

## Multi-Object Detection of Chinese License Plate in Complex Scenes

Dan Liu<sup>1,3</sup>, Yajuan Wu<sup>1</sup>, Yuxin He<sup>2</sup>, Lu Qin<sup>2</sup> and Bochuan Zheng<sup>2,3,\*</sup>

<sup>1</sup>School of Computer Science, China West Normal University, Nanchong, 637002, China

<sup>2</sup>School of Mathematics and Information, China West Normal University, Nanchong, 637002, China

<sup>3</sup>Internet of Things Perception and Big Data Analysis Key Laboratory of Nanchong, China West Normal University, Nanchong, 637002, China

\*Corresponding Author: Bochuan Zheng. Email: zhengbc@vip.163.com

Received: 22 September 2020; Accepted: 27 October 2020

**Abstract:** Multi-license plate detection in complex scenes is still a challenging task because of multiple vehicle license plates with different sizes and classes in the images having complex background. The edge features of high-density distribution and the high curvature features of stroke turning of Chinese character are important signs to distinguish Chinese license plate from other objects. To accurately detect multiple vehicle license plates with different sizes and classes in complex scenes, a multi-object detection of Chinese license plate method based on improved YOLOv3 network was proposed in this research. The improvements include replacing the residual block of the YOLOv3 backbone network with the Inception-ResNet-A block, imbedding the SPP block into the detection network, cutting the redundant Inception-ResNet-A block to suit for the multi-license plate detection task, and clustering the ground truth boxes of license plates to obtain a new set of anchor boxes. A Chinese vehicle license plate image dataset was built for training and testing the improved network, and the location and class of the license plates in each image were accurately labeled. The dataset has 62,153 pieces of images and 4 classes of China vehicle license plates, almost images have multiple license plates with different sizes. Experiments demonstrated that the multi-license plate detection method obtained 83.4% mAP, 98.88% precision, 98.17% recall, 98.52 F<sub>1</sub> score, 89.196 BFLOPS and 22 FPS on the test dataset, and whole performance was better than the other five compared networks including YOLOv3, SSD, Faster-RCNN, EfficientDet and RetinaNet.

**Keywords:** Chinese vehicle license plate; multiple license plate; multi-object detection; Inception-ResNet-A; spatial pyramid pooling

### 1 Introduction

Automatic license plate recognition technology is one of the important technologies in the automobile intelligent traffic management system. It is widely used in many systems such as park toll management, traffic flow measurement, vehicle monitoring, detection of stolen vehicle, highway over-speed automatic supervision, capture system of running red light, highway toll management, etc. Vehicle license plate



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

detection is one of key steps in license plate recognition. The license plate images collected at some specific places (e.g., the entrance of the residential area, company gate, the entrance of a parking lot, etc.) are clear due to good acquisition equipment and short distance from vehicle to camera. There is only one license plate needed to detection, and the background is simple, and the size and location of the license plate are relatively fixed, so it is easy to detect those license plates even for the traditional license plate detection approaches. However, for multi-license plate images taken from far distance in complex scenes (e.g., streets, intersections, and people-vehicle mixed traffic, etc.), the traditional methods are difficult to construct a detection model suitable for complex image background, multi-license plate, and different sizes and locations of license plate due to the weak generalization ability [1].

Multi-license plate detection in complex scenes has widely practical applications: 1) Used to detect the multi-license plate before recognizing the vehicle license plate in complex scenes; 2) Instead of vehicle detection, multi-license plate detection can be used to auto count traffic flow at the roads, scenic areas, or other places; 3) Used to monitor all vehicles passing at street or other places. The class of license plate is a few and the plate shape is rectangular, license plate detection is easier than the generic object detection. Recently years, object detection methods based on deep Convolutional Neural Networks (CNNs) were used to detect license plate [2–5]. However, mostly research works for license plate detection only detect a single license plate per image in the simple background. The detection accuracy is not high enough to meet the requirements of practical application when the existing object detection method based on CNNs is directly used for multi-license plate detection in complex scenes.

YOLOv3 [6] adopt multiple scales detection layer and Darknet-53 network structure, which has shown fewer parameters, more accuracy and faster speed than YOLOv1 [7] and YOLOv2 [8]. Despite YOLOv3 can detect multiple objects belonging to different classes at the same time, but the state-of-the-art accuracy cannot be obtained when it is used to directly detect multi-license plate in complex scenes. In the other hand, there are only four classes of license plate needed to detect in our dataset, the repeating blocks of YOLOv3 can further reduced. So, we modified the network structure of YOLOv3, made it suitable for multi-license plate detection in complex scenes, at same time made it meet the requirements of practical application, such as speed and detection accuracy. The main contributions of this paper are as follows:

1. The Inception-ResNet-A [9] is integrated into the backbone network of YOLOv3 to obtain a new feature extraction network for extracting license plate image feature;
2. The SPP (Spatial Pyramid Pooling) [10] is added to the detection network of the YOLOv3 to obtain a new multi-scale detection network;
3. The anchor boxes are designed again according to the cluster result of the license plate region size;
4. To minimize parameters and detection time under ensuring the excellent detection performance, cutting experiments are carried out to obtain appropriate number of the repeating blocks in the improved network.

