ISSN 2616-5775 Vol. 6, Issue 3: 75-83, DOI: 10.25236/AJCIS.2023.060310

Research on Traffic Scene Recognition Algorithm Combining Object Detection and Semantic Segmentation

Yifan Deng, Shaoqing Mo, Haiyun Gan, Jiangjiang Wu

School of Automobile and Transportation, Tianjin University of Technology and Education, Tianjin, China

Abstract: *At present, the detection of instances in traffic scenes based on deep learning is mainly divided into two mainstream directions: object detection and semantic segmentation. Among them, object detection realizes the specific location of a single object in the road scene, and semantic segmentation realizes the pixel level classification of objects and background categories in the road scene. However, when pedestrian and other objects have occlusion problems, semantic segmentation is difficult to directly separate a single instance, and the anchor frame generated by object detection contains redundant information. To solve this problem, this paper proposes a method combining target detection and semantic segmentation. This method first uses YOLOv5 model to complete the target detection, and detects people, vehicles and other objects in the captured traffic scene image. At the same time, the improved DeepLabv3+ network model is used to capture the semantic and regional information of roads in the captured traffic scene image. Finally, the prediction results of the output of the two task branches are drawn in the image to be detected, and finally the drawing results are combined and output uniformly. This method can effectively distinguish different people, vehicles, roads and other information in the traffic scene, "complement each other" to understand the driverless road scene, and improve the detection accuracy. The experimental results show that the average mAP of this method is 79.11%, and the segmentation accuracy is high, which is suitable for the unmanned driving scene on real urban roads.*

Keywords: *Target detection; Semantic segmentation; Instance segmentation; Complex traffic scenarios*

1. Introduction

Scene understanding based on visual images[1] is a basic and common technology in the field of computer vision. This technology provides sufficient visual clues and analytical information for the unmanned vehicle perception module to realize the perception and understanding of the surrounding road environment. Target detection and semantic segmentation are the only way to realize the understanding of the road scene at the instance level and pixel level[2].

Among them, target detection includes two stages: classification and positioning. First, it determines whether there are target instances such as pedestrians and vehicles in the image, and then uses the detected anchor frame to determine the specific position of the target in the scene to distinguish different instances of the same category, so as to ensure the stable operation of perception modules such as obstacle detection and multi-target tracking. Semantic segmentation assigns a clear semantic category label to pixels in the image to realize pixel and positioning, which can be used for fine segmentation of the road, sidewalk, background environment and other stationary background categories with no fixed shape, so as to extract the driving area, so as to regulate the safe driving of vehicles within the road range. It can be seen that accurate, stable and real-time target detection and semantic segmentation road scene understanding are the guarantee for the safe operation of unmanned driving system and have important research significance[3].

For detection of different instances in traffic scenes (including but not limited to pedestrians, vehicles, etc.), traditional machine learning object detection algorithm or deep learning-based object detection algorithm can be generally used. The traditional machine learning object detection algorithm is based on feature extraction. The general process is as follows: on the image to be detected, candidate boxes of different sizes are set by means of sliding window, and feature extraction methods such as HAAR[4], HOG[5] and DPM[6] are used to send the features extracted from the candidate areas into classifiers such as SVM[7] for classification and judgment. In the process of classification determination, if single-category detection only needs to distinguish whether the objects contained in the current candidate boxes are background or targets, and multiple categories inevitably produce overlapping candidate boxes, in this case, the non-maximum suppression of NMS[8] is required to combine the candidate boxes, remove redundant candidate boxes, calculate candidate boxes of each category, and finally complete the target detection.

Some shortcomings can be seen from the traditional target detection algorithm process: firstly, the features designed by hand generally acquire the bottom and middle features, and the expression ability is relatively poor, not good robustness to the change of diversity, and the region selection strategy based on the sliding window is not targeted, and the detection results are unsatisfactory.

