Therefore, there will be a big error in feature extraction in this kind of scene.

In order to solve the shortage of traditional machine learning methods in feature mining, a new method of deep learning is proposed, which takes feature as multiple abstractions and automatically adjusts feature weight through feedback to realize automatic feature construction. We can use deep learning algorithm for medical image segmentation and medical image recognition. In medical image segmentation field, different segmentation methods are usually combined with different machine learning or deep learning methods, such as feature learning and ensemble learning for layered retinal vascular segmentation from Mocanu *et al.* [17], dynamic model and deep neural network for tracking left ventricular endocardial ultrasound number. In addition, 3D CNN and fully Convolutional Neural Networks for sematic segmentation are often used for image segmentation. Such as Segmentation of Spine on MR Image by 3D CNN from Korez *et al.* [18] and segmentation of brain and chest muscles on MRI images from Moeskops *et al.* [19]. In classification and recognition field, there are detection of retinal base in diabetic patients by transfer learning from Gulshan *et al.* [20], classification of skin cancer by transfer learning from Esteva *et al.* [21], automatic classification on DCE-MRI by Sparse Autoencoder from Mansanet *et al.* [22] and the detection and evaluation of the grade of knee osteoarthritis using CNN on X-ray image from Antony *et al.* [23]. Reddy *et al.* [24] proposed an ensemble based machine learning model for diabetic retinopathy classification, which can quickly classify different lesions. With the continuous development of computer vision technology, it has also made good progress in assistant diagnosis, as analysis of PAP-Smears and recognition of Cervical Cancer using Deep Belief network and SVM from Bengtsson and Malm [25] and identification of benign and malignant breast cancer by CNN from Kooi *et al.* [26], which is far superior to the traditional computer aided diagnosis method. Besides, deep learning algorithms are often used to judge prostate diseases, breast cancer, lung cancer, cell micronucleus detection, classifier model for intrusion detection [27] and so on.

However, most of the segmentation methods of thyroid tumors cannot realize end-to-end image segmentation, which requires human intervention. Firstly, the Region of Interest (ROI) is marked by the doctors with rich clinical experience in the original ultrasound images. Traditional methods used various edge segmentation algorithms to extract the contour of ROI, which needs a lot of manpower and energy, requiring high clinical experience and personal ability of doctors.

## III. OUR APPROACH

At present, there are two methods for target detection in depth learning image recognition: a two-stage model based on ROI extraction, such as R-CNN, SPP-Net, Fast R-CNN, Faster R-CNN, R-FCN, etc. The other method is the one-stage model without ROI extraction, such as YOLO, SSD, etc. As an essential part of computer vision field, the main part of these image recognition methods is to use CNN for feature learning. In this paper we combined with some traditional visual methods to achieve target detection.

The imaging principle of ultrasonic image is to use ultrasound to penetrate human organs, and then result as a black-and-white image. Generally, there are not only fuzzy, but also vulnerable to interference and easy to form the noise areas. Fig. 1 shows an example of thyroid tumor ultrasound image.

**FIGURE 1.** Ultrasound image of thyroid tumors.

To solve the thyroid cancer image recognition problem, the technical framework presented in this paper is shown in Fig. 2.

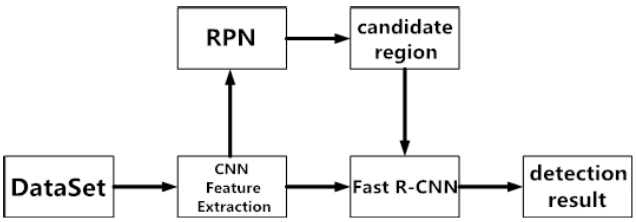



**FIGURE 2.** The structure of thyroid tumor image recognition model based on Faster R-CNN.

Based on the idea of Faster R-CNN, we proposed to construct RPN and Fast R-CNN respectively to extract and detect regions from images. Different convolution networks such as ZF, VGG, ResNet50 are used to extract features. In the training of fast R-CNN, we propose a network-training framework of approximate joint training, which is specifically through the integration RPN and Fast R-CNN into one network. In forward propagation, Fast R-CNN uses the ROI generated by RPN for training, while in backward propagation; the gradient of sharing layer comes from RPN and Fast R-CNN networks. In this way, thought it has ignored the gradient

of the frame coordinates, it can reduce the training time by 25% - 50%, overcoming the problem that Alternate training works too slowly. In the face of huge medical image data set, such as thyroid tumor ultrasound image, this training method has more advantages. The most effective network will be used as the shared network of RPN and Fast R-CNN. The candidate areas generated by RPN are input into Fast R-CNN for Softmax classification and border regression.

### A. CNN MODEL AND ULTRASONIC IMAGE MINING

The basic structure of the convolution neural network is shown in Fig. 3, which mainly includes input layer (original ultrasound image), convolution layer, pool layer, full connection layer and classification layer (softmax layer).

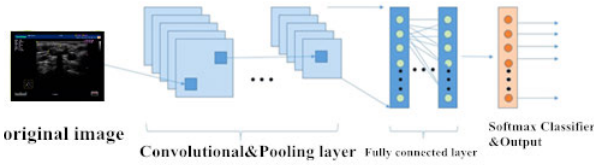



**FIGURE 3.** Convolution neural network structure.

#### 1) CONVOLUTION LAYER

Convolution layer is an important part of convolution neural network, which is responsible for a lot of computing tasks of convolution neural network. Each convolution layer contains multiple convolution kernels (filter), which scan the entire image by translation and sliding. Moreover these cores can be self-learning by forward propagation and backward propagation. The result of image scanning through a convolution kernel is called feature graph or feature map, to represent the response to the convolution kernel input at each space position. The working process can be described as follows:

$$x_j^l = f(\sum_{j \in M_j} x_i^{l-1} * w_{ij}^l + b_j^l) \tag{1}$$

The superscript $l$ represents the $l$-th convolution layer. $i, j$ represent the $i$-th characteristic graph of the $l$-th layer and the $j$-th characteristic graph of the $l-1$ layer. $M_j$ represents all characteristic graphs in the $l-1$ layer connected to the $j$ characteristic graph of layer $l$. $w_{ij}^l$ represents the convolution kernel parameter of the $i$-th feature map of the $l$-th layer corresponding to the $i$-th feature map of the $l$-1th layer. $b_j^l$ denotes the bias of the $j$-th characteristic graph. $F$ is the activation function. $*$ is the convolution operation.

Let a two-dimensional image $I$ be an input, a characteristic diagram obtained after a convolution operation using a two-dimensional convolution kernel $K$ is shown in the following equation:

$$\begin{aligned} S(i,j) &= (I * K)(i,j) \\ &= \sum_m \sum_n I(m,n) K(i-m, j-n) \end{aligned} \tag{2}$$

In general, the structure of the convolution layer can be determined by four superparameters namely the size of the convolution kernel, the number of convolution kernels, the stride size and the filling. Different convolution kernels can extract different features. The length of step determines the length of each movement of the convolution kernel. The filling can extract the edge information of the image. It is based on two principles: local perception and parameter sharing. The dimensionality reduction of the high dimensional input data is realized, and the abstract and deep features of the original data can be extracted.

