ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0090

Research on License Plate Detection Method based on Improved YOLOv4

Lifen Li, Haitao Li

School of Control and Computer Engineering, North China Electric Power University, Baoding 071000, China

Abstract

Aiming at the low accuracy and slow speed of license plate detection in complex environments, an improved YOLOv4 license plate detection algorithm is proposed. First, the original backbone feature extraction network was replaced with MobileNet, and dilated convolution was added to the latter layers of the backbone network, which reduced the amount of parameters and computation of the network model, and improved the detection accuracy and speed; then, an additional pair of small The target-sensitive feature layer strengthens the extraction of shallow feature information and improves the detection rate of small targets; then, the ordinary convolution in the enhanced feature extraction network is replaced with a depthwise separable convolution to further reduce the number of parameters of the model; Finally, the K-means clustering algorithm is used for the license plate dataset to obtain the appropriate number and size of a priori boxes to improve the license plate recognition accuracy. Experiments are carried out on the LP-CCPD data set, and the results show that the improved algorithm has an accuracy rate of 97.58%, and the detection speed can reach 88FPS. Compared with the original YOLOv4 algorithm, the improved algorithm has better accuracy and real-time performance, which can basically meet the actual needs of license plate detection and better provide help for the intelligent traffic management system.

Keywords

Target Detection; YOLOv4; License Plate Detection; MobileNet.

1. Introduction

In recent years, with the rapid development of society and economy, the number of private cars, buses, online car-hailing and taxis has increased year by year. However, the accompanying problem is that traffic management and vehicle management are becoming more and more difficult, in order to facilitate the management of numerous vehicles, a modern intelligent transportation system has emerged. As an important part of the development of smart cities, automatic license plate recognition (ALPR) technology in intelligent transportation systems is widely used in many scenarios, such as high-speed automatic ETC, parking lot fee management, traffic law enforcement, community access control, etc.

License plate recognition is generally divided into two parts, license plate location detection and license plate character recognition[1]. The former is particularly important in license plate recognition, and the accuracy of license plate detection will largely affect subsequent character recognition. Therefore, license plate detection is the first and most important stage of license plate recognition. At present, there are two main types of license plate detection technologies at home and abroad: license plate detection technology based on traditional image processing and license plate detection technology based on deep learning.

Traditional license plate detection methods mainly include detection based on edge features, color features and texture features[2]. The detection method based on edge features is mainly based on the difference in image density. The edge density of the area near the license plate is generally higher than that of other areas. Sobel and Canny are usually used. and other edge detection operators for license plate detection. This method is simple and efficient in calculation, but for blurred images, the edge is not obvious, and the detection effect will be poor. The detection method based on color features is mainly aimed at the inconsistency of the color of the license plate and the body. Although this method can be detected in a fuzzy situation, if the color of the license plate and the body are very similar, other parts of the body will be mistakenly detected as the license plate. In addition, changes in light intensity can have a large impact on the performance of this method. The detection method based on texture features mainly uses the difference in the distribution of texture information in the image. Generally speaking, the texture information of the area where the license plate is located is much richer than that of other places. This method is more suitable for images with strong texture feature discrimination[3]. In most cases, the performance is good, the detection speed is fast, and the detection accuracy is relatively high, but its anti-interference ability is weak and requires higher image resolution.

The license plate detection method based on deep learning mainly uses deep convolutional neural network (CNN) for learning and feature extraction to detect the license plate in the image[4]. Compared with the traditional detection method based on edge information and other features, this method has better performance. The generalization, robustness, and detection effect of the system can achieve end-to-end training and detection. Target detection algorithms based on deep learning are more and more widely used. At present, there are mainly two types, two-stage detection network and one-stage detection network. The former mainly includes R-CNN series, such as R-CNN, Fast R-CNN, Faster R-CNN CNN[6], etc. The characteristics of these networks are to divide the target detection task into two steps: first detect possible target candidate bounding boxes (Region Proposal), and then extract the features of these candidate boxes, infer the confidence of these candidate boxes corresponding to each target, so as to Detect the relative position and type of the target in the picture. The latter method directly obtains the category probability and position of the target through one feature extraction, which is faster than the two-stage detection method. Typical one-stage detection methods include SSD[8], YOLO series[9], Retina-Net, etc.

After the target detection algorithm based on deep learning is applied to license plate detection, its accuracy has been greatly improved, but in complex scenarios[10], there is still room for improvement. In order to improve the accuracy of license plate detection in complex scenarios, this paper an improved license plate detection algorithm is proposed and compared with other target detection algorithms. The experimental results show that the improved algorithm improves the accuracy and detection speed, which proves the effectiveness of the algorithm.

2. Related Algorithms

2.1 YOLOv4 Algorithm

YOLOv4 is the fourth version of the YOLO (You Only Look Once) series of target detection algorithms[11]. Two years after YOLOv3[12] was proposed, the original author issued a statement that he would not continue to update the YOLO algorithm. After that, Alexey and others from Russia communicated with the original author. Later, they officially named their research YOLOv4. Compared with the YOLOv3 algorithm, its network structure has not changed much, but many practical tips have been added. On the basis of the original YOLOv3 target detection algorithm, from data preprocessing, loss function, activation function and network structure, etc. In terms of optimization methods, YOLOv3 has been improved, and the accuracy is greatly improved without reducing the detection speed of the current algorithm.

