drones

MDPI

Article

Improved YOLOv7 Target Detection Algorithm Based on UAV Aerial Photography

Zhen Bai [1,2], Xinbiao Pei [1,2], Zheng Qiao [1,2], Guangxin Wu [1,2] and Yue Bai [1,2,*]

[1] Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China; baizhen21@mails.ucas.ac.cn (Z.B.); peixinbiao@ciomp.ac.cn (X.P.); qiaozheng19@mails.ucas.ac.cn (Z.Q.); wuguangxin20@mails.ucas.ac.cn (G.W.)
[2] University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: baiy@ciomp.ac.cn

Abstract: With the rapid development of remote sensing technology, remote sensing target detection faces many problems; for example, there is still no good solution for small targets with complex backgrounds and simple features. In response to the above, we have added dynamic snake convolution (DSC) to YOLOv7. In addition, SPPFCSPC is used instead of the original spatial pyramid pooling structure; the original loss function was replaced with the EIoU loss function. This study was evaluated on UAV image data (VisDrone2019), which were compared with mainstream algorithms, and the experiments showed that this algorithm has a good average accuracy. Compared to the original algorithm, the mAP0.5 of the present algorithm is improved by 4.3%. Experiments proved that this algorithm outperforms other algorithms.

Keywords: UAV; object detection; YOLOv7; dynamic serpentine convolution; SPPF

Citation: Bai, Z.; Pei, X.; Qiao, Z.; Wu, G.; Bai, Y. Improved YOLOv7 Target Detection Algorithm Based on UAV Aerial Photography. *Drones* **2024**, *8*, 104. https://doi.org/10.3390/drones8030104

Academic Editor: Abdessattar Abdelkefi

Received: 1 January 2024
Revised: 12 March 2024
Accepted: 17 March 2024
Published: 19 March 2024

1. Introduction

In recent years, the application of UAV remote sensing in various fields has become more and more extensive. For example, UAV remote sensing has outstanding performance in scenarios such as battlefield inspection, disaster rescue, environmental survey, electric power overhaul, and monitoring and inspection. Through the use of drones, the efficiency of accomplishing tasks has been greatly improved. Remote sensing images have a significant improvement in the resolution and accuracy of remote sensing images compared to traditional satellite remote sensing and other means, but they still have not solved the problems of having a long distance from the target, shooting a small target, serious occlusion, and weak recognizable features. In addition, because of the limited load of the UAV, it is difficult for the airborne edge computing platform to meet the arithmetic demand of common deep learning algorithms, which also poses a problem for applications.

The reason for the difficulty of target detection under the UAV perspective is that UAV images have scale changes, sparse and dense distribution, and a higher proportion of small targets, especially the contradiction between the high computational demand of UAV high-resolution images and the limited arithmetic power of the current stage of low-power chips is difficult to balance. Compared with the natural images taken from the ground viewpoint, the wide field of view from the UAV viewpoint provides richer visualization information but also implies more complex scenes and more diverse targets, bringing more useless noise interference to the target detection. Moreover, in the sky view, targets in the image are often more difficult to detect due to factors such as remote shooting, background occlusion, or the influence of lighting; therefore, it is necessary to use high-resolution images. This greatly increases the computational overhead and memory requirements of target detection algorithms, and the direct use of general-purpose target detection algorithms that have not been specially designed will bring unbearable computational overhead and memory requirements, further exacerbating the difficulty

of target detection. In real-world application scenarios, which are often faced with fine-grained classification problems similar to identifying vehicle types, these similar targets pose a huge challenge for the model to recognize the target correctly.

Traditional target detection consists of feature extraction, classifier, and region selection. A candidate region is first searched in the image to be detected; then, features are extracted and classified. Since the target may appear at any position in the image, and its aspect ratio and size cannot be determined beforehand, it is necessary to set a sliding window with different scales to traverse the image to be detected. This strategy can determine the location of possible targets, but it has the problems of high time complexity, redundant windows, and poor region matching, which seriously affect the speed and effect of subsequent feature extraction. In fact, affected by the time complexity problem, and for targets with large floating aspect ratios, it is difficult to obtain matching feature regions even if the whole image is traversed. In the feature-extraction stage, features such as local binary patterns, scale-invariant feature transforms, and directional gradient histograms are often used. Because of the uncertainty of the target morphology, the diversity of lighting changes, and the complexity of the target background, it is very difficult to make the features robust. In summary, the effect of traditional detection methods is unstable, easily affected by a variety of conditions, and difficult to put into practical use.

As technology continues to advance and mature, visual target detection plays a key role in practical applications. In recent years, numerous related tech unicorn companies, such as Shangtang Technology and Kuangwei Technology, have emerged in the industry. Meanwhile, computer vision has become crucial in the field of autonomous driving, and some tech companies, especially representative companies such as Tesla, serve as representatives of visual perception leading the development of autonomous driving. Despite many advances in UAV visual inspection, it still faces many challenges. Mainly because, for one thing, aerial images are different from images of natural scenes, making it difficult to identify targets accurately. Second, the human–machine target-detection task has high requirements for real-time and accuracy.

To solve the above problems, this algorithm improves YOLOv7. This algorithm was experimented on VisDrone-2019 on a public dataset, proving the algorithm has high detection accuracy. First, the improved algorithm incorporates dynamic snake convolution (DSC) in Backbone, which significantly improves the model-detection accuracy. Secondly, an improved SPPF instead of the original spatial pyramid pooling structure is used. At last, the original loss function was replaced with EIOU.

2. Related Work

2.1. Targeted Detection

Target detection, as a relevant application of computer vision, directly affects the performance of these vision tasks and applications. Therefore, target detection techniques have received focused attention from various industries and fields. In academia, target detection is a key area of interest in computer vision journals, and many papers on target detection are published yearly. According to Google Scholar, more than 15,000 papers on target detection have been published in the past decade. In industry, many technology companies, such as Google, Shangtang and Kuangyi, Facebook, Huawei, and Baidu, have invested money and R&D staff in research. In the government, target detection is considered one of the key technologies for artificial intelligence, and countries are actively competing and developing this field.

At the very beginning, target detection algorithms generally used hand-designed features combined with shallow classifiers, like AdaBoost [1]. At this stage, a series of classical feature descriptors for target detection emerged, such as Haar features and histogram of orientation gradient features. Since 2012, deep learning techniques have developed rapidly, computational resources have improved, and large-scale datasets and evaluation criteria have emerged and become publicly available. A series of classic research efforts have emerged, including regional convolutional neural networks [2], SSD [3], YOLO [4],

and DETR [5]. Compared with the traditional method of manually designing features, it solves the tedious problem of the manual design process and can automatically learn the features of interest. Meanwhile, deep learning-based methods integrate classifier learning and feature extraction in a single framework. These innovations drive rapid development and progress.

The single-stage target detection algorithm divides the image into a number of cells, each of which determines whether it contains an object and the object's category and location, such as the YOLO algorithm and the SSD algorithm [3]. The two-step target detection method divides the task into two steps. Firstly, some proposal frames that potentially contain targets are generated in the first step, followed by the classification and location localization of these proposal frames in the second step before finally determining whether a target exists in the image or not. Examples are Faster R-CNN algorithms [6]. Two-stage target detection algorithms are less real-time, more accurate, and have excellent detection performance on many datasets.

2.2. Unmanned Aerial Vehicle Target Detection

The UAV perspective brings problems to multi-target detection, such as increasing the number of small targets, insufficient features contained in single-dimensional information, low detection efficiency due to sparse and inhomogeneous distribution of target categories, interference in target detection, target omission and misdetection due to scale changes, slow inference speed, etc. This section describes the improvements proposed by scholars for the above problems from the two perspectives.