#### 2) POOLING LAYER

When the image features are extracted by convolution layer, it is usually necessary to insert pool operation between adjacent volume bases. The pooling layer function is a statistical function, which examines a statistical feature of all elements in a subregion of the input data, such as the maximum value, the mean value, the $L2$ norm, and so on. The common pool operations mainly include maximum pool and average pool. The maximum pool takes the maximum value of all elements in the region that the filter slips through, while the average cell takes the average value of all the elements that the filter slips through the region. The pool operation can gradually reduce the size of the feature graph, thus reducing the number of network parameters and computational overhead, preventing over-fitting, while maintaining the invariance of local linear transformation.

#### 3) FULLY CONNECTED LAYER

In the fully connection layer, the two-dimensional feature map of the last convolutional layer is tiled to one-dimensional feature vector as the input of the full connection layer. Output is obtained by weighting summation of input and the activation function.

$$x^l = f(w^l x^{l-1} + b^l) \tag{3}$$

Among them, $x^l$ is the output of layer $l$. $x^{l-1}$ is the output of layer $l-1$. $w^l$ is the weight parameter of layer $l$. $b^l$ is the bias of layer 1.

#### 4) CLASSIFICATION LAYER

Softmax classifier is often used as the classification layer of $CNN$. Compared with logistic classifier, softmax has the function of multi-classification. The form of Softmax function is as follows:

$$\sigma_i(z) = \frac{e^{Z_j}}{\sum_{j=1}^m e^{Z_j}} \tag{4}$$

where $Z_j$ is the input of softmax and $m$ is the number of input neural units. The Softmax function can map an arbitrary real number vector of $m$ dimension to another $m$ dimensional vector with a value range of [0,1], which can be expressed as the probability value of each classification. Cross entropy is usually used as a loss function by Softmax, following is the

expression of it:

$$J(\theta) = -\frac{1}{N}(\sum_{i=1}^{N}\sum_{c=1}^{C} t_{ic} \log \frac{e^{Z_j}}{\sum_{j=1}^{m} e^{Z_j}}) \quad (5)$$

where $N$ is the number of samples. $C$ is the number of classes. $t_{i0} = 1$ if and only if the $i$-th sample belongs to class $c$.

### B. CONSTRUCTING RPN NETWORK

#### 1) RPN STRUCTURE

RPN can input a feature map of any size and output candidate regions corresponding to the original image by anchor mechanism, in which each region contains a class probability and coordinate position. The structure is shown in Fig. 4. Assuming that all networks share the same convolution layer, we first use the classical convolution network (such as ZF, VGG) to extract the features of the image. A special convolution layer is attached to the last convolution layer of the convolution network as the special structure of RPN. We use the window of $n \times n$ (for example $3 \times 3$) to convolution the last layer convolution layer, and then we use $1 \times 1$ convolution to reduce the dimension. It is mapped into two branches, one as softmax classification and the other as bounding box regression. Therefore, we use the idea of faster r-cnn to build the core RPN network of faster r-cnn to build the basic model for the next step of classification regression.

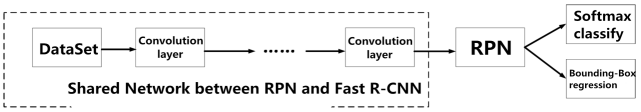



**FIGURE 4.** RPN network structure.

There is a special concept in RPN, called *anchor*, as shown in Fig. 5. Because the target size and the ratio of length to width are different in the sense field corresponding to the sliding window, we need a sliding window of multiple scales. Anchor is the size of a given datum window. Windows of different sizes can be obtained according to multiple and aspect ratio. In this paper, we used three kinds of anchor, of different sizes ($128^2$, $256^2$, $512^2$) and three different aspect ratios (1:1 1:1:2:2:1). That means 9 anchors for each location of the feature graph. Each anchor is mapped to a low-dimensional vector by convolution sliding (for example, ZF is 256-dimensional, VGG is 512-dimensional), and then it is classified and regressed by inputting low-dimensional vectors into two independent fully connected layers.

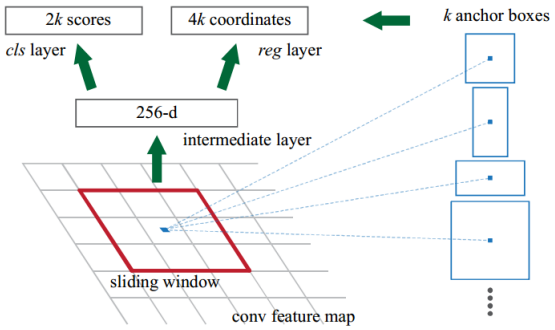



**FIGURE 5.** Anchor in RPN networks.

#### 2) LOSS FUNCTION OF RPN

As Fig. 5 shows, the final two outputs of RPN, one of which is the Softmax classification output, indicate whether the current anchor contains our target object. It is a two-class, and $k$ anchors are used. The final output is a vector of $2k$ in length. The other is the regression output, where each anchor corresponds to a vector of 4 in length $(x,y,w,h)$, where $(x,y)$ is the central coordinate of anchor and $(w,h)$ is the length and width of anchor, so it contains a total of $4k$ outputs.

If the IoU of the anchor box and the ground-truth box is greater than a threshold (set to 0.7 in the experiment), then the anchor contains the target object. The IoU defines the overlap of the two rectangular boxes.

$A$ is a boundary box of anchor and $B$ is a ground-truth frame, then the overlap of a anchor is defined as the proportion of the intersection of $A,B$ to the union of $A,B$, as shown in Eq.(6):

$$IoU = \frac{A \bigcap B}{A \bigcup B} \quad (6)$$

According to the definition above, the multitasking loss function of RPN can be described in the Eq.(7):

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_t L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_t p_i^* L_{reg}(t_i, t_i^*) \quad (7)$$

where $I$ is the subscript of each batch anchor. $p_i$ is the first anchor is the prediction probability of the target class. If $p_i^*$ is the target class, then $p_i^* = 1$, otherwise, $p_i^* = 0$. $t_i$ is a vector of length 4, representing the parameterized coordinates of the predicted border. $t_i^*$ is a vector with a length of 4 that represents the coordinates of the real border. $N_{cls}$ is the size of small batch sampling pairs. $N_{reg}$ is the number of anchor. $\lambda$ which is the weight of the balance loss function is usually set to 10. $L_{cls}$ is the loss function of the category. $L_{reg}$ is the regression loss function. $L_{cls}$ and $L_{reg}$ are showed as Eq.(8) to Eq.(10):

$$L_{cls}(p_i, p_i^*) = \log[p_i^* p_i + (1 - p_i)(1 - p_i^*)] \quad (8)$$

$$L_{reg}(t_i, t_i^*) = smooth_{L1}(t_i - t_i^*) \quad (9)$$

$$smooth_{L1}(x) == \begin{cases} 0.5x^2 & if\,|x| < 1 \\ |x| - 0.5 & if\,|x| \geq 1 \end{cases} \quad (10)$$

For the calculation of $t_i$ and $t_i^*$, see Eq.(11) to Eq.(18):

$$t_x = (x - x_a)/w_a \quad (11)$$

$$t_y = (y - y_a)/h_a \quad (12)$$

$$t_w = \log(w/w_a) \quad (13)$$

$$t_h = \log(h/h_a) \quad (14)$$

$$t_x^* = (x^* - x_a)/w_a \quad (15)$$
$$t_y^* = (y^* - y_a)/h_a \quad (16)$$
$$t_w^* = \log(w^*/w_a) \quad (17)$$
$$t_h^* = \log(h^*/h_a) \quad (18)$$

where $x$, $y$, $w$, $h$ are the central coordinates, length and width of the predicted border. $x^*$, $y^*$, $w^*$, $h^*$ are the real borders, and $x_a$, $y_a$, $w_a$, $h_a$ are the anchor borders. In order to adapt to the actual situation of our data set, the number of anchors and the size of sampling pairs need to reach a relatively balanced level. Therefore, we set the parameters of RPN loss function as follows: $N_{cls} = 256$, $N_{reg} = 2400$, $\lambda = 10$, so that the weight of category loss and regression loss are basically the same.

