ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 1 - 2026

Deep Learning Based Target Detection in UAV Images-Drone-YOLO Algorithm

Jingping Guo[1,*]

[1] China People's Police University (Guangzhou), Guangzhou, Guangdong, 510663, China

SUMMARY: *Along with the evolution of scientific knowledge, UAV aerial photography technology has become an important tool for acquiring information, and the research of UAV target detection will contribute to the decision-making and guidance of traffic management and diversion, with great prospects in the future field of intelligent transportation. This paper proposes an algorithm model of improved YOLOv5 (Drone-YOLO) for target detection based on the expansion of UAV image data by utilizing the improved cyclic generative adversarial network (CycleGAN). The Drone-YOLO algorithm model improves the adaptability to the size changes of UAV image data targets through the increase of detection branches, integration of multi-level information, and fusion of multi-scale features. Additionally, the introduction of the multi-scale attention mechanism increases the attention ability of the network on the target objects; the decoupling of classification and regression tasks promotes the detection accuracy, and the optimization of the loss function improves the efficiency of training. According to the experiments conducted on the drone image data set of VisDrone, the precision of the Drone-YOLO algorithm reaches 51.5%, an increase of 5.6% from the original model. Meanwhile, the recall rate was improved from 34.2% to 39.6%, mAP@.5 from 34.5% to 39.8%, and mAP@.5:.95 from 18.3% to 23.1%, all of which are enough to complete the detection task of target objects in UAV scenarios.*

KEYWORDS: *generative adversarial networks; multi-scale attention; target detection; YOLO algorithm; UAVs*

1 Introduction

UAV is a kind of unmanned aircraft that utilizes self-contained program control device and radio remote control equipment. It is characterized by small size, easy operation and flexibility, which makes its application scenarios become more and more diversified, such as agricultural operations, logistics and other fields [1, 2]. Due to UAV technology's maturity, it is extensively employed in many areas, including traffic management, environmental detection, and hazardous area search, which safely and conveniently accomplishes a range of difficult aerial flight duties including loading and monitoring [3-5]. The application of UAV technology is basically inseparable from the images taken by the UAV, it is different from the manned flight equipment, the UAV can only interact with the personnel through the image, in this weak interaction between man and machine environment want to realize the relevant tasks, the first thing is to analyze and understand the image information, and the enhancement of the understanding of the scene information must be based on the target detection [6-9]. As early as in the 1990s, target detection technology has been applied to the UAV field, limited by the early

*18680555590@163.com
https://doi.org/10.65102/is2026411
Published: 30 April, 2026

computing power, although the algorithm has been optimized but still slow development [10, 11]. After these decades of development, with the significant increase in computing power today, it has become more convenient to use target detection techniques to solve UAV application problems.

One of the most popular areas of computer vision research is target identification, as well as one of the most challenging branches, and nowadays it is widely used in people's lives, such as surveillance security, industrial inspection, drone scene analysis, traffic monitoring, automatic driving, etc. The goal is to limit the use of human resources, which is crucial for both industrial production and daily living, and to identify the target quickly and effectively using a computer [12-15]. In practical applications, object detection broadly falls into two research directions: “research-oriented object detection” and “specialized application object detection.” The former primarily investigates how object detection networks allocate attention to various objects, aiming to simulate human visual cognition. The latter focuses on specialized detection of specific objects in particular environments, such as pedestrian detection, face recognition, and behavior detection [16-20]. Nevertheless, because of their distinct location, weather, and flying distance, the objects captured on UAV aerial images vary widely from those of the generalized target detection database. For instance, the size of the same object will be significantly different in different images, and the ratio of sizes among various objects within the same image cannot correspond to reality, and so forth. On the one hand, the high degree of flexibility that comes with the UAV helps in the completion of UAV aerial imaging tasks, but on the other hand, it also causes the same object to appear differently due to varied angles [21-24]. In addition, the height and moving speed of the UAV are fast, and there are also a large number of small-sized targets in the UAV aerial images, which all bring great test for the target detection task [25, 26].The development of deep learning technologies has led to an unprecedented research frenzy in the target identification profession in recent years.