For multi-target detection under the UAV perspective, the single-stage detectors YOLO series and SSD are widely used due to their clear advantages. Numerous scholars have addressed the problem of viewpoint algorithms from the UAV perspective. (1) For the situation where there are many small targets in the sky view, Liu Res Unit_2 is added to the Backbone network and ResNet unit in YOLO, and two ResNet units are merged in the Resblock of Darknet, and at the same time, the probability due to the restricted sensing field is reduced, so that the problem of small target omission due to the limited sensing field is solved [7]. (2) Saetchnikov et al. proposed the YOLOv4 eff network, which uses four sets of cross-stage partial for connecting the Backbone network and neck network, and uses the Swish function as the activation function, and letter-box is set to 1 to maintain the efficiency of use. The letter-box is set to 1 to maintain the utilization efficiency [8]. (3) In order to solve the problem of target misdetection in UAV aerial images due to scale variations, Li et al. proposed an SSD algorithm combining an attention mechanism with extended convolution, using extended convolution to replace the original, and combining low-order feature maps of small targets with higher-order feature maps [9].

Two-stage target detection algorithms are different from single-stage ones. The direct migration of the algorithm from the conventional perspective to the UAV aerial video is less effective and needs to be optimized according to the target characteristics of the UAV aerial video. The main improvements are summarized as follows: (1) Avola proposed a multi-stream structure for multi-scale image experiments in order to cope with the sky environment with many small targets. Using structure as the Backbone of the Fast R-CNN network, an MS-Faster R-CNN target detector was designed to consistently and stably detect targets in the UAV video sequence. Stadler used a Cascade R-CNN network as the target detector, halved the size of the default anchor frame to account for smaller targets, and doubled the number of predicted targets [10]. (2) To address the problem of insufficient single-dimensional information inclusion features, Azimi et al. used a Siamese network to extract visual features and work with graph convolutional neural networks and LSTM to incorporate the appearance, graphical, and temporal of targets [11]. (3) Coping with the slowness brought about by the dispersion of targets in the sky environment, Yang added the clustering idea to target detection and proposed the ClusDet network, which firstly generates target cluster regions using the clustering network CPNet, estimates the target proportion in these regions using the Sca-leNet network and then feeds the clustered

regions into the DetecNet network to perform target detection, which reduces the detection computation. Finally, the clustered areas are fed into the DetecNet network for target detection [12].

3. Principles and Improvements

3.1. YOLOv7

YOLOv7 belongs to the single-stage target detection algorithm, which is one of the most advanced algorithms and exceeds the previous YOLO algorithms. The YOLOv7 network comprises four main components: Input, Backbone, Neck, and Head. Four parts are constructed [13].

The Backbone consists of Conv2D_BN_SiLu(CBS), a high-efficiency layer aggregation network (ELAN), a maximum-pooling layer (MP), and SPPCSPC. The convolution part comprises Conv2D, batch normalization (BN), and Selu, which combines Sigmoid and linear rectification functions to extract image features of different sizes. The maximum-pooling layer (MP) is divided into two branches. The left branch is downsampled through maximum pooling first and then undergoes a convolutional part to reduce the number of channels; the right branch passes through a 1×1 convolutional part, then immediately connects with a 3×3 convolutional part for downsampling, and the left and right branches are stacked to enhance target-specific extraction. The high-efficiency layer aggregation network (ELAN) uses a stack of four branches, each with a different number of convolution parts (Conv2D_BN_SiLu), which corresponds to a denser residual structure, making it easier to optimize and mitigate the problem of gradient explosion that is inherent in neural nets that increase in network depth.

The Neck part uses the path-aggregation network (PA-Net) structure to extract three feature layers, middle, middle-lower, and bottom, with different network depths on the Backbone and performs a number of convolutional, maximal up-sampling, maximal down-sampling, and high-efficiency layer aggregation network operations on them to achieve the enhancement of feature information.

The Head part performs a RepConv operation on each of the three reinforced feature layers obtained from the Neck part, which is used to introduce a special residual structure to achieve the effect that the network prediction performance does not decrease but the network complexity decreases, which in turn is then passed into YoloHead to complete the prediction of the categories and the anchoring of the target bounding box. Figure 1 below shows the overall network structure of YOLOv7.

3.2. Improvement

3.2.1. Dynamic Snake Convolution

UAV remote sensing contexts are characterized by the presence of many elongated and convoluted tubular strong structures, thin and weak local structures, and fickle and complicated global patterns. The commonly used standard convolution kernel aims to extract some of the features, such as local features. On top of that, deformable convolution enriches its application by adapting to the geometric deformation of different objects. Nonetheless, focusing on slender and curvy tubular structures is difficult because of the previous challenges. Therefore, we incorporate the new framework, DSCNet, which includes a tubular-aware dynamic snake convolution kernel and a multi-angle feature integration scheme, and not only that but also a topological continuity constrained loss function, and in the following, we discuss the derivation of the formulas for dynamic snake convolution [14].

First, given the standard convolutional coordinates K, the center coordinates are given. Then, it is expressed as:

$$K = \{(x-1, y-1), (x-1, y), \cdots, (x+1, y+1)\} \quad (1)$$

In order to allow it to keep focusing on the geometrically complex features of the target, the deformation offset Δ is added. Assuming that the above model is completely free to learn offsets, the offsets tend to be unconstrained and especially ripe to deviate from the target when dealing with elongated structures. Therefore, an iterative strategy is adopted to sequentially make predictions about the next position of the target to be dealt with, thus ensuring the continuity of dealing with the target and not having relatively large offsets.

