Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 3, No. 1, 2023

Vehicle target detection algorithm based on yolov5

Xing He

College of Electrical Engineering, Southwest Minzu University. Chengdu, Sichuan 610041, China

Abstract: With the rapid advancement of automobile manufacturing technology and the intensification of competition among automobile brands, the development of autonomous driving has been pushed to the forefront of automobile development, which causes approximately $650 billion in losses due to traffic accidents worldwide every year. For in the complex vehicle target scene, because of its many vehicle targets, dense targets often exist between the occlusion, overlap, a variety of different weather reasons, rain, fog, sunny days, etc. resulting in obstruction of the field of view, and the vehicle camera often intercepted images mostly exist blurred, ghosting and other problems, making the vehicle target detection for detailed feature extraction requirements are high, the detection accuracy is often difficult to meet the requirements, proposed based on YOLOv5s vehicle target detection algorithm is proposed. ACmix attention mechanism is introduced to make the model achieve the purpose of expanding the target perception field, adaptively focusing on different target regions, and capturing more information features, etc. The test shows that the accuracy of the model increases by 1.1% on a subset of BDD100K data set after improvement. The accuracy of the improved model increased by 1.1% on the subset of BDD100K dataset and by 1.2% on the PASCAL VOC2007 general dataset.

Keywords: Yolov5s; ACmix; Deep learning.

1. Introduction

Machine learning class algorithms were the first deep learning algorithms, and in 2000, the SVM algorithm was proposed by Osuna [1], which aims to find a reasonable curve or a plane to segment the binary objects perfectly; in 2001, Jones[2] proposed the VJ algorithm. In 2004, the SIFT[3] algorithm was refined to preserve the robustness of the size and rotation variations of the features extracted from the image; these were the early target detection algorithms. 2013, Girshick et al. In the network, the candidate regions (Region Proposals) in the image are obtained by Selective Search, and then the candidate regions are input to the Convolutional Neural Network (CNN) to extract features, and finally the SVM completes the region classification. However, R-CNN uses selection search also generates a large number of redundant candidate regions, which lowers the detection speed of the model[4] .

The SPP-Net model was introduced in 2015 by K. Ho et al. This model solves the subsequent problems caused by images with inconsistent sizes of candidate regions, and it can accept any size of image input into the convolutional neural network to perform convolutional operation on the whole image, eliminating the redundant computation of each region. In addition SPP-Net also adds SPP Pooling behind the convolutional network, which makes the feature maps of different sizes converted to fixed size size. However, the classifier of SPP-Net is still an SVM, which cannot be trained end-to-end, so a large number of pixels need to be stored. In addition, it is difficult to adjust the whole convolutional neural network by adding SPP-Net, resulting in the convolutional neural network is not well used in practical applications

2015 Girshick et al.[6]borrowed the practice of global feature extraction of feature maps in SPPNet, and improved on the basis of R-CNN to propose Fast R-CNN, which solved the problem of cumbersome training of R-CNN and each training stage being independent of each other, and secondly solved the problem of requiring consistent input image size, and finally replaced the original SVM classifier with softmax. faster R- CNN[7] Compared with Fast R-CNN, its biggest innovation is to abandon the use of selective search network and use Region ProPosal Network (RPN) as a replacement. The RPN, as the core network of Faster R-CNN, uses a fully convolutional layer instead of a fully connected layer. Its biggest advantage is that it can automatically learn to distinguish the foreground and background of the image during candidate region extraction by the algorithm to extract valid candidate regions and avoid generating candidate regions with negative samples, thus reducing the number of candidate regions.

The YOLO (you only look once) series is one of the representative works of single-stage target detection methods. Single-stage target detection does not have the process of finding candidate regions, and its can be unified into a regression problem. Compared with, for example, the Faster R-CNN algorithm, YOLO can better distinguish the foreground and background of an image by this way, and the detection speed is faster. However, the YOLOv1[8] algorithm requires a fixed size of the input image, and when the input image is divided into a grid, each grid can only predict a single object, and when multiple objects appear inside the same grid, the NMS algorithm will be used to merge, filter, and delete the bounding box, and finally leave the bounding box with the highest confidence, which will lead to the target increase of missed detection rate. Later, scholars improved YOLO, and there are YOLOv2 version, YOLOv3 version, YOLOv4 version, etc.