As hardware technology advances, target detection development may be classified into two groups. There are two types of target detection: deep learning-based target detection and classical target detection during a time when computer power was constrained. Traditional target detection and modern target detection vary in a few ways because deep learning convolutional neural networks are capable of efficiently extracting features automatically. On the other hand, unlike SIFT and HOG, the former necessitates manual feature extraction, which makes the procedure difficult and ineffective [27, 28]. In contrast, deep learning-based target detection algorithms are robust, accurate, and quick. To enhance the identification of tiny targets, literature [29] suggested a self-attention steering and multi-scale feature fusion-based unmanned aerial vehicle image target identification network. Additionally, the model made use of inverse residual feature augmentation, parallel sampling feature fusion, and global-local feature guiding. Literature [30] utilizes a single-stage detector to recognize UAV images, sets the detection target (divided into certain and suspicious areas) with a threshold of confidence, captures image features under a visual geometry group architecture, fuses feature maps and suspicious area information to increase the confidence of the suspicious area, performs secondary detection, and combines the results of the first and second detections as the final detection results. According to literature [31], the convolutional neural network (CNN)-based UAV image target identification approach reportedly achieves 97.5% detection accuracy, and its computational efficiency and classification effect are also high. Literature [32] extracted UAV image hierarchical features with cross-shaped window transformer, input them into feature pyramid network for fusion, and constructed CNN-based hybrid block embedding module and inference method for image low-level feature extraction, and thereafter combined the original and sliced images to improve the performance of target detection algorithms for UAV images with multi-scale and inhomogeneous distribution. Literature [33] proposed a MS-

Faster R-CNN object detector called novel multi-stream architecture and combined with a deep online real-time tracking algorithm for target detection and tracking of UAV images, which is spoken to continuously detect and track targets in video sequences in real time.

The speed and accuracy of UAV image target recognition algorithms have been significantly increased with the iterative updating of the YOLO series of algorithms. In order to increase the real-time accuracy of UAV image target recognition, literature [34] suggested an enhanced YOLO (You Only Look Once) technique based on target frame size clustering, pre-trained network classification, multi-scale detection training, and candidate frame filtering criteria. Literature [35] used YOLOv4 and multi-target tracking algorithms to formulate an automatic UAV detection and tracking method for vehicles in urban environments, combined with maneuvering target state estimation to localize the vehicles, as a way to design deep learning based algorithms for UAV target detection, tracking, and localization. According to the literature [36], the P2 detection module is added to the YOLOv8 model, and a fusion process of various layers features through the upgraded bi-directional feature pyramid network is performed, followed by the addition of the RCS-OSA (Reduced Channel Spatial Object Attention) module to the YOLOv8 model to form an enhanced YOLOv8 UAV image target detection algorithm. Along with the evolution of the YOLO algorithms, for UAV image target detection, the goals of miniaturization targets and lightweight real-time have been considered. Literature [37] prunes and adjusts the YOLO algorithm and establishes a small UAV target detection network, optimizes the feature extraction accuracy and success rate of SIFT based on the methods of adaptive thresholding and minimum distance, and tracks and monitors the target under the correlation of scale and position information. In literature [38], an uncomplicated parameter-free attention block has been proposed to enhance the YOLOv7 network model for achieving improved results in tiny object detection from UAV image datasets. In literature [39], the space to depth convolution module and various attention strategies have been presented in the UAV image detection framework that relies on the YOLOv5 architecture, with the aim of minimizing the loss of sampling data and improving target detection and small object detection. In literature [40], an enhanced target detection algorithm has been developed for UAV aerial images by using YOLOX-X. The slicing inference strategy is applied during data preprocessing, along with shallow features and a newly developed detection head that aims at focusing on small targets.

Literature [41] offers a target detection algorithm for UAV images using the improved YOLOv8 method, where the focus was shifted to the feature extraction of the orthogonality of the enhancement module with emphasis on the target regions by means of the local attention module. This algorithm achieves the detection of small targets as well as being lightweight while maintaining computational efficiency. On the basis of the YOLOv5s algorithm, the paper [42] developed an efficient lightweight-YOLO architecture and has achieved a detection accuracy rate of up to 96.8%. Literature [43] presents a YOLO lightweight target detector algorithm with an effective receptive field module capable of preserving the local features in the process of feature extraction. In addition, the paper optimizes the network structure of path aggregation networks using the module. This algorithm can perform small object detection under complex backgrounds with the assistance of the added detection head. Literature [44] designed a lightweight multi-scale infrared vehicle target image feature extraction network-YOLOv7-MobileViT, which combines content-based feature reorganization to extract image semantic features, enhance network performance and optimize anchor frame size under C3 structure and K-means++ clustering method, thus improving the accuracy and computational efficiency of vehicle target detection in infrared images from UAVs . Literature [45] designed a new feature extraction module using sensory wild attention convolution to optimize the performance of small object extraction from UAV aerial images, combining spatial pyramid

pooling with large divisible convolutional kernel attention to reduce the interference of the target image, replacing the dynamic detector head, and multi-scale target localization with a bounding box regression loss function based on the dynamic focusing mechanism, thereby achieving a YOLOv8-based object detection algorithm with 94.2% and 95.4% detection accuracy and precision. Literature [46] designed the YOLO algorithm for low-cost edge hardware for the task of vehicle detection in UAV aerial images, which breaks through the problems of small target size and real-time performance. Literature [47] compared several network architectures based on ResNet50 for real-time detection of cracks in UAV paved sidewalk images, with OLOv2 and YOLOv4-tiny having better detection accuracy and speed, as well as detection ability under environmental interference. Literature [48] replaces the backbone network of the YOLO model with ShuffleNetV2, combines the multiscale expansion attention module to promote the model's multiscale target detection robustness and accuracy, introduces phantom convolution technique and composite sensory field lightweight convolution method to optimize the model structure and the efficiency of the multiscale feature fusion, and the Wise-loU loss function to optimize the model anchor frames, which balances the UAV image target detection accuracy and operational efficiency, and realizes lightweight, low cost and real-time.

