Research on pedestrian detection algorithm in driverless urban traffic environment

Xinchao Liu[1,*], *Ying* Yan[1], and *Haiyun* Gan[1]

[1]Tianjin University of Technology and Education, Tianjin 300222, China

Keyword: TidyYOLOv4, Pedestrian detection, Model pruning, Unmanned.

Abstract. Pedestrian detection in urban traffic environment is an important field of driverless vehicle research. Due to the variability of traffic flow, target detection algorithm cannot extract complete feature information, which brings great challenges to driverless pedestrian detection. Target detection algorithm YOLOv4 has excellent detection performance in object detection, but it is not perfect in identifying semi-blocked pedestrians. In this paper, the Spatial Pyramid Pooling was added in front of the third yolo detection head module of YOLOv4 to optimize the extraction of deep network features. Then, on the basis of optimizing the network, pruning strategy was adopted to simplify the target detection algorithm, which was called TidyYOLOv4.TidyYOLOv4 and YOLOv4 (network set input image size is 864×864) were compared on the self-made human head data set. Total BFLOPS decreased by 95.04% and Inference time decreased by 82.82%. The above experimental results show that the optimized TidyYOLOv4 algorithm is more suitable for driverless pedestrian detection in urban traffic environment.

1 Introduction

With the progress of artificial intelligence, driverless vehicles have become one of the main research and development directions. Unmanned driving adopts a number of technology fusion detection, among which visual detection is one of the most important detection technologies. Pedestrian detection in urban roads is the basic task of visual perception applied to driverless cars in various traffic scenes. Because when the driverless vehicle does not detect the pedestrian in the road accurately, it may harm the life and safety of the pedestrian. Therefore, it is very important to ensure the accuracy of pedestrian detection. With the progress and improvement of deep learning algorithm, the detection of road pedestrian has been further improved, but it still needs to be further improved in practical application. There are two main problems: (1) The deep neural network vision algorithm needs strong computing power and running space. Currently, it is mainly used to test and verify its detection performance on the server, which is difficult to store and run on

* Corresponding author: liuxinchao_com@sina.com

the on-board chip. (2) The complex traffic flow will make the target detection algorithm fail to extract complete feature information (for example, the body part of the pedestrian is blocked by other vehicles or traffic signs), so it is necessary to rely on part of the acquired information to determine the characteristics of the target.

In order to solve the problem that deep learning target detection algorithm cannot be applied to unmanned chip, pruning algorithm is developed to reduce the spatial volume of target detection algorithm and reduce the consumption of computing force, so as to realize the reasonable deployment of target detection algorithm on unmanned chip. In order to improve the detection effect of obscured the pedestrian, pedestrian in the presence of block data set to validate the performance of the improved algorithm, due to the characteristics of the legs, hands and body information exists strong uncertainty, so choose to identify with high degrees of the head as an object of annotation besides has remarkable characteristics in the road obscured the probability is relatively low. Such annotation does not exist in the common open source pedestrian data set, so we made the head annotation data set with the human body partially obscured. The experimental results show that TidyYOLOv4, an optimization algorithm based on this data set, is more suitable than YOLOv4[1] to be applied to the detection of pedestrians by driverless vehicles in urban traffic environment.

2 Related work

Machine vision is mainly divided into two categories: (1) Classifying the element information in the image; (2) To locate the object information in the image, and target detection is the fusion problem of classification and positioning. The initial target detection algorithm mainly extracts the target information in the image through the sliding window, and then analyzes the target positioning and classification. The result of the analysis cannot achieve satisfactory results. Until the advent of R-CNN aroused the interest of a large number of researchers, and became one of the hot research areas in the field of vision.Now more excellent target detection algorithms have been developed on the basis of R-CNN, such as R-CNN[2], Fast R-CNN[3], R-FCN[4], SSD[5], YOLO[6], YOLOv2[7], YOLOv3[8], YOLOv4[1], etc.

