A Nuclear detection method Based on improved You Only Look Once v5s Framework

Jiaheng Li¹, Jingcun Wang¹* and Shen Feng¹

¹Wuhan University of science and technology, Wuhan Hubei 430081, China

*Corresponding author’s e-mail: wangjingcun@wust.edu.cn

Abstract. Extracting human cell smear samples through automated scanners, using target detection algorithm to detect its nucleus is an important content in cell detection. Due to the different shapes, different sizes and complex backgrounds of nuclei in the samples, automatic detection of nuclei is a challenging task. In order to solve the problem of complex small target recognition in nuclear detection and improve detection accuracy, based on the YOLOV5S framework of one-stage target detection, make certain improvement, add new residual module to the original network Neck layer, at the same time carry on network widening operation to the subsequent framework. The algorithm visualizes the results of training and testing through Tensorboard. Experimental results show that the accuracy of the improved YOLOV5S network is 1.6% and 4.4% higher than that of the original YOLOV5S network and YOLOV3 network. the improved network has strong real-time performance and high accuracy, which can meet the actual use requirements.

1. Introduction
Nucleus detection of samples extracted from human tissues is an important task in cell testing. It plays an important role in the fields of biomedical experiments and clinical medical diagnosis. For example, cell smear is a common method in clinical medical testing. Because of the differences in the nucleus morphology before and after cancer, by detecting the shape of the cell nucleus, the possibility or degree of cancerous transformation of the cell can be judged. In order to take a timely and effective treatment plan.

The acquired data set comes from the cell staining samples obtained by a fully automated machine through a series of processes. Then use the microscope camera to take pictures of the obtained samples to obtain cell picture samples with diverse shapes and complex background environments. However, it is still challenging to use image detection algorithms to automatically identify the nuclei. There are two main types of methods currently available for nuclear detection:

One is traditional image processing methods, such as automatic cell recognition technology based on Hough transform [1], Threshold-based method [2], K-means clustering [3] and other classic algorithms. Although these traditional methods are easy to implement and have high efficiency, when there is a large number of overlapping nuclei and cell adhesion on the detection image, it is easy to cause missing information and imperfect segmentation, there are many complicated situations in the actual target detection application. Traditional image processing methods are difficult to solve these problems.

The other is the target detection algorithm based on deep learning, this type of algorithm is divided into two categories due to its candidate frame generation steps: Tow-stage target detection algorithm, One-stage target detection algorithm. Two-stage target detection algorithm representatives are: RCNN [4], Fast R-CNN [5], Faster R-CNN [6], etc. Although the Tow-stage target detection algorithm has high
accuracy, it is difficult to solve the problems of slow detection speed and large model size. The network structure is also more complicated. One-stage target detection algorithm representatives are: SSD [7], YOLO [8], etc. One-stage target detection algorithm can produce higher quality detection frames, has faster detection speed and model training speed, and can also obtain better detection accuracy, compared with the Tow-stage target detection algorithm, the One-stage target detection algorithm can obtain a smaller weight model and faster detection speed. In practical applications, a smaller weight model is beneficial to deploy in CPU and embedded systems.

2. Network selection and testing process
This paper chooses YOLOv5s in the One-stage target detection algorithm as the basic framework. The YOLOv5 network framework launched by Ultralytics in May 2020 has the advantages of small size, fast speed, and flexibility. It can be built in the ecologically mature PyTorch deep learning framework. The deployment is simple to implement. YOLOv5 draws on the CutMix method, The Mosaic data enhancement method effectively solves the most difficult small target problem in model training and the problem that only part of the object is in the image. There are 4 versions of YOLOv5. The models from small to large are s, m, l, x. This article selects the YOLOv5s network with the smallest model size and the fastest detection speed. On the premise of having a faster detection speed, make improvements to it to improve detection accuracy. Through the experimental verification of the self-made nuclear data set, the detection effect of the method in this paper has been significantly improved after comparing with YOLOv3 and the original YOLOv5s. In order to further verify the improvement of the improved algorithm, the original algorithm and the algorithm in this paper are trained and verified with a public data set. Compared with the original algorithm, the improved algorithm has achieved better results on the public data set than on the self-made data set. The process of network training and detection is shown in Figure 1.

