ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

Vehicle Detection for UAV Images based on Fine-grained YOLOv3

Xingyao Yu[1, *], Yuhan Jia[2, a], Hongzheng Ni[3, b], Chao Qi[4, c], Hehao Wu[5, d]

[1] Soochow University, Suzhou, Jiangsu, China

[2] Beijing Institute of Technology, Beijing, China

[3] Xiamen University Malaysia, Selangor Darul Ehsan, Malaysia

[4] Nanchang University, Nanchang, Jiangxi, China

[5] University of Toronto, YTO, CAN

[a]1137752706@qq.com, [b]1192365942@qq.com, [c]1822419927@qq.com, [d]3128169102@qq.com

[*]Corresponding author: 1366259534@qq.com

These authors contributed equally to this work

Abstract

Unmanned aerial vehicles (UAVs) have a wide span of applications on different fields, especially on city depression angle shooting and traffic monitoring. We choose Yolov3 as our algorithm to detect tiny objects like vehicles shot by UAVs is the main problem. Based on Yolov3, we tried 3 methods and combined their advantages. First, since the big objects can be hardly found in pictures from UAVs, the biggest feature map is deleted to reduce the time this algorithm needs. Second, smaller strides are chosen to make detection more accurate without missing. Finally, anchor boxes are shrunk to detect smaller objects, improving the precision rate. It turns out smaller strides and anchor boxes showed good results, which made mAP rates rise by 3% and 5% respectively. Hence, we combined these two methods and the mAP rate rose by 12.1%. In conclusion, with smaller strides and anchor boxes, training results can be better.

Keywords

UAV; CNN; Yolov3; Small Object Detection; Vehicle Detection.

1. Introduction

Today, the applications of drones can be seen in a wide variety of industries which already deploy drones for their repspective purposes. For example, being equipped with thermal sensors, drones are able to locate the position of lost persons. They are also able to work in dark and inside a challenging terrain; In the field of agriculture, drone can keep an eye of failing plants and study the large sized farm lands along with proper monitoring of irrigation systems, thus helping farmers to save money. Because of the use of drones, we have the ability to safely and quickly gather data and to access inaccessible locations. It's no doubt that as a High-tech cutting-edge technology industry, the applications of drones are attracting more and more people to put their energy into it.

So we begin to study the application of gathering and detecting images. Based on the drone image dataset and yolov3 algorithm, we want to achieve the detection of vehicles and make improvements.

Our main research direction is the application of gathering and detecting images. Based on the drone image dataset and yolov3 algorithm, we want to achieve the detection of vehicles and make

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

improvements. According to the different characteristics of aerial images, the following difficulties are brought to the Drones object detection:

Due to the different object scales, the evaluation criteria are different during detection. For example, most of the time, small objects are the main object, but occasionally medium and large objects appear.

Since the drone is shot from the top view angle, the model needs to have higher accuracy in the top view angle.

Drone photography is now widely used in various fields. In the specific drone photography process, it is necessary to reasonably analyze the drone object detection and tracking methods. Using appropriate object detection and tracking methods, drone photography is effectively used to obtain relevant information. drone photography has a wide range of adaptability and is applied in different fields, which brings a lot of convenience. However, there are still some problems in drone technology. For example, the existing algorithms still have the problem of missing detection. When the detection object is in a dense background, or there are a large number of objects similar to the detection object in the background, the detection effect will not be very good. There is still much room for improvement in the current drone technology. However, with the increasing attention of drone object detection algorithm, the future drone object detection technology will also develop rapidly.

This paper studies the vehicle detection of drone image based on YOLOv3:

1) We take YOLOv3 as the basic algorithm for our research. However, some hyperparameters originally owned in YOLOv3 algorithm are not suitable for vehicle detection. We changed some hyperparameters to make the algorithm suitable for our research.

2) Through the training model, we found that the algorithm has good applicability to vehicle detection although it still has some improvement. The algorithm can better enable drone to carry out vehicle detection work.

3) In this paper, the hyperparameters of YOLOv3 algorithm are changed and we explain YOLOv3 based on data sets.