The remainder of this paper is organized as follows. In Section 2 we briefly review related methods for vehicle license plate detection. Our proposed method is given in Section 3, where we describe four improvements which can increase the performance of license plate detection. The experimental results are shown in Section 4. Finally, Section 5 summarizes our conclusions.

## 2 Related Works

### 2.1 LicensePlate Detection Based on Traditional Methods

The traditional license plate detection methods mainly adopt the edge, texture, gray projection, color and other features of the license plate region. In [11], the license plate is detected by using mathematical morphological operation and color feature, and then the accurate position is further obtained according to

the texture features. In [12], the vehicle license plate detection is carried out by using the combination of HOG features and genetic algorithm. The input image is scanned randomly by using the fixed detection window repeatedly until the area with the highest prediction probability score is obtained, which is the vehicle license plate region. Jiang et al. [13] proposed a license plate detection method integrating texture features, color information and structural features of the license plate. This method firstly detects the edge of the vehicle gray image using the Canny operator, and then obtain the license plate regions through judging the number of edge jump variable. In [14], an algorithm is proposed to detect the license plate by analyzing the object pixel color. This method is based on the modified template matching technology, which firstly detect a group of object pixels, and then obtain the license plate location by verifying if its shape and aspect ratio match those of the standard license plate. Dun et al. [15] proposed a license plate detection method based on the color information for mainly detecting blue and yellow license plate. Zhang et al. [16] presented a new AdaBoost algorithm combined with color difference model for license plate detection, which can detect license plate even at night.

## **2.2 LicensePlate Detection Based on CNNs**

In recent years, the object detection methods based on CNNs had been popping up, such as a series of Faster Region-Convolutional Neural Network (Faster R-CNN) [17–19], a series of Single Shot MultiBox Detector(SSD) [20–24], and a series of You Only Look Once (YOLO) [6–8]. The object detection methods based on CNNs mainly includes two classes: one stage and two stages. One stage object detection methods directly detect objects from the input image through regression. YOLO and SSD serials belong to one stage method. Two stages object detection method has two stages to detect object, which first using the region recommendation extraction algorithm to obtain the object candidate regions, and then obtaining true object regions by judging all candidate regions. Due to good performance for object detection, the object detection methods based on CNNs are also used to detect vehicle license plates. In [2], an integrated recurrent neural network with connectionist temporal classification was proposed to detect vehicle license plate, where VGG16 is used to extract image features. In [3], a CNN was proposed to obtain score for each image sub-region, and then the obtained score is combined with the results obtained from the sparse overlapping region to detect license plate. In [4], a vehicle license plate detection method based on the cascaded CNN was proposed, where a single classifier arranged in a cascaded way is used to detect the license plate from the front of the vehicle. In [5], a method based on region CNNs and morphological operation is used to detect vehicle license plate, which can detect license plate in a real-time mode.

## **3 The Proposed Method**

### **3.1 Improved Network Structure**

YOLOv3 is a composition of two CNNs: Darknet-53 backbone network for extracting image features and detection network for multi-scale prediction. Our improved YOLOv3 multi-scale detection network is illustrated in Fig. 1. The backbone network consists of 6 convolutional layers and 17 Inception-ResNet-A blocks, where there are 5 convolutional layers with filter of size  $3 \times 3$  and stride 2 that reduces the input dimensionality by a factor of 52, and the Inception-ResNet-A block is used to replace the residual block in YOLOv3. Each convolutional layer includes Conv2d + BN + LeakyReLU, which is the same as that in YOLOv3. The new backbone network is named as Darknet-74 due to having 74 layers in total. In the object detection network, three feature maps with different sizes (i.e.,  $13 \times 13$ ,  $26 \times 26$ , and  $52 \times 52$ ) in the backbone network are used to detect objects of different size. To make the shallow feature map contain the abstract semantic information contained in the deep feature map, we upsample the feature map of size  $13 \times 13$ , and then merge it with the feature map of size  $26 \times 26$  to get a new feature map of size  $26 \times 26$  for medium object detection. At the same time, we continue to upsample the new feature

map and merge it with feature map of size  $52 \times 52$  to get a new feature map of size  $52 \times 52$  for small object detection. A set of convolution operations are performed on each detection feature map before detecting objects, the set of convolution operations is illustrated in the upper left corner of Fig. 1. SPP in the convolutional set is a newly added block, which is mainly used to obtain the local salient features among different scales of receptive field in the same layer. The final object detection result is obtained by using non-maximum suppression (NMS).