2.1. The network model of the original YOLOv5s
The overall network structure of YOLOv5s is similar to the YOLOv3 and v4 networks, and can be roughly divided into four parts: the input layer, the Backbone layer, the Neck layer, and the Prediction layer.

- Input layer: used to input the original picture and perform pre-processing operations on the original picture, including adaptive picture scaling, using data enhancement methods such as Mosaic, and obtaining adaptive anchor boxes through clustering methods.

- Backbone layer: used to extract features, including Focus structure, SPP structure, and CSP structure. The Focus layer slices the input 640*640*3 image into 4 pieces and then undergoes a convolution operation of 32 convolution kernels into a 304*304*32 feature map. The CSP (Cross Stage Partial) structure refers to the idea of dense cross-layer jump of Densenet, and performs local cross-layer feature fusion, and the residual network is embedded inside to further strengthen the extraction and fusion of features. A feature map with richer information can be obtained. The SPP (Space Pyramid Pooling) module uses three cores of different sizes to perform the maximum pooling operation, and then performs tensor splicing with the input feature map.
Neck layer: used to perform feature fusion on feature maps of different layers, FPN (Feature Pyramid Network k) With PANet (PathAggregation). Through the top-down integration of high-level semantic features into the shallow feature map, and then integrate the shallow location features into the high-level feature map from the bottom up, and use the CSP structure for further feature extraction on the way.

Prediction layer: output 80*80, 40*40, 20*20, a total of three detection layers of different sizes, which is conducive to detecting targets of different sizes. GIOU_Loss is used instead of IOU_Loss in YOLOv3 as the box loss function, which solves the prediction value and Ground truth does not overlap, resulting in the problem that IOU is always 0 and cannot be optimized.

3. Improve YOLOV5s model

3.1. Residual block

For traditional deep learning networks, the deeper the network layer, the stronger the learning ability of the network, but the deepening of the network will result in slower model convergence. The training time will also increase, and a too deep network will even cause problems such as reduced accuracy, decreased learning rate, gradient disappearance, and gradient explosion. The proposed residual network can effectively solve this series of problems. The formula of the residual block is shown in formula (1).

\[ H(x) = F(x) + x \]  

\( F(x) \) is the feature map outputted by a series of convolution modules, and \( x \) is the original feature map entered into this series of convolution modules. The two keep the same dimension, and they are added according to the dimension one-to-one correspondence. Convolution can be compared to a compression process, in which part of the features will be lost, and the original image \( x \) has more complete features. The combination of the two can make the network learn more abundant content, \( H(x) \) is the residual Block output feature map. The residual network is easy to optimize, and can alleviate problems such as the disappearance of gradients caused by the excessive depth of the network through the jump connection method. The structure of the residual block is shown in Figure 2 below.

![Residual block](image)

A residual block is composed of a first-order convolution module and a third-order convolution module. Each convolution module contains a BN layer (Batch Normalization) to solve the problem of gradient disappearance and explosion and speed up training. In the module The BN layer is followed by a SiLU activation function.

Although the residual network helps to deepen the number of network layers and improve the detection accuracy, too many residual blocks will increase the volume of the network model and reduce the training and testing speed of the network. In order to keep the network light enough, only one residual module is inserted into each of the last three CSP layers of Neck to improve the detection accuracy while ensuring a faster speed.