2. Related Work

2.1 Object Detection

The task of Object Detection is to find all the objects which could be a object in the image, and to determine their categories and positions, which is one of the core problems in the field of computer vision. Since various objects have different appearances, shapes and postures, coupled with the interference of factors such as illumination and occlusion during imaging, object detection has always been the most challenging problem in the field of computer vision.[1].

In the research of object detection, there are two series of models that are currently the most widely used.

1) R-CNN is the first-generation algorithm in the whole R-CNN series. It combines deep learning with traditional computer vision, uses selective search to extract region proposals, and uses SVM to achieve classification. [2][3].

As the R-CNN algorithm improving, Fast-RCNN appeared. It is an improvement based on R-CNN and SPPnets. Instead of calculating once for each candidate area, SPPnets' innovative point is to perfrom image feature extraction only once, and then map the feature map of the candidate area to the entire image feature map according to the algorithm. [3][4].

Ross B. Girshick proposed a new Faster-RCNN in 2016 after the accumulation of R-CNN and Fast-RCNN. The Faster-RCNN structurally integrates feature extraction, region proposal extraction, bbox regression, and classification into one network to achieve comprehensive performance, and it has been greatly improved, especially in terms of detection speed. [5].

2) YOLO(You Only Look Once) is another framework proposed by Ross Girshick for DL object detection speed after RCNN, fast-RCNN and faster-RCNN[6]. The Yolo algorithm solves the object

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

detection as a regression problem. It can directly predict the output of the object position bounding box and class probabilities in a single neural network from the original image. It is a single End-to-End network. But the previous object detection method first needs to generate a large number of priori boexs that may contain the object to be detected, which is more complicated. [7][8].

The YOLOv1 version is the pioneering work of the YOLO series. The core idea is to treat object detection as a single regression task. It first divides the image into SxS grids, and on which grid the center of the real frame of the object falls, the anchor frame corresponding to the grid is responsible for detecting the object.

And the YOLOv2 is optimized on the basis of YOLOv1, including the backbone network 448448 Darknet19, full convolutional network structure Conv+BatchNorm. And it using Kmeans to cluster Anchor on the COCO data set, which can achieve the better detection results. Furthermore, it also introduce muti-scale training to improve the generalization ability and detection effect of the network[9].

The YOLOv3 is further optimized on the points to be improved in YOLOv2, including the Backbone network Darknet53, the Muti-scale prediction, and the Cross-scale feature fusion. And it also collect 9 anchors of different scales on the COCO dataset, with 3 anchors for each sacle.[10].

As of today, YOLO has evolved to the v5 version, but the research on object detection is far from over.

2.2 Drones Object Detection

Based on computer vision and artificial intelligence, drones can accurately identify the name of the target and track it by shooting the target, which is the target detection technology of drones.

In the shooting of drones, due to the large image, the area occupied by the target in the image is small. The background is complex and the target is prone to rotation, and illumination, occlusion interference and the influence of drones camera jitter, which makes drones target detection very complex.

At present, there are many drones target detection technologies.

Intra-frame subtraction method mainly determines whether the target moves by analyzing the gray difference between pixels at the same position in two consecutive frames. The algorithm is simple and easy to implement, but it is only suitable for target detection under static background and single target.

Background subtraction method is to preset the background before shooting. Then, the target is obtained by making a difference between the detected image and the preset background image. This method is affected by the quality of the detection image and the preset background image.

Feature matching method mainly extracts the features of the target to be detected, and then establishes the target template. Then, in the detection image, the similarity matching is carried out between the detection feature map and the target template, so as to realize target detection. And it has good robustness.

Many technical difficulties will be encountered in the research of drones target detection. The following introduces these problems according to several characteristics of drones image.

1) The problem of complex background in drones images

There may be a large number of objects with similar targets in the background of drones image, which leads to the increase of missed detection or false alarm in detection, and it makes the size and number of features extracted in multi-scale cavity convolution very important.