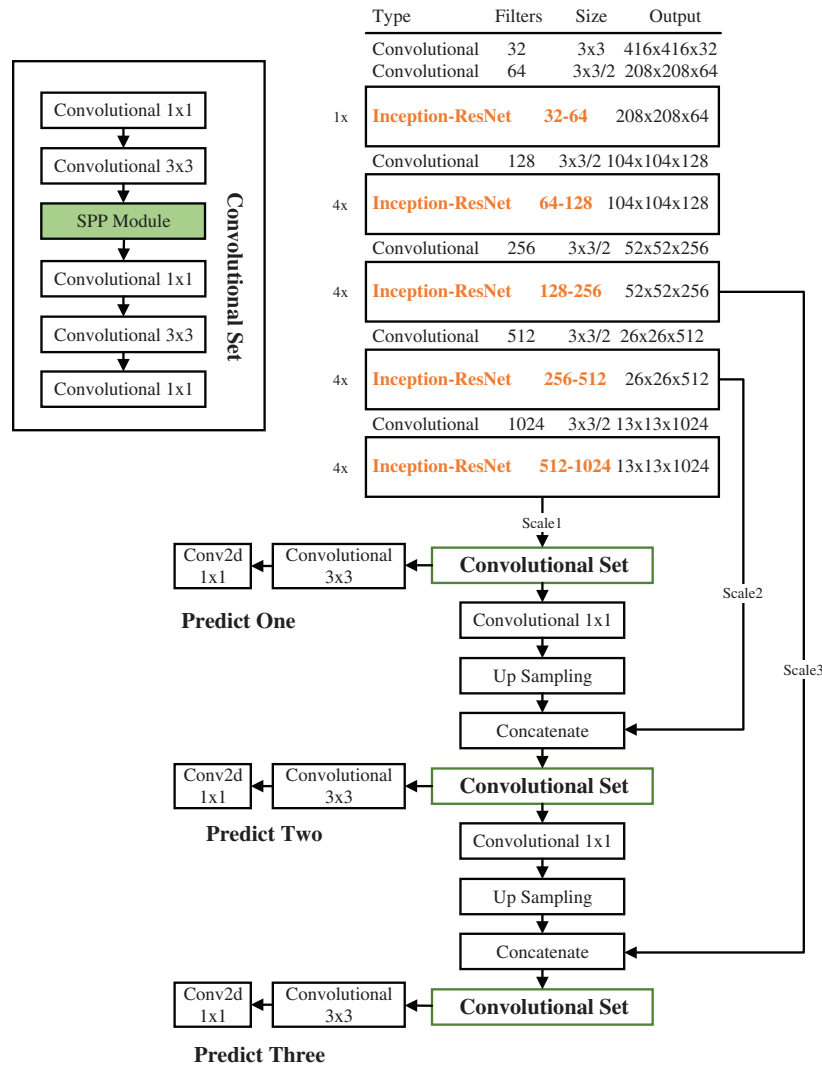


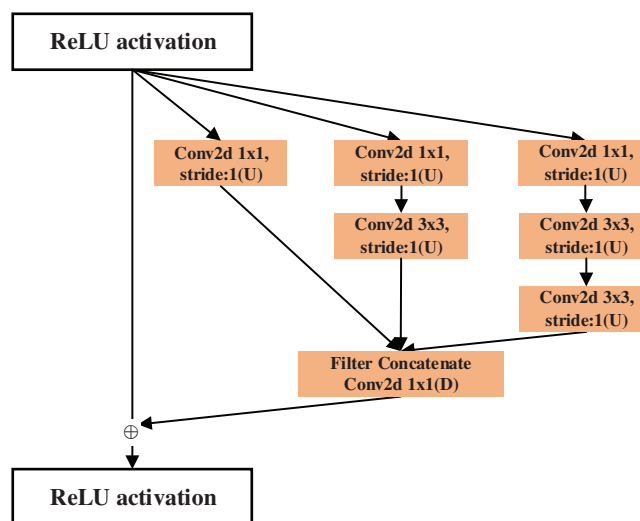
Figure 1: The improved network structure

### 3.2 Inception-ResNet-A Block

Inception block can obtain features from receptive fields of different scales through concatenating several feature maps which is obtained by using filters of different size. Inception block has been improved from Inception-v1 [25], Inception-v2 [26], Inception-v3 [27] to Inception-v4 [9]. Inception-v4 thoroughly modify the architecture of Inception-v3: simplify module design, eliminate unnecessary integration, and each Inception block is designed uniformly. Inception-ResNet [9] is the residual version

of Inception-v4. The shortcut connection makes the training speed greatly improved, which combines the advantages of residual block and Inception block.