 Volume 8 Issue 4, 2022
DOI: 10.6919/ICJE.202204_8(4).0090

2.2 Principles of the YOLOv4 Algorithm

YOLOv4 first performs feature extraction on the input image through the backbone network (feature extraction network), and then divides the input image into M×M grids. If the center of the target falls in a grid, then the main task of the grid is the detection target. After feature extraction, the features of three scales are output, and then feature fusion and feature stacking are performed through the SPP network and PANet network, and finally sent to the YOLOHead part for prediction, regression analysis is performed on the features of the three scales, and the NMS algorithm is used to compare confidence. The low prediction frame is deleted, and the frame with higher confidence is retained as the target detection frame, and finally the position and category of the target are obtained.

2.3 YOLOv4 Network Structure

The network structure of YOLOv4 is shown in Figure 1. CSPDarkNet53 is used as the backbone network. The Neck part includes the SPP module and the PANet module. The SPP module is an additional module, and the PANet module is a feature fusion module. Finally, YOLOHead uses the obtained features for prediction.

(1) Backbone network: The backbone feature extraction network of YOLOv4 adopts CSPDark-net53 network, and YOLOv3's Compared with the Dark-net53 network, CSPDark-net53 has 5 more CSP modules. The CSP module divides the feature map of the base layer into two parts, and then merges them through a cross-stage hierarchical structure. The CSP module can achieve richer gradient combinations, greatly reducing the amount of computation while ensuring the speed and accuracy of detection.

(2) SPP module: Spatial Pyramid Pooling. After the convolution of the last feature layer of the backbone feature extraction network, the SPP module uses the convolutional feature layers of four different scales. Pooling is used for processing. The maximum pooling cores are 13×13, 9×9, 5×5, and 1×1, of which 1×1 is equivalent to no processing. The max pooling operation is then stacked to form a one-dimensional vector. The SPP structure makes the aspect ratio and size of the input feature map arbitrary, and then converts it into a fixed-size feature vector, which can also greatly increase the receptive field and significantly separate important contextual features.

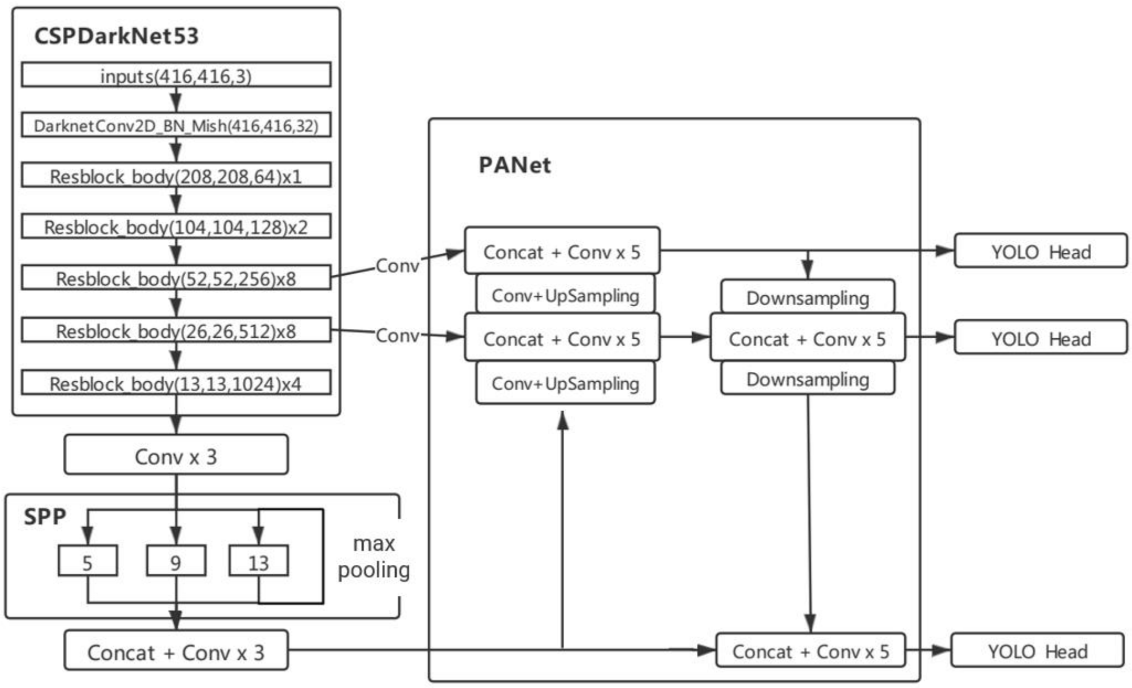

Figure 1. YOLOv4 network structure

 Volume 8 Issue 4, 2022
DOI: 10.6919/ICJE.202204_8(4).0090

(3) PANet structure: Path Aggregation Network, including FPN module and PAN module. In YOLOv4, PANet is used instead of FPN in YOLOv3 as a parameter aggregation method, so that the FPN structure and PAN structure are combined to operate, and the FPN layer Strong semantic features are conveyed top-down, while feature pyramid conveys strong localization features bottom-up, suitable for different detector levels for parameter aggregation from different backbone layers. And the original PANet method is modified, using tensor connection (Concatenation) instead of the original addition connection (Addition). Different from the upsampling method used by FPN in the YOLOv3 algorithm, YOLOv4 first transmits the high-level semantic feature information to the low-level network through upsampling, and then fuses it with the low-level high-resolution semantic feature information to improve the accuracy of small targets. detection effect.