These deep target detection algorithms are mainly divided into two categories according to their different network architectures: one is a two-stage target detector represented by R-CNN and Fast R-CNN, which is composed of three major modules, namely, regional recommendation module, backbone network and detection head. First of all, the region detection module of the two-stage target detector will generate suggestions with regions of interest, and the detection head will conduct information classification based on these suggestions. Finally, position regression will be carried out to accurately locate the target object. The two-stage target detector achieves excellent detection accuracy through region suggestion. Its running process not only requires huge loss of computing power and running memory, but also leads to slow real-time target detection. In the other category, the single-stage target detector represented by YOLO series and SSD is set with k prior boxes densely covering each specific position of the image at each position of the feature graph, and no branch network similar to the regional suggestion is used. Therefore, the single-stage detector is faster than the two-stage detector in reasoning. In the single-stage target detector, YOLOv4 target detection algorithm has excellent detection speed and advanced detection accuracy. Therefore, In this study, YOLOv4's target detection algorithm was selected as the basic algorithm model for pruning. In combination with pruning strategy, a more efficient target detection model, TidyYOLOv4, was learned to improve the real-time detection of pedestrians by driverless vehicles in urban traffic.

3 Network

3.1 Network optimization

YOLOv4 is an advanced algorithm that is constantly optimized and improved from YOLO algorithm.YOLOv4 algorithm is mainly based on YOLOv3 combined with the existing advanced optimization strategy to complete, it has been greatly improved in speed and accuracy. For the sake of better detection effect of network model, YOLOv4 combines the thought of CSP-Net[9] on the basis of Backbone-53 to construct CSPDarknet-53 to greatly improve the transmission effect of network algorithm, while the Neck combines the advantages of SPP[10] and PAN[11] to strengthen the extraction of deep network, and Head uses the detection method of YOLOv3 for reference.

In order to fully enhance the feature extraction of deep structure in the experiment, an SPP module was added between the 5th and 6th convolutional layers in front of the third detection head of YOLOv4 to improve the detection effect, and YOLOv4-SPP1 was combined. As shown in Figure 1 below:

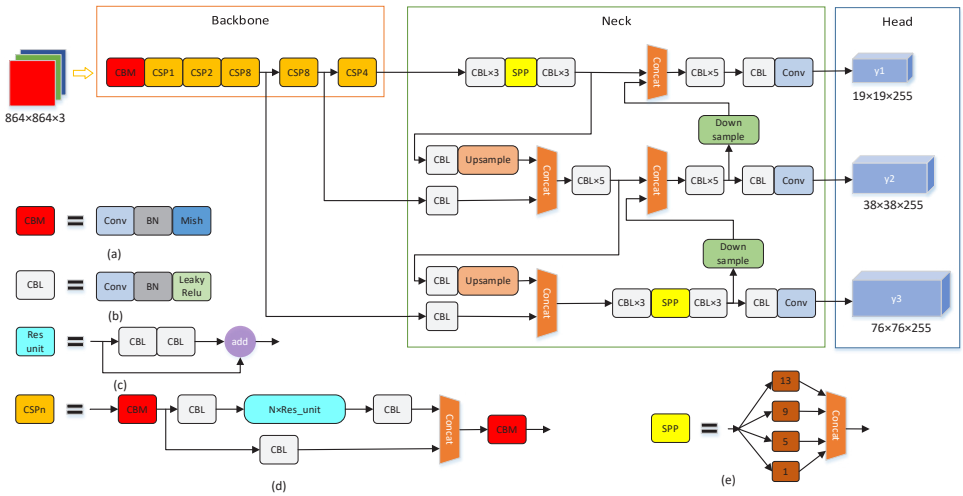

Fig. 1. Schematic diagram of TidyYOLOv4 architecture.

3.2 Network pruning

Pruning strategy is adopted to reduce the running resource consumption of target detection algorithm and improve detection efficiency. Based on the optimized network of YOLOv4-SPP1, the model was simplified, and the optimized network TidyYOLOv4 was obtained through the iterative process of network pruning in Figure 2.

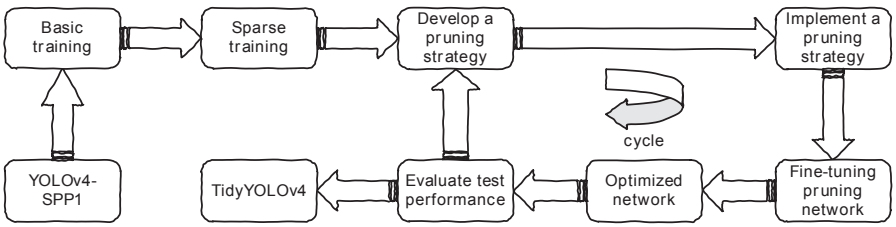

Fig. 2. Iterative process of TidyYOLOv4.