With the rapid development of deep learning in recent years, the object detection algorithm based on convolutional neural network[9] has also made great breakthroughs. Compared with the traditional machine learning object detection algorithm, the most important thing is that the detection accuracy has been greatly improved, which benefits from the powerful deep-level feature extraction ability. Object detection algorithms based on deep learning can be mainly divided into two categories: one is R-CNN[10], Fast-RCNN[11] and other typical two-stage recognition algorithms. Based on feature extraction, such algorithms first generate a large number of candidate regions by independent network branches, and then conduct classification and regression. Another type of algorithm is typical one-stage recognition algorithm such as SSD[12] and YOLO[13], which carries out classification and regression at the same time when generating candidate areas. The advantages of the two-stage recognition algorithm are mainly reflected in scalability and high accuracy, while the one-stage recognition algorithm is faster and more suitable for target detection requiring real-time detection. The YOLOv5 model belonging to the one-stage recognition algorithm takes into account high detection accuracy on the basis of ensuring real-time target detection. Therefore, considering the real-time performance of the final algorithm, YOLOv5 is selected as the algorithm of target detection in this paper.

Traffic scenes in the real society are complex, including roadway, sidewalk, zebra crossing, etc. Some areas are traffic sections, while others are pedestrian sections. In this case, it is necessary to determine whether the traffic instances in the detected scene will obstruct the traffic[14], so it is a very challenging task.

In order to divide each traffic instance, this paper uses semantic segmentation algorithm. Semantic segmentation is the classification at the pixel level to determine whether a certain pixel in an image is in a certain object class, so semantic segmentation is to understand the image from the pixel level. Before deep learning is applied to the field of computer vision, researchers generally use Texton Forest or Random Forest method to build classifiers for semantic segmentation. In 2014, the FCN[15] proposed by Long et al. was the first to use CNN[16] network for semantic segmentation. This network changed the last full connection layer of AlexNet network for target recognition into convolutional layer, used deconvolution layer for up-sampling, and proposed skip connection to improve up-sampling. Compared with the traditional image segmentation method using region feature extraction, the performance is significantly improved. Subsequently, a series of excellent semantic segmentation models emerged one after another, such as SegNet[17], PSPNet[18], DeepLabv3+[19] and so on. In this paper, the semantic segmentation model DeepLabv3+ with high precision is used.

2. YOLOv5 target detection network

The YOLOv5 model has the characteristics of fast identification speed and high detection accuracy. However, in order to realize proprietary detection for different targets in traffic scenarios and further improve the accuracy of target detection, further parameter adjustment of the network structure of the YOLOv5 model is required.

The image reasoning speed of YOLOv5(You Only Look Once) can reach 0.007s at the fastest, that is, it can process 140 frames per second, which can meet the requirements of real-time image detection when the computing power is sufficient. Meanwhile, the structure of YOLO series is smaller than that of previous generations. The weight data file of YOLOv5s is 1/9 of YOLOv4 version. Compared with previous generations of YOLO models, it is lighter and more suitable for transplantation and deployment on hardware platforms, so it has a better prospect of industrial landing.

Figure 1 shows the brief forward propagation process of the YOLOv5 model. It can be seen from the figure that the entire YOLOv5 model is composed of four parts: input end, BackBone benchmark network, Neck network and output end. The central idea is to divide the image to be detected into N×N pixel blocks, and then predict the object category of each pixel block separately. Bounding box information and class owning probability information of fixed trees are output, and directly classified and returned after feature extraction. The whole process requires only one stage, with higher detection speed compared with the two-stage target detection algorithm.

The backbone network in YOLOv5 model is CSPDarknet-53, which has strong feature extraction ability. The neck stem is designed to better extract the features in the backbone network and make reasonable use of the feature map at different stages of processing. By means of FPN+PAN, Spatial Pyramid Pooling [20] (SPP), upper/lower sampling branch composition, the feature map is integrated by splicing, point sum or dot product. The Head part of the classification network is responsible for detecting the position and category of the target by extracting the feature map. The Head part contains the prediction results of F1, F2 and F3 output feature maps, and calculates the category probability confidence for different results to complete the classification of objects and output the prediction result map.

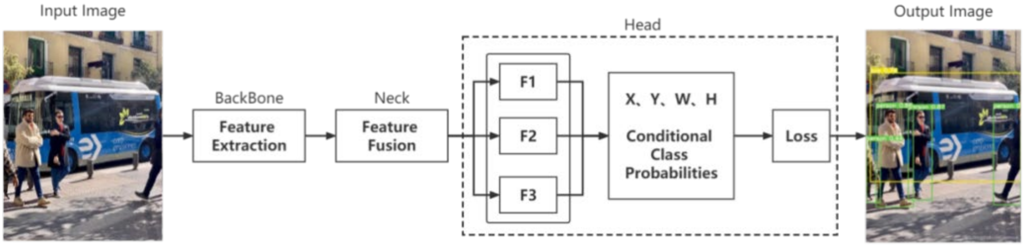


Figure 1: Brief network structure diagram of YOLOv5

3. Image semantic segmentation network

Semantic segmentation is a basic task in computer vision. Compared with object detection, semantic segmentation is a more sophisticated work, which is pixel-level classification. DeepLabv3+ builds on DeepLabv3, the flagship semantic segmentation network.

