An effective object detection algorithm for UAV-based urban regulation

Rui Qian

Nanjing University of Aeronautics and Astronautics, Jiangsu, 210016, China

qianrui@nuaa.edu.cn

Abstract. Target detection from the perspective of UAV has great potential in the field of urban regulation, limited by the dense small targets, severe environmental obstructions, camera shake, and changes in lighting conditions in the aerial view of drones, the existing object detection algorithms cannot effectively undertake this task. This paper introduces two lightweight feature extraction modules based on YOLOv5, which are C3-Faster with PConv and COT3 with transformer structure. Meanwhile, an extra small detection head is added to the output layer. These approaches enhance accuracy while maintaining the advantages of being lightweight and easy to deploy. The ablation experiments and comparative experiments are designed to verify the effectiveness of these modules. The algorithm presented in this paper can be deployed into embedded systems of small UAVs to assist UAVs in completing various regulatory tasks in complex urban scenarios.

Keywords: UAV, Urban regulation, Object Detection, YOLOv5.

1. Introduction

With the development of electronic communication technology, unmanned aerial vehicles (UAVs) have been rapidly promoted. In urban settings, UAV object detection technology is mainly used in traffic intelligence monitoring, urban security management, urban environmental supervision, and other aspects, making it an important tool for urban regulation.

However, compared to object detection tasks under general shooting perspectives, object detection tasks under the UAV perspective in urban scenes face the following challenges [1]: (1) Uneven and multiscale distribution of targets, with many small targets densely distributed. (2) Severe environmental occlusion and complex image backgrounds. (3) Restricted by the camera, the image light changes frequently, and some targets are seriously blurred. (4) UAVs require embedded computers for real-time synchronous processing of aerial image data to achieve automatic obstacle avoidance and task planning. Therefore, object detection tasks from the UAV perspective place higher demands on the accuracy and speed of algorithms.

Currently, deep learning-based object detection algorithms can mainly be divided into two categories: one category is the two-stage algorithms represented by R-CNN [2], SPP-NET, Fast-RCNN [3], Mask-RCNN, Cascade RCNN [4], Faster-RCNN [5], etc.Two-stage algorithms have high accuracy but large parameter and computational requirements, resulting in slower processing speeds and making deployment in industry challenging. The other category is the one-stage algorithms represented by the YOLO series [6], RefineNet, SSD, etc. The YOLO series has multiple versions that can meet real-time

© 2024 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

requirements. In complex and dynamic urban scenarios, the YOLO algorithm, with its fast processing speed and easy deployment, has a competitive edge.

Researchers have made many improvements based on one-stage algorithms for UAV aerial photography tasks. Zhu et al. [7]. proposed the Deformable Detection Transformer (DDETR) based on the transformer. However, due to the use of residual networks for feature extraction, lower-level feature information loss still hinders the ideal accuracy of small target detection. The TPH-YOLOv5 [8] model introduces a transformer prediction head and an attention model (CBAM) into YOLOv5 by adding a prediction head, achieving better results in detecting small objects in UAV images. Nevertheless, this model's use of the YOLO maximum parameter network, makes deployment much more challenging.

2. Improve YOLOv5 algorithm

The structure of the improved YOLOv5 object detection network is shown in Figure 1. This article introduces the Partial Convolution (PConv) into the YOLOv5's C3 module, forming a new lightweight feature extraction module called C3-Faster. The Contextual Transformer (CoT) Block is introduced into the neck network and combined with the C3 module to form the CoT3 module. An additional small object detection layer is introduced to the output layer, while the original large object detection layer is removed. The modified YOLOv5 network achieves a good lightweight effect while improving detection accuracy.

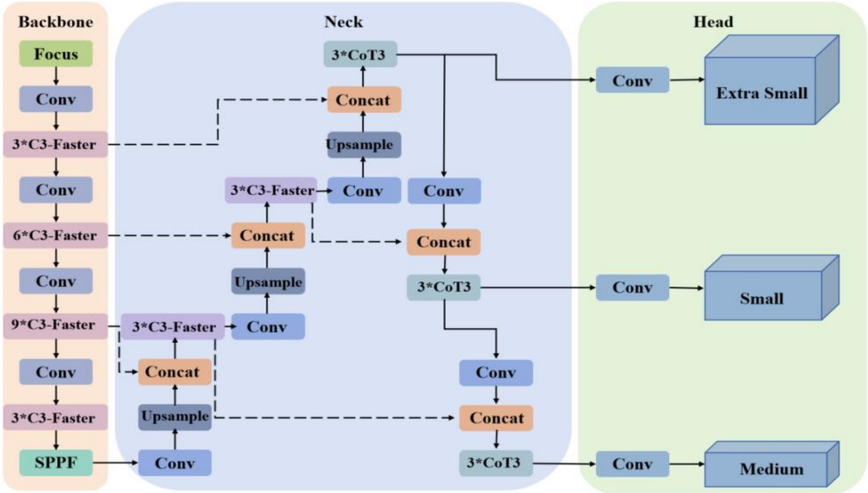


Figure 1. Improvement of YOLOv5 structure diagram.

