Road object recognition method based on improved YOLOv3

Yichi Zhang[1,*], Boyu Hu[2,a], Xinyi Yuan[3,b], Yupeng Li[4,c]

[1]University of Nottingham, Ningbo, Zhejiang, China
[2]Beijing Information Science and Technology University, Beijing, China
[3]Xihua University, Chengdu, Sichuan, China
[4]The Woodlands, Mississauga, Ontario, Canada
[a]huby2000@bistu.edu.cn, [b]784976442@qq.com, [c]yupeng04.yl@gmail.com
**Corresponding author: biyyz74@nottingham.edu.cn*
These authors contributed equally to this work.

Abstract: *Based on the emergence and development of autonomous driving technology, the identification of obstacles on the road is a very important and challenging task. And there are many difficulties in the realization of this task, for example, there are many types of road targets, and the scale span of the road obstacles is large. In view of these problems, this experiment proposes three improvement directions for the YOLOv3 algorithm to perform the task of road target prediction: one is to improve the up-sampling multiple to use more shallow spatial information to improve the accuracy of small target detection. The second is to change the way of feature fusion of the feature pyramid. Thirdly, the convergence direction of the model is changed by clustering learning. Experiments on the BDD100K data set show that the yolov3_10cls_tiny algorithm proposed in this paper has the best detection performance and better average accuracy than YOLOv3.*

Keywords: *Object recognition, YOLOv3, Clustering learning*

1. Introduction

In the past decade, with the continuous development of mobile processors, network communications, big data, robots, sensors, and other new technologies, has brought a new development direction for automotive technology-automatic driving. The emergence of automatic driving technology will bring a new change to automobiles, transportation, electronics, network, and other industries and technologies, and will also have a great impact on human lifestyle [1]. Today's unmanned systems are not yet mature, and the recognition technology for different objects is not yet accurate. The implementation of automatic driving technology will inevitably change the transportation and transportation industry first. The need for automatic driving technology will greatly promote the development of intelligent transportation. Combining automatic driving technology with intelligent traffic management can reduce traffic congestion, accident rate, and casualties, improve road traffic efficiency, improve vehicle operation efficiency, reduce energy consumption, and improve air pollution. And can effectively reduce the labor intensity of drivers, improve transportation efficiency, improve transportation safety, greatly improve the production efficiency of the transportation industry, and reduce the transportation cost.

Firstly, through the research of obstacle recognition, the classifier is used to execute the detection, determine the boundary box of spatial separation and the related class probability, and study the collision prediction and representation method of unmanned vehicles and moving obstacles. For different detection objects, pedestrians, vehicles, lane lines, and roadside signs, there are some problems in each category. For example, in the actual driving process, when overtaking or turning, only a part of the vehicle can be displayed in the visual range, and it is difficult to accurately determine the detection target. This may require the judgment and fusion of context information and target tracking of the previous and subsequent multi-frame images [2]. In the later research, a new obstacle occupancy grid map is proposed, which can not only represent the occupancy information of static obstacles but also represent the predicted collision information between the unmanned vehicle and moving obstacles in the future. In addition, the localization of data sets is another problem. At present, most of the open-source data sets that can be found on the Internet are based on foreign data sets, and the data sets of the actual road environment in China are relatively few [3].