3.2. Partial dimensional improvement

The four models of YOLOv5, YOLOv5s, m, l, and x, are mainly different in two points: the depth and width of the network. The depth changes the number of residual blocks in each CSP module, and the width changes the size of the convolution dimension of the entire network. The larger the model, the
greater the depth and width. Increasing the depth and width of the network easily leads to an increase in the model volume. This article chooses to only increase the dimensions of part of the network to increase the learning ability of the network. Starting from the insertion of the first residual block in the Neck layer, the dimension of the subsequent network is doubled. This method of widening the network by only increasing the dimensions of a small part of the network only increases a small amount of the network volume and has a small impact on the speed. The dimensions of the three feature maps of different sizes that the improved Neck layer inputs to the Prediction detection layer are increased from the original 128, 256, and 512 to 256, 512, and 1024. The size of the three detection layers remains the same as 80*80, 40*40, and 20*20. After experiments (see 5.1 for experimental results), Compared with v3 and the original network, the accuracy of the improved network has been improved. The overall network structure after the improvement is shown in Figure 3.

4. Data set and environment

4.1. Experimental dataset
The data set of this experiment is taken from the stained pictures of cell smears extracted from the hospital through a microscope. The original picture has a resolution of 4384*3288. It cannot be directly scaled or put directly into the network for training and testing. The obtained pictures, Due to the different magnifications and brightness changes of the pictures obtained by the industrial microscope, the brightness of the picture environment is different, and the size of the cell nucleus is different. The original picture is shown in Figure 4.
4.2. Data processing
Due to the large resolution of the original image, training directly into the network will cause the overall network model to be huge, resulting in a decrease in training and detection speed. This is contradictory to our pursuit of speed and accuracy, and there are many blurry and invalid images. The area will also affect the effect of picture training. Directly scaling the image to 640*640 will cause serious loss of information on the image. There are too many small targets in a smaller picture, which will greatly increase the difficulty of training and detection. Therefore, it is necessary to cut the obtained image in advance to increase the number of data sets under the premise of ensuring complete information.

4.3. Picture cutting method
For the processing of the original image, we use the image cutting method with overlapping areas instead of the traditional geometric cutting. The cutting method is shown in Figure 5(left).

![Figure 5. Cutting method and Picture after cutting](image)

The image cutting method with overlapping regions can prevent some targets from being segmented and truncated, and can fully learn the targets in each picture after cutting.

After segmenting and excluding some invalid samples, we obtained 880 images of appropriate size. Use Labelling software for labeling, and convert the labelled xml suffix file into a txt file. Randomly select 780 samples as the training set and 100 samples as the test set. The pictures obtained after cutting are shown in Figure 5(right).

4.4. Environment and configuration
The experimental environment uses the Ubuntu20.04 operating system, Python3.8 is selected as the compilation language, the deep learning framework is Pytorch, the acceleration environment is CUDA11.2, The graphics card uses RTX 3080.

5. Network model training
In the network model training phase, the total number of iterations is set to 500, and the parameter training adopts the SGD optimization algorithm; the batch size is 12; the momentum factor is 0.937; the weight attenuation coefficient is 0.0005. Use GIOU Loss as the loss function. And compare the detection accuracy. The recall and precision are shown in Figures 6.

The recall rate is the probability that the correct class of the sample is predicted correctly. \( FN \) means predicting the correct class as a negative class, and \( TP \) means predicting the correct class correctly. As shown in formula (2), it is stable when the model iteration reaches 50 times, and the value is close to 0.95.

\[
recall = \frac{TP}{TP + FN}
\]
Accuracy, $FP$ indicates that the negative category is predicted to be the correct category, that is, the total number of samples that are actually correct in the test set is divided by the total number of samples that are predicted to be correct, as shown in formula (3). It is stable after 50 iterations of the model, and the value reaches 0.96.

$$precision = \frac{TP}{TP + FP}$$

(3)

From the data performance evaluation of Figures 6, it can be concluded that the improved YOLOv5s network model has a faster convergence speed. If you need to pursue higher detection accuracy, you can replace the YOLOv5s model with a larger m, l, x series model. But it will reduce the training and detection speed of the model. The purpose of this article is to achieve the fastest detection speed while maintaining adequate detection accuracy.