The fundamental ideas of deep learning and YOLO object identification are initially presented in this study. Additionally, the CycleGAN network model is improved in terms of its architecture, activation function, and the addition of style loss to increase UAV photos in order to address the problem of insufficient UAV images. The Drone-YOLO algorithm model is suggested to improve YOLOv5 and increase its capacity to identify various layers of the model in the situation of missing target detection and false positive target detections in the UAV photos. Network structure of the feature pyramid is designed for information integration at multiple levels. Feature fusion strategy based on multi-level channel attention is proposed to further emphasize small targets. The information of the VisDrone small target dataset used in experiments is illustrated. Commonly-used performance metrics in this paper are introduced. Then a comparative analysis between Drone-YOLOe and several baseline models such as YOLOv5 is carried out.

2 UAV image target detection based on Drone-YOLO algorithm

2.1 Deep learning based YOLO series algorithm

2.1.1 Deep learning

In order to replicate the hierarchical information processing mechanism of biological neural systems and accomplish autonomous feature extraction and pattern recognition, deep learning, a significant subfield of machine learning, focuses on the bionic construction of multilayer neural network architectures [49]. Neural network technology has evolved from the basic theoretical framework of artificial neural networks (ANN), which is essentially a typical connectionist computational model that builds a distributed computing architecture through densely interconnected neuron-like units, with neuron inputs of n $x_1, x_2 \ldots x_n x_1, x_2 \ldots x_n$, $x = [x_1, x_2 \ldots x_n]$ denotes the input.

$$\sum_{i=1}^{n} w_i x_i + b \tag{1}$$

Equation (1) represents the weighted sum of input signals x obtained by a neuron, where w is the weight corresponding to each input and b is the bias, after which the weighted sum is passed through the activation function f to obtain the output.

In order to construct information processing systems, artificial neural networks [50] are built on the bionics concept. Their usual design consists of input, hidden, and output layers, with the number of hidden layers dictating the model's depth level. The network realizes the information transmission between neurons through the synapse-like connection structure, and the neurons in each layer transmit the input data to the subsequent layers after the hierarchical nonlinear transformation. In the structure of feed-forward neural network, the data flow strictly obeys the one-way propagation rule from input nodes to output nodes, without any reverse communication pathway. To increase the information transfer rate, a parameter adjustment technique is used for tuning the weight associated with each connection link to enhance the model accuracy through reducing the gap between actual and predicted outputs. As the training process proceeds, the weight matrix is iteratively modified according to the gradient descent approach to make the network output converge to the true value.

Two processes exist whereby the weight values are changed; these processes include forward and backward propagation. In the process of forward propagation, it involves the propagation of signals from the input to the hidden layer, where the learning occurs, and then to the output layer from which the predicted value is produced. If there is a difference between the actual and expected results and the difference exceeds a certain threshold value, then the error value will be fed back through backward propagation.

2.1.2 YOLO series of target detection algorithms

The YOLO series is a significant turning point in the evolution of target detection methods, and its most notable features are its real-time target identification and single-shot target detection capabilities.Its unique advantage is embodied in the process of reformulating the task of target detection into a global regression task. It directly obtains the target's location information and category distribution using a unified structure of fully convolutional networks simultaneously, thus enabling synergies of accuracy and real-time speed.

The YOLOv1 algorithm splits the input image into 7 × 7 grids, wherein each cell predicts the object whose centroid falls inside that cell. The number of predicted bounding boxes and the corresponding categories are set at B and C respectively. The five elements included in one bounding box are the center coordinates (x, y) and dimensions (w, h). The last output tensor size is 7 × 7 × 30, with 30 = B × 5 + C, where (x, y) refers to the displacement of the bounding box center relative to the upper left point of the grid that should be normalized into the range of 0-l. (w, h) represents the proportion of the bounding box width and height regarding the entire image. Confidence refers to the confidence degree of the object within the grid and the precision of the prediction, which is defined in Eq. (2), in which Pr(Object) is the probability that there is an object within the grid, considered either zero or one. In the case that the bounding box includes the object, then Pr(Object) equals one, and therefore the confidence degree equals the IoU value, otherwise, the confidence degree equals zero.

$$confidence = \Pr(Object) \times IoU_{pred}^{truth} \tag{2}$$

YOLOv5, YOLOv5, as the fifth generation algorithm of YOLO series, inherits the genes of the single-stage detection framework, and at the same time achieves a double breakthrough in speed and accuracy through modular architecture design and engineering optimization. The main idea is global regression, which uses one-time forward propagation to identify and forecast the target bounding box, class, and confidence.