#### 3) TRAINING METHOD OF RPN

The training of RPN can combine backward propagation and SDG to achieve end-to-end training. Small batches of samples are extracted from an image each time. Each batch of samples includes positive and negative samples. It can bias to negative samples when sampling because the number of negative samples is much larger than the number of positive samples, for example, 256 samples are sampled in a positive-negative ratio of 1 : 1. If the positive sample does not satisfy 128, the negative sample can be used for filling. For a special convolution layer of RPN, a Gaussian distribution with a standard deviation of 0.01 and a mean of 0 can be used to initialize it. Others can be initialized with pre-training models such as ZF, VGG, ResNet that have been trained in large-scale image databases as Pascal VOC or ImageNet and fine-tuned. Due to the limited amount of experimental data, we can not reach the level of large-scale training, so we all use the network pre training model on Imagenet, and then fine tune the model in detail.

### C. CONSTRUCTING FAST R-CNN NETWORK

#### 1) FAST R-CNN STRUCTURE

The main difference between Faster R-CNN and Fast R-CNN is that RPN is used instead of selective search while the detection network remains unchanged. So the Fast R-CNN network still uses ROI Pooling Layer and obtains candidate regions from RPN network, then pools the last convolution layer of shared convolution network according to candidate regions, and then connects the full connection layer to classify and regress. Because the size of ROI extracted by RPN or selective search method is inconsistent, and the input of full connection layer must be fixed, SPP-net uses SPP layer to make the output of fixed dimensions. Fast R-CNN uses a simplified SPP layer, called ROI pooling layer, and uses only one scale sliding window.

#### 2) ROI POOLING LAYER

ROI pooling layer uses maximum pooling operation to transform ROI features into feature maps with fewer dimensions and fixed sizes. The size of it is $H \times W$, with $H$ and $W$ are both hyperparametric needed be set up beforehand. Each ROI is determined by four parameters $r$, $c$, $h$, $w$, where $r$, $c$ is the upper left coordinate and $h$, $w$ is the height and width. ROI can transform ROI of $h \times w$ with size approximation $h/H \times w/W$ as sliding window into a feature map of $H \times W$, so that POI pooling layer can process feature map of any scale. For the customized ROI pooling layer, in order to share the weight, so that the final full connection layer can also continue to be used, the best setting is $(512 \times 7 \times 7)$. When all ROIs are pooled into a $(512 \times 7 \times 7)$ feature map, reshape it into a one-dimensional vector, then we can use the weight of VGG16 pre training to initialize the first two layers of full connection layer. We set the parameters of ROI pooling layer as follows: $H \times W = 7 \times 7$.

#### 3) LOSS FUNCTION OF FAST R-CNN

The loss function is bellow:

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u \geq 1]L_{loc}(t^u, v) \quad (19)$$

where $p = (p_0, p_1, \ldots, p_k)$ represents the predictive probability of each category. $K$ is the number of categories, including $K + 1$ categories. The category 0 is the background. $u$ is the category of the real border. $t^u = (t_x^u, t_y^u, t_w^u, t_h^u)$ is the coordinate of the predictive border of the category $u$. $v = (v_x, v_y, v_w, v_h)$ is the coordinate of the real border of the category $u$. $[u \geq 1]$ is the indicator function, when $u \geq 1$, it is $[u \geq 1] = 1$, otherwise $[u \geq 1] = 0$. $\lambda$ is the weighting parameter. $L_{cls}$ is the loss function of the category, see Eq.(20). $L_{loc}$ is the loss function of border regression, see Eq.(21).

$$L_{cls}(p, u) = -\log p_u \quad (20)$$
$$L_{loc}(t^u, v) = \sum_{i \in x,y,w,h} smooth_{L1}(t_i^u - v_i) \quad (21)$$

We set the parameter of fast R-CNN loss function at $\lambda = 1$ to balance the normalized weight of classification loss and regression loss.

#### 4) TRAINING METHOD OF FAST R-CNN

The pooling layer of ROI can be trained by forward and backward propagation. The calculation process of forward propagation is shown in the Eq.(22)- Eq.(23):

$$y_{rj} = x_{i*(r,j)} \quad (22)$$
$$i^*(r, j) = \arg\max_{i' \in R(r,j)} x_{i'} \quad (23)$$

where $x_i$ is the $i$-th input of ROI pooling layer. $y_{rj}$ is the $j$-th output of the $r$-th ROI. $R(r, j)$ is the input set of pooled subwindows of output unit $y_{rj}$. The pooling layer of ROI calculates the partial derivative of $x_i$ by backward propagation, see Eq.(24:

$$\frac{\partial L}{\partial x_i} = \sum_r \sum_j [i = i^*(r, j)] \frac{\partial L}{\partial y_{rj}} \quad (24)$$

where $[i = i^*(r, j)]$ is the indicator function, if $i = i^*(r, j)$ holds, it is 1, otherwise 0. Because it takes a long time to

calculate the full connection layer, we use Singular Value Decomposition (SVD) to reduce the computation cost. In the optimization algorithm of neural network, we propose to use SGD optimization strategy, which is to calculate the gradient of mini batch every iteration, and then update the parameters. By introducing momentum, SGD can accelerate the operation in a certain direction, suppress the oscillation, and accelerate the convergence, so as to make the model reach the optimization as soon as possible, so as to achieve a stable and efficient effect, overcoming the problem that the model is difficult to converge or miss the optimal solution. In the treatment of tumor images, because of the similarity of features, we think that accelerating convergence will have better effect.

### D. CDFI BLOOD FLOW SIGNAL CLASSIFICATION MODEL BASED ON CNN FOR THYROID TUMORS

CDFI blood flow signal, as an important index for judging benign and malignant tumors in ultrasound, expresses the results of the detection of blood flow by color Doppler ultrasonography. And Pseudo-color coding technology is used to show the direction of blood flow to the back using red and blue, and the depth of color represents the speed of blood flow.

In general, the blood flow of malignant tumors is more abundant than that of benign ones, because when the tumors have blood flow supply, they tend to grow faster. According to the ultrasonographic examination report provided by Sun Yat-sen University Cancer Center, CDFI blood flow signals are classified into three categories as shown in Fig.6.


**FIGURE 6.** CDFI blood flow signals in ultrasound images of thyroid tumors. Red and blue denote the color doppler flow imaging of CDFI. Red is the direction of blood flow toward the probe, and blue is the direction of blood flow away from the probe. Color depth represents the speed of blood flow.