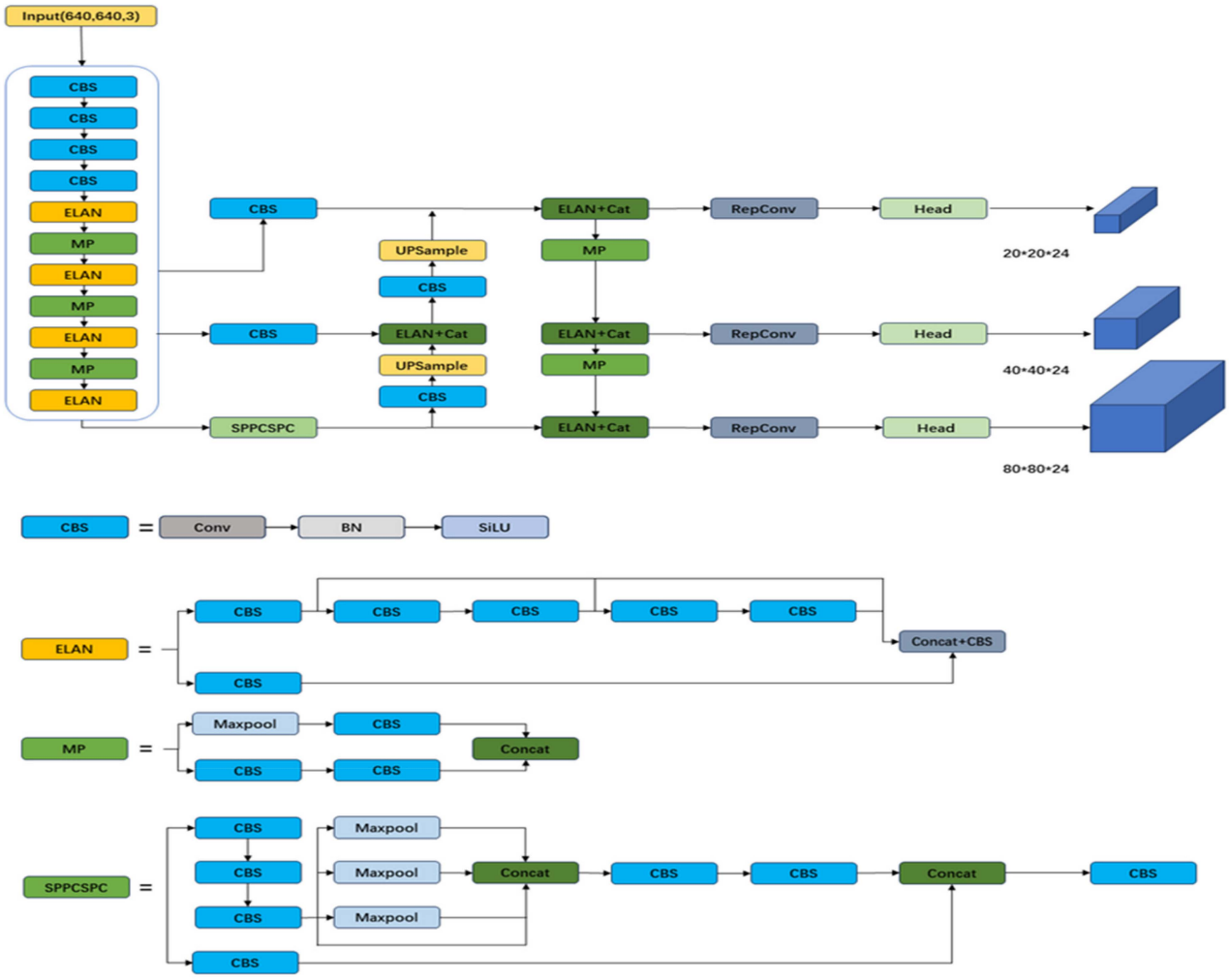

Figure 1. YOLOv7 network structure diagram.

Second, the convolution kernel is linearized not only in the x-axis but also in the y-axis. The selection of each position in the convolution kernel K has a superposition effect. Starting from the starting position K_i, the positions of the meshes far from the center often depend on the position of the previous mesh K_{i+1}, adding an offset $\Delta = \{\delta|\delta \in [-1, 1]\}$ with respect to K Thus, the offsets accumulate Σ and are used so that the convolution kernel does not violate the linear morphological structure.

As shown in Figure 2, the change in the x-axis direction is

$$K_{i\pm c} = \begin{cases} (x_{i+c}, y_{i+c}) = (x_i + c, y_i + \sum_i^{i+c} \Delta y), \\ (x_{i-c}, y_{i-c}) = (x_i - c, y_i + \sum_{i-c}^{i} \Delta y), \end{cases} \tag{2}$$

The change in the y-axis direction is

$$K_{j\pm c} = \begin{cases} (x_{j+c}, y_{j+c}) = (x_j + \sum_j^{j+c} \Delta x, y_j + c), \\ (x_{j-c}, y_{j-c}) = (x_j + \sum_{j-c}^{j} \Delta x, y_j - c), \end{cases} \tag{3}$$

Because offsets are essentially small numbers, but coordinates tend to be integers, bilinear interpolation is used as follows:

$$K = \sum_{K'} B(K', K) \cdot K' \tag{4}$$

In the above equation, K denotes the decimal positions of the convolution kernel on the coordinate axis, and then all positions in the integer space, and B is a bilinear interpolation kernel and can thus be expressed as a product of one-dimensional convolution kernels:

$$B(K, K') = b(K_x, K'_x) \cdot b(K_y, K'_y) \tag{5}$$

Unlike deformable convolution in Figure 3, deformable convolution gives the network complete freedom to learn geometric variations, thus leading to perceptual area roaming, especially on fine tubular structures [15].

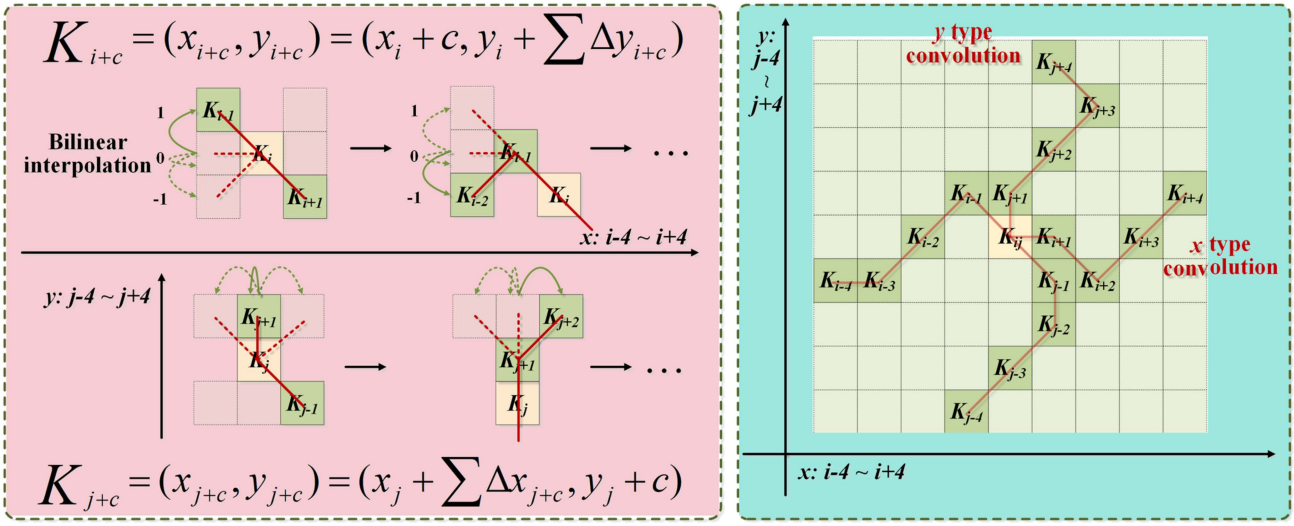

Figure 2. Schematic diagram of the dynamic serpentine convolution kernel coordinates computation and optional receptive fields.

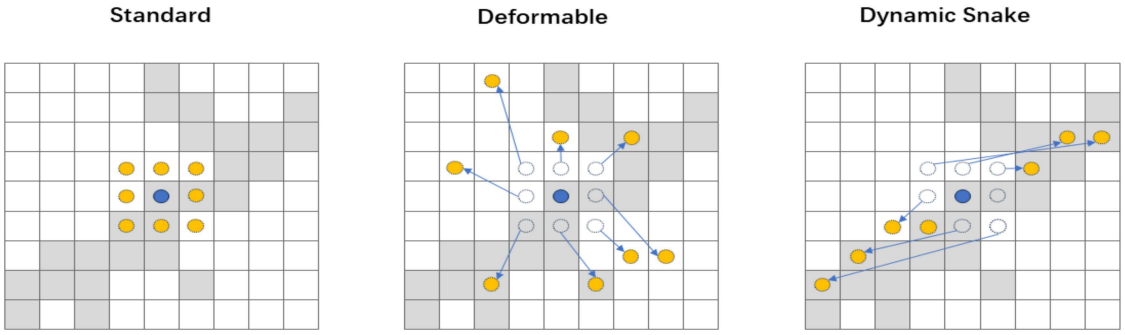

Figure 3. Feeling fields for standard convolution, variability convolution, and dynamic serpentine convolution.

DSConv focuses on the curvilinear properties of pipe shapes and specifically enhances the perception of pipe morphology through qualification-assisted adaptive learning. Cross-viewpoint attribute fusion techniques are adopted when faced with the challenge of unstable holomorphology. The program formulates a diverse morphological core model through DSConv, interrogates the structural attributes of the target in multiple dimensions, and aggregates key elements to achieve effective attribute integration. The increase in feature aggregation may cause the network processing to rise and trigger data redundancy, so the hierarchical and arbitrary exclusion method is implemented in the attribute integra-

tion training phase to reduce the network operation load and avoid model over-matching. A topological coherence loss function (TCLoss) based on persistent homotopy (PH) is proposed to address the problem of pipeline structural cuts tending to be disconnected. PH tracks the formation to dissipation of topological features and refines important topological details in cluttered data.