Redmon [9] et al. proposed the YOLOv2 algorithm to address the shortcomings of YOLO, which improves on YOLO in three main aspects. The first aspect is that YOLOv2 adopts the Darknet-19 network as the backbone network, mainly using 3 × 3 convolution and 1 × 1 convolution, and discards the Dropout network and adds a batch normalization after each convolution, which not only accelerates the convergence speed of the model in the training phase, but also effectively reduces the possibility of overfitting the model. The second aspect is the use of convolutional layers instead

of fully connected layers in YOLO, which reduces the sensitivity of the network structure to the size of the input image, and thus allows multi-scale training by changing the size of the image, and then fine-tuning the initial classification network with a high-resolution classifier, which can reach 448×448 in YOLOv2. The third aspect is the introduction of the anchors box in The idea of anchors box is introduced in prediction, and the direct regression method is improved by introducing anchors box. Compared with the YOLO algorithm, the YOLOv2 algorithm has all improved performance in detection speed and detection accuracy, but the YOLOv2 algorithm is still poor in detecting small targets in images.

YOLOv3 [10] makes improvements on the basis of YOLOv2 and has better performance. the YOLOv3 algorithm uses Darknet53 network in the feature extraction part, and its structure is referred to the residual network, and layer-hopping connections are made in different network layers to fuse deep and shallow features, which increases the depth of the network layers and not only reduces the loss of feature information between feature layers in the convolution process This not only reduces the loss of feature information during the convolution process, but also reduces the number of parameters and the amount of operations, and improves the detection speed. In addition, YOLOv3 uses a similar operation to FPN network in the detection process, after upsampling the shallow features, 1×1 convolution is performed, and then the two feature maps are summed to complete a feature fusion. After repeated such top-down feature fusion three feature maps of different sizes are obtained. Using multi-scale prediction, the target is predicted on three different sizes of feature maps. In addition, for label classification, YOLOv2 uses the softmax function, but when the target of the detected image has more than one category, the softmax function will only output the prediction type of a certain category, so softmax is not suitable for multi-category label prediction. YOLOv3 uses an independent logistic classifier instead of softmax to compensate for this drawback. The classifier will assign a certain threshold range to each category, and when the resulting data is within a certain threshold range, it corresponds to a certain category prediction. Compared with the YOLOv2 algorithm, YOLOv3 greatly improves the detection accuracy with guaranteed speed.

YOLOv4[11], as an enhanced version of YOLOv3, has made some changes in the network structure to address the shortcomings of YOLOv3. YOLOv4 divides the whole network into three parts, including backbone, Neck and Head, to make the overall structure of the network clearer, and YOLOv4 adopts the idea of SPPNet to increase the perceptual field by adding a SPP layer after YOLOv4 adopts the idea of SPPNet to increase the perceptual field, adding an SPP layer after backbone, which actually adopts different size of pool_size and strides to realize the feature output of different perceptual fields.

2. YOLOV5 framework

Yolov5 contains four versions: yolov5s, yolov5m, yolov5x, and yolov5l. Because yolov5s has the smallest model structure, the smallest number of parameters, and the fastest running speed among the four versions, this paper selects yolov5s as its backbone network, and the overall framework of yolov5s is shown in the following figure, which can be distinguished into three parts Backbone, Neck, and Head.

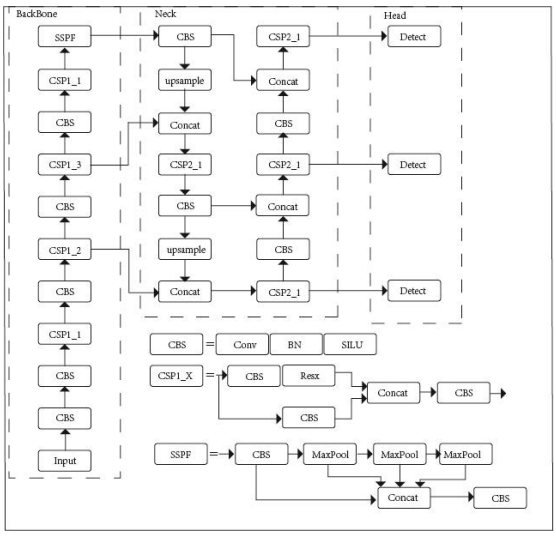

Figure 1. Yolov5s structure diagram

Backbone is the backbone network of the model, and the Focus module is removed after V6.0. yolov5 adopts the same Mosaic data enhancement as yolov4, i.e., four images are randomly selected from the original image for stitching, which can prevent data overfitting and reduce the local loss caused by data sampling. the cross-stage localization (CSP) module is used in yolov5. There are two designs in yolov5, CSP1_X structure and CSP2_X structure, the former contains N residual structure to solve the degradation problem of deep network, the latter mainly connects Neck part to fuse the semantic information of high level with the location information of bottom level, the output has 3 different sizes of probe head, the generated prediction frame is NMS suppressed to save the higher confidence prediction frames.