### E. TRAINING

#### 1) INTERVAL TRAINING

RPN and Fast R-CNN networks are initialized by pre-training model, then RPN is trained to generate ROI which will train Fast R-CNN network, and then RPN is initialized by Fast R-CNN network. Unlike RPN initialization at the beginning, the parameters of shared convolution layer are fixed here, only the parameters of RPN are fine-tuned. Then RPN training is used to initialize the Fast R-CNN network, so that RPN and Fast R-CNN network can alternately train. The training effect of this method is better but it takes a long time.

#### 2) APPROXIMATE JOINT TRAINING

This training method tries to integrate RPN and Fast R-CNN network into a mutual network. Fast R-CNN network uses ROI generated by RPN to train in forward propagation. In backward propagation, the gradient of shared layer comes from RPN and Fast R-CNN network. This situation ignores the gradient of the frame coordinates, therefore it is approximate, but it can reduce the cost of time by 25% to 50%. After building RPN network and fast R-CNN network, combined with the reference of efficiency, we choose approximate joint training.

#### 3) TRAINING SETTING

The parameters of the RPN Network Loss Function are set as $N_{cls} = 256$, $N_{reg} = 2400$, $\lambda = 10$ with the ROI Pooling Layer's are set as $H \times W = 7 \times 7$ and the Fast R-CNN Network Loss Function are set as $\lambda = 1$. In the experiment, we use the approximate joint training to train the model, and adopt the stochastic gradient descent optimization algorithm based on momentum. The updating method is shown as follows.

$$v_t = \gamma v_{t-1} + \eta \times g(\theta) \tag{25}$$

$$\theta = \theta - v_t \tag{26}$$

where $\theta$ is the parameter that needs to be updated. $v_t$ is the momentum at the $t$ moment. $\gamma$ is the momentum parameter and $\gamma \in [0, 1]$. $\eta$ is the learning rate, with $g(\theta)$ is the gradient of $\theta$. In the experiment, we will set $\gamma = 0.9$. The initial learning rate $\eta$ will be 0.001, and it will decay 0.0005 per 50000 iteration.

## IV. EXPERIMENTS

### A. EXPERIMENT SETUP

All the sonographic images of thyroid nodules in this paper come from head-neck department of Sun Yat-sen University Cancer Center. These included 16 cases, 368 ultrasound images, including benign and malignant cases. However, because the benign and malignant tumor is not marked, this paper only considers the location of tumor, not the recognition of benign and malignant tumor. 368 ultrasonic images were divided into training set, verification set and test set according to 5 : 2 : 3 scale.

With reference to the format of the VOC2007, the data is packaged into the *Annotations*, *ImageSets* and *JPEGImages* folders. The *Annotations* folder contains the XML files with all the pictures, labels and coordinate information of each image's target, etc., as shown in Fig. 7. The tag < size > records the size of the picture and the number of channels. The tag < name > records the class label of the object. The tag < bndbox > records the position of the objects, where < xmin > and < ymin > are the upper left coordinates of the objects. The <xmax > and < ymax > contain all training pictures for the lower-right coordinates of the objects. The ImageSets file contains four txt files containing the corresponding picture names for the training set, the test set, the validation set and the training-validation set respectively.

*Performance Evaluation Index:* In the depth learning target detection algorithm, Mean Average Precision (mAP) is mainly used as the performance evaluation index of the

```
<?xml version="1.0"?>
- <annotation>
    <folder>JPEGImages</folder>
    <filename>000001.jpg</filename>
    <path>D:\VOCdevkit2007\JPEGImages\000001.jpg</path>
  - <source>
      <database>Unknown</database>
    </source>
  - <size>
      <width>768</width>
      <height>576</height>
      <depth>3</depth>
    </size>
    <segmented>0</segmented>
  - <object>
      <name>doubt</name>
      <pose>Unspecified</pose>
      <truncated>0</truncated>
      <difficult>0</difficult>
    - <bndbox>
        <xmin>306</xmin>
        <ymin>110</ymin>
        <xmax>482</xmax>
        <ymax>207</ymax>
      </bndbox>
    </object>
</annotation>
```


**FIGURE 7.** Sample XML file in the Annotations folder.

detection algorithm. In multi-class object detection, each class draws the Receiver Operating Characteristic Curve (ROC curve) by the rate of recall and precision, AP is the area under the ROC curve, and the mAP is the average value of APs as follows.

$$mAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q} \tag{27}$$

where $Q$ is the number of classes. $AveP(q)$ is the AP value of the $Q$ class. Usually, the value of map is in the range of [0, 1], which indicates the effective of the given model.

### B. EFFICIENCY COMPARISON OF DIFFERENT NETWORK MODELS AND ITERATIONS

According to the above experimental setup, we use three kinds of models (ZF, VGG16, ResNet50) and different iterations to train our models. The amount of experimental data set is so small that we all adopt the pre-training models of network on ImageNet then fine-tune them. The *mAP* of the experimental results is shown in TABLE 1 and Fig. 8.

**TABLE 1.** *mAP*(%) for different iterations and different network models.

| Network Model | Iterations | | |
|---|---|---|---|
| | 1000 | 5000 | 10000 |
| ZF | 0.691 | 0.762 | 0.697 |
| VGG16 | 0.732 | 0.786 | 0.786 |
| ResNet50 | 0.720 | 0.799 | 0.772 |

The effect of VGG16 and ResNet50 is better than that of ZF, because VGG16 and ResNet50 are both large networks. The number of layers is 16 (VGG16) or 50 (ResNet50) but ZF is only 7. The introduction of convolution network model above shows that by increasing the number of network layers the network can extract more information with deeper semantics, thus improving the accuracy of the model.

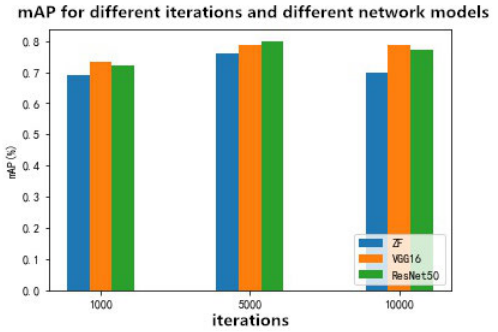



**FIGURE 8.** *mAP* histogram with different iterations and different network models, where the ordinate is *mAP*(%) and the abscissa is three different iterations, which indicate the efficiency difference of ZF, VGG16 and ResNet50 under the iteration times of 1000, 5000 and 10000 respectively.

The experiment also shows that the best result appears when the number of iterations is 5000, but it had been worse when the number of iterations is 10000. The main reason is that the model is overfitting at 10000 times of iteration and the under-fitting of 1000 times of iteration. Because the experimental data is limited, the whole model is easy to converge and the number of iterations should not be too much. From Fig. 9, it can be found that the three networks have basically converged at 3000 times, and then tended to be stable, in which the vgg16 will have a large fluctuation. ResNet50 works best, not only with the smallest Loss, but also with the smoothest. So we use ResNet50 as feature extraction network in Faster R-CNN in the later system. Fig. 9 also shows us how to adjust the training times of the network model later in order to prevent overfitting and reduce time overhead.