3.1 DeepLabv3+ Semantic Segmentation Network overview

Semantic segmentation DeepLabv3+ model for encoding - decoding structure. The encoding module in the DeepLabv3+ model in the original paper uses Xception(the lighter MobileNetV2 is used in this paper) as a classification network for feature extraction, and then connects with the ASPP module for extracting and fusing multi-scale features of the image. The decoding module fuses the encoding module feature graph with the improved Xception intermediate feature graph, and then upsamples to get the segmentation result. Among them, the Xception model of the backbone network uses depth-separable convolution to increase the network width, which not only improves the classification accuracy, but also enhances the network's learning ability for advanced semantic features.

After the image is collected by the classification network, the feature map with the size of 1/16 of the original image is obtained. Then, the feature map with the size of 1/16 of the original image is entered into the global pyramid pool ASPP module. In this module, four parallel hollow convolution with different convolution kernel sizes are used for feature extraction to obtain four feature maps with different sensitivity fields. The fusion of these feature maps makes the feature maps output by the network part of the coding module have multi-scale image semantic information. The advantage of this kind of network is that it can effectively distinguish and learn high-level semantic information from low-level semantic information, so that the model has stronger feature learning ability. However, the disadvantage is that multiple downsampling and unsynchronously long convolution calculation will lead to the loss of object boundary information, which will lead to rough object boundary and low precision in the final output result.

As shown in Figure 2, the deep separable convolutional network MobileNetV2 is used in this paper to replace Xception as the backbone network, which can effectively improve the image boundary information lost due to downsampling.

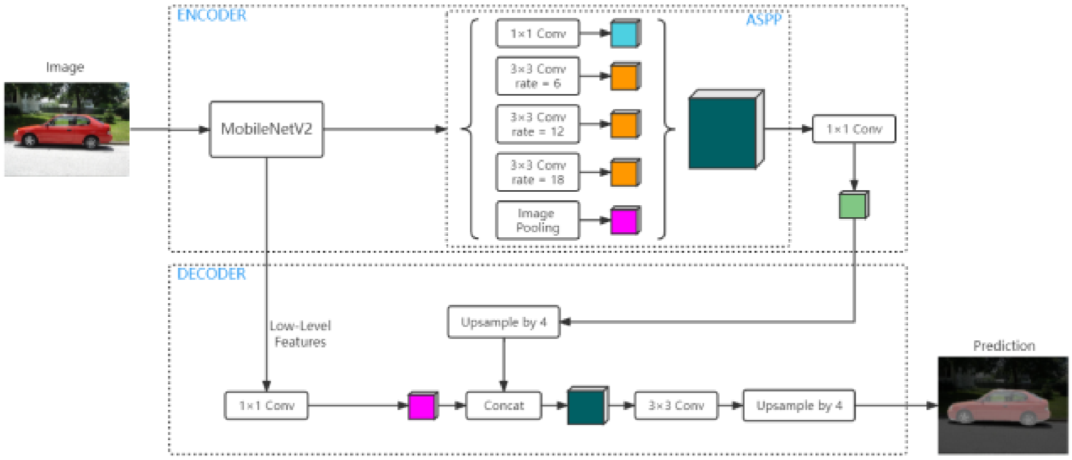


Figure 2: Network structure diagram of the improved DeepLabv3+

3.2 Comparison of Xception and MobileNetv2 network architecture

Figure 3 shows the structure of the Xception network, and you can see that the Xception network uses a depth separable convolution of step 2 to replace the maximum pooling layer where the original downsampling is done. Deep separable convolution decomvolution of standard convolution into depth wise convolution and pointwise convolution effectively reduces the amount of computation and parameters of models.