3.2.2. Improvement of SPPCSPC

PANET uses SPPCSPC in the bottom layer for the Backbone part, which consists of spatial pyramid pooling (SPP) and cross-stage partial (CSP) pyramid junctions, which are divided into 1×1, 5×5, 9×9, and 13×13-sized convolutional kernels for the Maxpool operation for distinguishing different targets, increasing the receptive field, and extracting important feature information, the performance of which is due to the SPPF module proposed via yolov5, but has a greater impact on the speed of network inference [16]. So, this study refers to the idea of SPPF simplifying its design, which greatly improves the inference efficiency with little impact on detection accuracy. Therefore, in Figure 4, this study refers to the network model of SPPF to make the following improvements to SPPCSPC: three convolution kernels of 5×5, 9×9, and 13×13 sizes are made to perform serial Maxpool operations to simplify the structure of the model and collect the target data at each scale [17].

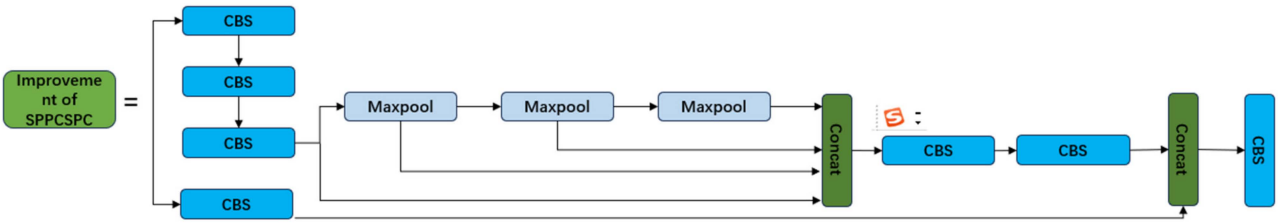

Figure 4. Improved SPP network.

3.2.3. Improvement of IOU LoSS

The original YOLOv7 used CIOU as the coordinate loss regression function [18]. The original loss function combines three geometric elements: overlap region, centroid interval, and aspect ratio in Figure 5. The loss function is defined as such between the predicted and actual frames.

$$L_{CIOU} = 1 - IOU + \frac{p^2(b, b^{gt})}{c^2} + \alpha v \tag{6}$$

where b and b^{gt} denote the centers of B and B^{gt}, respectively, $p(\cdot) = \left|\left|b - b^{st}\right|\right|_2$ denotes the Euclidean distance, and c is the minimum enclosing diagonal length that covers both boxes. $v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}\right)^2$ and $\alpha = \frac{v}{(1-IOU)+v}$ measure the difference in aspect ratios.

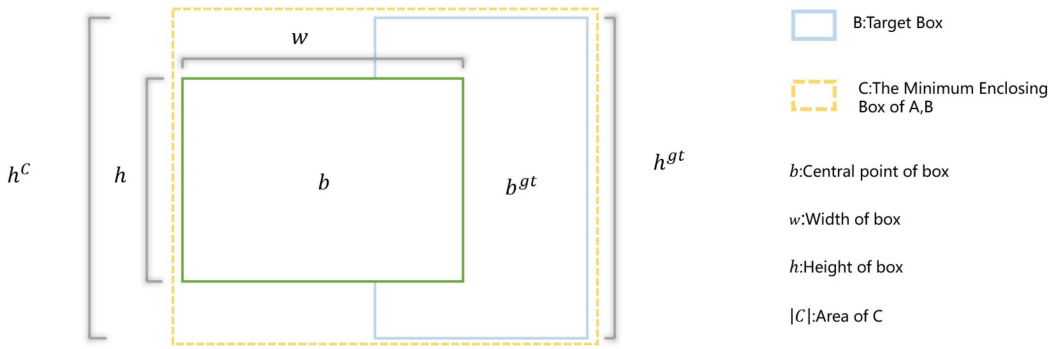

Figure 5. Loss function.

Compared with the earlier loss function, the CIOU loss is significantly enhanced in terms of convergence efficiency and recognition accuracy. However, it remains to be further clarified to define and optimize the last term, which, on the one hand, reduces the convergence speed of CIOU. (1) v reflects only the difference in aspect ratio, not the real relationship between w and w_{gt} or h and h_{gt}; that is, with all the bounding boxes with the nature of $\{(w = kw^{gt}, h = kh^{gt})|kR^{+}\}$ and $v = 0$, it is impossible. (2) In the $\frac{\partial v}{\partial w} = -\frac{h}{w}\frac{\partial v}{\partial h}$ form, $\frac{\partial v}{\partial v}$ is the opposite of $\frac{\partial v}{\partial h}$. If one of w or h increases, the other decreases, which is inconsistent with common sense. (3) Because v reveals aspect ratio dissimilarity, the loss or gain will optimize similarity in a non-ideal way.

So, we propose that the EIOU loss follows [19].

$$\begin{aligned} L_{EIOU} &= L_{IOU} + L_{dis} + L_{asP} \\ &= 1 - IOU + \frac{p^2(b,b^{gt})}{(w^c)^2+(h^c)^2} + \frac{p^2(w,w^{gt})}{(w^c)^2} + \frac{p^2(h,h^{gt})}{(h^c)^2} \end{aligned} \tag{7}$$

Here, h^w and h^c represent the width and height of the minimum enclosing box covering the two boxes. That is, the sub-damage function has three sections: IOU and L_{IOU}, spacing loss L_{dis}, and facing loss L_{asp}. As a result, the original old damage characteristics are maintained, while the function directly reduces the height–width difference between the target and the anchor, facilitating more rapid convergence and more accurate localization. The overall improved network structure is shown below in Figure 6.

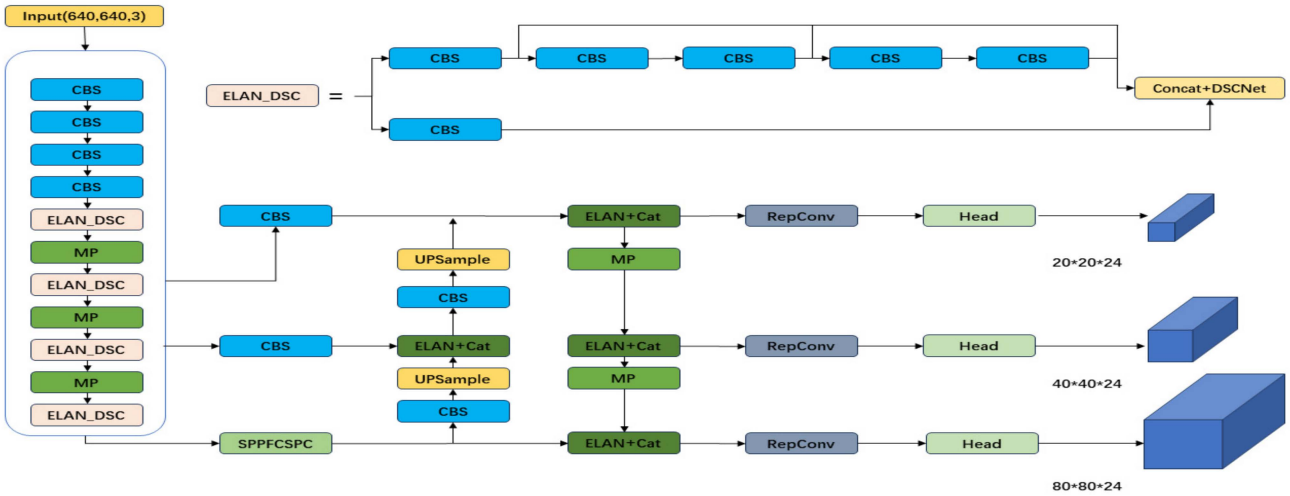

Figure 6. Improved YOLOv7 overall network.