Although YOLOv6 and YOLOv7 improve the overall performance, YOLOv6 and YOLOv7 have poor engineering support and the complex structure is difficult to be efficiently deployed on edge devices, which is a limitation in small target detection scenarios of UAVs, whereas YOLOv5 becomes a better choice due to its lightweight, flexibility and engineering advantages.

The core idea of YOLOv8 is to directly predict the bounding box and category probability of a target through end-to-end regression without the need of region proposal or multi-stage processing. Compared with its predecessor model, YOLOv8 has made several optimizations in architectural design, loss function and training strategy, which significantly improves the detection accuracy and speed.

2.1.3 YOLOv5 neural network modeling

1) Backbone Network

Four essential modules make up the backbone network of the YOLOv5 model: the Focus module, the CBL module, the C3 module, and the SPPF module. These modules collectively constitute the basis of the feature extraction process.

Before image processing, the Focus module generates four small images that complement each other by selecting one pixel point every other pixel from the original image by neighborhood sampling. The width and height of these small images are reduced by half but still retain all the image information. Subsequently, the width and height information, which belongs to the image space, is converted to the channel dimension so that the number of input channels is expanded from three to twelve in rgb. Then, the spliced image is processed by convolution operation to generate a downsampled twice feature map without loss of information.

2) Intermediate layer

The intermediate layer of the YOLOv5 model is enhanced to recognize targets of various sizes by combining the FPN and PAN architectures, a technique that is appropriate for multi-scale target identification applications.

The key characteristic of the FPN architecture is that it uses a top-down method to fuse the feature maps to integrate information. Specifically, based on up-sampling and lateral connection, the FPN can efficiently integrate deeper semantic information and shallower spatial information into a feature map to achieve effective information integration. When using the FPN architecture in the model, fpn firstly utilizes the deepest feature map output by the backbone network, and then gradually enlarges the scale of the feature map at a double ratio through up-sampling operations. These up-sampled feature maps are reduced to fewer channels by applying the 1×1 convolution kernel, and concatenated with the feature maps from the previous layer to strengthen the feature representation. Then, convolution operations with a 3×3 kernel are employed to extract features and improve feature quality. By doing so, the FPN model can effectively transfer the rich semantic information in the deep layer to the shallow layer to reinforce the semantics of shallow layers.

The PAN uses a bottom-up feature synthesis path to enhance the localization performance of each layer's feature map at different scales. The bottom-up process, which starts at the bottom-most layer of the feature map, progressively integrates more positional information upward through layer-by-layer convolution and fusion operations. At each level of the feature map, it is combined with the down-sampled version of the upper level of the feature map after convolution to preserve location information while incorporating deep semantics. The bidirectional flow of feature maps not only increases efficiency in utilizing features but also makes less effort to pass through multiple layers to ensure that the accurate location information at the lower layer influences the upper layer's feature representation.

3) Head Network

The head network, also known as the detection module, consists of multilayer convolutional layers that process three different scales of feature maps from the backbone network. The head network achieves target detection through anchor frames, convolutional layers, prediction layers, and non-maximal suppression.

2.2 UAV image target detection data expansion

Data expansion is realized by training an improved CycleGAN to change the background of the training set images as follows: the first step is to establish different style datasets, the second step is to train the generative adversarial network according to different styles and migrate the original training set to different image styles; the third step is to merge different kinds of migrated images to complete the data expansion. Compared with the detection model trained only with the original training set, the detection accuracy of the target detection model with the style migrated images is improved.

2.2.1 Feature Perturbation Improvement

The original CycleGAN network is mainly used for image cross-domain conversion task, but the network can only realize one-to-one mapping after training can not realize one-to-many mapping, that is, an image as input, the output image is uniquely determined. In this paper, CycleGAN network is utilized for image style migration for data expansion, if one-to-many mapping is realized, i.e., an image is used as input and the output image is indeterminate, it will be more conducive to the task of data expansion, therefore, this paper proposes the method of feature perturbation, which is carried out for the shallow, medium, deep, and multi-layer feature perturbation, respectively.

This article adds a noise layer after the feature extraction module in CycleGAN. The variables in the noise layer follow a normal distribution with a mean of 0 and a standard deviation of 1. By adding noise at different depths of the network, it achieves feature perturbations at shallow, middle, deep, and multiple layers. The purpose of introducing the noise layer is to introduce perturbations to the extracted features, so that even after the model is trained, when the images are upsampled and reconstructed through the extracted features, they can still produce different output images.