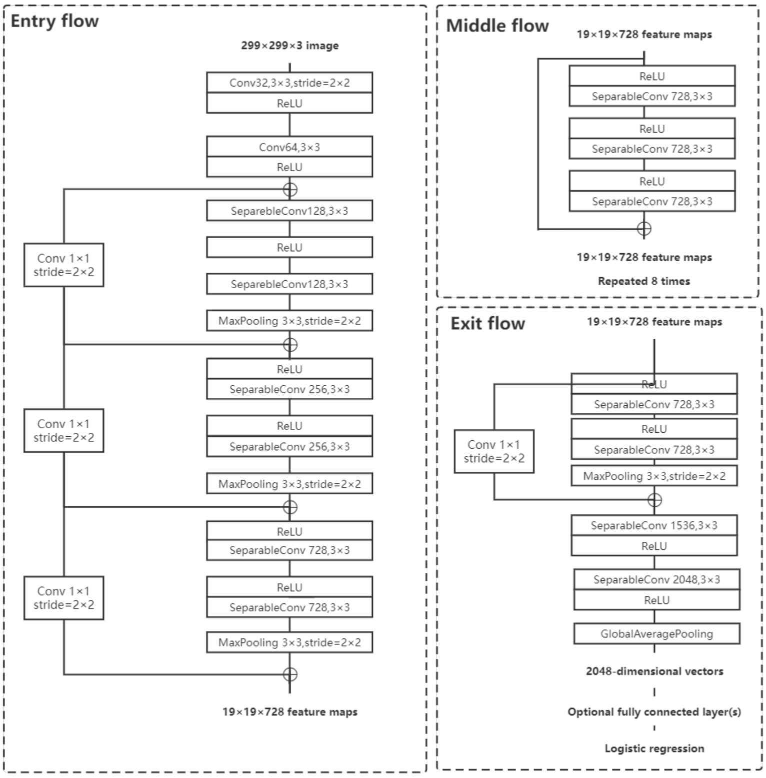


Figure 3: The network structure diagram for Xception

Figure 4 shows the structure diagram of MobileNetV2 network, which is a lightweight convolutional neural network based on the improved MobileNetV1 network. Its biggest feature is that deep separable convolution is used to reduce the number of parameters and the amount of computation, so as to accelerate the calculation and improve the performance of the network. Compared with MobileNetV1, MobileNetV2 adds a 1×1 expansion layer before the depth separable convolution to increase the number of channels to obtain more features. At the end, no ReLU transformation is used to directly output the convolution result linearly.

The core of Xception network and MobileNetV2 network both use deep separable convolution to improve the running speed. The difference is that Xception uses deep separable convolution and increases the number of network parameters to compare the results, so as to investigate the effectiveness of network structure for optimization. The MobileNetV2 network uses deep separable convolution for compression and speed, not for performance but for speed. In Xception, ReLU transform after depth-separable convolution will lead to information loss, resulting in loss of high semantic features in the learning process, while in MobileNetV2, ReLU transform after depth-separable convolution is cancelled to prevent feature destruction.

To sum up, MobileNetV2 has a relatively shallow network structure, and the effect of the model in the learning process is better than Xception. Therefore, compared with Xception, MobileNetV2 can have faster reasoning speed and calculation speed theoretically.

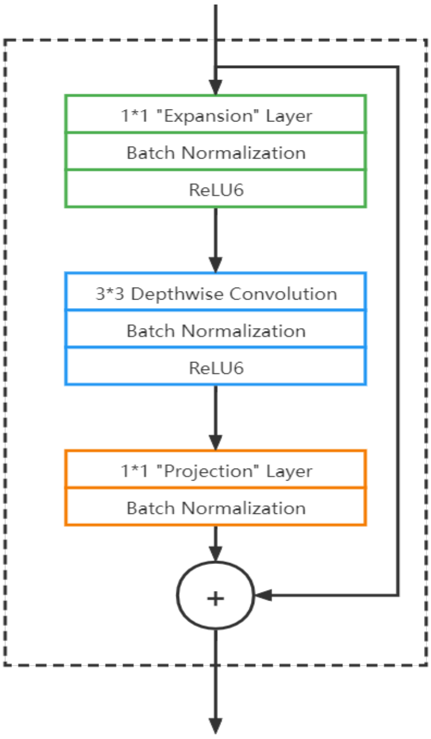


Figure 4: The network structure diagram for MobileNetv2

3.3 Mask R-CNN instance splits the network

In general, object segmentation refers to semantic segmentation, while instance segmentation is a small field independent from the object segmentation field. Compared with semantic segmentation, instance segmentation is more complex. On the basis of semantic segmentation, it needs to carry out more detailed segmentation of similar objects, that is, to count the segmented objects in class order.