Through the research on the collision avoidance method of moving obstacles of unmanned vehicles, an improvement of the searchable continuous domain is proposed, which linearly interpolates the grid and changes the search direction to a continuous arbitrary direction, which can solve the problem that the shortest path solved by the traditional algorithm in the grid graph is not optimal. Then a dynamic and static obstacle collision avoidance method of an unmanned vehicle based on this improved algorithm is studied, which realizes the safe and intelligent autonomous driving of unmanned vehicles in a dynamic environment [3]. Target detection algorithms based on deep learning can be roughly divided into two categories: one is SSD, YOLO, YOLOv2, and other single-channel network structures, which are fast enough in specific experiments, but easy to lose details, so the accuracy needs to be improved. The other is RCNN, faster-RCNN, RFCNN, and so on. Although this kind of network has enough accuracy in the experiment, it runs slowly and is difficult to meet the actual environment of automatic driving. By comparing different algorithms, it is found that YOLO imposes strong spatial constraints on bounding box prediction, and there are some similarities between YOLO and R-CNN. Each grid cell proposes potential bounding boxes and uses convolution features to score these boxes [2]. Other Fast Detectors Fast and Faster R-CNN focus on accelerating the R-CNN framework by sharing computations and using neural networks to propose regions rather than a selective search [4]. Although the RCNN algorithm has made great progress, its shortcomings are also obvious: the redundant calculation of the features of overlapping boxes (more than 2000 candidate boxes in an image) makes the detection speed of the whole network very slow (it takes about 14 S to detect an image using GPU). Compared to low resolutions, YOLOv2 operates as a cheap, fairly accurate detector, which makes it ideal for smaller GPUs, high frame rate video, or multiple video streams [5]. while YOLOv2 is the most advanced and fastest of other detection systems in various detection datasets, and can also run at a variety of image sizes. Compared to YOLOv3, it is comparable to the SSD variant in terms of average Ap metrics, but three times faster [6].

Through four indicators, this paper evaluates the improvement of the original model algorithm of YOLOv3, compares different algorithms, uses computer language to detect and distinguish different objects, constructs multi-scale feature expression, and especially puts forward three key improvement directions for the YOLOv3 algorithm to implement road target detection. Change the way of feature fusion of the feature pyramid, improve the accuracy by trying to sample, and combine three different methods. A new improved model is obtained. This paper introduces the technical route of the development of a target detection algorithm based on CNN: one-stage, two-stage, and two-stage detection algorithm including R-CNN and one-stage detection algorithm including YOLO [7]. Through the multi-scale target detection in this study, the comparison of different algorithms, the design of parallel branches with different parameters in the network, the construction of spatial pyramids, and the solution of the scale problem of target detection based on multi-scale features, whether there are over-fitting and under-fitting problems of different scale targets in the data set are discussed.

2. Related Works

2.1. Two-stage target detection algorithm

The two-stage detection algorithm divides the detection problem into two stages. Firstly, the candidate regions (region proposals) are generated, and then the candidate regions are classified (generally need to be refined). The typical representative of such algorithms is the R-CNN algorithm based on region proposals, such as R-CNN [8], SPPNet, Fast R-CNN [9], Faster R-CNN [10], FPN, R-FCN, etc. R-CNN is an early target detection algorithm using CNN. The idea is to use the selective search [11] to extract about 2000 preselection boxes, then resize them to a unified scale for CNN feature extraction, and finally classify them with FC.

Compared with RCNN, Fast R-CNN only needs one feature extraction of the whole image, and directly uses FC for classification and regression to realize end-to-end multi-task collaborative optimization. The specific process is to extract the feature map of the image through the Backbone network first; extract RoI features from features using the preselection box of selective search; transform RoI features into a fixed size by RoI Pooling; finally enter the classification and regression head for classification and regression. Fast R-CNN's RoI still searches through Selective Search, which is slower. Based on Fast R-CNN, RPN (Region Proposal Network) is proposed to automatically generate RoI, which greatly improves the efficiency of preselection box generation.

2.2. One-stage target detection algorithm

One-stage detection algorithm, which does not require the region proposal stage, directly generates the category probability and position coordinates of the object. After a single detection, the final detection results can be directly obtained, so it has a faster detection speed. Typical algorithms such as YOLO [12], SSD [13], and Retina-Net. The one-stage target detection algorithm eliminates the region proposal link of the two-stage method, that is, there is no pre-classification and regression process, directly divides the specific categories, and regresses the border.

The whole SSD network adopts the idea of one stage to improve the detection speed. The anchor's idea in Faster R-CNN is integrated into the network, the feature hierarchical extraction is performed, and the border regression and classification operations are calculated in turn, which can adapt to the training and detection tasks of multiple scale targets. RetinaNet directly completes the whole set of target detection tasks by the RPN network. Its network structure is actually that the FPN network extracts multi-scale features, and then connects the detection head based on multi-scale features to predict the classification and location regression of the target.