2) The problem of small target in drones images

Due to the large image shooting range of drones, the detection target may be small. The resolution of small target in the picture is limited, which leads to difficulty in detection. The feature fusion method can combine multi-layer features to predict and improve the detection effect of small targets.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

At present, drones target detection algorithm can achieve good detection effect, but there is still much room for improvement. Methods are needed to improve accuracy. It need to increase the receptive field and intensively generate characteristics of different scales. It should also be possible to adaptively fuse characteristics and generate ROI.

3. Method

3.1 Backbone Network

In this project we use YOLOv3 as our basic model and we use Darknet-53 as our backbone network.

As shown in its name, Darknet-53 uses 53 convolution layers where the network uses consecutive 3x3 and 1x1 convolutional layers. It is a hybrid approach of Darknet-19, the network structure used in YOLOv2, FPN (feature pyramid) and Residual networks (ResNet). Being stacked with 53 more layers for the detection head, the YOLOv3 is a 106 layer fully convolutional underlying architecture as a whole. Therefore, the larger architecture, though making it slightly slower than YOLOv2, increase the accuracy significantly.

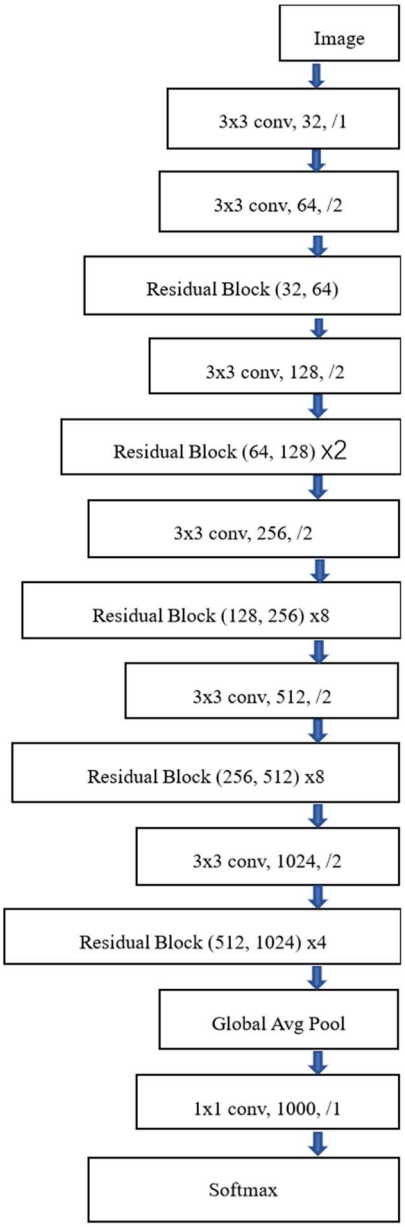

Fig 1. Structure of DarkNet53

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

The YOLOv3 network is designed to predict bounding boxes of each object associated with the probability of the class which the object belongs to. The model divides every input image into an $S \times S$ grid of cells and each grid predicts bounding boxes and class probabilities of the objects whose centers fall inside the grid cells.

The bounding box B is associated with the number of anchors it used. Each bounding box has $5 + C$ attributes, where 5 refers to the five bounding box attributes: center coordinates b_x and b_y, height b_h, width b_w and confidence score. The output format as $[S, S, B * (5 + C)]$ because we are working on an $S \times S$ image grid.

3.2 Improve

To better serve our project, we make some improvement.

First, we change the size of anchor boxes. The original size of anchor boxes is too large to fit in the objects UAVs shoots, so we reduce the size of anchor boxes. The mAP rate and recall rate increase as a result, but the precision rate decreases. That means our improvement can better serve for simple images which contains a few of vehicles and behave less satisfying when progressing more complex images with smaller and more vehicles.

Second, we delete the 32 times downsampled map for accelerating the progressing speed. As we are detecting objects through a UAV, we notice that the size of the objects is relatively small, so it might be inefficient to use the 32 times downsampled feature map. After doing so we find that the progressing speed rises indeed, but the precision rate, recall rate and mAP rate decrease without expectation. We check the training set to get the explanation that some pictures contain relatively big objects, and that causes the decrease of mAP rate, which we care the most.