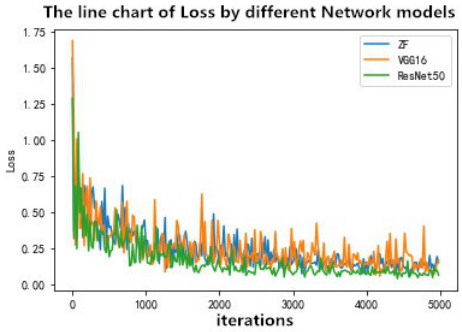



**FIGURE 9.** The line chart of Loss by different Network models, where the ordinates are the value of Loss and the abscissa are iterations in the range of (0, 5000). This indicates the convergence of the three models under the condition that the iterations increase gradually.

From the above, the best results of the three models are only 79.9%, and there is room for improvement. Many factors will affect our results, such as data set partition ratio, number of iterations, learning rate, data augmentation and so on. Next, we mainly improve the model of this paper by dividing the proportion of data sets, data augmentation and learning rate.

### C. EFFICIENCY COMPARISON OF DIFFERENT PARTITION RATIOS OF THE DATASETS

Different partition of dataset will make the proportion of training set and test set different, which will also affect the results of the model. If the training set is much larger than the test set, there may be a fitting, while the training set is smaller or less different than the test set, then there may be under-fitting. We set the number of iterations to 3000. In order to reduce the training time and use the network ZF to experiment, four different proportions are adopted, which are $2.5:2.5:5, 5:2:3, 4:4:2, 4:3:3$. The experimental results are shown in TABLE 2 and Fig. 10.

**TABLE 2.** ***mAP*(%) for different data partitioning ratios.**

| Network Model | Partition ratio (Training Set:Verification set:Test set) | | | |
|---|---|---|---|---|
| | 2.5:2.5:5 | 5:2:3 | 4:4:2 | 4:3:3 |
| ZF | 0.691 | 0.762 | 0.697 | 0.778 |

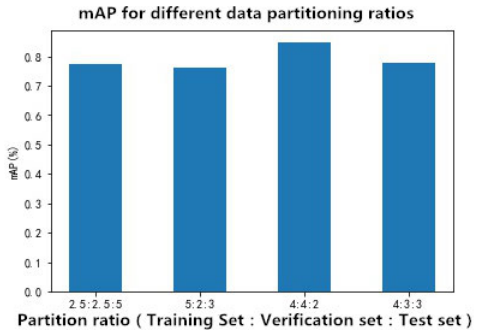



**FIGURE 10.** ***mAP* histogram with different dataset partitioning ratios, where the ordinate is mAP(%) and the abscissa are four different partition ratio of the dataset, which indicate that the ratio of (4 : 4 : 2) has the best effect.**

It is shown that the accuracy rate of $4:4:2$ is increased from 76.2% ($5:2:3$) to 84.8%. And the uniform ratio of the training set to the verification set can prevent the model from overfitting and improve the generalization ability of the model. As shown in Fig. 11, the loss curve of these four partition ratios has little effect on the convergence rate, and they all converge basically at 3000 times to reach a stable state.

### D. EFFICIENCY COMPARISON OF WHETHER USE IMAGE DATA AUGMENTATION

In order to improve the generalization of the model, we use the Data Augmentation mentioned above such as horizontal flipping, vertical flipping, random interception, fixed interception, scale transformation, rotation, color jitter, edge enhancement and so on. Effective Data Augmentation can not only expand the number of training samples, but also enhance the diversity of samples. On the one hand, it can avoid overfitting. On the other hand, it will improve the performance of the model and increase the translation invariance and rotation invariance of images. The robustness of convolution neural network to object scale and direction is also increased.

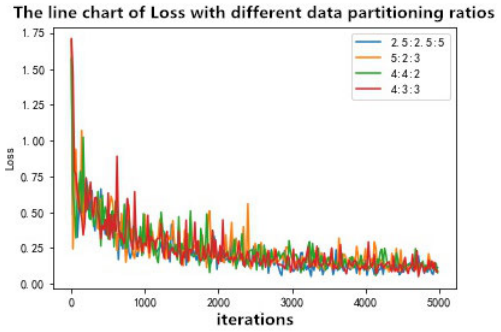



**FIGURE 11.** **The line chart of Loss with different partition ratios of dataset, showing the convergence of four partition ratios. As the iterations increased, Loss became smaller and less volatile.**

LeNet normalized the test image in testing, intercepting four angles and five regions in the center according to the fixed size, flipping the image horizontally, increasing the rotation invariance and translation invariance of the image, and improving the generalization performance of the model. After these operations, the image dataset can be expanded to ten times of the original, and then the voting mechanism is used to select the most suitable category. Using this idea for reference, we could increase the data set by random interception and horizontal inversion, where the size of random interception is $224 \times 224$. In addition, edge enhancement in image processing is used to increase the contrast of ultrasound image, and it is also used as a way of data augmentation. We use the above method to enhance the data set and use ZF in four different data sets. The experimental results are shown as TABLE 3 and Fig. 12.

**TABLE 3.** ***mAP*(%) comparison of data Augmentation with different data set partitioning ratios.**

| Data Augmentation | Partition ratio (Training set:Verification set:Test set) | | | |
|---|---|---|---|---|
| | 2.5:2.5:5 | 5:2:3 | 4:4:2 | 4:3:3 |
| False | 0.775 | 0.782 | 0.848 | 0.778 |
| True | 0.862 | 0.902 | 0.802 | 0.889 |

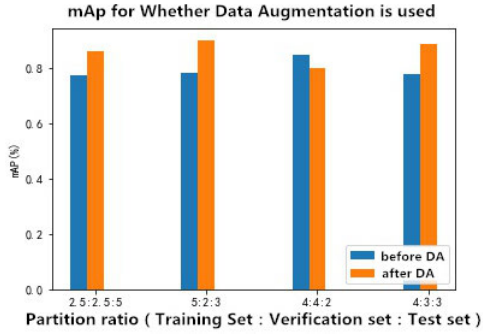



**FIGURE 12.** ***mAP* histogram with whether use Data Augmentation of different partition ratio for datasets, "DA" means Data Augmentation, After data augmentation, the translation invariance and rotation invariance of the image are increased while the number of samples is expanded, and the robustness and accuracy of the model are improved.**

The accuracy rate of the four kinds of partition ratios is better and all above 80%. Significantly, the accuracy of the 5 : 2 : 3 division was increased from 78% to 90%. After data augmentation, the translation invariance and rotation invariance of the image are increased while the number of samples is expanded, and the robustness and accuracy of the model are improved. From Fig. 13, the convergence rate of the loss with four different partition ratios is basically the same after the data augmentation, which is stable at about 3000 iterations. Fig. 14 shows the loss changes of different partitioning ratios before and after data augmentation, and the convergence rate of loss of the four partitioning ratios before and after data augmentation is basically unchanged.

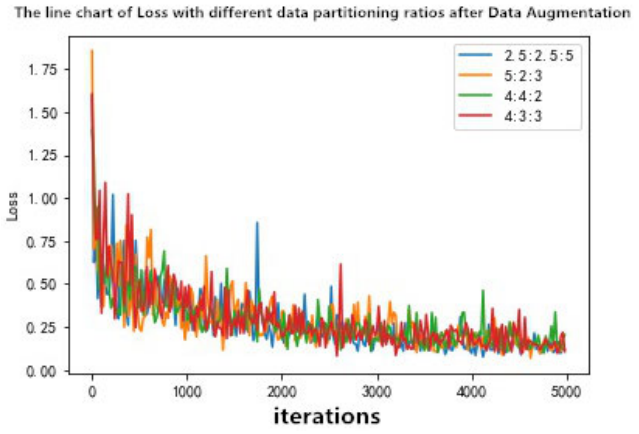



**FIGURE 13.** The line chart of loss with different partition ratios after data augmentation, the convergence rate of the loss with four different partition ratios is basically the same after the data augmentation, which is stable at about 3000 iterations.