YOLO target detection algorithm directly completes the prediction from feature to classification and regression. Classification and regression are realized by the same full connection layer. The object detection task is treated as a regression problem, and the coordinates of the bounding box, confidence, and class probabilities of objects in the box are obtained directly by all pixels of the whole image. Through YOLO, each image can be obtained by looking at what objects are in the image and where they are. As a representative of the one-stage detection algorithm, YOLOv3 [14] has the advantages of simple structure, good robustness, fast detection speed, and high detection accuracy. However, the recognition accuracy of small and medium targets is still not ideal.

2.3. Road Object Detection

To enhance the detection performance of small targets and improve the original SSD network structure, the research group of Southwest Jiaotong University introduced the improved SFPN network to load the SSD feature extraction network [15] and fused the feature pyramids with different scales and resolutions. The shallow and deep feature map receptive fields were fused to improve the detection performance of small targets, and then the classification and regression were performed to predict the targets. At the same time, in order to improve the overall network performance, ResNet50 is selected to replace VggNet16 [16] as the main feature extraction network of the improved model, deepening the number of network layers and improving the performance.

Luo Jianhua proposed a small target detection method based on improved YOLOv3 [17]. Firstly, a new feature fusion structure is designed to reduce the miss rate of small targets, and the DIOU loss is used to improve the positioning accuracy. At the same time, the clustering algorithm in the YOLOv3 algorithm is improved. The K-means + + algorithm is used to improve the extraction of the center point of the cluster prior box, and a more suitable Anchor Box is selected to improve the average accuracy and speed of the detection.

Liu Yunxiang studied the structural reform and parameter tuning of the model and proposed an RF-YOLOv3 network model [18]. The algorithm uses the K-means clustering algorithm to determine the number of target candidate boxes and the dimension of the width-height ratio according to the inherent characteristics of the vehicle. Then, the model parameters are reset according to the clustering results, so that the RF-YOLOv3 network has certain pertinence in vehicle detection.

3. Method

3.1. YOLOv3 backbone structure

Based on the feature pyramid, Yolov3 divides the input image into coarse, medium, and fine grids to realize the prediction of a large, medium, and small objects respectively. Each grid corresponds to an ROI area. If the center of an object just falls in this grid, the grid is responsible for predicting the object. If the size of the input picture is 416x416, the coarse, medium, and fine grid sizes are 13x13, 26x26, and 52x52 respectively. In this way, the length and width dimensions are scaled by 32, 16, and 8 times respectively, and these multiples are exactly the size of these ROIs.

The figure 1 is the overall structure of the YOLOv3 network, which can be seen in the figure: The input images with the size of 416x416 were entered into the Darknet-53 network to obtain three branches, which finally obtained three feature maps with different sizes after a series of operations such as convolution, up-sampling, and merging. The shapes are [13, 13, 255], [26, 26, 255] and [52, 52, 255].

3.2. Improvement of the algorithm

The improvement of this paper mainly has three directions. One way is to try to improve the up-sampling multiple to use more shallow spatial information to improve the accuracy of small target detection. Second, try to upgrade the feature pyramid to PaNet, adding a new sampling fusion process to improve performance. The third is to try to change the anchor hyper-parameter setting to the result of k-means clustering on the data set.

Firstly, this paper attempts to use over-sampling to improve the sampling accuracy, because, in the case of over-sampling, every four times increase in the sampling frequency will increase the accuracy by one bit. So this paper attempts to change the third layer of the feature pyramid from the original 2 times sampling to 4 times sampling (yolov3_10cls_tiny). In addition, this paper attempts to upgrade the feature pyramid to PaNet, which adds a top-down fusion sampling process to the original feature pyramid structure to transfer shallow spatial information (yolov3_10cls_panet). PaNet can transmit the low-level information to the high-level, and reduce the convolution layers that the information flow from the high-level to the low-level needs to pass through. It can also carry out RoI Pooling on multiple levels at the same time, and integrate the multi-level information to predict. In addition, this paper attempts to change the anchor hyper-parameter setting to the result of k-means clustering on the data set (*_anchor). At the same time, the three methods are combined to obtain a new improved model (yolov3_10cls_tiny_anchor, yolov3_10cls_panet_tiny).