Last but not least, we use GIoU to be the loss function of prediction box instead of MSE function. The benefit of GIoU is that GIoU has a better performance compared to MSE. This is inspired by Generalized Intersection over Union: A Metric and A Loss for Bounding Box [12]. We tested on COCO 2018 and the result shows improvement. Therefore, we believe that using GIoU can give us a relatively accurate result.

3.3 Loss Function

YOLOv3 predicts an objectness score for each bounding box using logistic regression. [11] Unlike YOLO, YOLOv3 change the way of calculating loss function. Since we use GIoU instead of MSE to calculate the loss for Center x, Center y, Width w and Height h of bounding box, the total loss function formular should change due to that.

The total loss function is:

$$Loss = L_{GIoU}\left(x_i^j, \hat{x}_i^j, y_i^j, \hat{y}_i^j\right) + L_{GIoU}\left(\sqrt{w_i^j}, \sqrt{\hat{w}_i^j}, \sqrt{h_i^j}, \sqrt{\hat{h}_i^j}\right) + \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} BCE(y_{\hat{c}_i^j}, p(C_i^j)) +$$
$$\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{noobj} BCE(y_{\hat{c}_i^j}, p(C_i^j)) + \sum_{i=0}^{S^2} I_{ij}^{obj} \sum_{c \in classes} BCE(y_{\hat{c}_i^j}, p(C_i^j)) \quad (1)$$

For different parts we use different loss functions. The mean two kind of loss functions are BCELoss (Binary Cross Entropy) and GIoU (Generalized Intersection over Union).

BCELoss.

The loss function is used for multi-label classification: $Loss = -(y * \ln(p) + (1 - y) * \ln(1 - p))$, where p is the probability $[0,1]$, and y is the class correct 1 or not 0.

GioU.

The GIoU algorithm works as follow:

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

0) input are predicted and ground truth bounding box coordinates B^p and B^g:

$$B^p = \left(x_1^p, y_1^p, x_2^p, y_2^p\right), B^g = \left(x_1^g, y_1^g, x_2^g, y_2^g\right) \tag{2}$$

1) For predicted box B^p, ensuring $x_2^p > x_1^p, y_2^p > y_1^p$, calculate for $\hat{x}_1^p, \hat{x}_2^p, \hat{y}_1^p, \hat{y}_2^p$:

$$\hat{x}_1^p = \min\left(x_1^p, x_2^p\right), \hat{x}_2^p = \max\left(x_1^p, x_2^p\right), \hat{y}_1^p = \min\left(y_1^p, y_2^p\right), \hat{y}_2^p = \max\left(y_1^p, y_2^p\right) \tag{3}$$

2) Calculate area of B^g and B^p:

$$A^g = \left(x_2^g - x_1^g\right) * \left(y_2^g - y_1^g\right), A^p = \left(\hat{x}_2^p - \hat{x}_1^p\right) * \left(\hat{y}_2^p - \hat{y}_1^p\right) \tag{4}$$

3) Calculate intersection I between B^g and B^p:

$$x_1^I = \max\left(\hat{x}_1^p, x_1^g\right), x_2^I = \min\left(\hat{x}_2^p, x_2^g\right), y_1^I = \max\left(\hat{y}_1^p, y_1^g\right), y_2^I = \min\left(\hat{y}_2^p, y_2^g\right) \tag{5}$$

$$I = \begin{cases} (x_2^I - x_1^I) * (y_2^I - y_1^I) & if\ x_2^I > x_1^I, y_2^I > y_1^I \\ 0 & otherwise \end{cases} \tag{6}$$

4) Find the coordinate of smallest enclosing box B^c and calculate area of B^c:

$$x_1^c = \min\left(\hat{x}_1^p, x_1^g\right), x_2^c = \max\left(\hat{x}_2^p, x_2^g\right), y_1^c = \min\left(\hat{y}_1^p, y_1^g\right), y_2^c = \max\left(\hat{y}_2^p, y_2^g\right) \tag{7}$$

$$A^c = (x_2^c - x_1^c) * (y_2^c - y_1^c) \tag{8}$$

5) Union and Intersection over Union

$$U = A^p + A^g - I, IoU = \frac{I}{U} \tag{9}$$

6) Geberalized Intersection over Union

$$GIoU = IoU - \frac{A^c - U}{A^c},\ L_{GIoU} = 1 - GIoU[12] \tag{10}$$

The loss function is composed by five parts. The first two parts are loss function for Center x, Center y, Width w and Height h of bounding box, and we use GIoU to calculate it. The third part is the loss function for objectness score prediction for bounding boxes with predicting objects, the fourth part is the loss function for no objectness score of a bounding box, and the last part is the loss function for multi-class predictions of a bounding box.