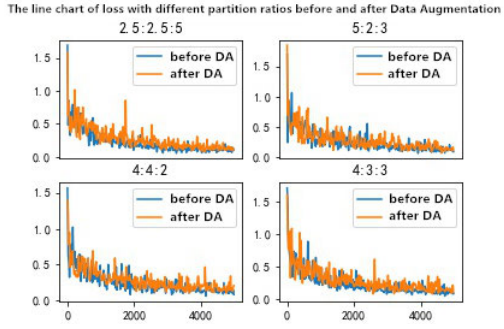



**FIGURE 14.** The line chart of loss with different partition ratios before and after data augmentation, the convergence rate of loss of the four partitioning ratios before and after data augmentation is basically unchanged.

The division ratio of different data sets can have a certain influence on the training of the model when the data set is too small. We can increase the data set by means of data augmentation.

### *E. EFFICIENCY COMPARISON OF DIFFERENT MODE OF CROP*

The method of random crop is used to enhance the data, but the disadvantage of random crop is that the information is probably incomplete. As shown in Fig. 15, only half of the tumors intercepted are invalid information.


**FIGURE 15.** Ultrasonic images before and after random crop.

We use morphological filtering to intercept the image. Firstly, we enhance the edge of the original image to enhance the contrast of the image, and then use the convex hull in the morphology to find the maximum boundary (the connected region) of the image. Then it is a kind of instructive interception instead of random interception. Fig. 16 shows the process of using morphological filtering to intercept an image.

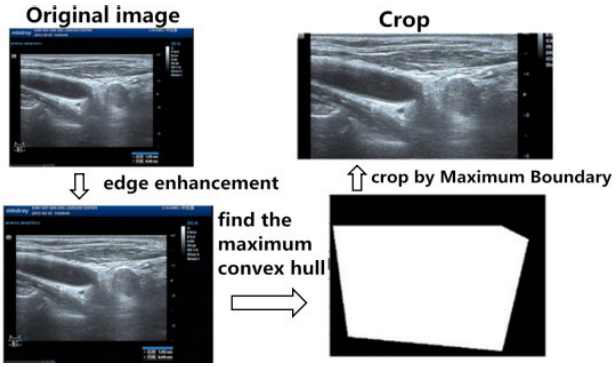



**FIGURE 16.** An example of cropping an image based on morphological filtering.

Based on the method of morphological filtering, we enhance the data set again. The experimental results are shown as TABLE 4 and Fig. 17.

**TABLE 4.** *mAP*(%) comparison among different partition ratios before and after two ways of data augmentation.

| Data Augmentation | Partition ratio (Training set:Verification set:Test set) | | | |
|---|---|---|---|---|
| | 2.5:2.5:5 | 5:2:3 | 4:4:2 | 4:3:3 |
| False | 0.775 | 0.782 | 0.848 | 0.778 |
| True (random crop) | 0.862 | 0.902 | 0.802 | 0.889 |
| False (morphological filtering) | 0.902 | 0.877 | 0.899 | 0.896 |

From TABLE 4 and Fig. 17, we can see that data augmentation based on morphological filtering can improve the accuracy of the model. The accuracy of the four partition ratios is close to 90, which is better than the data augmentation based on random crop, and more stable. In addition, we analyze the change of loss convergence of four partition ratios of three

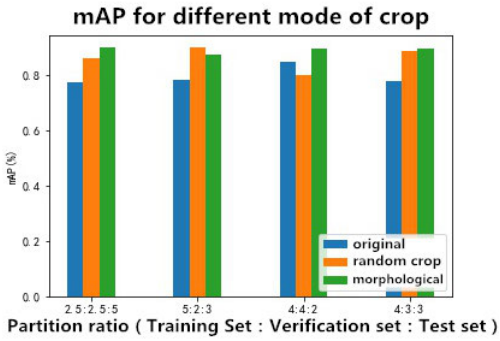



**FIGURE 17.** *mAP* **histogram with different data set partitioning scale based on different interceptions, data augmentation based on morphological filtering can improve the accuracy of the model.**

different interception methods, as shown in Fig. 18. we can see that different interception methods have no effect on the convergence rate of the four partition ratios, and it is basically achieved at 3000 iterations.

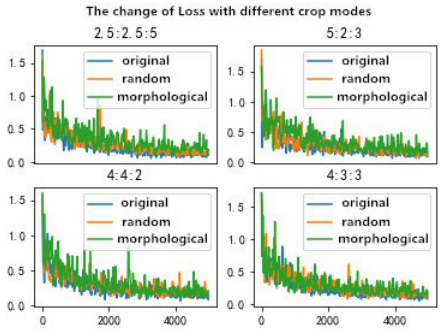



**FIGURE 18.** **The line chart of Loss with different crop modes for different partition ratios of data set, but different interception methods have no effect on the convergence rate of the four partition ratios, and it is basically achieved at 3000 iterations.**

### *F. EFFICIENCY COMPARISON OF THE DIFFERENT LEARNING RATE*

We divide the data set into training sets, validation sets, and test sets at 4 : 3 : 3 scale. The data set based on data augmentation by morphological filtering is trained by faster R-CNN based on ZF network with the learning rate of 0.1,0.005,0.001,0.0001 for 5000 iterations. The experimental results are shown in TABLE 5 and Fig. 19.

**TABLE 5.** ***mAP*(%) comparison of different learning rates.**

| Network Model | Learning Rate | | | |
|---|---|---|---|---|
| | 0.1 | 0.005 | 0.001 | 0.0001 |
| ZF | None | 0.902 | 0.889 | 0.760 |

From TABLE 5 and Fig. 19, it can be found that the result is none when the learning rate is 0.1, because of the floating point overflow caused by the excessive learning rate. The accuracy rate is the highest when the learning rate is 0.005, and the accuracy is decreased when the learning rate

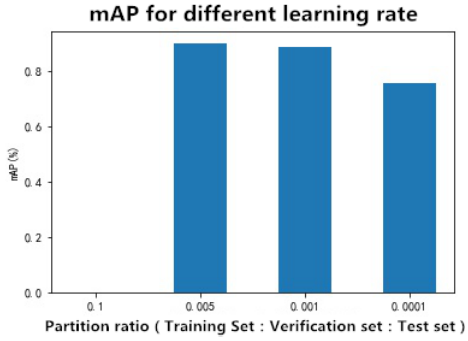



**FIGURE 19.** *mAP* **histogram with different learning rates, the result is null when the learning rate is 0.1 because of the floating point overflow caused by the excessive learning rate.The accuracy rate is the highest when the learning rate is 0.005.**

is 0.0001. We analyze the loss convergence rate of three learning rates. As shown in Fig. 20, when the learning rate is 0.0001, the network converges very slowly and had not converged at 5000 iterations. Besides, the learning rate of 0.001 or 0.005 has basically converged to a stable state at about 5000 times.