3. Hybrid Self-Attention Mechanism (ACmix)

The ACmix hybrid attention mechanism [12] combines the advantages of both traditional convolutional and self-attentive mechanisms. The former uses an aggregation function over the local receptive field based on convolutional filter weights, which are shared throughout the network structure to obtain indispensable inductive bias with its inherent properties. The latter uses a weighted averaging operation based on the input feature context, and the attention weights are dynamically calculated by the similarity function between neighboring pixel pairs, so as to expand the perceptual field of the network model, focus on different target regions adaptively, capture more information features, etc.

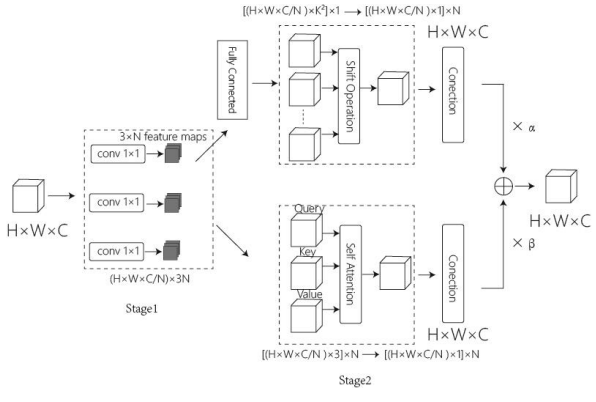

Figure 2. ACmix Hybrid Self-Attention Structure

Traditional convolution and self-attention modules actually share the same operation when projecting input feature maps by 1×1 convolution, and these two operations take up the main computational overhead. Based on this idea, the ACmix hybrid self-attention mechanism is mainly divided into two stages, as shown in the figure, in the first stage, the input tensor of H×W×C is projected in three 1×1 convolutions and reshaped into N to obtain an intermediate feature set containing 3N feature maps, and in the second stage, for the self-attention module, ACmix aggregates the intermediate features into N groups, each containing three feature maps, each from 1 × 1 convolution, and the corresponding feature maps denote Query, Key and Value, respectively, following the traditional multi-headed self-attention model, as shown in Equation (1)

$$g_{ij} = \mathop{\Big\|}_{l=1}^{N} \left(\sum_{a,b\in N_{k(i,j)}} A(q_{ij}^{(l)}, k_{ab}^{(l)}) V_{ab}^{(l)}\right) \quad (1)$$

where g_{ij} denotes the projection tensor corresponding to pixel (i, j), || denotes the concatenation of N attention head outputs, and $N_{k(i,j)}$ denotes the local region of pixel space centered at (i, j) ranging over K$A(q_{ij}^{(l)}, k_{ab}^{(l)})$ Corresponding to$N_{k(i,j)}$ the attentional power of the features within weight, the size of the attention weight assigned to value depends on the matching similarity between query and key, if the similarity is higher, the size of the weight assigned is consequently larger and vice versa; for the traditional convolutional path, the convolution with a convolutional kernel of K as shown in Equation (2) (3) is connected by a lightweight fully connected layer to obtain k^2 feature maps, generated by moving and aggregating the features, shown in Equation XX, to process the input features in a convolutional manner and collect information from the local receiver domain as in the traditional receiver domain.

$$g_{ij}^{(p,q)} = Shift(\tilde{g}_{ij}^{(p,q)}, p - [\tfrac{k}{2}, q - [\tfrac{k}{2}]]) \quad (2)$$

$$g_{ij=\sum_{p,q} g_{ij}^{(p,q)}} \quad (3)$$

Finally, the two paths of the conventional convolution and the self-attentive module are summed, and the intensity is controlled by the two learnable scalars α, β controlled by Equation 4

$$F_{out=}\alpha F_{att} + \beta F_{conv} \quad (4)$$

where F_{out} denotes the final path output of the ACmix module, and F_{att} denotes the path output of the second-stage self-attentive module branch, and F_{conv} denotes the path output of the branch of the second-stage convolution module, and the intensity controllable scalarα, β range from 0 to 1. In this paper, the learnable scalar α, β values are 1.

4. Experimental results and analysis

4.1. Data set

The dataset used for the experiment is BDD10K, a subset of BDD100K released by the University of California, Berkeley, which is the largest and most diverse autonomous driving dataset to date. This experimental dataset contains a total of 10,000 images, with a total of 10 categories of vehicles, buses, pedestrians, bicycles, trucks, motorcycles, etc. It is divided into a training set, a test set, and a validation set according to 7:2:1. An example of the dataset is shown in Fig 3.