In this research, we replaced residual block in the Darknet-53 network with the Inception-ResNet-A block of the Inception-ResNet-v2 to improve extracting feature ability. The Inception-ResNet-A block is shown in Fig. 2, in which the letter ‘U’ denotes the number of reduced channels dimension, and the letter ‘D’ denotes the number of output channels. Taking the first Inception-ResNet-A block in Fig. 1 as an example, the size of input feature map is  $208 \times 208 \times 64$ , and the filter of size  $1 \times 1$  is used to change the dimension, the number of reduced channels dimension is 32, and the number of output channels is 64. Three feature maps with 32 channels are concatenated to become a feature map having 96 channels, and then 96 channels are reduced to 64, finally the feature map with 64 channels is added with the input feature map to obtain the final output.



**Figure 2:** Inception-ResNet-A block

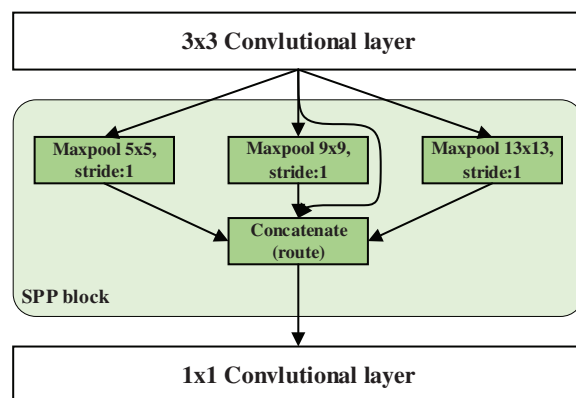
### 3.3 SPP Block

SPP block proposed in SPPNet [28] can transform feature images of different sizes into the feature vectors of same size. Max-pooling is used in SPP to extract the maximum responsive feature in a certain region. The largest and strongest response in the receptive fields of different sizes can be obtained by mixing the max-pooling features of grids of different sizes. The multi-scale detection network of YOLOv3 fuses the global features of multiple convolution layers of different scales, but it ignores to fuse the local multi-scale features in the same convolution layer. Added to the convolutional set of the YOLOv3 detection network, SPP block is used to pool and connect multi-scale local feature, which help to filter the optimal features in the feature map, focus on the most important information and improves the accuracy of object detection [28]. The structure of the SPP block is shown in Fig. 3. The max-pooling of  $5 \times 5$ ,  $9 \times 9$ , and  $13 \times 13$  scales is used respectively, and then the three pooling feature maps are combined with the original feature map to form a new feature map. To keep the size of the feature map unchanged after the max-pooling, the overlapping max-pooling is adopted, and the sliding stride is set to 1.

### 3.4 Anchor Boxes Cluster Analysis

YOLOv3 network realize the object detection with three different scales through the Feature Pyramid Networks (FPN) [29], and the sizes of the three feature maps are  $13 \times 13$ ,  $26 \times 26$ , and  $52 \times 52$ ,

respectively. There are only 4 classes of license plates in our dataset, including blue, yellow, green, and white, the size of the predicted tensor corresponding to each cell grid is  $(4 + 4 + 1) \times 3 = 27$ , where each cell grid has 3 anchor boxes, and each anchor box is composed of 4 class predictions, 4 bounding box offsets and 1 objectness prediction.



**Figure 3:** SPP block is added into detection network

The anchor boxes set in [6] are obtained by clustering all ground truth boxes of generic objects into 9 clusters, but the shape and size of these anchor boxes do not match with the size of license plate well, which affect the detection performance. In this research, we adopt same cluster algorithm as the literature [6] to obtain new anchor boxes set by clustering 106,265 ground truth boxes of license plate objects in our dataset into 9 clusters. The two anchor box sets and their impact on the IoU calculated with the validation dataset are listed in Tab. 1. Adopting different anchor boxes set greatly impact on the IoU, it is only 49.99% while using the anchor boxes set in [6], but the IoU can be stable at around 83.00% if the new anchor boxes is adopted (here, the network structure in Fig. 1 is used to train).