Mask R-CNN [21] is a very flexible framework, which can complete different tasks by adding different branches, such as target segmentation, target detection, instance segmentation, etc., and has a high accuracy of instance segmentation. Therefore, Mask R-CNN was selected as the instance segmentation model in this paper to conduct a comparative experiment with the joint model YOLOv5-DeepLabv3+.

Instance segmentation is a very challenging task, which consists of two independent functions: object detection and instance segmentation. In 2017, He et al. changed the backbone network to ResNet-101-FPN based on the Faster R-CNN algorithm. By combining the multi-task loss with the split branch loss, classification and boundary box regression loss, On the basis of target classification and border regression, a Region of interest pooling (RoI) layer is added to predict the segmentation of Mask network branches, so as to realize real-time target detection and instance segmentation. Due to ROi-pooled integer quantization, the feature map region and the original image region do not align, resulting in a bias when predicting pixel-level masks. In order to solve the error problem caused by the rounding operation of the feature map scale in the process of scaling such as downsampling and ROi-pooling layer, He et al. proposed the ROi-align layer instead of the RoI pooling layer, and used bilinear difference to fill non-integer positions to achieve pixel-level alignment, in order to improve the accuracy of target detection branches. Figure 5 shows the overall processing flow of Mask R-CNN.

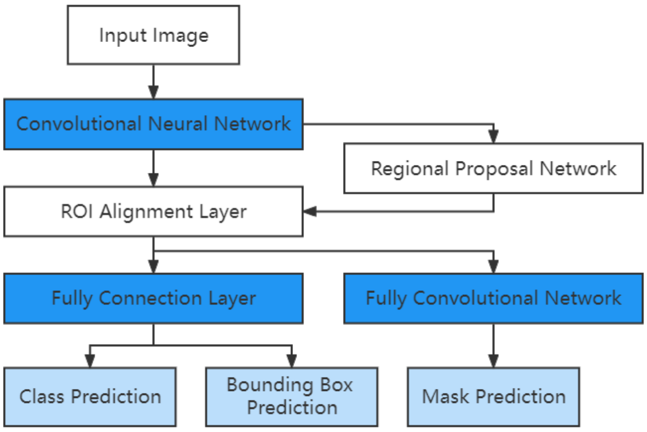


Figure 5: The overall processing flow of Mask R-CNN

4. Experimental setup and result analysis

4.1 Dataset introduction

Due to the heavy workload of semantic segmentation and annotation in real road surveillance video scenes, it is difficult to carry out mass image annotation in this paper. Therefore, the data set used in this paper is cityscapes dataset, which is a commonly used semantic segmentation dataset, including 50 European cities with 5000 finely annotated images. There are 2975 training sets, 500 verification sets (training set: verification set ≈6:1), and 1525 test sets. There are 19 categories in the Cityscapes dataset. This paper selects this data set for semantic segmentation training, then uses scripts to process the data set, and generates a new target detection data set based on the annotation of semantic segmentation. The volume is the same as that of the semantic segmentation data set, but the difference is that the data set for target detection contains 10 categories such as car and traffic sign.

4.2 Network training

The running environment of this experiment was as follows: CPU was AMD Ryzen 5 3600, GPU was Ge Force GTX 2060, memory was 6G, operating system was windows10, CUDA version 11.1, development language was python, and deep learning framework was pytorch.