5.1. results and analysis

In order to verify the effectiveness of the improved YOLOv5s, the improved algorithm was compared with YOLOv3 and YOLOv5s under self-made nuclear data.

| Algorithm       | Accuracy (%) | Frame (/s) |
|-----------------|--------------|------------|
| YOLOv3          | 92.4         | 77         |
| YOLOv5s         | 95.2         | 125        |
| Improve YOLOv5s | 96.8         | 118        |

From the performance evaluation of the algorithm in Table 1, it can be concluded that the improved YOLOv5s network model training test results and the accuracy of YOLOv3 and the original YOLOv5s network have increased by 4.4% and 1.6% respectively, and the detection speed has reached 118 frames per second, maintaining sufficient detection. The speed and accuracy have been improved.

The detection results of different network structures are shown in Figure 7. The improved yolov5s network in this paper has improved the detection effect of too large, too small nuclei, adhered nuclei, and some edge nuclei.

Figure 7. Comparison of detection results
In order to further verify the versatility of the improved yolov5s, the improved algorithm and the original algorithm are trained and tested on the public datasets coco128 and VOC2014. The test results are shown in Table 2.

| Algorithm      | coco128(%) | VOC2014(%) |
|----------------|------------|------------|
| YOLOv5s        | 79.0       | 73.7       |
| Improve YOLOv5s| 86.4       | 76.3       |

From the comparison in Table 2, it can be seen that the improved YOLOv5s in this paper can also achieve better detection results on public data sets than the original algorithm.

6. Conclusion

This paper is based on the improved YOLOv5s framework to train on the self-made nuclear data set, and obtain the nuclear detection model. The detection network model obtained in this experiment is small in size, fast in speed, and high in accuracy, and can extract rich feature information from the image. The experimental results show that the detection accuracy of the improved YOLOv5s constructed in this experiment reached 96.8%, and the detection speed of a single image reached 8.5ms, and a good detection effect was obtained. However, due to the small amount of data, the model still has certain limitations. Next, we will further strengthen the collection of images, increase the amount of data, and integrate with embedded systems and promote applications.

References

[1] Reddy V H. (2014) Automatic red blood cell and white blood cell counting for telemedicine system[J]. International Journal of Research in Advent Technology, 2014(1): 294-299.

[2] Arteta C, Lempitsky V, Noble J A, et al. (2012) Learning to detect cells using non-overlapping extremal regions[C]//Proceedings of the 15th International Conference on Medical Image Computing and Computer-Assisted Intervention-Volume Part I. Berlin, Heidelberg: Springer, 348-356.

[3] VAISHALI UGHADE, NISHCHOL MISHRA, SANJEEV SHARMA. (2011) Improved KMean Clustering with Steepest Ascent 'Gradient' Method for Image Retrieval[J]. International Journal of Computer Applications, 2011(1):8.

[4] Girshick R, Donahue J, Darrell T, et al. (2014) Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops. Washington, DC: IEEE Computer Society, 2014: 580-587.

[5] Girshick R. (2018) Fast R-CNN[EB/OL]. [2018-01-26].http://cn.arxiv.Org/pdf/1504.08083.pdf.

[6] Ren S, He K, Girhick R, et al. (2015) Faster R-CNN: towards realtime object detection with region proposal networks[C]// Proceedings of the 28th International Conference on Neural Information Processing Systems. Cambridge, MA: MIT Press,2015: 91-99.

[7] Liu W, Anguelov D, Erhan D, et al.(2015)SSD: single shot multibox detector[C]//Proceedings of the 14th European Conference on Computer Vision. Berlin: Springer, 2015: 21-37.

[8] Redmon J, Divvala S, Girshick R, et al. (2015) You only look once: unified, real-time object detection[C]//Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2015: 779-788.