**Table 1:** IoU Comparison of using different anchor boxes set

	Anchor boxes set in [6]	New anchor boxes set
Size of anchor boxes	$(10 \times 13)$ , $(16 \times 30)$ , $(33 \times 23)$ , $(30 \times 61)$ , $(62 \times 45)$ , $(59 \times 119)$ , $(116 \times 90)$ , $(156 \times 198)$ , $(373 \times 236)$	$(12 \times 17)$ , $(16 \times 9)$ , $(18 \times 11)$ , $(21 \times 12)$ , $(24 \times 14)$ , $(28 \times 16)$ , $(33 \times 20)$ , $(41 \times 24)$ , $(54 \times 32)$
IoU	49.99%	83.00%

For 3 feature maps, each feature map is allocated 3 anchor boxes according to the size of anchor boxes. The anchor boxes of size  $(33 \times 20)$ ,  $(41 \times 24)$ , and  $(54 \times 32)$  are used in the feature map of size  $13 \times 13$  to detect large objects. The feature map of size  $26 \times 26$  use the anchor boxes of size  $(21 \times 12)$ ,  $(24 \times 14)$ , and  $(28 \times 16)$  for the middle objects. The feature map of size  $52 \times 52$  use the anchor boxes of size  $(12 \times 17)$ ,  $(16 \times 9)$ , and  $(18 \times 11)$  to detect the small objects.

### 3.5 Repeating Block Cutting

The combination scheme of the repeating residual blocks in Darknet-53 backbone network is  $1 + 2 + 8 + 8 + 4$ , which are suitable for detect a lot of classes of generic objects. More the blocks are, the network parameters are more. Naturally, network training and testing will consume more time. There are only

4 classes of vehicle license plates, so the needed network structure should be simpler than the network structure used to detect the generic objects. We obtained the most suitable number of repeating Inception-ResNet-A block in our Darknet-74 backbone network through repeating block cutting experiments without detection accuracy loss. The most suitable combination scheme of the repeating Inception-ResNet-A block is  $1 + 4 + 4 + 4 + 4$ , as shown in Fig. 1.

## 4 Experiments

The hardware configuration of the experimental computer includes dual core Intel® Xeon® CPU E5-2650 v4 @ 2.20 GHz, the 24 GB memory, 4 pieces of Tesla P40, each of them has 24 GB memory. Software system configuration has Ubuntu 16.04LTS, CUDA 9.0, CUDNN 7.3, Python3.6 programming language, Darknet deep learning framework.

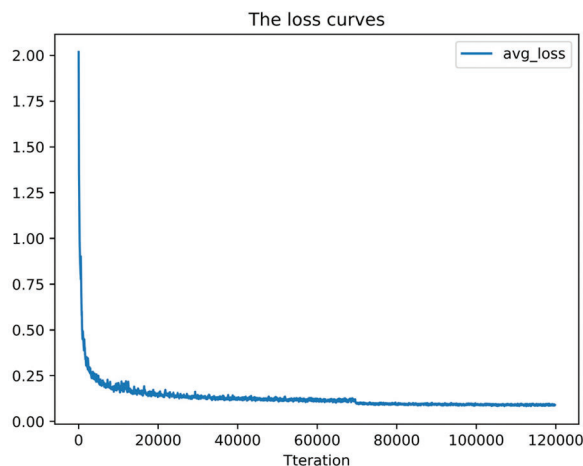
### 4.1 Vehicle License Plate Dataset

A large amount of data is necessary to train CNNs. But to our knowledges, there is no public China vehicle license plate dataset that mostly contain multi-license plate images taken in complex scene. At present, a few open Chinese vehicle datasets [30] contain only one license plate which accounts for a large proportion of pixels in the image and the characters are clear. In addition, license plates are not also classified. So, we built a China vehicle license plate dataset, which was collected from different scenes such as overpass, underpass, tunnel, floating car, fork in the road, intersection, bus station, and so on. The weather includes cloudy day, rainy day and sunny day. Almost each image contains multiple license plates. Consequently, our task is to detect multiple license plates from images with complex background.

The process of labeling dataset included two stages: manual labeling and automatic labeling. In the stage of manual labeling, volunteers manually labeled all vehicle license plates appeared in the images using the LabelImg software. The labeling requirement is that the characters of the labeled license plate should be recognized by human. 27,606 images had been labeled in the stage of manual labeling. Considering that the license plate with unrecognized characters is unnecessary, we did not label these plates. To obtain more labeled data, we adopted an automatic labeling method to label the rest of the dataset. Firstly, the manually labeled images were fed into the vehicle license plate detection network (Fig. 1) for training. The network model trained on 27,606 images was used to automatically detect license plates from the rest of dataset in turn. Finally, simple manual labeling process was carried again to correct the wrong detection of license plate. The automatic labeling method greatly reduces the labor and energy consumed by manual labeling, which made us obtain a lot of labeled data easily. At last, our experimental dataset had 62,153 labeled images, which including 106,265 pieces of license plate. There are 4 classes of China vehicle license plates, i.e., blue, yellow, white and green license plate. The 106,265 license plates include 97,410 blue plates, 7,157 yellow plates, 217 white plates, and 1,481 green plates.