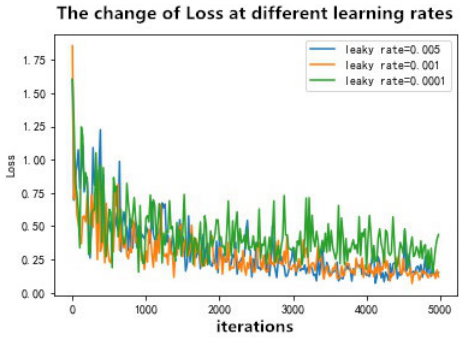



**FIGURE 20.** **The change of Loss at different learning rates, where shows that the network converges very slowly and had not converged at 5000 iterations when it was 0.0001, and the learning rate of 0.001 or 0.005 has basically converged to a stable state at about 5000 times.**

Because of our relatively simple classification, we use a shallow model of caffenet, which converges faster and after 1000 iterations it had basically converged. Of course, we can also use some deep networks such as VGG16, ResNet to train the model, which might extract richer semantic information, and be helpful for classification results.

### *G. THE EFFICIENCY BY COMBINING CDFI VLOOD FLOW SIGNAL*

In the experiment, we use the pre-training Caffenet model which has been trained in the large-scale competition of ImageNet, and then fine-tune it. We recorded the above three blood flow signals in class 0, 1, 2 respectively. The whole data set includes 364 ultrasound images, of which 255 were in class 0, 53 were in class 1 and 56 were in class 2. Some parameters are set as follows: batch size is 256; learning rate is 0.001; iteration number is 5000; momentum parameter is 0.9; learning rate is attenuated by 0.0005 per 1000 times.

After 5000 iterations, the accuracy of the network is about 87%. However, the accuracy is not ideal. We find that the network predicts most of the pictures into class 0 when we predict some pictures, that is, no obvious blood flow signals are detected, which means that some classification errors are made obviously. It's easy to find that the imbalance of sample categories is the main reason for the errors. It can be observed that the proportion of the three categories is 5 : 1 : 1. Unbalanced training samples will cause the training model to focus on the category with more samples, while ignoring the category with fewer samples, which will affect the generalization ability of the model in test data.

Usually, data resampling is to solve the imbalance of sample categories. Resampling includes upsampling and subsampling. For the class with fewer samples, upsampling can be used, that is to replicate the sample or increase it to the category with the largest number of samples by means of Data Augmentation, and for the class with more samples, subsampling can be used.

We do upsampling for class 1 and 2 and subsampling for class 0. The specific way is to deal with the pictures for class 1 and 2 by means of edge enhancement, and then to intercept the image after edge enhancement based on morphological filtering. The number of pictures of class 1 and 2 can be increased to three times of the original; 3/5 of the pictures are randomly selected from the class 0, and then used as batch processing training object. After treatment, the proportion of three classes was 1 : 1 : 1. After data resampling, the accuracy of the model is promoted from 87% to 93%, with a well test result. As can be seen from Fig. 21, because of the small data set, the model has begun to converge in 1000 iterations, and the speed of convergence is faster.

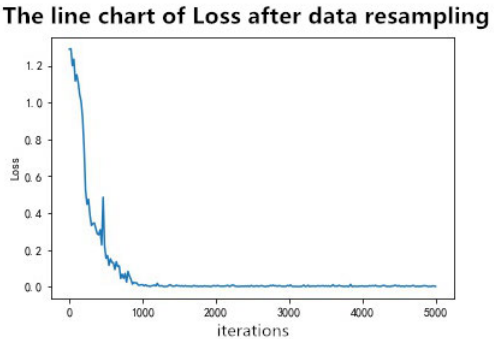



**FIGURE 21.** The line chart of Loss after data resampling, the model has begun to converge in 1000 iterations, and the speed of convergence is faster.

From the experiments we can find that:

**a) Usually deeper network is helpful to model learning.** By increasing the number of network layers, the network can extract deeper and richer semantic information, thus improving the accuracy of the model.

**b) Different data set partitioning proportion will have a certain impact on the training of the model when the data set is too small.** And it is easy to cause over-fitting or under-fitting when the partitioning proportion is too large or too small, which is not conducive to improving the generalization ability of the model. We can increase the data set by data augmentation, or we can use image processing based on morphological filtering to intercept images to reduce incomplete information. Data augmentation can often improve the accuracy of the model, enhance the robustness to image deformation, and enhance the generalization ability of the model. Data augmentation based on morphological filtering makes the model more stable while improving the accuracy.

**c) The learning rate will affect the convergence rate of the model.** When the learning rate is too large, the network will oscillate. It can not get the extreme value, and may overflow. If the learning rate is too small, the network convergence speed would be too slow with the long training time, but also may fall into the local optimal. So we need to train the model with the appropriate learning rate to achieve better results.

## V. CONCLUSION

In this paper, we take the ultrasound image of thyroid nodule as the main research object, using Faster R-CNN integrated with PRN based on deep learning to detect the CDFI blood flow signals, focusing on the efficiency comparison of the model selection, different iterations, different partition ratios of the data set, different setting of learning rate and whether to use data augmentation and morphological filtering, and build an efficient tumor-recognition model by a series appropriate parameters. Based on the real world ultrasound image, our experimental results show that our proposed approach outperformed the other methods in accuracy and stability.

In future work, the distributed training and classification of this algorithm in Map-Reduce framework will be studied continuously. In this way, the recognition results can be obtained quickly and the application of this research is advanced. Moreover, we can further apply this technology framework to ultrasound video recognition application.

## REFERENCES

[1] J. A. Noble and D. Boukerroui, "Ultrasound image segmentation: A survey," *IEEE Trans. Med. Imag.*, vol. 25, no. 8, pp. 987–1010, Aug. 2006.

[2] Y.-L. Huang and D.-R. Chen, "Watershed segmentation for breast tumor in 2-D sonography," *Ultrasound Med. Biol.*, vol. 30, no. 5, pp. 625–632, May 2004.

[3] R. Gaetano, G. Masi, G. Poggi, L. Verdoliva, and G. Scarpa, "Marker-controlled watershed-based segmentation of multiresolution remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 2987–3004, Jun. 2015.

[4] H. Naimi, A. B. H. Adamou-Mitiche, and L. Mitiche, "Medical image denoising using dual tree complex thresholding wavelet transform and Wiener filter," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 27, no. 1, pp. 40–45, 2015.

[5] V. N. Pham, T. N. Long, and T. D. Nguyen, "Feature-reduction fuzzy co-clustering algorithm for hyperspectral image segmentation," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Jul. 2017, pp. 1–7.

[6] Y. Zhao, L. Rada, K. Chen, S. P. Harding, and Y. Zheng, "Automated vessel segmentation using infinite perimeter active contour model with hybrid region information with application to retinal images," *IEEE Trans. Med. Imag.*, vol. 34, no. 9, pp. 1797–1807, Sep. 2015.

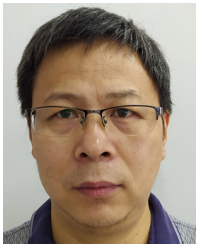
[7] X. Fan, L. Ju, X. Wang, and S. Wang, "A fuzzy edge-weighted centroidal Voronoi tessellation model for image segmentation," *Comput. Math. Appl.*, vol. 71, no. 11, pp. 2272–2284, Jun. 2016.