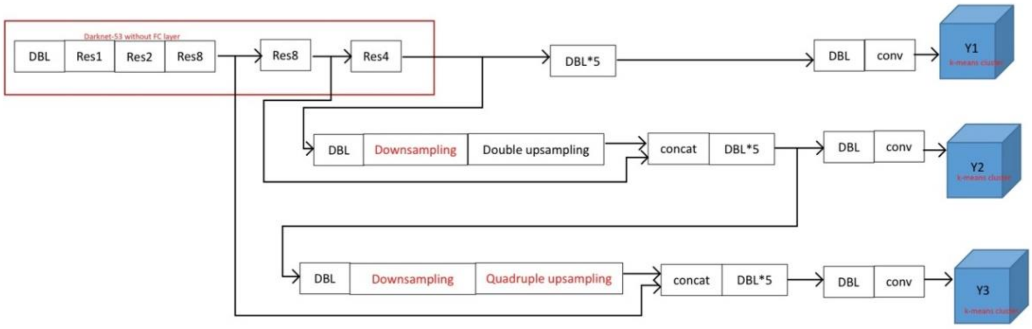


Figure 1: The improved YOLOv3 model

3.3. Prediction of target bounding box

In the three feature maps of YOLOv3 network, convolution prediction is performed by (4+1+c)×k convolutional kernels of size 11, k is the number of predefined bounding box prior (k is 3 by default), c is the category number of predicted targets, among which 4k parameters are responsible for predicting the offset of target bounding boxes, k parameters are responsible for predicting the probability that targets are contained in the target boundary box, and ck parameters are responsible for predicting the probability that k preset boundary boxes correspond to c target categories.

Figure 2 below shows the prediction process of the target bounding box. The dotted rectangular box in the figure is the preset boundary box, and the solid rectangular box is the predicted boundary box calculated by the offset predicted by the network.

In the figure, $(\mathrm{c_x}, \mathrm{c_y})$ is the central coordinate of the preset boundary frame on the feature graph, and $(\mathrm{p_w}, \mathrm{p_y})$ is the width and height of the preset boundary frame on the feature graph. $(\mathrm{t_x}, \mathrm{t_y}, \mathrm{t_w}, \mathrm{t_h})$ are the center offset of boundary box predicted by network: $(\mathrm{t_x}, \mathrm{t_y})$ and width to height scaling ratio $(\mathrm{t_w}, \mathrm{t_h})$. $(\mathrm{b_x}, \mathrm{b_y}, \mathrm{b_w}, \mathrm{b_h})$ is the prediction of the ultimate target boundary box, the conversion process from the preset boundary box to the final predicted boundary box is shown below:

$$b_x = \sigma(t_x) + c_x$$

$$b_y = \sigma(t_y) + c_y$$

$$b_w = p_w e^{t_w}$$

$$b_h = p_h e^{t_h}$$

In addition, the transformation of the process including $\sigma(x)$ function is the sigmoid function, its purpose is to predict the offset and scaling to between 0 and 1 (this can be preset bounding box center Acoordinate is fixed in a cell, accelerate network convergence) [19].