### 4.2 Dataset Training

The dataset is divided in the proportion of 1:9, so the 10% of dataset is the test set, 90% of dataset continue to be divided in the proportion of 1:9 to obtain the validation set and the training set respectively. The original size of the collected image is  $1080 \times 1920$ . Before the training, the image is resized to  $416 \times 416$ , the iteration batch size is set to 4, the momentum is set to 0.9, and the small batch random gradient descent is used for optimization. The initial learning rate is set to 0.001, the epoch is set to 64, the decay is set to 0.0005. Data augmentation tricks include saturation, exposure, hue, and multi-scale training. After 100000 iterations, the network converges and the average loss decreases to 0.050756. Fig. 4 shows the average loss curve in the process of network training, it is obviously that the network loss decreases steadily and finally converges. After the network model converges, the IoU calculated on the validation set is stable at about 0.83, indicating that the accuracy of classification and location are very good.



**Figure 4:** Loss function curve

### 4.3 Ablation Experiment

We did ablation experiments depending on the three improvements proposed in Section 2. The mAP, P, R, F<sub>1</sub>, BLFOPs(billion float operations), and FPS(frames per second) of vehicle license plate detection are calculated from test dataset, as shown in Tab. 2, the first row is the original YOLOv3 without any improvements. In our ablation experiments, the combination scheme of repeating Inception-ResNet-A block that we adopted to make experiments is 1 + 2 + 8 + 8 + 4.

**Table 2:** Performance comparison in ablation experiments

NA <sup>1</sup>	IR-A <sup>2</sup>	SPP	mAP	P/%	R/%	F <sub>1</sub> /%	BFLOPS	FPS
			81.5	96.37	94.41	95.37	65.326	23
√			81.6	98.64	94.51	96.53	65.326	23
√	√		82.2	98.25	96.10	97.16	104.135	19
√		√	82.8	98.82	96.16	97.47	72.211	25
√	√	√	<b>83.1</b>	<b>98.85</b>	<b>97.46</b>	<b>98.15</b>	109.931	18

<sup>1</sup>NA = New anchors, <sup>2</sup> IR-A = Inception-ResNet-A.

The combination of new anchor boxes set, Inception-ResNet-A and SPP has the best mAP, P, R, and F<sub>1</sub>, but its network structure is more complex than the other combinations, the detection speed is relatively the slowest, as shown in Tab. 2.

### 4.4 Repeating Block Cutting Experiment

We design 4 combination schemes for the repeating Inception-ResNet-A block in Fig. 1. For each scheme, we trained the network and calculated the license plate detection performance on the test dataset. The experimental results of 4 combination schemes are shown in Tab. 3.

Obviously, the combination scheme of 1 + 4 + 4 + 4 + 4 is the best suitable scheme, which obtain the highest mAP, R, and F<sub>1</sub> and relatively the best P, BFLOPS, and FPS. Therefore, the combination scheme of 1 + 4 + 4 + 4 + 4 is used in our final improved network structure, as shown in Fig. 1.



**Table 3:** Performance comparison of different combination schemes

Repeated block	mAP	P/%	R/%	F <sub>1</sub> /%	BFLOPS	FPS
1 + 2 + 8 + 8 + 4	83.1	98.85	97.46	98.15	109.931	18
1 + 2 + 6 + 6 + 4	80.8	98.92	98.13	98.52	96.107	13
1 + 4 + 4 + 4 + 4	<b>83.4</b>	98.88	<b>98.17</b>	<b>98.52</b>	<b>89.196</b>	22
1 + 2 + 4 + 4 + 2	82.6	97.94	97.26	97.60	75.371	23

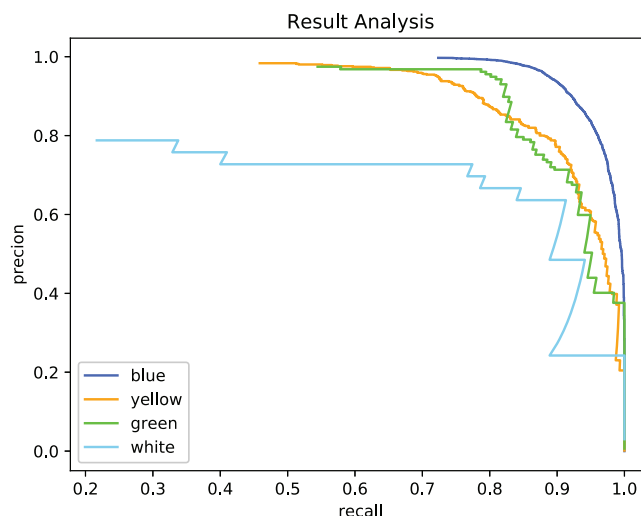
#### 4.5 Test Result

When we applied the final improved YOLOv3 network proposed in this research to detect license plates from the images of the test dataset, the detection performance is: 83.4% mAP, 98.88% precision, 98.17% recall, 98.52 F<sub>1</sub> score, 89.196 BFLOPS, and 22 FPS, as shown in [Tab. 3](#).