4. Analysis of Experimental Results

4.1. Datasets

In order to test the algorithm's performance, the group conducted tests on the Visdrone2019 dataset. The AISKYEYE team of Tianjin University summarized the dataset, containing 288 videos, 261,908 frames, and 10,209 images. The data were obtained from various drone photography sources, covering a wide range of scenarios, including 14 cities in China with long distances, different environments in urban and rural areas, and various objects (e.g., pedestrians, vehicles, etc.) with varying densities.

VisDrone2019 contains 6471 training images, 548 validation images, and 1610 test images, covering a wide range of traffic scenarios, such as highways, intersections, and T-intersections, as well as a wide range of climatic backgrounds, from day and night to hazy and rainy days. The set can be used to validate the UAV ground-based small target detection performance. All methods within the experiment are trained in the training set and evaluated in the validation set.

4.2. Experimental Steps

Table 1 displays the experimental hardware setup: an Intel(R) Core(TM) i9-113500FCPU @ 3.50 GHz, with model training on a GeForce GTX 3090 featuring 24 GB of video RAM and 40 GB of system RAM. The experiment ran on a Windows operating system, utilized Python 3.8.6 for programming, and was built on the Pytorch 1.11.0 framework, incorporating CUDA 11.6 for enhanced processing.

Table 1. Experimental parameter configuration.

Configuration	Name	Type
Hardware	CPU	Intel(R) Core(TM) i9-113500F
	GPU	NVIDIA GeForce GTX3090
	Memory	40GB
Software	CUDA	11.6
	Python	3.8.6
	Pytorch	1.11.0
Hyperparameters	Learning Rate	0.01
	Image Size	640 × 640
	Workers	8
	Batch Size	16
	Maximum Training Epochs	300

The experiment leveraged the Adam optimizer for model training to refine and update the network's weights. The optimizer was configured with specific settings: a 16-bit size, a learning rate of 0.01, a momentum of 0.937, a weight decay of 0.0005, and a training duration extended to 300 epochs to ensure comprehensive training of the network.

This research adopts the Adam optimizer to refine our model, blending the strengths of Momentum and RMSprop algorithms for effective weight adjustment. The essence of Adam, or adaptive moment estimation, lies in its ability to calculate both the mean (first-order moment) and the gradients' uncentered variance (second-order moment). It then dynamically tailors the learning rate for each parameter based on these calculations. This approach enables Adam to adjust its step size based on the parameter update history, thus offering faster convergence and enhancing both the training's efficiency and stability, in contrast to conventional stochastic gradient descent (SGD) methods.

4.3. Evaluation Indicators

The model employs precision (P, recall (R), average precision (AP), and mean average precision (mAP) as metrics to assess its performance. AP serves as the metric for evaluating the accuracy of detecting individual categories, while mAP is calculated by summing the AP values across all categories and dividing by the total number of categories. In the study, mAP0.5 is the mAP with a threshold of 0.5, where IoU measures the overlap ratio between the predicted and actual bounding boxes.

$$P = TP/(TP + FP) \tag{8}$$

$$R = TP/(TP + FN) \tag{9}$$

$$AP = \int_0^1 P(R)dR \tag{10}$$

$$mAP = \frac{1}{N}\int_0^1 P(R)dR \tag{11}$$

In the model's performance evaluation context, TP is the number of positive samples correctly identified as positive by the model. FP represents the count of negative samples incorrectly classified as positive. Meanwhile, FN denotes the number of positive samples that were mistakenly categorized as negative. These metrics are crucial for calculating precision, recall, and other related performance indicators.

4.4. Ablation Experiments

To demonstrate the efficacy of the introduced components, this study performed ablation tests on the VisDrone2019 dataset using the YOLOv7 as the foundational algorithm. These experiments focused on measuring mean average precision (mAP), parameter counts, and frames per second (FPS) to gauge performance enhancements. The outcomes of these tests are summarized in Table 2 below.

Table 2. Ablation experiment of improved point.

Method	P%	R%	Parameters/M	mAP%
YOLOv7	59.7	50.6	35.51	50.47
YOLOv7_dscnet	62.3	53.1	51.83	53.07
YOLOv7_SPPF	60.8	51.8	30.01	51.37
YOLOv7_EIOU	60.3	51.2	35.51	50.90
YOLOv7_dscnet_SPPF	62.8	53.5	46.82	53.45
YOLOv7_dscnet_EIOU	62.5	53.4	51.83	53.28
YOLOv7_SPPF_EIOU	61.3	52.3	30.01	52.01
Ours	64.1	54.9	46.82	54.7

Seven sets of ablation experiments were performed under equivalent conditions, as detailed below in Table 2:

- The first set of experiments for the baseline model, i.e., the YOLOv7 algorithmic model, is used as a reference, which has a mAP value of 50.47% on the Visdrone2019 dataset;
- The second group is to replace the ELAN of the benchmark model with the improved ELAN_DSC; the number of parameters increases by 16.32M, but the mAP is improved by 3.4%, the accuracy is improved by 2.6%, and the recall is improved by 3.5% compared with the benchmark model, and the main reasons for the model's enhancement include the following: The target as a fine structure accounts for a very small percentage of the overall image, with a limited pixel composition, and it is easily affected by the complex background. The main reasons for the model improvement are: the target is a small proportion of the overall image, the pixel composition is limited, and it is easily interfered with by the complex background, which makes it difficult for the model to accurately identify the subtle changes of the target, but the addition of the dynamic serpentine convolution to ELAN can effectively focus on the slender and curved target, thus improving the detection performance. Since dynamic serpentine convolution has better segmentation performance and increased complexity compared to normal convolution, the number of parameters in the improved module rises compared to the original model;
- The third group is replaced by the improved SPPF module, which improves 1.3% over the baseline model, improves 1.1% accuracy, improves 1.2% recall, and reduces the parameters. mAP, P, and R improvements and parameter reductions are analyzed as follows: the improved SPPF module performs Maxpool operations on convolutional kernels of different sizes to differentiate between different targets, increase the receptive field, and extract more important feature information; therefore, mAP, P, and R are improved; the improved module performs serial operations on convolutional kernels of different sizes and therefore reduces the model complexity, so the number of parameters decreases. The improved SPPF module uses different sizes of convolutional kernels for the Maxpool operation to distinguish different targets, increase the receptive field, and extract more important feature information, so the mAP, P, and

R are improved; the improved module operates the different sizes of convolutional kernels in a serial manner, so the complexity of the model is reduced, and the number of parameters decreases;

- The fourth group replaces the loss function with EIOU, the mAP improves by 0.43%, the accuracy improves by 0.5%, the recall improves by 0.6%, and the number of parameters is unchanged because of the unaltered network model and, therefore, unchanged compared to the baseline model. Analysis of the reasons for improving the detection performance: the loss function directly minimizes the difference in the height and width between the target box and the anchor box, which results in faster convergence and a better localization effect;
- From the second to the seventh set of ablation experiments, the introduction of DSCNet provided the key improvement, with a 3.4% improvement in mAP in Figure 7.

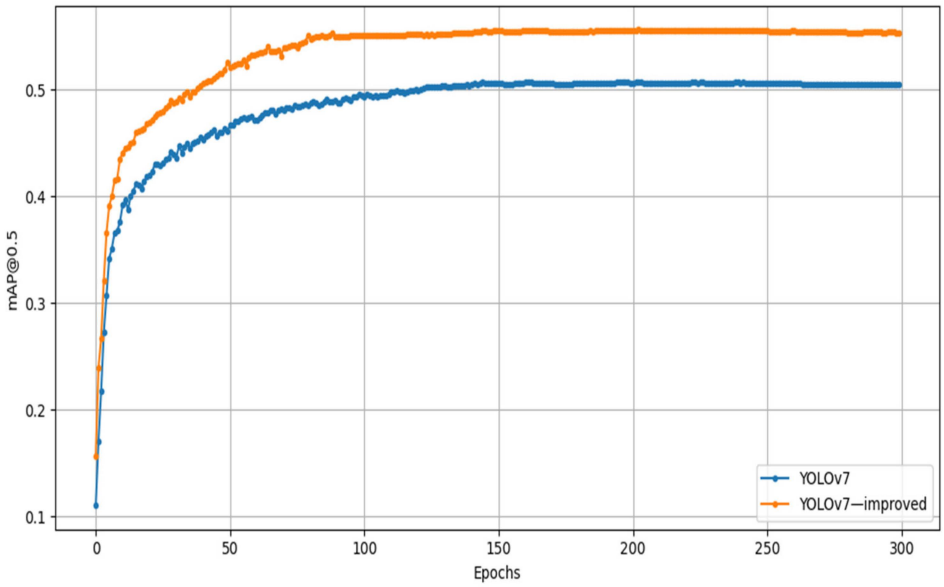

Figure 7. Comparison curve of mAP between the benchmark algorithm and the improved algorithm during the training process.