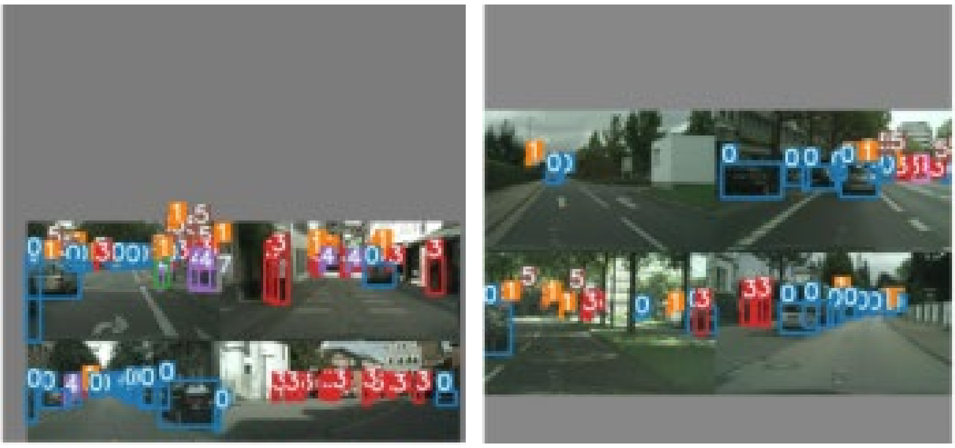
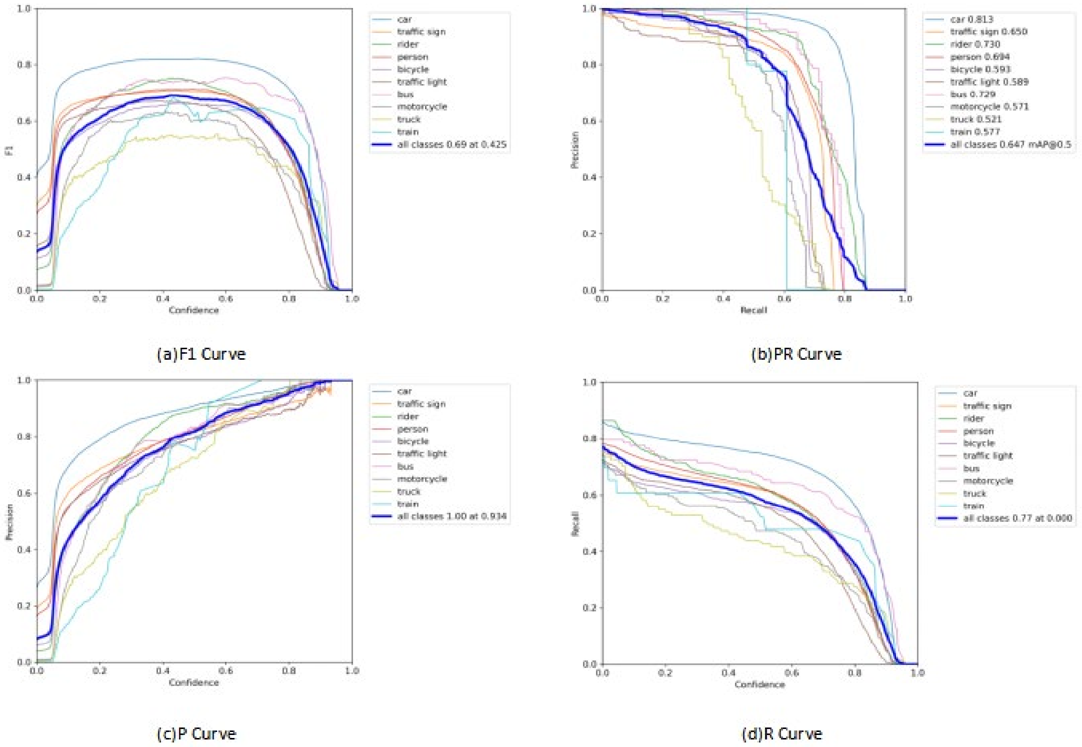


Figure 6: Subsamples during training

The training results on the COCO dataset initialized the network parameters of YOLOv5. The SGD optimization algorithm was adopted for parameter training, and the parameters were set as follows: BatchSize is 2; The maximum number of iterations is 300 epochs. Momentum factor was 0.937;

Weight attenuation coefficient is 0.0005; Cosine annealing strategy was adopted to dynamically adjust the learning rate, and the initial learning rate was 0.0015. GIOU Loss is used as the loss function. The size of the input picture is 2048*2048, and four pictures are trained at the same time. In the training process, operations such as scaling, cropping and splicing are carried out randomly, as shown in Figure 6.