Some multi-license plate detection results of test dataset are shown in [Fig. 5](#).

**Figure 5:** Exhibition of some multi-license plate detection results

It can be seen from Fig. 6 that our trained network model can accurately detect multiple license plates from images in complex scenes. Our network not only accurately locate the license plate, but also accurately label the class of license plate. In addition, our network can accurately detect both big and small license plates even these plates have fuzzy characters, or dirty.



**Figure 6:** P-R curve

The results of comparing experiments are shown in Tab. 4, the other 4 compared methods are SSD, Faster RCNN, EfficientDet [31] and RetinaNet [32]. It is obviously that our method obtains the best mAP and relatively better FPS. The SSD obtain the best FPS due to the smallest input size, but the mAP is far lower than that of ours. The closest method to ours is YOLOV3, but the whole performance of our method is better than YOLOV3, as shown in Tab. 2.

**Table 4:** Comparison with others methods on our test dataset

Method	Backbone	Input size	mAP/%	FPS
SSD	VGG16	$300 \times 300$	63.94	<b>31</b>
Faster-RCNN	ResNet50	$512 \times 512$	47.34	13
EfficientDet	Efficientdet-D1	$512 \times 512$	69.35	10
RetinaNet	ResNet50	$600 \times 600$	71.97	13
YOLOv3	Darknet53	$416 \times 416$	81.5	23
Ours	Darknet74	$416 \times 416$	<b>83.4</b>	22

## 5 Conclusions

In this research, some improvements were carried out based on the YOLOv3 multi-scale detection network structure, and multi-license plate detection experiments were carried out on the Chinese vehicle license plate dataset collected and labeled by ourselves. The improvements include adopting the Inception-ResNet-A block in the backbone network, adding SPP block into the detection network, cutting the redundant repeating block, and clustering new anchor boxes. Experiments show that the improved

YOLOv3 multi-scale detection network has better robustness and generalization ability than the original YOLOv3. The precise, recall, and  $F_1$  can reach 98.88%, 98.17%, and 98.52%, respectively. At same time our improved network can ensure good detection speed. Our improved YOLOv3 multi-scale detection network can be used to detect multiple license plates under the complex scenes. Our work is a useful attempt of artificial intelligence technology in Chinese license plate detection, which can greatly improve work efficiency and save labor. Our dataset exists class-imbalance problem, which the number of blue plates is far larger than the number of other three classes, the AP of blue plate is bigger than the other three. In future work, we should collect more images that have yellow, white, and green license plate. At same time class-imbalance learning tricks will be considered.

**Funding Statement:** This study was supported by the China Sichuan Science and Technology Program under Grant 2019YFG0299, the Fundamental Research Funds of China West Normal University under Grant 19B045, and the Research Foundation for Talents of China Normal University under Grant 17YC163.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] K. J. Zhu, Q. H. Hao, L. S.Y. and C. Y. Hu, "Overview of license plate recognition," *Modern Information Technology*, vol. 2, no. 8, pp. 4–6, 2018.
- [2] H. Li, P. Wang and C. H. Shen, "Toward end-to-end car license plate detection and recognition with deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 1126–1136, 2019.
- [3] F. D. Kurpiel, R. Minetto and B. T. Nassu, "Convolutional neural networks for license plate detection in images," in *Proc. of the IEEE Int. Conf. on Image Processing*, Beijing, China, pp. 2381–8549, 2017.
- [4] Q. Fu, Y. Shen and Z. H. Guo, "License plate detection using deep cascaded convolutional neural networks in complex scenes," in *Proc. of the Int. Conf. on Neural Information Processing*, Sydney, NSW, Australia, pp. 696–706, 2017.
- [5] M. A. Rafique, W. Pedrycz and M. Jeon, "Vehicle license plate detection using region-based convolutional neural networks," *Soft Computing China*, vol. 22, no. 19, pp. 6429–6440, 2018.
- [6] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018. [Online]. Available: <https://arxiv.org/abs/1804.02767>.
- [7] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: unified, real-time object detection," in *Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 779–788, 2016.
- [8] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 7263–7271, 2017.
- [9] C. Szegedy, S. Ioffffe, V. Vanhoucke and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. of the Thirty-First AAAI Conf. on Artificial Intelligence (AAAI)*, San Francisco, USA, pp. 4278–4284, 2017.
- [10] K. M. He, X. Y. Zhang, S. Q. Ren and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.
- [11] C. Tan and J. Cao, "An algorithm for license plate location based on color and texture," in *Proc. of the 2013 5th Int. Conf. on Intelligent Human-Machine Systems and Cybernetics*, Hangzhou, China, pp. 356–359, 2013.
- [12] J. Muhammad and H. Altun, "Improved license plate detection using HOG-based features and genetic algorithm," in *Proc. of the 2016 24th Signal Processing and Communication Application Conf. (SIU)*, Zonguldak, Turkey, pp. 1269–1272, 2016.