The following figure visualizes the change and improvement in mAP during the training process of the final improved algorithm and the benchmark model. It can be clearly seen that the original algorithm's mAP increases rapidly in the first 50 rounds and increases slowly from the 50th round until it reaches the final training mAP value around 150 rounds, and after that, it reaches convergence. In comparison, the improved algorithm has a rapid increase in the mAP in the first 30 rounds and a slow increase from the 30th round to around 90 rounds, and after that, reaches convergence. It can be clearly seen that the improved algorithm converges faster, and the mAP increases by 4.33% over the benchmark algorithm.

4.5. Comparative Experiments

Various UAV aerial image target detection algorithms, such as YOLOv4, YOLOv3-LITE, YOLOv5s, Faster RCNN, DMNet, etc., are selected to be compared and analyzed with the improved algorithm of this study on the Visdrone2019 test set. In Table 3, it can be seen that the comparison of this algorithm with others, with 33.0% improvement in mAP compared to Faster RCNN, 11.6% compared to YOLOv4, 24.4% compared to DMNet, and 23.6% compared to YOLOv5s. This algorithm not only improves significantly in mAP compared to mainstream target detection algorithms but is also significantly higher than other algorithms in AP; for example, car detection accuracy reaches 82.4%, van detection accuracy reaches 58.6%, and truck detection accuracy reaches 51.7. Due to other target detection algorithms, the experiments illustrate the effectiveness and practicality of this

algorithm for detecting weak and small targets in aerial images of UAVs. The experiment illustrates the effectiveness and practicality of this algorithm for detecting weak targets in UAV aerial images.

Table 3. Comparative experiments with different detection algorithms.

Method	AP%										mAP
	Pedestrian	Person	Bicycle	Car	Van	Truck	Tricycle	A-T	Bus	Motor	
Faster RCNN	21.4	15.6	6.7	51.7	29.5	19.0	13.1	7.7	31.4	20.7	21.7
YOLOv4	25.0	13.1	8.6	64.3	22.4	22.7	11.4	8.1	44.3	22.1	43.1
CDNet	35.6	19.2	13.8	55.8	42.1	38.2	33.0	25.4	49.5	29.3	34.2
DMNet	28.5	20.4	15.9	56.8	37.9	30.1	22.6	14.0	47.1	29.2	30.3
RetinaNet	13.0	7.9	1.4	45.5	19.9	11.5	6.3	4.2	17.8	11.8	13.9
Cascade R-CNN	22.2	14.8	7.6	54.6	31.5	21.6	14.8	8.6	34.9	21.4	23.2
CenterNet	22.6	20.6	14.6	59.7	24.0	21.3	20.1	17.4	37.9	23.7	26.2
YOLOv3-LITE	34.5	23.4	7.9	70.8	31.3	21.9	15.3	6.2	40.9	32.7	28.5
MSC-CenterNet	33.7	15.2	12.1	55.2	40.5	34.1	29.2	21.6	42.2	27.5	31.1
DBAI-Det	36.7	12.8	14.7	47.4	38.0	41.4	23.4	16.9	31.9	16.6	28.0
YOLOv5s	35.8	30.5	10.1	65	31.5	29.5	20.6	11.1	41.0	35.4	31.1
YOLOv7	53.3	46.5	26.1	77.3	53.4	46.5	39.8	19.9	59.7	55.8	50.47
Ours	57.6	51.1	29.3	82.4	58.6	51.7	44.4	24.6	64.7	60.4	54.7

Additionally, in order to reflect the advancement of this algorithm and to compare it with the current technical level of the YOLO V7 algorithm in the field of UAV, YOLOv7-UAV [20], PDWT-YOLO [21], and improved YOLOv7 algorithms are selected to compare with this algorithm [22]. It can be seen from Table 4 that this thesis shows that the YOLOv7-UAV algorithm is superior to this algorithm in terms of the parameters, but this algorithm in terms of this index is superior to the PDWT-YOLO and improved YOLOv7 algorithm. In addition, this algorithm is superior to the above algorithms in terms of the mAP metric, which can reach 54.7% in the VisDrone2019 dataset.

Table 4. Comparison of this algorithm with the latest improved algorithm based on YOLOv7.

Method	mAP%	Parameters/M
YOLOv7-UAV	52.21	3.07
PDWT-YOLO	41.2	24.2
Improved YOLOv7	45.30	26.77
Ours	54.7	16.82

4.6. Analysis of Detection Effects

The aerial images of UAVs in different complex scenes in the VisDrone2019 test set are selected for detection, and the detection effect is shown in Figure 8. It can be seen that this study's algorithm can attenuate the interference of trees and buildings in the complex background of the image and correctly segment and localize the target for the same small target in the complex background scene. It shows that this study's algorithm has better detection performance in actual scenes, such as lighting conditions, different backgrounds, and target distribution. Also, the confidence threshold is set to 0.25, below which the image confidence is not displayed.

To evaluate the detection performance on UAV aerial images, images under different scenes of very small targets, dark scenes, target occlusion, and complex backgrounds were randomly selected from the Visdrone2019 test challenge set and compared with the former algorithm in Figures 9–16.