(4) YOLO Head: In YOLOv4, the detection head of YOLOv3 is used as the head for multi-scale prediction, which improves the detection performance of targets of different sizes.

2.4 Introduction to MobileNetV1

MobilenetV1 is a streamlined lightweight convolutional neural network proposed by Google for mobile embedded devices such as mobile phones[13]. Its core idea is based on depthwise separable convolution. Different from the traditional convolution, the depthwise separable convolution is to decompose the standard convolution in the neural network into depthwise convolution and pointwise convolution (1×1 convolution kernel), as shown in Figure 2, which can reduce a lot of the redundant information of expressing features can also greatly reduce the number of parameters and the amount of calculation, and it is more used in embedded devices. Depth convolution is to apply different convolution kernels to each channel, and finally the convolution result of each channel is used as the final depth convolution result. Point-by-point convolution uses the result of depth convolution as input, and uses Linear combination is performed to generate new features. Compared with depthwise convolution, point-by-point convolution can change the number of channels, and can complete the function of dimension increase or dimension reduction. Through convolution decomposition, the amount of calculation can be reduced, a large number of parameters can be saved, the operation speed can be accelerated, and the training problems caused by overfitting can be reduced.

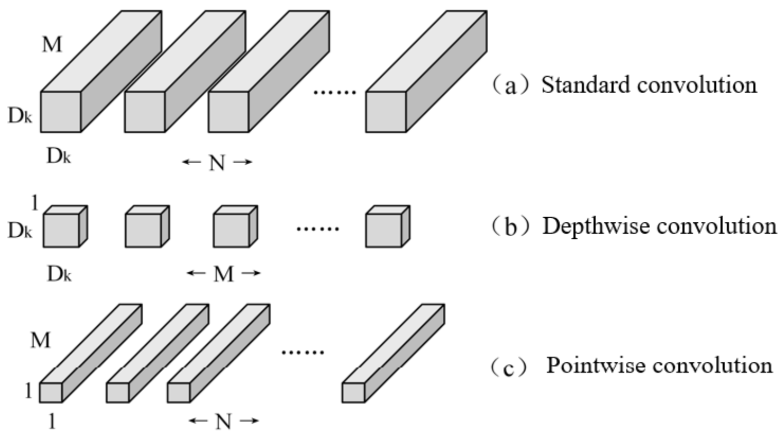

Figure 2. Schematic diagram of standard convolution and depthwise separable convolution

The computation ratio of standard convolution and depthwise separable convolution is:

$$\frac{D_k \times D_k \times M \times D_f \times D_f + M \times N \times D_f \times D_f}{D_k \times D_k \times M \times N \times D_f \times D_f} = \frac{1}{D_k^2} + \frac{1}{N} < 1 \tag{1}$$

 Volume 8 Issue 4, 2022
DOI: 10.6919/ICJE.202204_8(4).0090

where M and N are the number of channels of the input and output feature maps, respectively, the size of the input feature map is $D_f \times D_f \times M$, and the size of the output feature map obtained according to the convolution kernel $D_k \times D_k \times M \times N$ is $D_f \times D_f \times N$.

We can see that the ratio of the two is less than 1, which means that the computational complexity of the depthwise separable convolution is greatly reduced, which can improve the detection performance of the network.

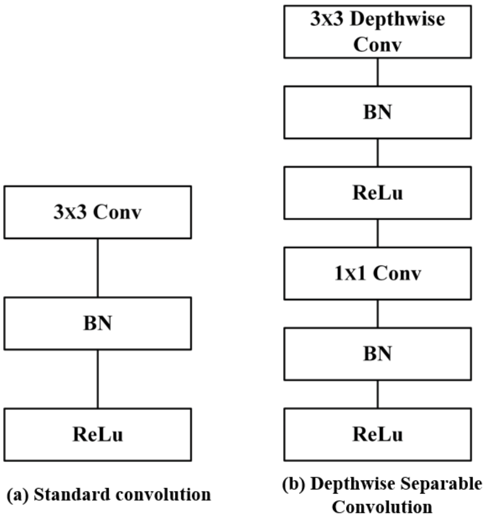

Figure 3. Comparison of standard convolution and depthwise separable convolution structures

Figure 3 is a structural comparison diagram of standard convolution and depthwise separable convolution. The left side is standard convolution, which first goes through 3×3 convolution, then goes through the normalization operation (BN, Batch Normalization) layer, and finally goes through the nonlinear The activation unit ReLu (Rectified Linear Unit) function; the right side is the depthwise separable convolution, and the standard convolution of 3×3 is decomposed into two independent modules, namely 3×3 depthwise convolution (Depthwise Conv) and 1×1 convolution, and the normalization operation and ReLu activation function are added after the output result. The combination of these two modules replaces the standard convolution, making it more efficient in theory, and a large number of convolution operations of 1×1 is used, which can be optimized by mathematical libraries to improve the speed of calculation.