2.2.2 Activation function improvement

Regardless of the number of layers in the neural network, the layers can be linearly stacked into one layer in the absence of an activation function. The activation function is defined to have a nonlinear impact on the neural network, enabling the network to fit to complex nonlinear functions. The ReLU function, which has the following functional equation, is the most commonly used activation function:

$$Relu(x) = \begin{cases} 0, x \le 0 \\ x, x > 0 \end{cases} \tag{3}$$

The expression of its derivative function is:

$$Relu(x) = 1, x > 0 \tag{4}$$

The advantage of the Relu function is that it is computationally efficient, allowing the network to converge quickly, and its nonlinearity in neural networks is mainly reflected in the unequal activation of the positive and negative regions, due to its complete inactivation on the

negative region, when the input is close to 0 or negative the output gradient of the Relu function will fall to 0, and the weights of the neurons will not be updated when back propagation is carried out, which is equivalent to the neurons being “necrosis”.

The original CycleGAN model uses LeakyRelu as the activation function, which is a kind of improvement function of Relu function, and its improvement goal is to solve the problem of neuron “necrosis” in the negative region of Relu function, and the main method is to carry out linear activation with small amplitude in the negative region, and its function is as follows The expression is shown in equation (5):

$$LeakyRelu(x) = \begin{cases} \alpha \cdot x, x \le 0 \\ x, x > 0 \end{cases} \tag{5}$$

The expression of its derivative function is:

$$LeakyRelu'(x) = \begin{cases} \alpha, x \le 0 \\ 1, x > 0 \end{cases} \tag{6}$$

where α is the slope of the negative region, generally take a small number between 0-1, the original CycleGAN uses $\alpha = 0.2$, the introduction of α makes the activation function also have a small activation value in the negative region.

Since the positive domain of the LeakyRelu activation function is similar to that of the Relu activation function, it offers all the benefits of using the latter. In addition, the LeakyRelu activation function addresses the issue of neuron death that arises when using the Relu activation function because the latter has a low positive slope in the negative domain. Nevertheless, it utilizes an exponential function in the negative lX domain, where activation values are high for negative input values.

In order to enhance the nonlinearity of the activation function and weaken its activation value in the negative region, this paper improves on the Relu activation function by using a convergent nonlinear function in the negative region, which makes the activation value converge to 0 when the input negative value is large, and at the same time, a small perturbation is applied to the activation value in the positive region, henceforth referred to as the RS activation function, to improve the model's resilience.

$$RS(x) = \frac{\beta \cdot x}{1 + e^{-x}} \tag{7}$$

where β is the float coefficient, in this paper, we take β between 0.8 and 1.2, and its derivative function expression is:

$$RS'(x) = \beta \cdot \left(\frac{e^{-x} \cdot x}{(1 + e^{-x})^2} + \frac{1}{1 + e^{-x}} \right) \tag{8}$$

2.2.3 Loss function improvement

In order to further improve the consistency of the style between the training set and the style dataset in the picture style migration, the improvement of the loss function in this study primarily incorporates the style loss. In the original CycleGAN model, the training set and the style dataset are directly converted to each other, and the image in the source domain is extracted from the deep features through a series of convolutional downsampling, and then

computed to the target domain through upsampling convolution, which lacks the control on the consistency of the style between the training set and style dataset in the process of computation, and in this paper we adopt the method of seeking the difference of the reconstructed features of the training set and the Gram matrix in the style dataset, and continuously reducing the difference between them during the training process. The method used in this paper is to find the difference between the reconstructed features of the training set and the Gram matrix of the features of the style dataset, and to continuously reduce the difference between them during the training process.

The Gram matrix is defined as the matrix consisting of the inner product of any k ($k \leq n$) vectors $a_1, a_2, ..., a_k$ in the n-dimensional Euclidean space, also called the Gram matrix of k vectors, with the expression as in equation (9):

$$\Delta(a_1, a_2, ..., a_k) = \begin{pmatrix} (a_1, a_1)(a_1, a_2)...(a_1, a_k) \\ (a_2, a_1)(a_2, a_2)...(a_2, a_k) \\ ..... \\ (a_k, a_1)(a_k, a_2)...(a_k, a_k) \end{pmatrix} \tag{9}$$

The Gram matrix calculates the covariance matrix of the input unsubtracted mean values, and calculates the image feature map The reason that the Gram matrix can reflect the general style of the image is that each value in the image feature map comes from the result of the calculation of a specific convolution kernel at that position, so each value in the feature map reflects the intensity of the features in a specific region, and in fact, the Gram matrix calculates the correlation between the features, i.e., which two features are mutually reinforcing, which two features are dominant, and so on. The Gram matrix actually calculates the correlation between features, i.e., which two features are mutually reinforcing, which two features are in a relationship, etc. Meanwhile, the diagonal elements of the Gram matrix of the feature map also reflect the intensity of individual features in the image, so the Gram matrix can be utilized to measure the approximate style of the image to a certain extent.