The iterative process of pruning optimization for YOLOv4-SPP1 network :(1) Select an appropriate basic network framework;(2) Basic training;(3) Conduct sparse training on the network after basic training;(4) Evaluate the importance of deep model and develop pruning strategies;(5) Pruning the network model by implementing pruning strategy;(6) Fine-tune the pruned model to fully improve the potential algorithm performance;(7) The network after pruning optimization;(8) Conduct pruning deployment again when the performance of the optimized model fails to meet the optimal requirements;(9) No pruning will be performed when the optimal network model required by the experiment is evaluated, namely, the optimal network TidyYOLOv4.

Optimization strategy: Firstly, By adding the L1 regularization on channel scale factor [12, 13] to enhance the sparse nature of channel level help structured pruning, after introducing the global threshold adjust the cutting rate of the channel, and then by cutting off the scaling factor average minimum convolution layer to further improve the detection efficiency and get optimal algorithm TidyYOLOv4. In this experiment, the improvement proposed in Liu [14] method was improved into a coarse-grained depth model search method to explore a more efficient target detector.

4 Experiment

After a series of optimization on the basis of YOLOv4, the target detection algorithm TidyYOLOv4 which conforms to the detection of pedestrians in urban roads was optimized. The validity of the algorithm is verified by the following experiments.

4.1 Experimental environment

During the experiment, the deep network algorithm running platform should be configured to meet the requirements of TidyYOLOv4.The operating environment is CPU/GHZ (Inter Xeon E5-2603 ,Memory/GB 16),GPU(Tesla P4， 8GB)and Operating System(Ubuntu 16.04).

4.2 Data set

In order to verify the effectiveness of the optimization algorithm for half-blocked pedestrian heads, a half-blocked pedestrian head data set[15] was made to verify the optimization algorithm TidyYOLOv4. This data set was a picture pixel of 1280×720 that was analyzed from the video recorded by DV. Then LableImg was used to make the head tag, and a total of 10,870 data sets were used. The data set is manually tagged, and the images are strictly completed in the same markup manner. The experiment was divided into 8696 pictures of training set, 1087 pictures of verification set and 1087 pictures of test set on a scale of 8:1:1.

Fig. 3. Shows an example of a dataset.

4.3 Model training

The training and verification of this experiment is implemented in PyTorch, a deep learning framework. During the training process, 4 information pictures were sent to the network for the training of 100 Epochs at a time. The learning rate of the initial training was 0.01, and the learning rate was ten times smaller when the network training iteration reached 70% and 90% of the whole process. The weight attenuation is set to 0.001 and the momentum is set to 0.9.

Sparse training: Each YOLO model trained 100 epochs. Sparse training of 300 Epochs on top of 100 Epochs was carried out to promote network pruning. Due to different learning rates, appropriate punishment factors were selected for training. In this experiment, 0.0001 punishment factor was set for training. Other setting parameters were the same as the basic training.

4.4 Performance indicators

The following indexes are used to analyze the performance of the optimization model: 1.Precision; 2.Recall; 3.mAP; 4.Total BFLOPS;5.Inference time (ms); 6.Parameters(M); 7.Volume(MB).

5 Analysis of experimental results

The experimental results in the table were used to analyze the experimental results of the basic model and the learning model with different optimization strategies, so as to select the optimal target detection model (YOLOv4-SPP1-X, SPP1 means a Spatial Pyramid Pooling was added on the basis of YOLOv4, X means pruning X%).

Table 1. Experimental results.