Figure 3. Example data set diagram

4.2. Training environment

The experimental platform on which this experiment is based is shown in Table 1

Table .1 Experimental platform environment

Parameters	Configuration
CPU	R7-5800H
GPU	3060
Framework	Pytorch 1.8.1
CUDA	11.1
Experimentalplatform	VScode
Language	Python 3.8.1
Batch-size	4
workers	8

The experimental algorithm is based on CUDA 11.1 and pytorch version 1.8.1. The hyperparameters are Batch-size=4, initial learning rate10^{-3} , Epoch=100, and the input image size is 640×640.

4.3. Experimental results and analysis

In this paper, we compare the detection results of the original yolov5 baseline model and the improved yolov5 network model on the BDD10K and PASCAL VOC2007

datasets as shown in Table 2

Table .2 Experimental results

Model	Test set	mAP@.5(%)	mAP@.5:.0.95(%)
YOLOV5	BDD10K	50.7	26.8
YOLOV5-ACmix		51.8	27.3
YOLOV5	VOC2007	84.7	65.1
YOLOV5-ACmix		85.9	65.3

Important evaluation indicators in target detection are average precision (AP) and mean average precision (mAP), which are calculated as follows.

$$AP = \int_0^1 P(R)dR \quad (5)$$

$$mAP = (\sum_{c=1}^{n} AP_c)/n \quad (6)$$

where P is the precision rate, which indicates the proportion of true positive samples to predicted positive samples; R is the recall rate, which indicates the proportion of true positive samples to actual positive samples; C is the predicted category; and n is the number of categories. mAP@.5 denotes the average precision of a class when the IOU confidence of the confusion matrix takes the value of 0.5. mAP@.5:.0.95 indicates mAP mAP The thresholds range from 0.5 to 0.95, and the mean value of mAP at a step size of 0.05.

The improved algorithm and YOLOV5 various algorithm metrics can be seen from Table 4-2 that The improved algorithm increased mAP@.5 by 1.1% and mAP@.5:.0.95 by 0.5% on BDD10K data, and its mAP@.0.5 increased by 1.2% on PASCAL VOC2007 dataset, the effect comparison graph is shown in fig

Figure 4. (a) Improved mode (b) YOLOV5 model

5. Conclusion

In this paper, we introduce the network structure of yolov5S and the ACmix hybrid attention mechanism, based on the yolov5 backbone network, add the hybrid attention mechanism ACmix into the network, and simulate the model in the BDD10k and VOC2007 datasets. The experiments show that the improved yolov5 model has improved the accuracy of the original model and the detection capability of small targets is greatly improved.

Acknowledgments

Funding: This research was funded by the Southwest Minzu University Graduate Innovative Re-search Project Grant No.(YB2022812)

References

[1] Osuna E, Freund R, Girosit F. Training Support Vector Machines: an Application to Face Detection[C]. IEEE Computer Society Conference on Computer Vision & Pattern Recognition, San Juan, PR, USA, 1997: 130-136.

[2] Viola P, Jones M. Rapid Object Detection using a Boosted Cascade of Simple Features[C]. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA, 2001: 511-518.

[3] Lowe D. Distinctive image features from scale-invariant key points[J]. International Journal of Computer Vision, 2004, 60 (2): 91-110.

[4] Girshick R, Donahue J, Darrell T, et a1. Rich Feature Hierarchies for Accurate Object Detection andSemantic Segmentation[C]. IEEE Conference on Computer Vision and Pattem Recognition, 2014: 580-587

[5] 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.

[6] Girshick R. Fast R-CNN [C]. IEEE International Conference on Computer Vision, 2015: 1440-1448

[7] Ren S, He K, Girshick R, et al. Faster R-CNN: Towards Real Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[8] Redmon J, Divvala S, Girshick R and Farhadi A. 2016. you only look once: Unified, real-time object detection / / Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA: IEEE: 779-788 [DOI: 10. 1109 / CVPR. 2016. 91].

[9] Redmon J, Farhadi A. YOLO9000: Better, Faster, Stronger[C]// IEEE Conference on Computer Vision & Pattern Recognition. IEEE, 2017:6517-6525

[10] Redmon J, Farhadi A. YOLOv3: An Incremental Improvement[J]. arXiv e-prints, 2018.

[11] Bochkovskiy A, Wang C Y and Mark Liao H Y. 2020. Yolov4: optimal speed and accuracy of object detection [EB / OL]. [2020-04-23]. https: / / arxiv. org / pdf / 2004. 10934. pd

[12] PAN X R, GE C J, LU R, et al. On the integration of selfattention and convolution [C]//Proceedings of 2022 IEEE/CVF Conference on Computer Vision and PatternRecognition. Washington D. C., USA: IEEE Press, 2022: 805-815.