- [13] X. Jiang, Y. D. Huang and G. Q. Li, "Fusion of texture features and color information of the license plate location algorithm," in *Proc. of the 2012 Int. Conf. on Wavelet Analysis and Pattern Recognition (ICWAPR)*, Xi'an, China, pp. 15–19, 2012.
- [14] A. H. Ashtari, M. J. Nordin and M. Fathy, "An Iranian license plate recognition system based on color features," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 4, pp. 1690–1705, 2014.
- [15] J. Dun, S. Zhang, X. Ye and Y. Zhang, "Zhang Chinese license plate localization in multi-lane with complex background based on concomitant colors," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 3, pp. 51–61, 2015.
- [16] Y. M. Tian, J. Song, X. D. Zhang, P. Y. Shen, L. Zhang *et al.*, "An algorithm combined with color differential models for license-plate location," *Neurocomputing*, vol. 2125, no. 2016, pp. 22–35, 2015.
- [17] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. of 2014 the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Colombia, pp. 580–587, 2014.
- [18] R. Girshick, "Fast R-CNN," in *Proc. of 2015 the IEEE Int. Conf. on Computer Vision (ICCV)*, Santiago, Chile, pp. 1440–1448, 2015.
- [19] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Advances in Aerial Information Processing Systems*, vol. 39, no. 6, pp. 1137–1149, 2017.
- [20] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed *et al.*, "SSD: Single shot multibox detector," in *Proc. of the European Conf. on Computer Vision (ECCV)*, Amsterdam, Holland, pp. 21–37, 2016.
- [21] C. Y. Fu, W. Liu, A. Ranga, A. Tyagi and A. C. Berg, "DSSD: Deconvolutional single shot detector," 2018. [Online]. Available: <https://arxiv.org/abs/1701.06659>.
- [22] S. Liu, D. Huang and Y. Wang, "Receptive field block net for accurate and fast object detection," 2017. [Online]. Available: <https://arxiv.org/abs/1701.06659>.
- [23] J. Ren, X. Chen, J. Liu, W. X. Sun, J. H. Pang *et al.*, "Accurate single stage detector using recurrent rolling convolution," in *Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 5420–5428, 2017.
- [24] Z. X. Li and F. Q. Zhou, "FSSD: Feature fusion single shot multibox detector," 2017. [Online]. Available: <https://arxiv.org/abs/1712.00960>.
- [25] C. Szegedy, W. Liu, Y. Q. Jia, P. Sermanet, S. Reed *et al.*, "Going deeper with convolutions," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Boston, MA, USA, pp. 1–9, 2015.
- [26] S. Ioffffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," 2015. [Online]. Available: <https://arxiv.org/abs/1502.03167>.
- [27] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, pp. 2818–2826, 2016.
- [28] Z. C. Huang, J. L. Wang, X. S. Fu, T. Yu, Y. Q. Guo *et al.*, "DC-SPP-YOLO: dense connection and spatial pyramid pooling based YOLO for object detection," *Information Sciences*, vol. 522, pp. 241–258, 2020.
- [29] T. Y. Lin, P. Dollar, R. Girshick and K. M. He, "Feature pyramid networks for object detection," in *Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 2117–2125, 2017.
- [30] S. M. Silva and C. R. Jung, "License plate detection and recognition in unconstrained scenarios," in *Proc. of the 2018 European Conf. on Computer Vision (ECCV)*, Munich, Germany, pp. 580–596, 2019.
- [31] M. G. Tan, R. M. Pang and Q. V. Le, "EfficientDet: Scalable and efficient object detection," 2020. [Online]. Available: <https://arxiv.org/abs/1712.00960>.
- [32] T. Y. Lin, P. Goyal, R. B. Girshick, K. M. He and P. Doll, "Focal loss for dense object detection," 2017. [Online]. Available: <https://arxiv.org/abs/1708.02002>.