In YOLOv1 the last three parts are squared errors, but in YOLOv3 we replace them by BCE loss function. And as mentioned before, we use GIoU to replace the original MSE loss function. Therefore, the total loss function forms as above.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

4. Experiments

4.1 Implementation Details

In this experiment, we know the superiority of yolo algorithm in the field of deep learning based object detection, so we select YOLOv3 algorithm, which is based on YOLO algorithm and applies residual structure and feature pyramid structure. For sure, we use python, whose version is 3.9, to program and the algorithm

Taking it into consideration that the participants of our experiment use Windows system, so we use anaconda to configure the environment of our experiment and carry out deep learning development.

Considering the background of the experiment, to implement on UAVs to detect vehicles, we choose to detect 4 kinds of vehicles, and they are cars, vans, trucks and buses. The number of images of the training set is 6471.Two kinds of hyperparameters, which are strides of up-sample and anchors are changed to find out the best way to improve the algorithm. Also, the biggest feature map is tried to delete for the same reason.

4.2 Data Processing

We get the training set containing about six thousand pictures and the validation set containing about five thousand pictures from VisDrone-DET2018 dataset, which is provided by Computer vision Foundation[13]. Every picture in the training set has a txt file to store all the objects in the picture. We describe every object with an integer representing the class, four decimal numbers between 0 and 1 representing center point x, center point y, the width and the height.

To achieve the one-to-one correspondence between image and label file, we place the image and label file in the same path. To tell the program how to access the data we need, a txt file is necessary to store the path of every image.

Then a data file is needed to describe what data is needed and a names file is needed to store the classes we want to distinguish in our experiment.

Now we can start our training process. Because of limitation of GPU. To make our results more accurate, we will train five different models, so we need to modify the corresponding parameters in model.py to assure the normal operation of program.

4.3 Experimental results

Table 1. Results of different hyperparameters

	precision	recall	mAP@0.5	F1
YOLOv3 (32,16,8)	0.511	0.453	0.427	0.472
YOLOv3 (32,16,4)	0.493	0.452	0.442	0.469
YOLOv3_test_anchors_(32,16,8)	0.481	0.477	0.45	0.472
YOLOv3 VisDrone (32,16,4)	0.494	0.504	0.466	0.494
YOLOv3 VisDrone (32,8,4)	0.515	0.499	0.48	0.504

The first column represents the type of model. “YOLOv3” means the algorithm we used in this experiment. The line having “VisDrone” means the anchor boxes in these kinds of model is deliberately changed for this experiment. “test_anchors” means anchor boxes are output based on training set from darknet, which are not very fit for this experiment. The number following “YOLOv3” like “(32,16,8) represents the size of downsampled feature map. Those only have two numbers means we delete one feature map to try to improve our algorithm.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