2.1. *C3-Faster module with Partial Convolution*

The core of C3-Faster is the Partial Conv (PConv) from FasterNet [9], which has lower Floating Point Operations Per Second (FLOPs) compared to conventional Conv, allowing for better utilization of device computational capabilities. The internal structure of the BottleNeck in C3-Faster and a schematic diagram of PConv are shown in Figure 2.

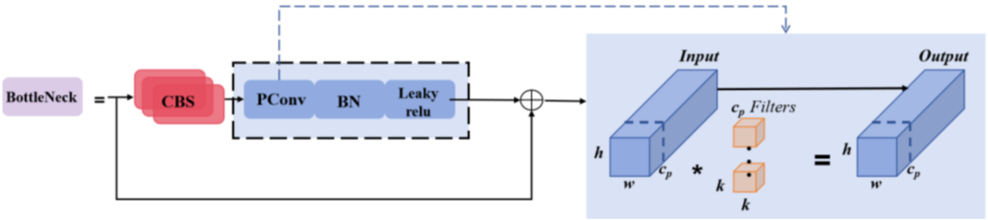


Figure 2. The structure of Bottleneck in C3-Faster

The working principle of PConv is to use only a portion of the input feature map channels for feature extraction, while keeping the rest of the channels unchanged. As shown in Figure 2, on a feature map with a total ofcchannels, a partial convolution kernel is used instead of the original 3×3 convolution to process the feature map with $\mathrm{c_p}$ channels, and then regular convolution is still used on the remaining channels. In the convolution operation with $\mathrm{c_p}$channels, the FLOPs of PConv can be represented as:

$$\mathrm{FLOPs} = h \times \mathrm{w} \times \mathrm{k}^2 \times \mathrm{c_p^2} \tag{1}$$

Where h and w represent the length and width of the feature graph, and k represents the size of the Pconv convolution kernel. Meanwhile, the amount of memory access of PConv can be expressed as:

$$h \times \mathrm{w} \times 2\mathrm{c_p} + \mathrm{k}^2 \times \mathrm{c_p^2} \approx h \times \mathrm{w} \times 2\mathrm{c_p} \tag{2}$$

The separation ratio is defined by the number of channels processed by PConv cp and the total number of channels c, if set $\mathrm{r} = \frac{1}{4}$, the FLOPs of PConv are only $\frac{1}{16}$ that of Conv, and the amount of memory access is also only $\frac{1}{4}$ that of Conv.

Under the processing method of PConv, the effective receptive field of the convolution operation is more like a T-shape, compared to regular convolution, this convolution method pays more attention to the center position. In the filter, the center position of the feature map is the most frequently accessed salient position, so the center position accounts for a larger proportion, which demonstrates the rationality of the partial convolution operation.

2.2. CoT3 module with Contextual Transformer Block

Many Vision Transformer (ViT) models with global self-attention mechanisms have demonstrated good performance in object detection tasks, and existing work also indicates that Transformer detection heads can achieve better accuracy and lightweight effects. This paper integrates the Vision Transformer module into C3, allowing for the full utilization of contextual information between input keys to guide the learning of dynamic attention matrices, thereby enhancing the ability to transmit semantic information.

On the input feature map, the CoT block [10] first encodes the input keys through contextual convolution to obtain static contextual representations. Then, for $\mathrm{k} \times \mathrm{k} \times \mathrm{C_h}$ convolution, the dynamic output of the CoT multi-head self-attention mechanism module can be represented as:

$$Y = V \odot Softmax(K \odot Q + P \odot Q) \tag{3}$$

where Q, K, and V represent keys, queries, and values, which are parameters in the standard self-attention mechanism, and P represents any position in the 2D convolutional kernel. The fusion of static and dynamic contextual representations yields the final output.

The CoT3 module, which incorporates the Transformer architecture, enables parallel computation and weight sharing in the self-attention mechanism. This approach not only captures dependencies in long-distance sequences and enhances the transmission of semantic information but also improves computational efficiency. Additionally, by leveraging contextual information more effectively, the module can help the algorithm better identify the occluded targets and enhance the model's generalization capability.

2.3. Improved muti-scale detection

Researchers have found that adding an additional smaller-scale detection layer and expanding the YOLO model to a four-scale detection system for small object tasks in drone-captured scenes can result in significant improvements.