The original CycleGAN loss function is:

$$L = L_{GAN} + \lambda \cdot L_{cyc} \tag{10}$$

The loss function of the improved CycleGAN function in this paper is:

$$L = L_{GAN} + \lambda \cdot L_{cyc} + \alpha \cdot L_{gram} \tag{11}$$

where L_{gram} is the style loss and α is the style loss coefficient, there:

$$L_{gram} = \sum (t_t \times t_f^T - s_t \times s_f^T)^2 \tag{12}$$

where t_f, s_f denote the training set features and style dataset features extracted by CycleGAN network, respectively, T denotes the matrix transpose, and Σ denotes the summation over all elements of the matrix.

2.3 Drone-YOLO modeling

2.3.1 New small target detection branch

The architecture of Drone-YOLO is illustrated in Fig. 1. The box with dotted lines is the newly designed detection branch called P2, which is responsible for detecting very tiny objects. Since the information contained in the feature map of P2 branch is mostly obtained through shallow layers, it retains a great deal of information related to shape, position, and size. However, due to the repeated convolution pooling operation, the feature map generated through deep layers loses a lot of useful information, making it possible that the large target features obscure the tiny target features. So the P2 branch which introduces the shallow information can effectively localize the position of small targets, and thus can better detect small targets.

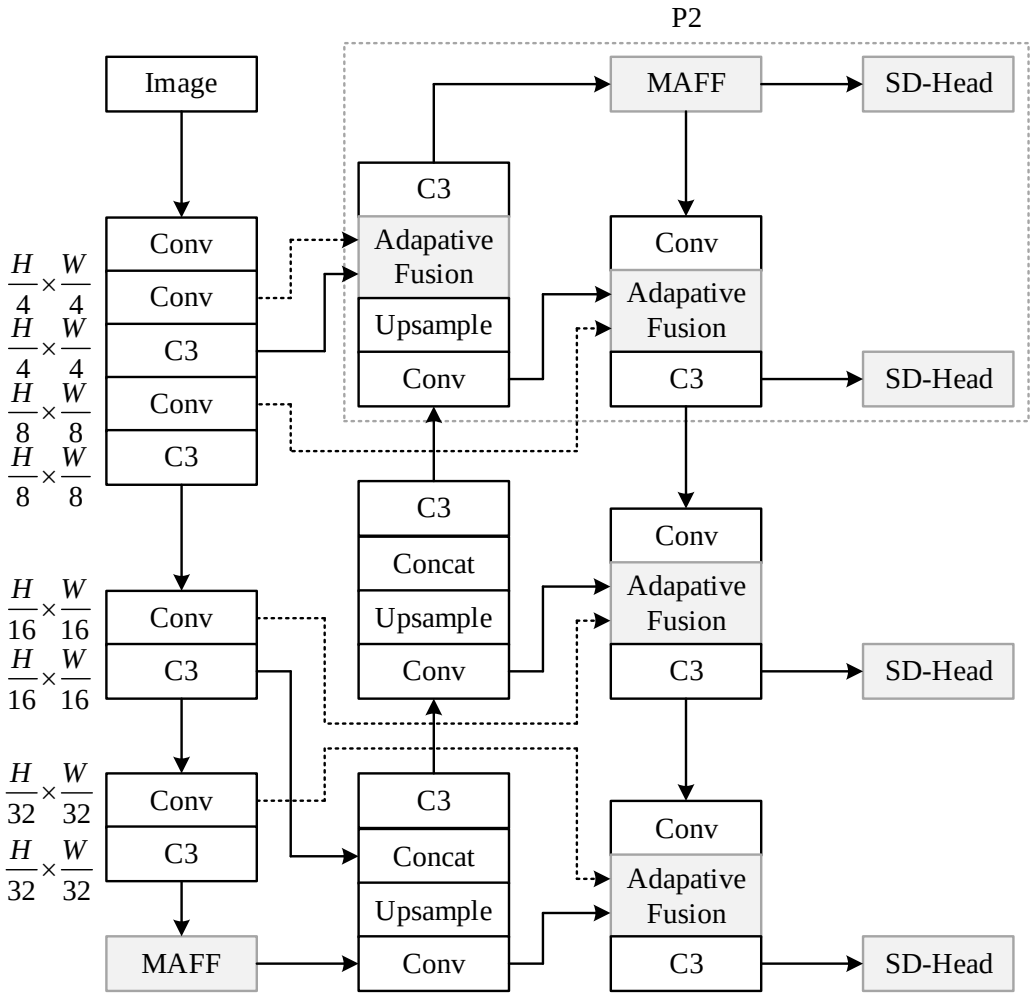


Figure 1: Structure of proposed Drone-YOLO

At the same time, the anchor frame-based model of this paper is more sensitive to the anchor frame setting, and the new P2 detection branch for predictive regression, the size of the anchor frame is set to the small target size obtained from the K-means cluster analysis of the dataset, and the anchor frame setting of each branch is shown in Table 1. In this way, the new P2 branch can reduce the situation that the object is ignored due to the object is too small and the anchor frame is too large, which can effectively alleviate the misdetection and omission due to the anchor frame setting.