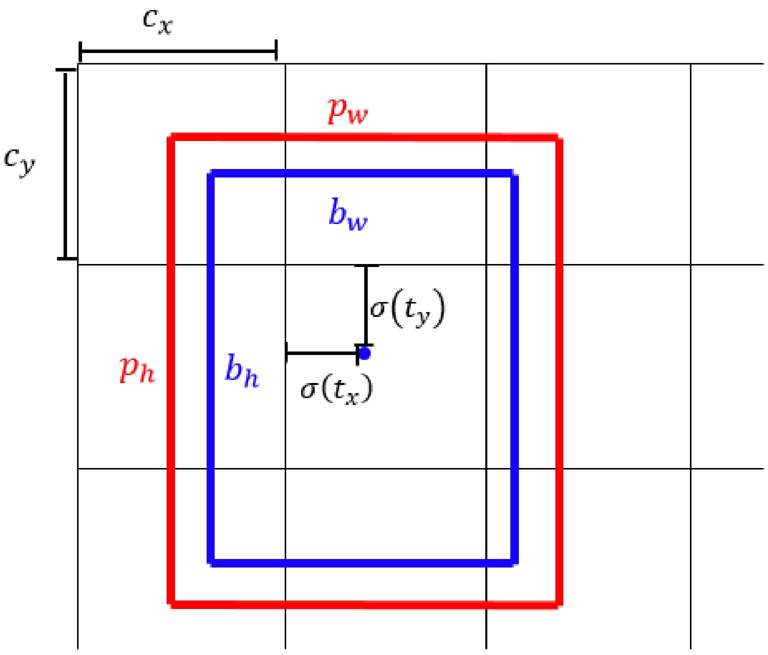


Figure 2: Bounding boxes with dimension priors and location prediction

3.4. Loss function

3.4.1. Fundamental loss function

The primitive loss function of YOLOv3 is as follows, which is mainly divided into three parts: boundary frame coordinate loss, classification loss and confidence loss:

$$Boundary\ frame\ coordinate\ loss = \lambda_{obj} \times (2 - truth_w \times truth_h) \times \sum_{r\in(x,y,w,h)} (truth_r - predict_r)^2$$

In the above formula, λ_{obj} represents the confidence to judge whether there are objects in the grid. In addition, similar to the bounding box coordinate Loss of YOLOv1, the Loss of (x, y, w, h) is calculated separately in YOLOv3 using the error square Loss function and then added together. In YOLOv1, the author takes the square root of width and height (w, h), to weaken the influence of boundary box size on loss value. In YOLOv3, the author did not take the method of square root but added a weight-related to the size of the object box, weight =2 - relative area, value range (1~2).

$$Classfication\ loss = \lambda_{obj} \times \sum_{r=0}^{k-1} ((r == truth_{class})?1:0 - predict_{class_r})^2$$

In the above formula, the error square loss function is used to calculate loss for the class.

$$Confidence\ loss = (truth_r - predict_r)^2$$

The error square loss function is used in the above formula to calculate the loss of the confidence conf.

3.4.2. Improved loss function

At present, the mainstream boundary box optimization in target detection adopts BBox regression loss (MSE Loss, L1-Smooth Loss, etc.). These methods calculate the loss value by the "proxy attribute" of the detection frame--distance while ignoring the most significant property of the detection frame itself--IoU.

$$IoU = \frac{Predict\ \cap GroundTruth}{Predict\ \cup GroundTruth}$$

The IoU is not suitable as a loss function because it has two fatal drawbacks:

- When there is no overlap between the prediction box and the real box, the IoU value is 0, leading to the gradient of 0 during the optimization of the loss function, which means that the optimization cannot be done.
- Even if the prediction box and the real box overlap and have the same IoU value, the detection effect is quite different

In that situation, GIoU continues the good features of IoU and eliminates the fatal weaknesses of IoU, which not only focuses on overlapping areas, but also on non-overlapping areas, so it can better reflect the degree of overlap between the two.

GIoU's calculation is very simple for two bounding boxes A and B. It can be calculated by calculating its minimum convex set (the minimum bounding box surrounding A and B) C:

$$GIoU = IoU - \frac{C-(A\cup B)}{C}$$ [20]

After introducing GIoU loss, the final loss function of this experiment is:

$$\text{Loss} = GIoU\ loss + Classfication\ loss + Confidence\ loss$$

where GIoU loss is:

$$GIoU\ loss = \lambda_{obj} \times (2 - \frac{GroundTruth_w \times GroundTruth_h}{width \times height}) \times (1 - GIoU)$$

4. Experiments

4.1. Data Set

The data set of this study comes from BDD100K [21] (A large-scare Diverse Driving Video Database), which is the largest public Driving data set with the most Diverse content released by the AI Lab of Berkeley University (BAIR) in May 2018. The BDD100K dataset contains 100,000 HD videos. BAIR researchers sampled key frames in the tenth second of each video and provided annotations for these key frames to help understand the diversity of data and object statistics in different scenes.

In this study, 3500 pieces were randomly selected as the training set and 500 pieces as the test set, so that the data set has the characteristics of complex background, multi-category and multi-scale.