3. Improved YOLOv4 License Plate Detection Algorithm

3.1 Improvement of Backbone Feature Network

The backbone feature extraction network of YOLOv4 is CSPDarknet53. Although this network can extract effective feature information, its network structure is complex and its parameters are complex, which is not very practical[14]. Therefore, adding MobileNet series networks can reduce parameters, and to a certain extent can Improve detection speed and accuracy.

3.1.1 MobileNetV1 as the Backbone Network

Replace CSPDarknet53 with MobileNetV1 as the new backbone feature extraction network. For deep networks, a series of convolution operations are often used for feature extraction[16]. During the convolution operation, several feature layers are obtained. While these feature layers are performing feature extraction, the height and width are usually constant. Compressed, but the number of channels is constantly expanding. For MobileNetV1, if you find the feature layer with the same height and width as CSPDarknet53, you can replace it, mainly replacing the traditional convolution part in CSPDarknet53 with depthwise separable convolution.

Volume 8 Issue 4, 2022
DOI: 10.6919/ICJE.202204_8(4).0090

3.1.2 Add Dilated Convolution

The fully convolutional neural network generally reduces the size of the image and expands the receptive field by downsampling the pooling layer, and then restores the image to its original size through upsampling. This process will inevitably lead to the loss of image information and expansion[17]. Convolution can avoid such problems to a certain extent.

Dilated convolution (also known as hole convolution), unlike the standard convolution kernel, the dilated convolution adds some holes to the convolution kernel, so that the number of parameters can be increased without loss of feature information. Expand the receptive field range of the feature map under the condition of. As shown in Figure 4, (a) represents standard convolution, (b) represents dilated convolution, the size of the convolution kernel of both is 3×3, and the dilated convolution also has a hyperparameter expansion rate (dilation rate), which represents the number of intervals of the convolution kernel, the dilation rate of the standard convolution is 1, and the dilation rate of the dilated convolution in (b) is 2.

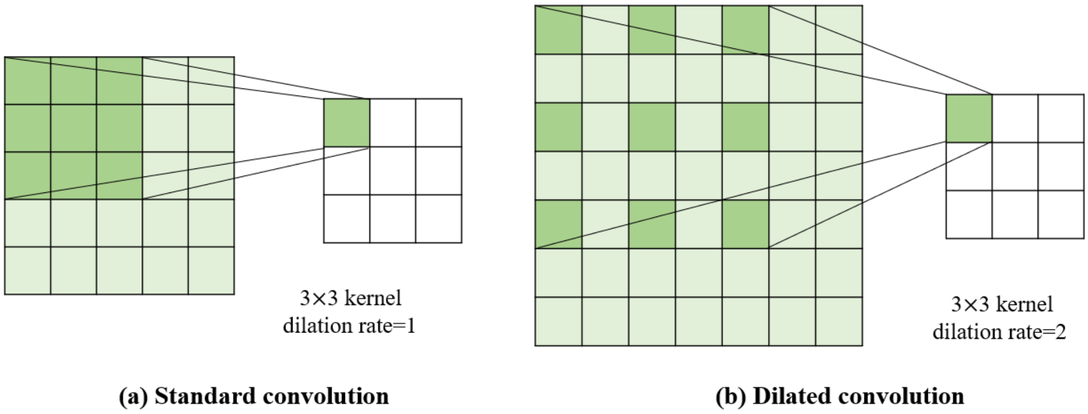

Figure 4. Standard convolution and dilated convolution

In order to improve the detection accuracy of small target objects, dilated convolution is added after the feature layer of the new feature extraction network, so that the low-level feature semantic information and high-level feature semantic information can be linked to avoid the loss of image information, increase the receptive field and accurately detect small target objects.

3.2 Enhanced Feature Extraction Network Improvements

3.2.1 Model Computation

Due to the large amount of calculation of the convolution block for enhancing the feature extraction network model, the speed of license plate detection is relatively slow. Therefore, the depthwise separable convolution can be used to replace the 3×3 general convolution, which can reduce the number of parameters of the model and greatly reduce the number of parameters. Improve the detection speed of the model.

3.2.2 Multiscale Feature Detection

Similarly, in order to enable the prediction network to obtain more small target feature information and improve the accuracy of license plate detection, this paper will improve on the basis of multi-scale feature detection, adding a low-dimensional feature layer to strengthen the learning of shallow features ability. As a result, the feature layers used for detection have changed from 3 to 4. According to the size of the input image, it is 416×416, and the added feature layer size is 104×104. In addition, an upsampling is added on the basis of the original downsampling feature map fusion, which is convenient for splicing with the feature map of the upper layer, and outputs the fourth detection head for target detection [18]. The improved overall network structure is shown in Figure 5.