This paper introduces an additional small object detection layer to focus on smaller objects, while adding its associated feature fusion layer to the neck network. The extra detection scale enhances the detection capability for tiny objects, provides relatively rich position information, and is more suitable for aerial scenes with a wide range of object sizes.

To balance the additional computational load introduced by the small object detection layer, this paper removes the original large object detection head, as large objects are rarely present in aerial photography. Additionally, certain structures in the neck network are removed to improve computational speed.

3. Research Design

To validate the improved performance of YOLOv5 in urban regulation scenarios, this paper conducted ablation experiments on the Visdrone dataset and compared it with some existing mature object detection algorithms.

3.1. Visdrone Dataset

The Visdrone dataset was collected by the team from Tianjin University. It contains 10,209 static images from 288 video clips, with 6,471 images in the training set, 548 images in the validation set, and 3,190 images in the test set. The aerial drone videos from different times, angles, and scenes in the Visdrone dataset can effectively support the training, validation, and testing tasks of the algorithm in this paper.

3.2. Performance Metrics

Performance metrics involved in this paper's algorithm include parameter count, FLOPs, average precision (mAP), and frame rate (fps). Precision metrics are obtained by integrating precision and recall, and mAP is the average accuracy across multiple classes. Two commonly used average precision metrics are AP50(IOU = 0.5) and AP50:95(IOU ∈ [0.5 ∶ 0.95]).

4. The improved YOLOv5 algorithm's performance and analysis in urban regulation scenarios

In the urban regulation scenario, this paper designs an ablation experiment and a comparison experiment to verify the performance of each module for the problems of dense small targets, multi-scale targets, occlusions from the perspective of drones, and model light weighting.

4.1. Ablation Experiments

On the Visdrone dataset, the role of each improvement module in this paper is reflected in the ablation experiments, and the results of the ablation experiments are shown in Table 1.

Table 1. Performance of each improved module in the ablation experiment

Models	**Param. (M)**	**FLOPs(G)**	**AP$_{50}$(%)**	**AP$_{50:95}$(%)**	**Fps(f·s^{-1})**
YOLOv5	7.2	16.5	32.2	17.3	84.0
YOLOv5+C3-Faster	6.5	14.4	32.6	17.5	92.3
YOLOv5+C3-Faster+CoT3	6.4	14.1	32.7	17.6	98.6
YOLOv5+Head	7.8	28.9	38.6	20.2	66.2
Improved YOLOv5	6.9	24.0	39.0	20.5	81.7

From Table 1, it can be seen that the design of the C3-Faster and CoT3 modules mainly aims to improve the detection efficiency of the neural network. The lightweighting effects of these two modules are quite significant. Compared to the original YOLOv5 model, the C3-Faster module reduces the algorithm's parameter count by 0.7M and FLOPs by 2.1G. Additionally, this module improves the accuracy parameters AP50 and AP50:95 by 0.4% and 0.2%, respectively. The inclusion of the CoT3 module further reduces the model size, decreasing the parameter count by 0.1M and FLOPs by 0.3G. The introduction of a global self-attention mechanism also leads to a 0.1% improvement in accuracy metrics. After integrating the C3-Faster and CoT3 modules, the computational speed increased by 14.6 frames per second.

Due to the abundance and density of small targets in the Visdrone dataset, the addition of an extra small object detection head significantly improves the precision of the object detection network. It results in a notable increase of 6.4% and 2.9% in AP50 and AP50:95, respectively.

The final model achieves a good balance between model size and detection accuracy. Compared to the original YOLOv5 model, the improved model shows a substantial increase in accuracy, with AP50 and AP50:95 improving by 6.8% and 3.2%, respectively. The model size decreases by 0.3M parameters, while FLOPs increase by 7.5G. The processing speed slows down by 2.3 frames per second, maintaining a relatively equivalent trade-off between model size and processing speed.

4.2. Comparative experimental results

In this paper, some mature object detection algorithms have been selected from the Visdrone dataset, including RetinaNet, Faster-RCNN, YOLOv4, YOLOv8 and so on, so as to verify the superiority of the algorithm in this paper.

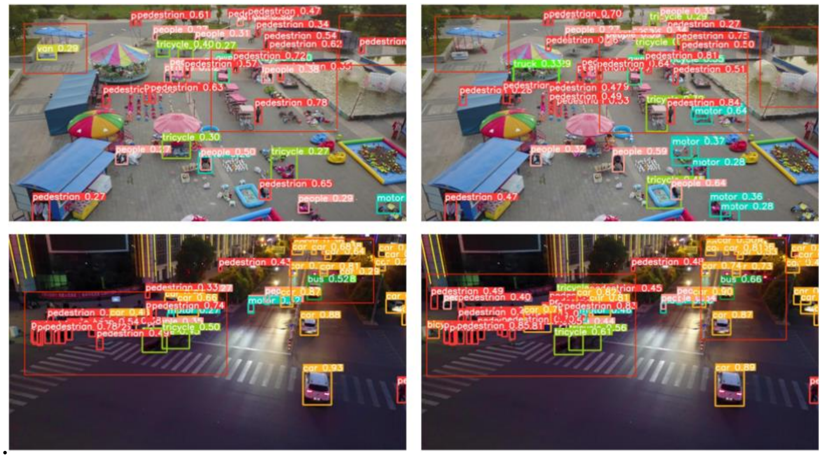
Table 2. Performance of the representative object detection algorithms on the Visdrone