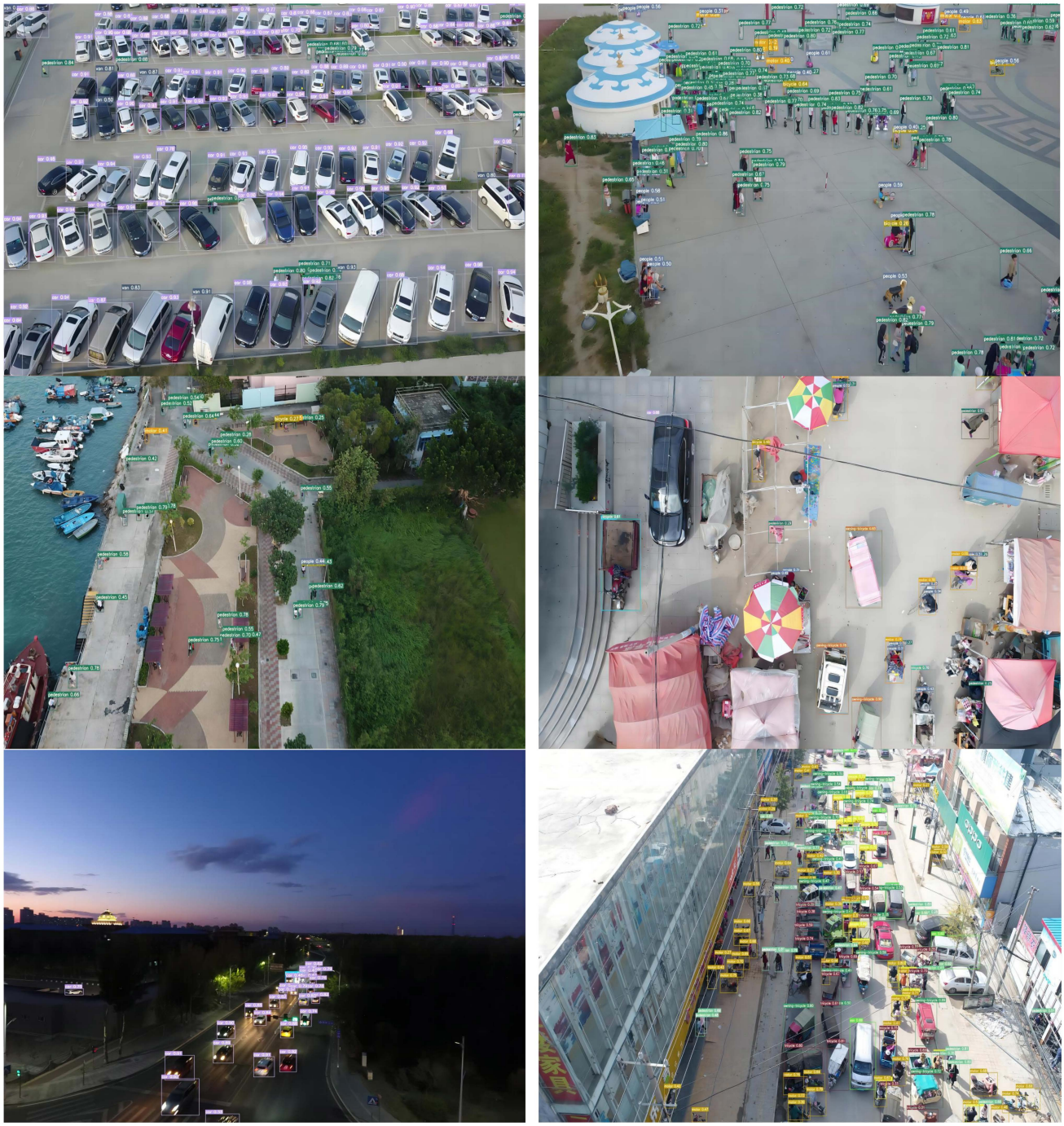

Figure 8. *Cont*.

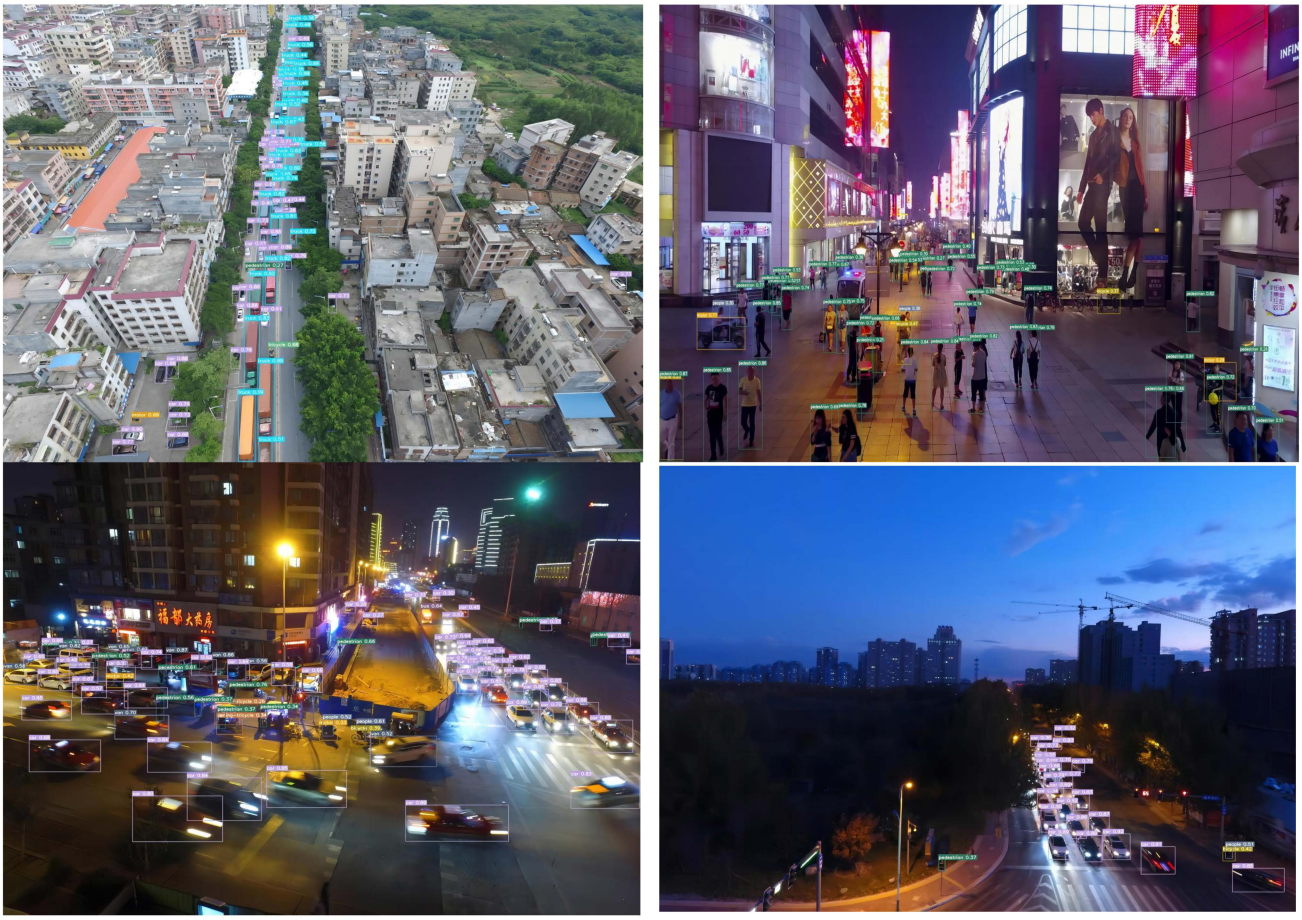

Figure 8. Effect of the improved algorithm in different scenarios.

Figure 9. Small targets (baseline algorithm).

Figure 10. Small targets (improved algorithm).

Figure 11. Complex background (baseline algorithm).

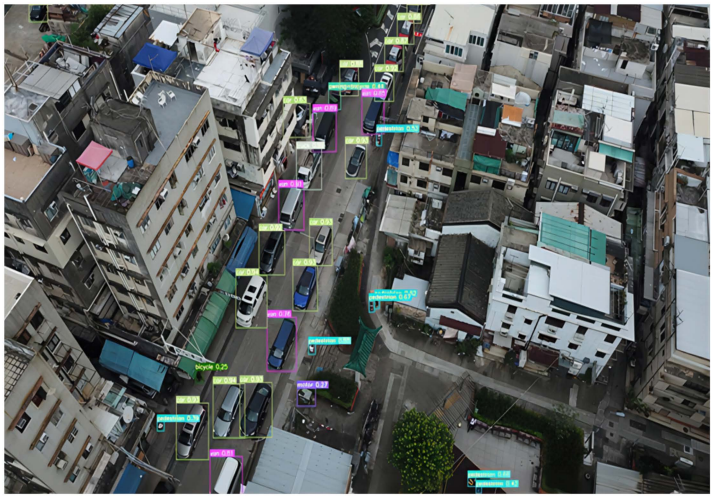

Figure 12. Complex background (improved algorithm).

Figure 13. Target occlusion (baseline algorithm).