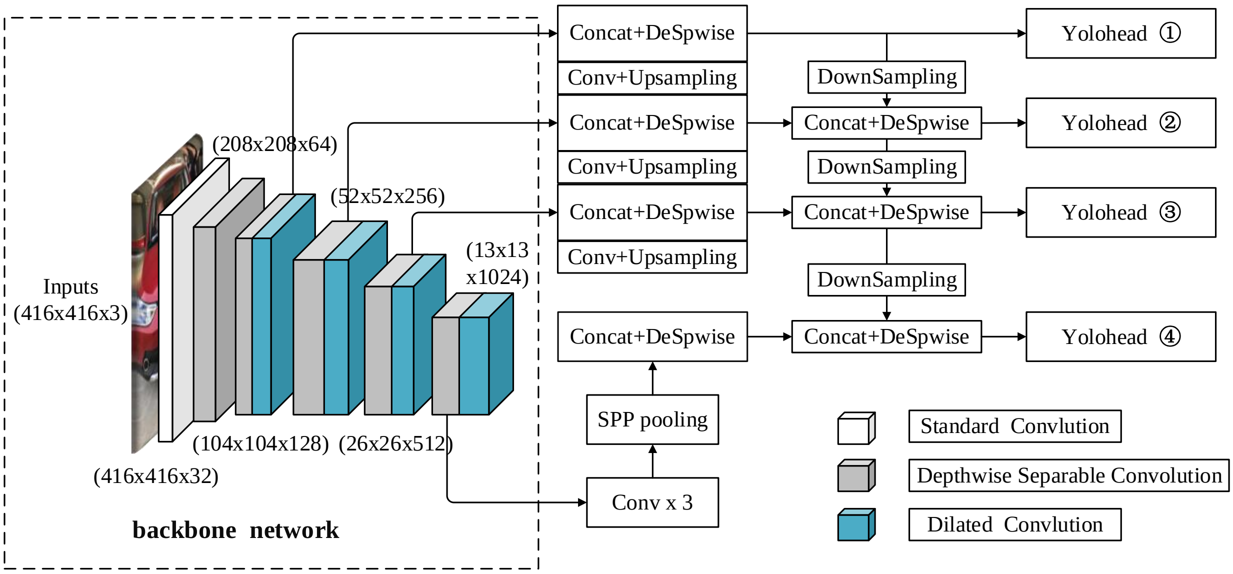

Figure 5. The overall network structure of YOLOv4 after improvement

3.3 Update of a Priori Box

YOLOv4 applies the idea of the anchor box when predicting the target, and uses the K-means clustering algorithm to obtain K suitable a priori boxes based on the manually labeled target frame (ground truth), which can improve the target detection efficiency[20]. effectiveness. In the traditional YOLOv4 network, 3 a priori boxes are set for the feature layer of each scale, and a total of 9 a priori boxes of different sizes can be clustered. In view of the situation where there are small targets in the license plate data set in this paper, a feature layer is added after the network improvement, so 12 a priori boxes need to be clustered, and the original a priori box parameters are obtained by clustering the public data set[21]. The types of sets are rich, the detection target is relatively large, and the size of the obtained a priori frame is relatively large, which is not suitable for the license plate data set in this paper. Therefore, it is necessary to use the K-means clustering algorithm to perform clustering analysis again according to the size of the license plate in the data set. Better license plate detection.

In the license plate detection task, clustering is to obtain a larger value of the intersection ratio (IOU) of the anchor box and the ground truth, and the appropriate a priori box size will also affect the accuracy of license plate detection, so general clustering cannot be used. The Euclidean distance of the class method, but the IOU distance formula is used instead of the Euclidean distance. The formula is as follows:

$$d(box, center) = 1 - IOU(box, center) \quad (2)$$

Among them, box represents the real box, center represents the cluster center, the larger the IOU, the smaller the distance, and the closer the anchor box is to the real box. The intersection and union ratio, IOU, represents the degree of overlap between the predicted frame and the real frame. The formula is:

$$IOU(D_P, D_T) = \frac{D_P \cap D_T}{D_P \cup D_T} \quad (3)$$

where DP represents the predicted box and DT represents the ground-truth box.

Through the clustering calculation of the labeled license plate data set in this paper, the relationship between the number of anchor boxes K and the intersection ratio IOU is obtained as shown in the

ISSN: 2414-1895 DOI: 10.6919/ICJE.202204_8(4).0090

figure. It can be seen from the figure that with the increase of the number of anchor boxes, the IOU is also Gradually increasing, when K >10, the curve gradually becomes flat. Since the algorithm in this paper selects 4 scale feature layers, 12 groups of anchors are selected, and the results are: (21,10),(29,15),(30,17),(37,20),(56,18),(57,13),(62,15),(79,36),(89,42),(95,45),(111,26),(158,105). Assign these anchors according to the feature layer scale, and use (21,10),(29,15),(30,17) on the 104 × 104 feature layer, mainly used to detect small license plate targets, in the 52 × 52 feature layer Use (37,20),(56,18),(57,13) on the 26 × 26 feature layer use (62,15),(79,36),(89,42), and finally feel at the maximum (95, 45),(111, 26),(158, 105) are used on the feature layer of the wild for detecting larger license plate targets.

4. Experimental Results and Analysis

4.1 Experiment Platform

This experiment is done using the TensorFlow framework under the Windows10 operating system. The relevant configuration of the computer used: CPU is IntelCoreTM i7-9700K, memory is 32G; GPU is NVIDIA GeForce RTX2080Ti, video memory is 8G, running platform is Pycharm, Python version is 3.6.2, and Cuda10.0 and Cudnn7.5.1 are installed at the same time the auxiliary GPU is used for accelerated operations. In order to support the smooth running of the code, a series of third-party libraries are also installed, such as Keras 2.1.5, OpenCV2.0, numpy1.19 and so on.