Model	Input size	Precision	Recall	mAP	Total BFLOPS	Inference time(ms)	Parameters(M)	Volume (MB)
YOLOv3	416	83.30	73.00	76.10	67.29	29.00	60.52	241.40
	864	86.50	88.10	88.90	285.65	88.20		
YOLOv4	416	86.40	81.70	81.40	68.67	30.70	63.67	255.60
	864	89.60	90.90	91.50	286.27	91.40		
YOLOv4-SPP1	416	86.91	82.60	83.90	74.17	35.96	65.75	257.80
	864	90.90	92.70	94.30	313.02	92.68		
YOLOv4-SPP1-95	864	88.50	88.80	90.20	17.10	19.00	0.80	3.90
YOLOv4-SPP1-96	864	87.80	84.40	87.50	14.20	15.70	0.66	2.70
	864	85.60	76.20	82.50	12.99	15.00	0.45	
YOLOv4-SPP1-97								1.80

Table 1 is the experimental results of several groups of different pruning degrees. It can be seen from the Input size column in the table that the evaluation indexes of YOLOv3, YOLOv4 and YOLOv4-SPP1 will also be significantly improved as the size of the network input picture increases from 416×416 to 864×864, among which the mAP of YOLOv3 increases by 12.8 and Recall increases by 15.1. The mAP of YOLOv4 increased by 10.1, and Recall increased by 9.2. The mAP of YOLOv4-SPP1 increased by 10.4, and Recall increased by 10.1. Therefore, the network model with the input picture size of 864×864 is selected for optimization and simplification of different pruning rates.

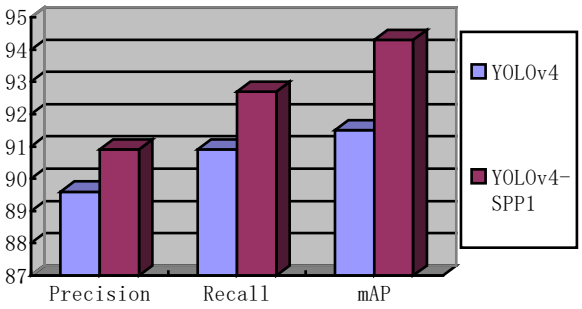

Fig. 4. Evaluation indexes of YOLOv4 and YOLOv4-SPP1 (Table 1, Input size is 864).

According to the comparison of evaluation indexes of YOLOv4 and Yolov4-SPP1 in Figure 4, it can be seen that the addition of evaluation indexes to the spatial pyramid presents an increasing trend, indicating that it is effective to improve feature extraction by adding the spatial pyramid module before the detection head of YOLOv4. Therefore, YOLOv4-SPP1 was selected as the pruning network.

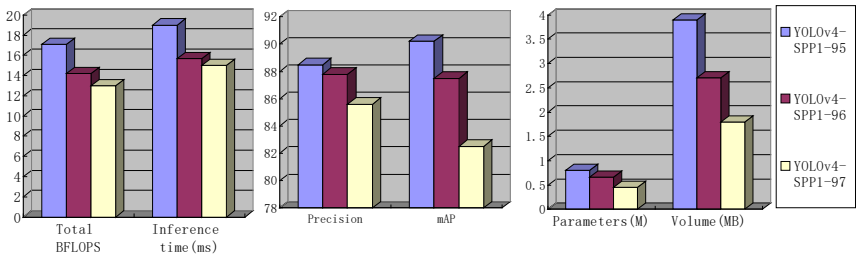

Fig. 5. The evaluation indexes of YOLOv4 and YOLOv4-SPP1 trimmed by 95%, 96% and 97% respectively (Table 1, Input size is 864).

Fig. 6. Visual detection effect of YOLOv4.

In the experiment, pruning of YOLOv4-SPP1 was carried out to different degrees. It can be seen from the figure that when the pruning rate of the model increased to 96%, it was the best. When the pruning rate reached 97%, the overall performance of the detection model began to show a downward trend. Based on the comparative analysis of evaluation indexes of YOLOv4-SPP1-95, YOLOv4-SPP1-96 and YOLOv4-SPP1-97 in Figure 5, YOLOv4-SPP1-96 was selected as the final optimized network model TidyYOLOv4.

Detection effect analysis: as can be seen from Table 1, TidyYOLOv4 was 272.07 less in Total BFLOPS and Inference time than YOLOv4 under the same network setting and

experimental environment, 75.70ms less in Inference time, 63.01M less in Parameters and 252.9MB less in Volume. Fig.6 and fig.7 of the visual detection effect of YOLOv4 and TidyYOLOv4 show that there is no obvious difference in detection effect. However, the inference time of each frame is reduced by 75.70ms, which greatly reduces the time of target detection and leaves more recognition and processing time for target detection.

Fig. 7. Visual detection effect of TidyYOLOv4.