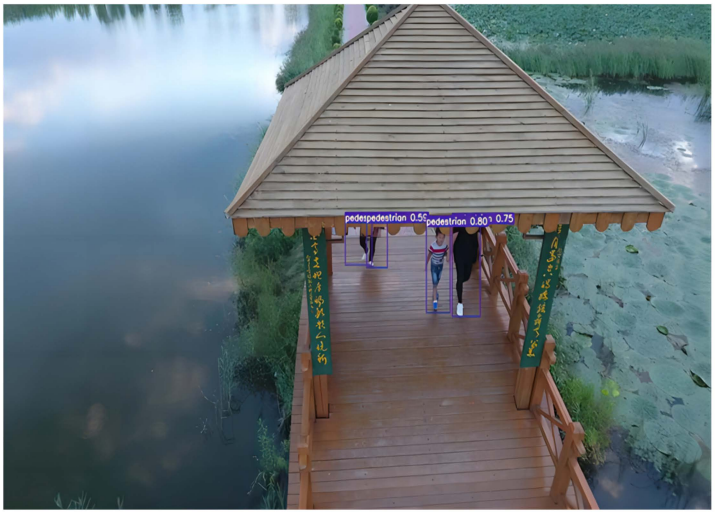

Figure 14. Target occlusion (improved algorithm).

Figure 15. Dark background (baseline algorithm).

Figure 16. Dark background (improved algorithm).

5. Conclusions

In this study, the problem of the difficult detection of small targets in complex backgrounds, which exists in UAV ground target detection, is successfully solved by introducing dynamic snake convolution (DSC), improved SPPCSPC based on the YOLOv7 model, and employing the EIoU loss function. After experiments, the improved algorithm in this study shows excellent detection effects in different aerial photography scenes and achieves optimal detection results in all nine categories, proving its strong practicality and effectiveness. Considering the similarity in processing requirements between satellite image analysis and UAV aerial images, especially in target detection, background complexity processing, and small target recognition, we believe it also applies to satellite image analysis. Satellite images are commonly used in the fields of geographic information systems (GIS), environmental monitoring, urban planning, and disaster management, where accurate detection and classification of small targets are also crucial. Although the resolution and scale of satellite images may differ from that of UAV images, the improved algorithm proposed in this study can still play an important role in satellite image processing by adjusting the algorithm's parameters appropriately or making slight modifications.

6. Future Work

Future research directions can be centered on the following core areas:

(1) Application migration and algorithm generalization: Exploring the migration of the improved algorithms developed in this study, such as dynamic serpentine convolution (DSC), improved spatial pyramid pooling structure (SPPFCSPC), and EIoU loss functions, to other models, for instance, YOLOv8, or SSD. Investigate how these improvements can be adapted to the characteristics of different algorithmic frameworks and how the parameters can be adjusted during the migration process to maintain or improve the accuracy and efficiency of target detection;

(2) Cross-domain application exploration: In addition to UAV image processing, explore the potential of the improved algorithms to be applied in other domains, such as satellite image analysis and traffic monitoring. In particular, study the performance of the algorithms in processing image data of different resolutions and scales and how to adapt the algorithms to the specific needs of these new fields;

(3) Real-time processing and edge computing: considering that UAV and satellite image analysis often requires real-time processing, future research could focus on optimizing the lightweighting and acceleration of models to adapt to edge computing platforms. Investigate how to deploy deep learning models to devices with limited hardware resources while maintaining efficient computational performance and accurate detection results;

(4) Multimodal data fusion: In UAV and satellite image analysis, multiple types of data (e.g., optical images, infrared images, radar data, etc.) are often involved. Future research can explore how to effectively fuse these different modalities of data.

Author Contributions: Conceptualization, Z.B. and Y.B.; methodology, Z.B. and X.P.; software, Z.B. and X.P.; validation, Z.B., X.P. and Z.Q.; formal analysis, Z.Q. and G.W.; investigation, Z.B. and G.W.; resources, Z.B.; data curation, Z.B.; writing—original draft preparation, Z.B.; writing—review and editing, Y.B.; visualization, Z.B.; supervision, Y.B.; project administration, X.P.; funding acquisition, Y.B. All authors have read and agreed to the published version of the manuscript.

Funding: Innovation Guidance Fund Project of Light Power Innovation Research Institute, Chinese Academy of Sciences (CXYJJ20-ZD-03); funded by National Key R&D Program (No. 2022YFF1302000).

Data Availability Statement: Data set: https://github.com/VisDrone. Access date: 1 December 2023.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hastie, T.; Rosset, S.; Zhu, J.; Zou, H. Multi-class adaboost. *Stat. Its Interface* **2009**, *2*, 349–360. [CrossRef]
2. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
3. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
4. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
5. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 213–229.
6. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
7. Liu, M.; Wang, X.; Zhou, A.; Fu, X.; Ma, Y.; Piao, C. Uav-yolo: Small object detection on unmanned aerial vehicle perspective. *Sensors* **2020**, *20*, 2238. [CrossRef] [PubMed]
8. Saetchnikov, I.; Skakun, V.; Tcherniavskaia, E. Efficient objects tracking from an unmanned aerial vehicle. In Proceedings of the 2021 IEEE 8th International Workshop on Metrology for AeroSpace (MetroAeroSpace), Naples, Italy, 23–25 June 2021; pp. 221–225.
9. Li, Z.; Liu, X.; Zhao, Y.; Liu, B.; Huang, Z.; Hong, R. A lightweight multi-scale aggregated model for detecting aerial images captured by UAVs. *J. Vis. Commun. Image Represent.* **2021**, *77*, 103058. [CrossRef]
10. Avola, D.; Cinque, L.; Diko, A.; Fagioli, A.; Foresti, G.L.; Mecca, A.; Pannone, D.; Piciarelli, C. MS-Faster R-CNN: Multi-stream backbone for improved Faster R-CNN object detection and aerial tracking from UAV images. *Remote Sens.* **2021**, *13*, 1670. [CrossRef]
11. Azimi, S.M.; Kraus, M.; Bahmanyar, R.; Reinartz, P. Multiple pedestrians and vehicles tracking in aerial imagery: A comprehensive study. *arXiv* **2020**, arXiv:2010.09689.
12. Yang, F.; Fan, H.; Chu, P.; Blasch, E.; Ling, H. Clustered object detection in aerial images. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27–28 October 2019; pp. 8311–8320.
13. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
14. Qi, Y.; He, Y.; Qi, X.; Zhang, Y.; Yang, G. Dynamic snake convolution based on topological geometric constraints for tubular structure segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 6070–6079.
15. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 764–773.
16. Li, C.; Li, L.; Geng, Y.; Jiang, H.; Cheng, M.; Zhang, B.; Ke, Z.; Xu, X.; Chu, X. Yolov6 v3. 0: A full-scale reloading. *arXiv* **2023**, arXiv:2301.05586.
17. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
18. Zheng, Z.; Wang, P.; Ren, D.; Liu, W.; Ye, R.; Hu, Q.; Zuo, W. Enhancing geometric factors in model learning and inference for object detection and instance segmentation. *IEEE Trans. Cybern.* **2021**, *52*, 8574–8586. [CrossRef]
19. Zhang, Y.-F.; Ren, W.; Zhang, Z.; Jia, Z.; Wang, L.; Tan, T. Focal and efficient IOU loss for accurate bounding box regression. *Neurocomputing* **2022**, *506*, 146–157. [CrossRef]

20. Zeng, Y.; Zhang, T.; He, W.; Zhang, Z. Yolov7-uav: An unmanned aerial vehicle image object detection algorithm based on improved yolov7. *Electronics* **2023**, *12*, 3141. [CrossRef]
21. Zhang, L.; Xiong, N.; Pan, X.; Yue, X.; Wu, P.; Guo, C. Improved object detection method utilizing yolov7-tiny for unmanned aerial vehicle photographic imagery. *Algorithms* **2023**, *16*, 520. [CrossRef]
22. Li, X.; Wei, Y.; Li, J.; Duan, W.; Zhang, X.; Huang, Y. Improved YOLOv7 Algorithm for Small Object Detection in Unmanned Aerial Vehicle Image Scenarios. *Appl. Sci.* **2024**, *14*, 1664. [CrossRef]