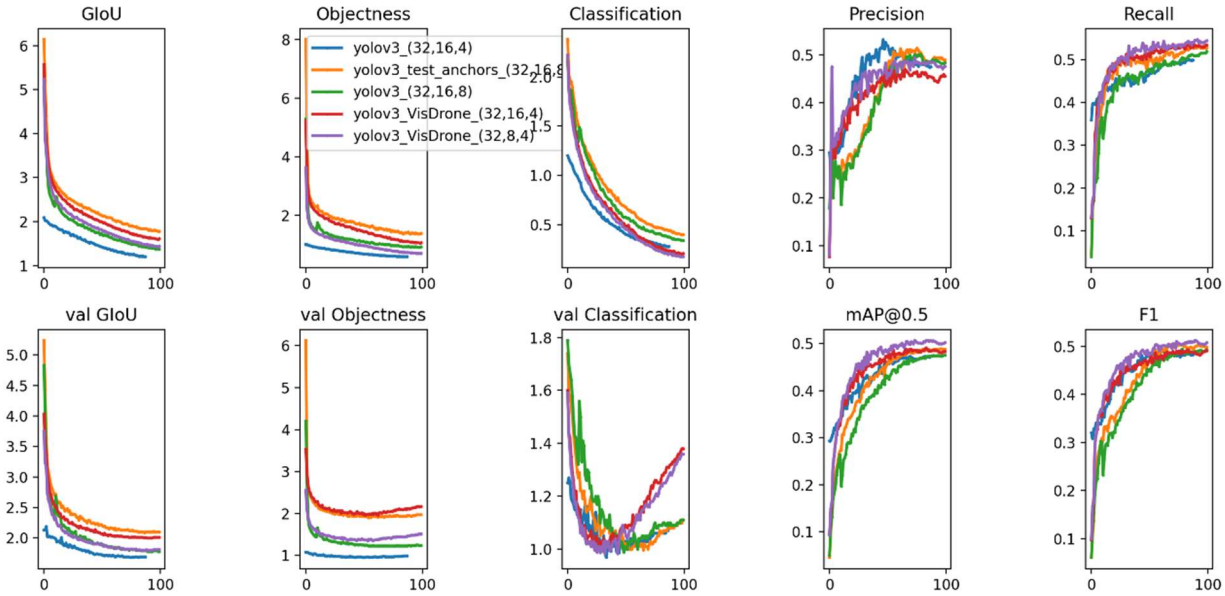

Fig 2. Comparsions_of_results

In regard of the number of outputs, we've tried to output only 2 feature maps, deleting the 32 times downsampled one, to accelerate the processing speed. The reason for doing this is that when we try to detect objects through a UAV, the size of them tends to be relatively small, so it's less likely to use the 32 times downsampled feature map. However, the truth is abandoning 32 times down can truly speed the process, while the precision rate, recall rate and mAP rate drop a little bit,about 5% to 6%, which can't afford the profits from acceleration of processing. So why did the mAP rate drop? After checking the training set, some images containing relatively big objects were found.

The left picture in the first row is the output with two feature maps, the picture on the right is the output with three feature maps and the same strides and anchor boxes of the left piture. Sometimes there actually are big objects.

With respect to strides of upsampling, we changed this hyperparameter due to the same reason for deleting the 32 times downsampled feature map. We found the mAP rate went up with the precision rate drops a little bit though.

Fig 3. Samples of different hyperparameters

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

The first picture in the second row above is the output with initial strides whose downsampling size is (32,16,8)and the second is with smaller strides whose downsampling size is (32,16,4). These two pictures clearly show us the reason for rising of the mAP rate and drop of the precision rate. Truly smaller objects can be detected, but some ambiguous cars can not be detected.

The last hyperparameter we alter is anchor boxes. The existing numbers for anchor boxes are actually not fit for the objects UAVs shoots. The size of anchor boxes is too big. The shape of vehicles tends to be a elongated rectangle or a rectangle of similar length and width when shot from directly behind. So we reduced the size of anchor boxes and used two sets of numbers. The first set is output from darknet applied to the training set. The mAP rate and recall rate have all risen by about 5%, but the precision rate has dropped by about 6%.

The same picture showed above can be accounted for the result of the change of anchor boxes. Compared with the first picture in the second row and the third one, the first picture above uses the initial anchor boxes and the one below uses the anchor boxes most fit for this training set and is with the same strides and feature maps. We can see ambiguous cars can now be detected again since the anchor boxes shrink in total. But this can make some information or feature lose so that it outputs the wrong type of vehicles.

In the end, we tried to combine the profits above. We avoided to deleting the biggest feature map, reduced the number of upsampling stride, and altered the size of anchor boxes with a set of more suitable numbers. The results look better. The mAP rate increased from 0.427 to 0.48. The recall rate increased from 0.453 to 0.499 and the precision rate from 0.511 to 0.515.