We preprocessed the data before the experiment: Use the normalization method to scale the feature to a specific [0,1] interval by dividing the length and width of the image, and then adjust the annotation accordingly in the following format: category, center point x, y, w, h (where x, y, w, h are normalized fractions between 0 and 1).

4.2. Evaluation Index

In this paper, the evaluation indexes of the target detection model on the road mainly include mAP@0.5, precision, recall, and F1-score. The mAP@0.5 (mean average precision) is a very important measurement index, which is used to measure the performance of the target detection algorithm. From the perspective of the prediction results, the accuracy describes how many of the positive cases predicted by the two classifiers are true positive cases, that is, how many of the positive cases predicted by the two classifiers are accurate. The recall describes how many real instances in the test set are selected by the two classifiers from the perspective of real results, that is, how many real instances are recalled by the two classifiers. F1-score is an indicator used in statistics to measure the accuracy of the binary (or multi-task binary) model. It also takes into account the accuracy and recall of the classification model.

4.3. Experimental Comparison of Different Improved Algorithms

These modified algorithms are trained and tested with 100 iterations using the bdd100k data set. According to the table, yolov3_10cls_tiny achieved 45.7% precision and 36.8% mAP, which is superior to all other improvements in accuracy.

Table 1: Comparison of experimental results of different improved algorithms

	precision	recall	mAP@0.5	F1-score
yolov3_10cls	0.424	0.403	0.345	0.409
yolov3_10cls_anchor	0.369	0.411	0.327	0.387
yolov3_10cls_tiny	0.457	0.4	0.368	0.422
yolov3_10cls_tiny_anchor	0.417	0.429	0.364	0.417
yolov3_10cls_panet	0.403	0.395	0.325	0.395
yolov3_10cls_panet_tiny	0.405	0.346	0.321	0.369

From table 1, the precision, recall, mAP, and F1-scores were all around 0.3 to 0.4 or ~30% to ~40%. In the precision category, yolov3_10cls_tiny had the most precision whereas the yolov3_10cls_anchor was the least precise model. As the precision value gets higher, this means that out of the machine's predictions, a higher amount is what we desire was detected by the model. yolov3_10cls_tiny_anchor had the highest recall score and yolov3_10cls_panet_tiny had the lowest amount of recalls. This shows that yolov3_10cls_tiny_anchor has a higher percentage of recalling the correct total amount of objects in each photo.

According to figure 3, the highest mean Average Precision score is yolov3_10cls_tiny and the lowest is yolov3_10cls_panet_tiny. Finally, the highest F1-score was yolov3_10cls_tiny and on the contrary, yolov3_10cls_panet_tiny had the lowest score. From this result we can conclude that if we change yolov3_10cls_anchor to k-means gathered data, the precision value would decrease while the recall value increases. However, yolov3_10cls_tiny would raise the overall object detection of yolov3 whereas, the structure of yolov3_10cls_panet would have a lower score in every category. In this scenario, we want to detect all everything such as the road, cars, humans, and traffic signs, that is within the dataset which means we want both a high recall and precision rate. This means that the model yolov3_10cls_tiny qualifies for this scenario the most as it has a high recall rate, high precision rate, high mAP@0.5, and high F1-score.