6 Conclusion

In this experiment, TidyYOLOv4, a target detection algorithm suitable for driverless urban traffic roads, was optimized. First, as shown in figure 1, this paper improves network feature extraction by adding SPP before the third detection head of YOLOv4.Secondly, the redundancy of YOLOv4-SPP1 training model is pruned through the joint pruning strategy of layer and channel to obtain a more efficient detection model. Finally, in order to automatically identify the unimportant parts of the training model, sparse L1 regularization is applied to the channel scaling factor to implement pruning strategy, and appropriate scaling factors are adjusted to trim the unimportant parts of the network model to improve the performance of the target detector. Based on this strategy, the TidyYOLOv4 model is optimized on the basis of the original model YOLOv4 (the size of the input image of network setting is 864×864).Compared with YOLOv3, TidyYOLOv4 not only has higher detection speed and better detection accuracy, but also has a 99.05% reduction in model space volume compared with YOLOv4.Therefore, it is concluded that TidyYOLOv4 is more suitable than YOLOv4 to be applied to the detection of pedestrians in the urban traffic environment by driverless vehicles.

This work was supported in part by the Tianjin University Discipline Leading Talent Training Program of China(No. SSW181030105) and the Tianjin Artificial Intelligence Project of China(No. 18ZXAQSF00090).

References

1. A. Bochkovskiy, C.Y. Wang , H.Y.M. Lia. "YOLOv4: Optimal Speed and Accuracy of Object Detection." (2020).arXiv:2004.10934v1
2. R. Girshick, J. Donahue, T. Darrell, J. Malik. Rich feature hier-archies for accurate object detection and semantic seg-mentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.
3. S. Ren, K. He, R. Girshick, J. Sun. Faster R-CNN: To-wards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Trans. Pattern Anal. Mach. Intell. **39**, 1137–1149 (2017).

4. J. Dai, Y. Li, K. He, Sun. J."R-fcn: Object de-tection via region-based fully convolutional networks." Advances in neural information processing systems. (2016).
5. J. Ye, X. Lu, Z. Lin, J.Z. Wang. Rethinking the smaller-norm-less-informative assumption in channel pruning of convolution layers[J]. arXiv:1802.00124, (2018).
6. J. Redmon，S. Divvala，R. Girshick，A. Farhadi. "You only look once: Unified, real-time object detection[J]." OALib Journal. (2016). ISSN: 2333-9721.
7. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, A.C. Berg. SSD: Single Shot MultiBox Detector.[C] European Conference on Computer Vision, 2016: 21-37.
8. J. Redmon, A. Farhadi. "YOLOv3:an incremental improvement[J]." arXiv:1804.02767v1 (1804),(2018).
9. C.Y. Wang, H.Y.M. Liao, Y. H. Wu, P.Y. Chen, I.H. Yeh. (2020). CSPNet: A New Backbone that can Enhance Learning Capability of CNN. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE.
10. K. He, X. Zhang, S. Ren, and J. Sun.Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analy-sis and Machine Intelligence (TPAMI), 37(**9**):1904–1916,(2015).
11. S. Liu, L. Qi, H. Qin, J. Shi, J. Jia. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages8759–8768, (2018).
12. A, S. S., A, D. C., B, A. H., A, A. U.: Group sparse regularization for deep neural networks. Neurocomputing, **241**, 81-89 (2017) arXiv preprint at https://arxiv.org/abs/1607.00485
13. Glenn Jocher, guigarfr, perry0418, Ttayu, Josh Veitch-Michaelis, Gabriel Bianconi, IlyaOvodov. (2019). ultralytics/yolov3:Rectangular Inference, Conv2d + Batchnorm2d Layer Fusion (Versionv6). Zenodo. http://doi.org/10.5281/zenodo.2672652
14. Z. Liu, J. Li, Z. Shen, G. Huang, S. Yan, C. Zhang.: "Learning efficient convolutional networks through network slimming." Proceedings of the IEEE International Conference on Computer Vision (2017). https://doi.org/10.1109/ICCV.2017.298
15. A. Ben Khalifa, I. Alouani, M. Ali Mahjoub, N. Essoukri Benamara, Pedestrian Detection Using a Moving Camera: A Novel Framework For Foreground Detection, Cognitive Systems Research (2019), doi: https://doi.org/10.1016/j.cogsys.2019.12.003.