Table 1: anchor settings for each detection branch

Test branch	Anchor frame setting
P2	(1,4),(2,9),(5,6)
P3	(5,13),(10,10),(8,20)
P4	(19,17),(15,31),(34,42)
P5	(30,61),(62,45),(59,119)

2.3.2 Multi-level information aggregation networks

The concept of this article is that the shallow feature map, the middle layer, and the deep feature map are combined so that all multi-level information will be preserved simultaneously. On the other hand, in order not to add more parameters to the model and to remain with the same number of channels, an adaptive fusion module is introduced in this paper that not only preserves the same number of channels without adding more parameters but also performs multi-level feature fusion.

The inputs of the module are three feature maps with the same shape and size as well as the number of channels, the size is assumed to be $H \times W \times C$, and its three inputs come from the shallow layer of the network, (X_1), the middle layer, (X_2), and the deep layer, the output from the previous Conv module, (X_3), respectively. There exists a pathway for the shallow feature map to be spliced directly, which can make full use of the shallow information, while the information from the middle layer and the deep layer is fused and then collocated with X_1 after 3×3 convolution, and then finally the channel dimensionality reduction is performed by using the 1×1 convolution. The calculation formula is shown in Eqs. (13)~ (16):

$$W_1 = X_1 \tag{13}$$

$$W_2 = f^{3\times3}(W_1 + X_2) \tag{14}$$

$$W_3 = f^{3\times3}(W_2 + X_3) \tag{15}$$

$$W = f^{1\times1}[W_1 : W_2 : W_3] \tag{16}$$

where $f^{1\times1}$ and $f^{3\times3}$ denote 1×1 convolution and 3×3 convolution operations, respectively. The direct pathway of this module can fully utilize the full shallow information, while the operations of summation and convolution mix the shallow, medium and deep information, and the fusion of multi-level information can improve the detection of small targets.

2.3.3 Multiscale Attention Feature Fusion Module

The ML-FPN framework presents the MAFF (Multi-scale Attention Feature Fusion) module, which is based on the multi-scale attention approach created in this study, to increase the model's sensitivity to tiny objects inside the backbone network and P2 branch networks.

To acquire feature information at various scales, the input feature maps pass through three maximum pooling layers with varying kernel sizes. In order to detect tiny targets, the kernel size at the P2 branch is set to (1,3,5), whilst the kernel size at the backbone network is set to (3,5,9).

The architecture of the MAFF module is given in Figure 2, where the output of the dashed box shows the channel attention weight, which is the proposed multi-scale attention method in

this work. The input for the dashed box comprises the multi-scale feature maps outputted from the three max-pooling layers that get collapsed again through GAP before forming the pass-attention weight with multiscale information by the fully connected layer (MLP) and Sigmoid activation function. The weight thus obtained gets multiplied by the original input feature maps, which are further convoluted with the output of the three max-pooling layers mentioned earlier in co-collapsed form.

This module obtains the multi-scale information with the maximum pooling layers of different kernel sizes, and at the same time, the multi-scale information guides the generation of the attentional weights, which realizes the cross-channel information interaction between the multi-scale features, and can effectively improve the model's attentionality to the small targets.

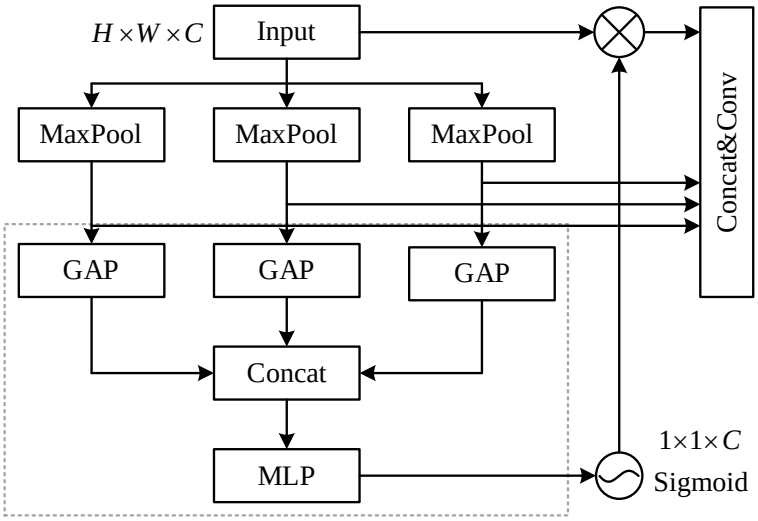


Figure 2: Structure of proposed MAFF module

2.3.4 Streamlined decoupling header

Regression and classification problems clash in target detection.YOLOX demonstrates that separating the classification and regression tasks can enhance the network's detection effect. The classification and regression tasks in the detection head of YOLOv5 are closely tied together with shared weights.