[8] H. Wang, T.-Z. Huang, Z. Xu, and Y. Wang, "An active contour model and its algorithms with local and global Gaussian distribution fitting energies," *Inf. Sci.*, vol. 263, pp. 43–59, Apr. 2014.
[9] U. Konur, F. S. Gürgen, F. Varol, and L. Akarun, "Computer aided detection of spina bifida using nearest neighbor classification with curvature scale space features of fetal skulls extracted from ultrasound images," *Knowl.-Based Syst.*, vol. 85, pp. 80–95, Sep. 2015.
[10] S. Farokhi, U. U. Sheikh, J. Flusser, and B. Yang, "Near infrared face recognition using zernike moments and Hermite kernels," *Inf. Sci.*, vol. 316, pp. 234–245, Sep. 2015.
[11] T. Weng, Y. Yuan, L. Shen, and Y. Zhao, "Clothing image retrieval using color moment," in *Proc. 3rd Int. Conf. Comput. Sci. Netw. Technol.*, Oct. 2013, pp. 1016–1020.
[12] J. Yamaguchi, A. Yoneyama, and T. Minamoto, "Automatic detection of early esophageal cancer from endoscope image using fractal dimension and discrete wavelet transform," in *Proc. 12th Int. Conf. Inf. Technol. (New Generations)*, Apr. 2015, pp. 317–322.
[13] J. Das and H. Roy, "Human face detection in color images using HSV color histogram and WLD," in *Proc. Int. Conf. Comput. Intell. Commun. Netw.*, Nov. 2014, pp. 1–6.
[14] X. Xu, C. Quan, and F. Ren, "Facial expression recognition based on Gabor wavelet transform and histogram of oriented gradients," in *Proc. IEEE Int. Conf. Mechatronics Automat. (ICMA)*, Aug. 2015, pp. 2117–2122.
[15] A. Emmanuel and O. O. Olugbara, "Lung cancer prediction using neural network ensemble with histogram of oriented gradient genomic features," *Sci. World J.*, vol. 2015, Feb. 2015, Art. no. 786013.
[16] S. R. M. Priya, P. K. R. Maddikunta, M. Parimala, S. Koppu, T. R. Gadekallu, C. L. Chowdhary, and M. Alazab, "An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT architecture," *Comput. Commun.*, vol. 160, pp. 139–149, Jul. 2020.
[17] D. C. Mocanu, H. B. Ammar, D. Lowet, K. Driessens, A. Liotta, G. Weiss, and K. Tuyls, "Factored four way conditional restricted Boltzmann machines for activity recognition," *Pattern Recognit. Lett.*, vol. 66, pp. 100–108, Nov. 2015.
[18] R. Korez, B. Likar, F. Pernuš, and T. Vrtovec, "Model-based segmentation of vertebral bodies from MR images with 3D CNNs," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2016, pp. 433–441.
[19] P. Moeskops, J. M. Wolterink, B. H. M. van der Velden, K. G. A. Gilhuijs, T. Leiner, M. A. Viergever, and I. Išgum, "Deep learning for multi-task medical image segmentation in multiple modalities," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2016, pp. 478–486.
[20] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. C. Nelson, J. L. Mega, and D. R. Webster, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *J. Amer. Med. Assoc.*, vol. 316, no. 22, p. 2402, Dec. 2016.
[21] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, Feb. 2017.
[22] J. Mansanet, A. Albiol, R. Paredes, and A. Albiol, "Mask selective regularization for restricted Boltzmann machines," *Neurocomputing*, vol. 165, pp. 375–383, Oct. 2015.
[23] J. Antony, K. McGuinness, N. E. O'Connor, and K. Moran, "Quantifying radiographic knee osteoarthritis severity using deep convolutional neural networks," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 1195–1200.
[24] G. T. Reddy, S. Bhattacharya, S. S. Ramakrishnan, C. L. Chowdhary, S. Hakak, R. Kaluri, and M. P. K. Reddy, "An ensemble based machine learning model for diabetic retinopathy classification," in *Proc. Int. Conf. Emerg. Trends Inf. Technol. Eng. (ic-ETITE)*, Feb. 2020, pp. 1–6.
[25] E. Bengtsson and P. Malm, "Screening for cervical cancer using automated analysis of pap-smears," *Comput. Math. Methods Med.*, vol. 2014, no. 2962, 2014, Art. no. 842037.
[26] T. Kooi, G. Litjens, B. van Ginneken, A. Gubern-Mérida, C. I. Sánchez, R. Mann, A. den Heeten, and N. Karssemeijer, "Large scale deep learning for computer aided detection of mammographic lesions," *Med. Image Anal.*, vol. 35, pp. 303–312, Jan. 2017.
[27] N. Khare, P. Devan, C. L. Chowdhary, S. Bhattacharya, G. Singh, S. Singh, and B. Yoon, "SMO-DNN: Spider monkey optimization and deep neural network hybrid classifier model for intrusion detection," *Electronics*, vol. 9, no. 4, p. 692, Apr. 2020.

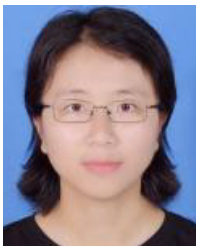
**WANRONG GU** received the Ph.D. degree in computer science and technology from the South China University of Technology, in 2015. He is currently an Assistant Professor with the Computer Science Department, South China Agricultural University. He has published a number of articles in these fields. His main research interests include big data mining and forecasting, machine learning, and big data analysis of bioinformatics.

**YIJUN MAO** received the Ph.D. degree in computer science and technology from Sun Yat-sen University, in 2016. He is currently an Assistant Professor with the Computer Science Department, South China Agricultural University. His main research interests include machine learning and bioinformatics and algorithm.

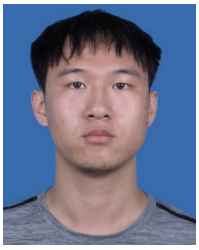
**YICHEN HE** is currently pursuing the degree with the College of Mathematics and Informatics, South China Agricultural University. He holds many software patents. He has done in-depth research in these fields, participated in many related software development projects. His main research interests include artificial intelligence, big data mining and forecasting, and machine learning.



**ZAOQING LIANG** received the Ph.D. degree from the School of Computer Science, Wuhan University, in 2007. He is currently a Lecturer with the Computer Science Department, South China Agricultural University. His main research interests include software modeling and machine learning.



**XIANFEN XIE** received the Ph.D. degree in statistics from Jinan University, in 2016. She is currently an Associate Professor with the Statistics Department, Jinan University. She has more than ten years of research experience and led a number of research projects in these fields. Her main research interests include financial risk forecasting, big data analysis, and policy decision in economics.



**ZIYE ZHANG** is currently pursuing the Ph.D. degree with the College of Mathematical, South China University of Technology. His main research interests include mathematical modeling, big data analysis of bioinformatics, and machine learning. He has participated in a number of mathematical contest in modeling and received awards.

**WEIJIANG FAN** received the B.S. degree from the Department of Computer Science and Technology, South China Agricultural University, in June 2020. He is currently pursuing the master's degree in software engineering with South China Normal University. His research interests include machine learning, include image recognition, artificial neural networks, and deep learning.