4.2 Experimental Dataset

The CCPD data set is an opensource data set in the field of license plate recognition in China[23], and it is also the largest public license plate data set at home and abroad. The data set was released in 2018 and updated in 2019. There are now more than 300,000 Chinese license plate pictures. Each picture is taken by the staff with handheld shooting equipment. The shooting location is flexible and the shooting angle is changeable, realizing the diversity of data, such as tilt angle, strong light, rain and snow weather, etc., which can fully meet the complex scene. experiment below. CCPD has a total of 9 sub-datasets, each of which has its own characteristics. Table 1 shows the number and characteristics of each sub-dataset.

Table 1. CCPD dataset

CCPD	Quantity/k	Feature Description
Base	200	regular license plate
FN	20	Poor distance, too far or too close
DB	20	Poor lighting, bright or dark, strong light
Rotate	10	Angle tilt, horizontal tilt 20° to 25°
Tilt	10	Angle tilt, horizontal tilt 15° to 45°
Weather	10	Bad weather, rain, snow and fog
Blur	5	blur caused by shooting
Challenge	10	Multiple Scenarios, Challenging
NP	5	No license plate

The data set of the experiment in this paper is to select 500 pieces of the first 8 sub-data sets, and collect 500 foreign license plate pictures from the Internet. These pictures are mixed together to form a new data set LP-CCPD. The data set is based on 8:2 ways to divide: 80% for the training set and 20% for the test set.

 Volume 8 Issue 4, 2022
DOI: 10.6919/ICJE.202204_8(4).0090

4.3 Data Annotation

The data set needs to be labeled before training. Refer to the format of the Pascal voc 2007 data set, and use the LabelImg labeling software to manually label the pictures. There is only one class License_plate in the data set. After the labeling, the corresponding XML format file is generated to facilitate the training of the network.

4.4 Evaluation Indicators

In the target detection task, Precision, Recall, Average Precision (AP) and mAP are generally used as the commonly used evaluation criteria to accurately evaluate the detection effect of the model.

in:

$$Precesion = \frac{TP}{TP+FP} \qquad Recall = \frac{TP}{TP+FN} \tag{4}$$

Precesion represents the proportion of correctly detected license plates among all detected targets; Recall represents the proportion of license plates detected by the network among all license plates, reflecting the detection capability of the model. TP represents the number of correctly detected license plates, FP represents the number of incorrectly detected license plates, and FN represents the number of license plates that were not detected.

4.5 Loss Function

The loss function of the YOLOv4 network consists of 3 parts, the bounding box regression loss L_{ciou}, the classification loss L_{class} and the confidence loss L_{conf}, when there is no predicted target in a bounding box, only the confidence loss is calculated, if there is a target in the box, all three losses are calculated. If the loss is smaller, it means that the difference between the predicted value and the real value is smaller, which means that the license plate detection network is well trained.

$$Loss = L_{ciou} + L_{class} + L_{conf} \tag{5}$$

$$L_{ciou} = 1 - IOU(D_P, D_T) + \frac{\rho^2(D,E)}{c^2} + \alpha\nu \tag{6}$$

$$\alpha = \frac{\nu}{(1 - IOU(D_P, D_T)) + \nu} \tag{7}$$

$$\nu = \frac{4}{\pi^2}(arctan\frac{w_T}{h_T} - arctan\frac{w}{h})^2 \tag{8}$$

$$L_{class} = \sum_{i=0}^{M^2} I_{ij}^{obj} \left\{ \hat{P}_i^j(c)\log P_i^j(c) + \left[1-\hat{P}_i^j(c)\right]\log\left[1-P_i^j(c)\right] \right\} \tag{9}$$

$$\begin{aligned} L_{conf} &= \sum_{i=0}^{M^2}\sum_{j=0}^{F} I_{ij}^{obj}\left[\hat{C}_i^j\log(C_i^j) + (1-\hat{C}_i^j)\log(1-C_i^j)\right] \\ &+ \lambda_{noobj}\sum_{i=0}^{M^2}\sum_{j=0}^{F} I_{ij}^{noobj}\left[\hat{C}_i^j\log(C_i^j) + (1-\hat{C}_i^j)\log(1-C_i^j)\right] \end{aligned} \tag{10}$$

ISSN: 2414-1895
DOI: 10.6919/ICJE.202204_8(4).0090

In the formula: $IOU(D_P, D_T)$ represents the intersection ratio between the predicted frame and the real frame; $\rho^2(D, E)$ represents the euclidean distance between the center E of the predicted box and the center point D of the real box ; c is the diagonal distance between the enclosed area containing the predicted frame and the real frame; α is the weight function; ν represents the aspect ratio similarity measure distance; w_T , h_T are the width and height of the real frame; w, h is the width and height of the prediction box; M^2 represents the number of grids; F is the number of prediction boxes generated on each grid; I_{ij}^{obj} indicates that there is a target in the jth prediction box generated on the ith grid, and the I_{ij}^{noobj} prediction box indicates that there is no target; $P_i^j(c)$ represents the predicted probability that the target in the box belongs to a certain category c, $\hat{P}_i^j(c)$ represents the true probability that the target in the box belongs to a certain category c; C_i^j represents the prediction confidence, $\hat{C}_i^j$ represents the true confidence; λ_{noobj} is the weight coefficient; n is the number of target categories.