(a)F1 Curve

(b)PR Curve

(c)P Curve

(d)R Curve

Figure 7: The graph was obtained after the training

As the number of iterations increases in the training process, various values change as shown in the figure 7. The meanings of each value in the figure are as follows: GIoU: When the value is close to 0, the precision of the regression frame is high; Objectness: When the value is close to 0, the target detection accuracy is high. Classiffccation: When the value is close to 0, the classification accuracy is high. Precision: Precision refers to the ratio between the number of correct targets detected and the total number of targets. The closer it is to 1, the more accurate it is. Recall: The ratio between the number of correct objects detected and the total number of objects to be marked. The closer it is to 1, the more accurate it is. mAP@0.5 and mAP@0.5:0.95: AP refers to the area enclosed after drawing with Precision and Recall as the two axes. The closer it is to 1, the more accurate it will be. The corresponding relationship between the final training result parameters and the epoch is shown in Figure 8.

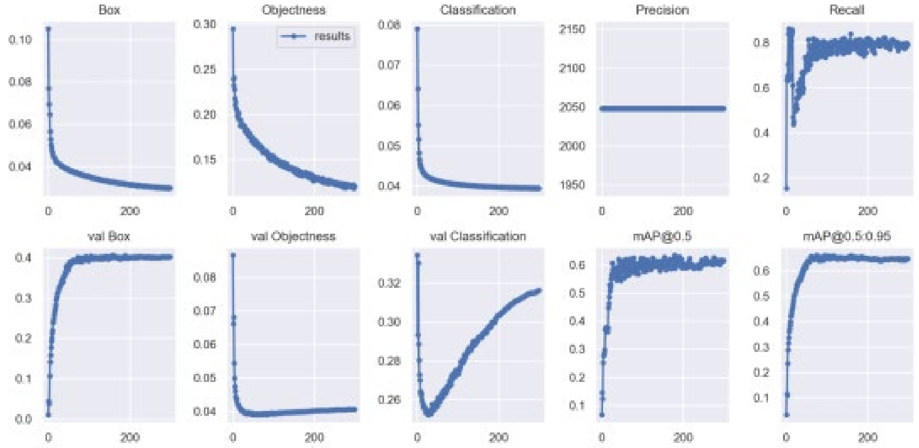


Figure 8: The result graph obtained after the training

4.3 Analysis of experimental results

The joint model YOLOv5-DeepLabv3+ model was used to test cityscapes verification set, and the test results were shown in Figure 9. From left to right are the input image, the target detection result, the semantic segmentation result and the prediction result after the combination of target detection and semantic segmentation.

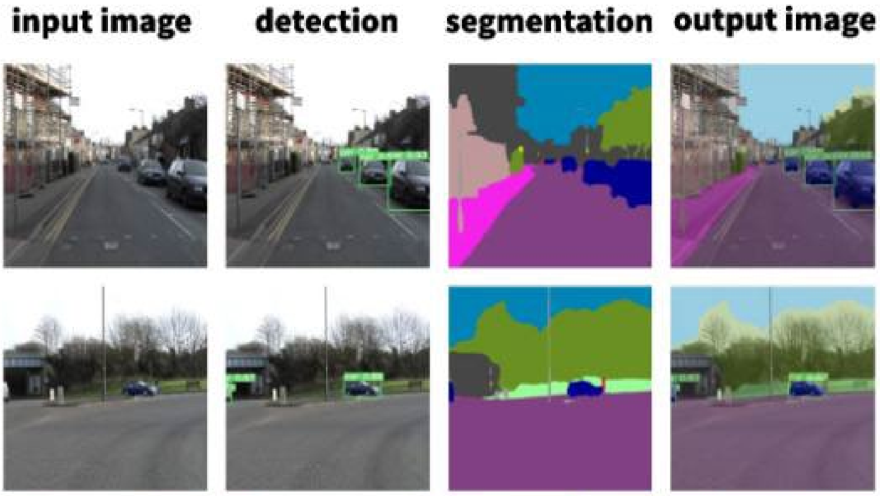


Figure 9: Graph of prediction results based on joint model

The average mAP of the model prediction results obtained after training is 79.11%, and the average detection speed of real-time video stream is 6.7fps/s, which basically meets the accuracy requirements and speed requirements of object detection in real traffic scenes. Mask R-CNN and the joint model YOLOv5-DeepLabv3+ were respectively predicted on the COCO dataset and cityscapes dataset. The prediction results are shown in Table 1 and Table 2. It can be seen that, Compared with Mask R-CNN, the joint model has higher detection accuracy and accuracy in COCO data set, and the detection speed of Mask R-CNN model is difficult to meet the real-time requirements.