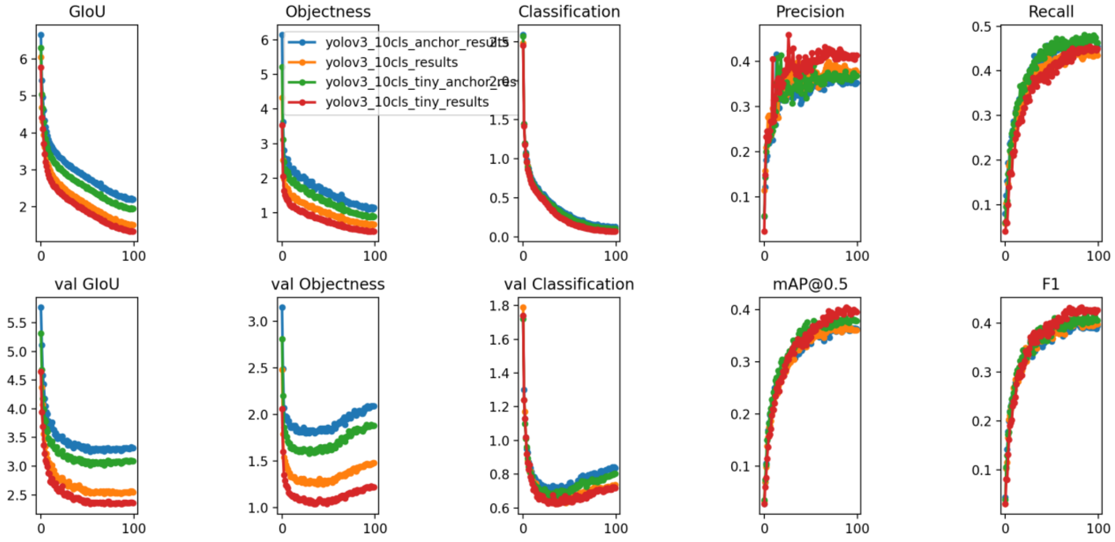
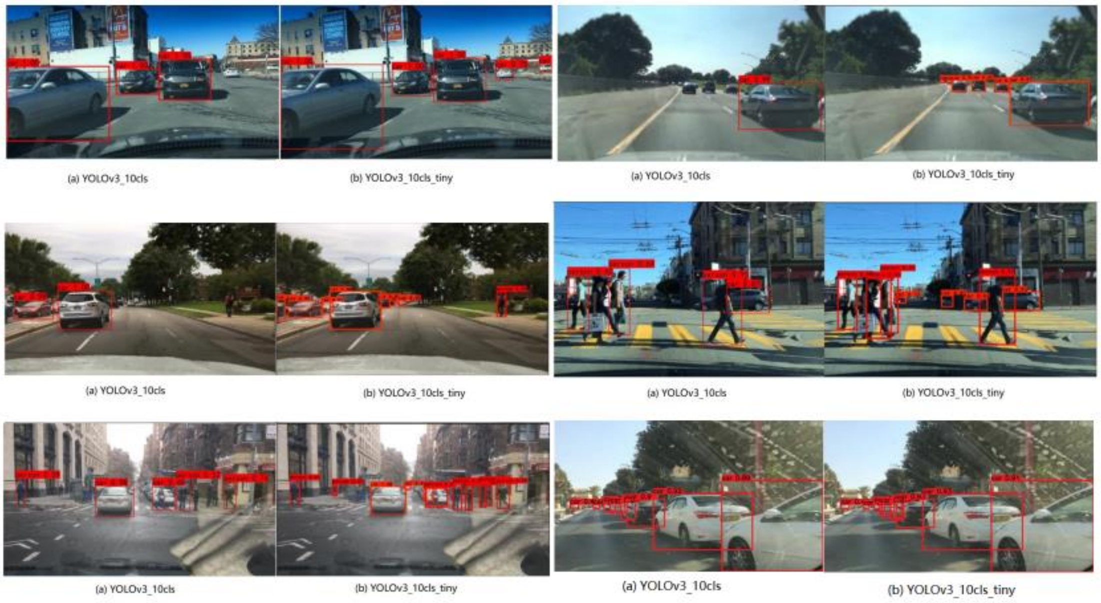


Figure 3: Performance comparison of different improved models

Figure 4 is the effect detection diagram of YOLOv3_10cls algorithm and YOLOv3_10cls_tiny algorithm. It can be seen that YOLOv3_10cls has missed detection for occluded pedestrian and vehicle targets. Missing detection also occurred for smaller pedestrian and vehicle targets YOLOv3_10cls in the distance. YOLOv3_10cls_tiny algorithm can correctly detect the traveler and vehicle targets without missing detection, and the detection effect is better than YOLOv3_10cls algorithm. It can be seen that the improved YOLOv3_10cls_tiny algorithm has better accuracy in target recognition and fewer missed detections.