4.6 Model Training

The training environment is based on the platform, framework and third-party related libraries described in 1. Before training, adjust the relevant parameters to meet the training requirements of the network data set in this paper. The adjusted parameters are shown in Table 2. The initial settings are as follows: The learning rate is 0.001, and the maximum number of iterations for training is 130. At the same time, in order to better converge the model, when the number of iterations is 70, the learning rate is adjusted to 0.001. The network construction is completed according to the improved part, and the YOLOv4-M1 model is trained first in the experiment.

Table 2. Part of the training parameters

Parameter name	Parameter value
learning rate	0.001
epoch	130
batch size	8

5. Experimental Results and Analysis

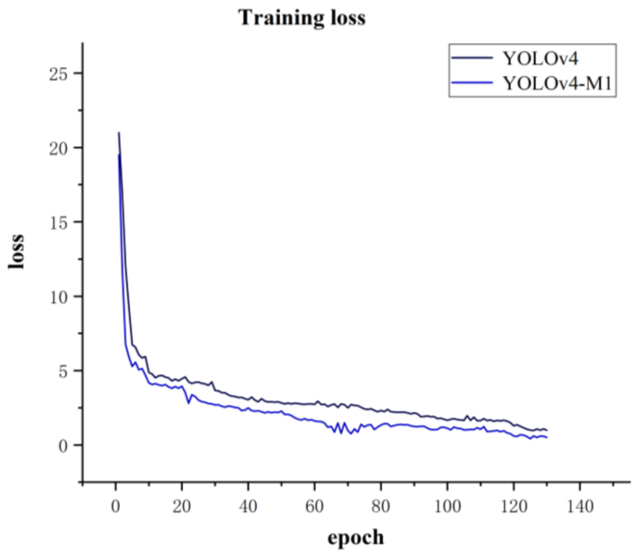

Figure 6. Training loss comparison curve

The experiment is to train the data according to the parameters in Table 2. Figure 6 shows the comparison curve of the loss change between YOLOv4 and the improved algorithm in this paper during the training process. As shown in the figure, the black represents the YOLOv4 algorithm curve, and the blue represents the YOLOv4-M1 algorithm curve. The abscissa represents the training round, and the ordinate represents the loss value.

As can be seen from the figure, the initial loss value of the original YOLOv4 is around 26, the initial loss value of the two improved algorithms is around 20, and the loss value of the two algorithms is decreased at the beginning, with the training rounds. Increase, the loss value fluctuates, and the loss value of the original algorithm is still relatively large. After the training round is 90, the fluctuation of the loss value becomes smaller, and finally it basically reaches a stable state. Compared with the original algorithm, the model loss of the improved algorithm is reduced, the accuracy and detection speed of the algorithm are significantly improved, and the detection effect is better.

Figures 7 and 8 are the comparison charts of the detection results before and after the algorithm improvement, showing the detection results in different situations in complex environments, including normal, inclined angle, bad weather, multi-vehicle and other scenarios, each group of experiments uses the same test picture.

As can be seen from the figure, the original algorithm can detect the license plate in the case of strong light, bad weather and tilt angle, but the accuracy is low, and it cannot detect all the license plates in the case of multiple vehicles; the improved algorithm It is slightly better than the original algorithm in terms of accuracy. In the multi-vehicle scenario, the detection effect is also due to the original algorithm, and the accuracy rate is higher. To sum up, the performance of the improved algorithm is improved to a certain extent compared with the original algorithm.

Figure 7. YOLOv4

Figure 8. YOLOv4-M1

6. Comparative Experiment

In order to verify the performance of the improved network, a comparison experiment was set up to compare with Faster R-CNN, SSD, YOLOv3, and YOLOv4 target detection algorithms in all aspects. The comparison experiment also used the same training set and test set, the same training batch, the input dimensions of the images are also resized to 416×416. Finally, the experimental results are analyzed, and the mean average precision (mAP), recognition frames per second (FPS) and model size of the above algorithms are counted, as shown in Table 3:

Table 3. Different algorithm detection results

Algorithm	Backbone Feature Extraction Network	FPS	mAP
Faster R-CNN	Resnet-50	0.97	97.28%
SSD	VGG16	35	95.41%
YOLOv3	Darknet-53	41	94.19%
YOLOv4	CSPDarknet-53	53	94.42%
YOLOv4-M1	MobileNetV1	88	97.58%

It can be seen from the table that the mAP values of the improved algorithm have reached 97.58%, which is similar to that of the Faster R-CNN algorithm, but the detection rate is much faster. Compared with SSD, YOLOv3, and the original YOLOv4 algorithm, mAP It can be seen that the improved algorithm in this paper is better than the above four algorithms in terms of license plate detection performance, and can simultaneously Taking into account the detection accuracy and detection rate, it can better complete the task of license plate detection and meet the real-time requirements as much as possible.