Models	**AP_{50}(%)**	**Fps($f\cdot s^{-1}$)**
RetinaNet	22.5	33.4
SSD512	26.8	37.2
Faster-RCNN	29.6	36.3
YOLOv8-S	34.8	62.4
Grid-RCNN	39.3	10.4
Ours	39.0	81.7

As can be seen from Table 2, YOLO series algorithms have achieved a good balance in terms of accuracy and processing speed, but the algorithm in this paper is more dominant than other YOLO series algorithms in terms of accuracy and speed. Some algorithms, such as Grid-RCNN, have higher accuracy, but the model is too large and the processing speed is slow, and they cannot undertake the task of city regulation with a complex and varied background. The improved YOLOv5 algorithm has better detection effects while fully guaranteeing real-time performance.

4.3. Detection results of different algorithms

The detection results of the proposed object detection algorithm in two typical urban scenarios are shown in Figure 3. As can be seen from Figure 3, in the complex urban scenario, the original YOLOv5 algorithm has many problems of false detection and missed detection, but the algorithm used in this paper significantly reduced the proportion of false detection and detected the target that could not be detected originally.



(a) The original YOLOv5 algorithm **(b)** The improved YOLOv5 algorithm

Figure 3. Comparison of the effect of the improved YOLOv5 algorithm and the original algorithm

5. Conclusion

UAV aerial photography technology has great potential in the field of urban supervision, but the existing object detection technology is still unable to deal with problems such as small and dense targets, multi-scale targets and environmental occlusion problems from the perspective of UAV. Due to the limitations of the UAV carrier itself, the optimal object detection algorithm should not only have high detection speed and detection accuracy, but also be easy to deploy. Based on the YOLOv5 network, this paper integrates PConv into the new feature extraction module C3-Faster. To make feature extraction faster and more efficient, a transformer architecture is introduced into the C3 module of the neck network to give the algorithm a global receptive field. This paper then modifies the original network's three-scale detection head to focus more on small targets in object detection. In the ablation experiments, the algorithm proposed in this article outperforms the original YOLOv5 on two accuracy metrics AP50 and AP50:95, with a minimal increase in model size, while maintaining processing speed comparable to the original model. In comparative experiments, this algorithm outperforms many current state-of-the-art object detection algorithms in both accuracy and speed. What can be seen from the experiments is that the structure not only achieves a good balance in precision and speed, but also has a light model size, which can cope with complex scenes in different urban environments. In practice, the algorithm can be deployed on the embedded systems of small UAVs to assist in urban supervision tasks such as traffic control and public security management.

References

[1] Wu X., Li W., Hong D. F., et al. Deep learning for UAV-based object detection and tracking: a survey [J]. IEEE Journal of Earth Science and Remote Sensing, 2022, 10 (1): 91-124.

[2] Girshick R., Donahue J., Darrell T., et al. Rich Feature Hierarchy for Accurate Object Detection and Semantic Segmentation [C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2014: 580-587.

[3] Girshick R. Fast R-CNN [C]// 2015 IEEE International Conference on Computer Vision (ICCV). Piscaaway: IEEE Press, 2016: 1440-1448.

[4] Ren S. Q., He K. M., Girshick R., et al. Faster RCNN: Towards real-time object detection with Region Proposal Networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39 (6): 1137-1149.

[5] Cai Z. W., Vasconcelos N. Cascade R-CNN: delving into the high-quality target detection [C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2018: 6154-6162.

[6] Redmon J., Divvala S., Girshick R., et al. You only look once: Unified, real-time object detection[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2016: 779-788.

[7] Zhu X., Su W., Lu L., et al. Deformable detr: Deformable transformers for end-to-end object detection [J]. arXiv preprint arXiv:2010.04159, 2020.

[8] Zhu X., Lyu S., Wang X., et al. TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios [C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 2778-2788.

[9] Chen J., Kao S., He H., et al. Run, Don't walk: Chasing higher FLOPS for faster neural networks[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 12021-12031.

[10] Li Y., Yao T., Pan Y., et al. Contextual transformer networks for visual recognition [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 45(2): 1489-1500.