5. Conclusion

In order to understand the reasons of traffic jams by analysis of picture taken from drones, we come up with the idea about usage of vehicle detection based on computer vision from an angle of depression.

Having YOLOv3 as our fundamental algorithm, we change some hyperparameters to better serve our research since the original hyperparameters in YOLOv3 do not fit the needs of vehicle detection. We make some changes in the size of anchor boxes, delete the 32 times downsampled map which is inefficient in vehicle detection and use GIoU instead of the original MSE loss function for prediction box. We use mAP as our standard and we make each change due to increase mAP. After training and data analysis, we find that our new model fit the vehicle detection better and the mAP increases as we expect. Though there are still rooms for improvement, the whole model serves as our expectation and the results shows the great fitness for vehicle detection. We believe the small changes we make can help us better achieve our initial goal.

References

[1] Girshick R, Donahue J, Darrell T and Malik J 2014 Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation CVPR. IEEE.

[2] Bakircioglu, H. and E. Gelenbe. “Feature-based RNN target recognition.” Defense, Security, and Sensing (1998).

[3] Girshick, Ross B.. “Fast R-CNN.” 2015 IEEE International Conference on Computer Vision (ICCV) (2015): 1440-1448.

[4] Liu B, Zhao W, Sun Q. Study of object detection based on Faster R-CNN[C]//2017 Chinese Automation Congress (CAC). IEEE, 2017: 6233-6236.

[5] RenShaoqing et al. “Faster R-CNN.” IEEE Transactions on Pattern Analysis and Machine Intelligence (2017): n. pag.

[6] Redmon, Joseph et al. “You Only Look Once: Unified, Real-Time Object Detection.” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016): 779-788.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202202_8(2).0023

[7] Liu, L., Ouyang, W., Wang, X. et al. Deep Learning for Generic Object Detection: A Survey. Int J Comput Vis 128, 261–318 (2020).

[8] Jiao, L. et al. "A Survey of Deep Learning-Based Object Detection." IEEE Access 7 (2019): 128837-128868.

[9] Redmon, Joseph and Ali Farhadi. "YOLO9000: Better, Faster, Stronger." 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017): 6517-6525.

[10] Redmon, Joseph and Ali Farhadi. "YOLOv3: An Incremental Improvement." ArXiv abs/1804.02767 (2018): n. pag.

[11] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

[12] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, Silvio Savarese. Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression. arXiv: 1902. 09630, 2019.

[13] Pengfei zhu, Longyin Wen, Dawei Du. VisDrone-DET2018:The Vision Meets Drone Object Detection in Image Challenge Results.

[14] JIANG Bo, QU Ruokun, LI Yandong, LI Chenglong. Object detection in UAV imagery based on deep learning: Review[J]. ACTA AERONAUTICAET ASTRONAUTICA SINICA, 2021, 42(4): 524519- 524 519.

[15] Yan Ke and R. Sukthankar, "PCA-SIFT: a more distinctive representation for local image descriptors," Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004., 2004, pp. II-II, doi: 10.1109/CVPR.2004.1315206.

[16] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool, Speeded-Up Robust Features (SURF), Computer Vision and Image Understanding, Volume 110, Issue 3, 2008, Pages 346-359, ISSN 1077- 314 2.

[17] LIN Wen. (2010). Research on Novel Moving Face Detection Algorithm Based on Frame Difference. Computer Simulation(10), 238-241.

[18] WANG Guoqiang etc. "Video target detection algorithms based on background subtraction." Journal of Engineering of Heilongjiang University 005.004(2014):64-68.

[19] LI Zhonghai, WANG Li, CUI Jianguo. Weak aerial target tracking algorithm based on Camshift and Particle Filter[J]. Computer Engineering and Applications, 2011, 47(9):192-195.

[20] DENG Jihong, WEI Yuxing. Location of Object Based on Local Feature Descriptor[J]. Opto-Electronic Engineering, 2015, 42(1):58-64.

[21] LUO Yi, JIN Lizuo. Moving Object Detection and Tracking Based on Aerial Video[J]. Industrial Control Computer, 2017, 030(003):24-25,28.