(a) YOLOv3_10cls (b) YOLOv3_10cls_tiny (a) YOLOv3_10cls (b) YOLOv3_10cls_tiny

(a) YOLOv3_10cls (b) YOLOv3_10cls_tiny (a) YOLOv3_10cls (b) YOLOv3_10cls_tiny

(a) YOLOv3_10cls (b) YOLOv3_10cls_tiny (a) YOLOv3_10cls (b) YOLOv3_10cls_tiny

Figure 4: Comparison of YOLOv3 _ 10cls and YOLOv3 _ 10cls _ tiny detection results

5. Conclusion

In this paper, an improved algorithm based on YOLOv3 is proposed to solve the problems of poor stability and insufficient generalization ability in pedestrian and vehicle detection of traditional feature extraction algorithms in the neighborhood of road target detection, which leads to inaccurate pedestrian and vehicle detection or missing detection. This paper makes a variety of improvements to YOLOv3, including trying to use oversampling to improve sampling accuracy, that is, increasing the number of samplings in the third layer of the feature pyramid; upgrading the feature pyramid to PaNet, adding a new sampling fusion process to improve performance; try changing anchor hyperparameter settings to k-means clustering results on data sets. By comparing the three improvements, it is found that the improvement effect of trying to increase the sampling number of the third layer of the feature pyramid is the best, and YOLOv3_10cls_tiny is proposed. At the same time, we use GIoU as the loss function, because GIoU continues the good features of IoU and eliminates the fatal weaknesses of IoU, which not only focuses on overlapping areas but also on non-overlapping areas, so it can better reflect the degree of overlap between the two. Compared with other improved algorithms, YOLOv3_10cls_tiny has better average accuracy and higher F1-score than YOLOv3_10cls_panet and YOLOv3_10cls_anchor.

References

[1] Li, F. (2016) Talks About the Development and Future of Automatic Driving Technology. Heilongjiang Science and Technology Information, 16, 59.

[2] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016) You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (pp. 779-788). IEEE.

[3] Xin, Y. (2014) Research on Methods in Dynamic Obstacles Detection, Prediction and Avoidance of Autonomous Vehicles [Unpublished doctoral dissertation]. University of Science and Technology of China.

[4] Girshick, R.B. (2015) Fast R-CNN. arXiv. https://arxiv.org/abs/1504.08083.

[5] Redmon, J. and Farhadi, A. (2016) YOLO9000: Better, Faster, Stronger. arXiv. https://arxiv.org/abs/1612.08242.

[6] Redmon, J. and Farhadi, A. (2018) YOLOv3: An Incremental Improvement. arXiv. https://arxiv.org/abs/1804.02767.

[7] Chen, K., Zhu, Z., Deng, X., Ma, C. and Wang, H. (2021) Overview of Deep Learning Research on Multi-scale Target Detection. Journal of Software, 32(4), 1201-1227.

[8] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2013). Rich feature hierarchies for accurate object detection and semantic segmentation. IEEE Computer Society.

[9] Girshick, R. (2015). Fast R-CNN. Computer Science.

[10] REN, S., HE, K., & GIRSHICK, R. (2017). Faster R-CNN: towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence.

[11] Uijlings, Rr, J., Sande, V. D., Ea, K., Gevers, & Smeulders, et al.(2013) Selective search for object recognition. International Journal of Computer Vision, 104(2):154-171.

[12] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: unified, real-time object detection. IEEE.

[13] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., & Fu, C. Y., et al. (2016). Ssd: single shot multibox detector. Springer, Cham.

[14] Redmon, J., & Farhadi, A. (2018). Yolov3: an incremental improvement. arXiv e-prints.

[15] Zou, H. and Hou, J., 2021. Research on road small target detection based on improved SSD algorithm. Computer Engineering.

[16] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv.

[17] Luo, J., Huang, J. and Bai, X., 2022. Improved YOLOv3 Road Small Target Detection Method. Mini-Micro Systems, 43(03), pp.449-455.

[18] Liu, Y., Zhang, G., Xu, Q. and Zhang, Y., 2021. Vehicle Detection Method Based on RF-YOLOV3 Algorithm. Modern Electric Technique, 44(13), pp.153-158.

[19] Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263-7271).

[20] Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., & Savarese, S. (2019). Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 658-666).

[21] Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., ... & Darrell, T. (2020). Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 2636-2645).