7. Conclusion

Aiming at the problems of low accuracy and slow speed of license plate location of existing algorithms in various complex scenarios, this paper proposes a license plate detection algorithm based on improved YOLOv4, and uses the public dataset CCPD to conduct license plate detection experiments. Use MobileNet lightweight network to replace CSPDarkNet53 as the backbone network for extracting feature information, appropriately reduce the amounts of parameters and computation for feature extraction; add dilated convolution to the new backbone network, improve multi-scale feature detection, a priori frame, improve the detection ability of small target objects; improving and strengthening the feature extraction network can further reduce the amounts of parameters. Finally, the accuracy, detection speed, mAP value, etc. are used as evaluation criteria, and a variety of algorithms are used to compare experiments on the data set to verify the feasibility of the improved algorithm. The experimental results show that the algorithm in this paper can meet the accuracy and real-time requirements of license plate detection on the basis of ensuring a high accuracy rate and a fast detection speed.

Although the algorithm in this paper has achieved a good detection effect on the license plate detection task, there is still room for improvement. When multiple license plates appear on a picture, there will be missed detection. The next step will be for this scenario, continue to optimize the algorithm model, further improve the network and improve the detection performance of the algorithm.

References

[1] Hashmi S N, Kumar K, Khandelwal S, et al. Real Time License Plate Recognition from Video Streams using Deep Learning[J]. International Journal of Information Retrieval Research. Vol. 1 (2019) No. 9, p. 65-87.

[2] YANG D, CHEN S, LIU J. License Plate Location Algorithm BasedOn Background and Character Color Feature[J]. Computer Applications and Software, 2018.

[3] ZHAO Wei, ZHANG Nan-nan, Simulation of License Plate Image Enhancement Algorithm in Severe Hazy Weather[J]. Computer Simulation. Vol. 3 (2019) No. 36, p. 207-211.

[4] XU C, H ZHANG, WANG W, et al. License Plate Recognition System Based on Deep Learning[C]// 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA). IEEE, 2020.

[5] PENG Huiling: Research on License Plate Location and Recognition Algorithm Based on Convolution Neural Network (Degree of Master of Engineering, Nanjing University of Posts and Telecommunications, China 2021).p.1-53.).

[6] WANG Yan, ZHANG Jikai, YIN Qian, License Plate Recognition Algorithm Based on Faster R-CNN[J]. Journal of Beijing Normal University (Natural Science) 2020−05.

[7] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[8] LIU W, ANGUELOV D, ERHAN D, et al. SSD: Single Shot MultiBox Detector[C]// European Conference on Computer Vision. Springer, Cham, 2016.

[9] REDMON J, DIVVALA S, GIRSHICK R, et al. You Only Look Once: Unified, Real-Time Object Detection[C]// Computer Vision & Pattern Recognition. IEEE, 2016.

[10] Di W, Tian Y, Geng W, et al. LPR-Net: Recognizing Chinese License Plate in Complex Environments[J]. Pattern Recognition Letters, 2018, 130.

[11] BOCHKOVSKIY A, WANG C Y, LIAO H. YOLOv4: Optimal Speed and Accuracy of Object Detection [J]. 2020.

[12] REDMONJ, FARHADI A. YOLOv3: An Incremental Improvement[J]. arXiv e-prints, 2018.

[13] HOWARD A G, ZHU M, CHEN B, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications[J]. 2017.

[14] GUO Jifeng, SUN Wenbo, PANG Zhiqi, et al. Improved Traffic Sign Recognition Algorithm for Yolov4 [J]. Journal of Chinese Computer Systems,2021.

[15] XIE Junzhang, PENG Hui, TANG Jianfeng, et al. Improve YOLOv4's dense remote sensing target detection[J]. Computer Engineering and Applications,2021.

[16] XU Hao, YANG Degang, JIANG Qianqian, et al. Improvement of lightweight vehicle detection network based on SSD[J]. Computer Engineering and Applications,2021.

[17] WANG Yingxuan, SONG Huansheng, LIANG Haoxiang, et al. Highway Vehicle Object Detection Based on An Improved YOLOv4 Method[J]. Computer Engineering and Applications, 2021.

[18] Liu Sheng, Ma Shexiang, Meng Xin, et al. Traffic Sign Detection Based On Multi-scale Feature Fusion Network[J]. Computer Applications and Software. Vol. 2 (2021) No. 38, p. 158-164.

[19] LI Chenghao, ZHANG Jing, HU Li, et al. Small object detection algorithm based on multiscale receptive field fusion[J]. Computer Engineering and Applications,2021.

[20] Wu Jiao, Wang Peng, Qiao Mengyu, et al. Target Detection Algorithm Of Land Battlefield Based On Improved Multi-Level Feature Pyramid Network[J]. Computer Applications and Software. Vol. 10 (2020) No. 37, p. 155-160.

[21] MA Qiaomei, WANG Mingjun, LIANG Haoran. License plate location detection algorithm based on improved YOLOv3 in complex scenes[J]. Computer Engineering and Applications,2020.

[22] Xu Guangzhu, Kuang Wan, Li Xingwei, et al. License Plate Location Based on YOLOv3 and Vertex Offset Estimation[J]. Journal of Computer-Aided Design & Computer Graphics.2021.

[23] Laroca R, Severo E, Zanlorensi L A, et al. A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector[C]// 2018:1-10.