Table 1: The joint model and Mask R-CNN were tested based on COCO data set

Model	mAP	mIoU	fps(frame/s)
Mask R-CNN	39.80%	50.12%	7.9
YOLOv5-DeepLabv3+	43.50%	56.46%	13.9

Table 2: The results of separate tests on the joint model and Mask R-CNN based on cityscapes dataset

Model	mAP	mIoU	fps(frame/s)	Target correct detection rate
Mask R-CNN	69.80%	79.67%	8.6	85.96%
YOLOv5-DeepLabv3+	79.11%	88.65%	15.1	94.21%

As can be seen from the above table, compared with Mask R-CNN, the joint model is more suitable for computer vision tasks in real traffic scenes, which can better detect and separate the target. Meanwhile, the processing speed is faster, and it is more suitable for hardware deployment.

5. Conclusion

For the complex scenes that unmanned driving needs to face, a single classification task has its own advantages and disadvantages, whether it is target detection and semantic segmentation, which is difficult to meet the requirements of autonomous driving vision. In this paper, the advantages of target detection algorithm and semantic segmentation algorithm are combined, and the two algorithms are integrated. By using the YOLOv5 to identify the categories and positions of various traffic facts such as pedestrians and vehicles in the image, and using the improved DeepLabv3+ network, the pixels in the scene are divided into different categories such as lanes, sidewalks and backgrounds. Finally, the two parts of information are integrated to determine the specific instances of people, cars and roads in the real traffic scene more accurately. Experiments show that the average monitoring accuracy of this method is 79.11%, and the real-time video stream detection speed is fps, which basically meets the detection requirements in real road scenes. However, it is not practical for this method to be applied in real driverless scenes. The real-time frame rate is difficult to break through due to the limitation of computing power. At the same time, there are certain differences between cityspaces data set and images collected on real roads, resulting in unsatisfactory segmentation effect of some instances. The next research work of this paper will focus on improving the real-time detection ability of the model and constructing semantic segmentation data set based on real traffic scenes.

References

[1] Franke U, Joos A. Real-time stereo vision for urban traffic scene understanding[C]//Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No. 00TH8511). IEEE, 2000: 273-278.

[2] Uhrig J, Cordts M, Franke U, et al. Pixel-level encoding and depth layering for instance-level semantic labeling[C]//German conference on pattern recognition. Springer, Cham, 2016: 14-25.

[3] Kuutti S, Bowden R, Jin Y, et al. A survey of deep learning applications to autonomous vehicle control [J]. IEEE Transactions on Intelligent Transportation Systems, 2020, 22(2): 712-733.

[4] Ning J, Yang J, Jiang S, et al. Object tracking via dual linear structured SVM and explicit feature map [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 4266-4274.

[5] Pang Y, Yuan Y, Li X, et al. Efficient HOG human detection[J]. Signal processing, 2011, 91(4): 773-781.

[6] Girshick R, Iandola F, Darrell T, et al. Deformable part models are convolutional neural networks[C]//Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2015: 437-446.

[7] Schuldt C, Laptev I, Caputo B. Recognizing human actions: a local SVM approach[C] //Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. IEEE, 2004, 3: 32-36.

[8] Hosang J, Benenson R, Schiele B. Learning non-maximum suppression[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4507-4515.

[9] Li P, Zhao W. Image fire detection algorithms based on convolutional neural networks[J]. Case Studies in Thermal Engineering, 2020, 19: 100625.

[10] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[11] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.

[12] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]//European conference on computer vision. Springer, Cham, 2016: 21-37.

[13] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[14] Li J, Mei X, Prokhorov D, et al. Deep neural network for structural prediction and lane detection in traffic scene[J]. IEEE transactions on neural networks and learning systems, 2016, 28(3): 690-703.

[15] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation [C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.

[16] Albawi S, Mohammed T A, Al-Zawi S. Understanding of a convolutional neural network[C]//2017 international conference on engineering and technology (ICET). Ieee, 2017: 1-6.

[17] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation [J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 39(12): 2481-2495.

[18] Zhao H, Shi J, Qi X, et al. Pyramid scene parsing network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2881-2890.

[19] Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 801-818.

[20] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition [J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(9): 1904-1916.

[21] He K, Gkioxari G, Dollár P, et al. Mask r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2961-2969.