Nevertheless, YOLOX's decoupled head adds multiple additional convolutional layers. In this paper, we redesigned a succinct and effective decoupled head, SD-Head, after carefully weighing the trade-off between accuracy and speed in order to reduce the inference delay brought on by the module while increasing accuracy.

Firstly, the decoupling head's input decreases the number of channels to 256 through 1×1 convolution and subsequently decreases the number of channels to 128 via 3×3 convolution. After the aforementioned reductions, the decoupling head will be split into two separate branches, which will undertake either the classification or the regression task. However, the regression task will be further split into two separate tasks, the one for the position and another for the confidence level. The decoupling of the detection head decreases the bias of predictions arising from differences among the tasks, and as a result, enhances the precision of the model's detection process.

2.3.5 Improved loss function

The model's loss function is often split into three components, which are computed using Equation (17):

$$\mathcal{L} = \mathcal{L}_{obj} + \mathcal{L}_{cls} + \mathcal{L}_{bbox} \tag{17}$$

where $\mathcal{L}_{obj}$ is the loss of object confidence, using a binary cross-entropy loss. $\mathcal{L}_{cls}$ on the other hand is the loss of object categorization, using the cross-entropy loss. And $\mathcal{L}_{bbox}$ is the loss of the position of the prediction box, in this paper, Alpha-IoU is used to optimize the calculation of the loss of the position of the prediction box $\mathcal{L}_{bbox}$, and in this paper, the loss function of the prediction box is calculated as shown in Eq. (18):

$$\mathcal{L}_{bbox} = 1 - IoU^{\alpha} + \frac{\rho^{2\alpha}(b, b^{gt})}{C^{2\alpha}} + (\beta v)^{\alpha} \tag{18}$$

where IoU is the intersection and concurrency ratio of the prediction frame (b) and the true frame (b^{gt}), and α is an adjustable hyperparameter by which the accuracy of the prediction frame can be adjusted by adjusting α. In addition ρ denotes the computation of the Euclidean distance between the centers of the prediction frame and the real frame, C denotes the length of the diagonal of the frame that minimally encloses the prediction frame and the real frame, and v is a parameter that measures the similarity of the aspect ratio of the prediction frame and the real frame, and its computation is shown in Equation (19):

$$v = \frac{4}{\pi}(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h}) \tag{19}$$

where (w^{gt}, h^{gt}) and (w, h) are the widths and heights of the real and predicted boxes, respectively. The β is then a weight, which is calculated as shown in Equation (20):

$$\beta = \frac{v}{(1 - IoU) + v} \tag{20}$$

In this paper's method, the distance between the real frame and the predicted frame, the overlap rate, the aspect ratio, and the scale are all taken into account in calculating the loss of the position of the predicted frame, so that the regression prediction is more accurate, and the model can be converged quickly during the training of the model. Meanwhile, the method in this paper maintains high robustness in small datasets and in the presence of data noise.

3 Experiments and analysis of results

3.1 Experimental setup

3.1.1 Experimental data

This chapter makes use of the VisDrone dataset, which was produced by the Chinese Academy of Sciences' Institute of Automation (CASIA). More than 11,000 video segments totaling more than 240,000 frames are included.The dataset collected data in various scenes, including urban areas, seaside, villages, parks, and nighttime. Due to the perspective of drone photography, the detection scenes in the VisDrone dataset are extremely complex, with significant issues such as severe target occlusion, varying light conditions, and targets that are too small. Additionally, the dataset includes data from different times of day, such as early morning, noon, and night,

which further increases the training difficulty. The dataset is labeled with a total of 540,000 targets with detection and contains 10 common targets to be detected in life, which are van, bus, bicycle, pedestrian, car, people, awning-tricycle,truck,motor and tricycle.It can be used in target detection and tracking tasks.

3.1.2 Algorithm evaluation criteria

(1) Precision and Recall

Precision rate and recall rate are very important indicators in the classification model, and their calculation formula is:

$$Precision = \frac{TP}{TP + FP} \tag{21}$$

$$Recall = \frac{TP}{TP + FN} \tag{22}$$

where TP stands for the value of predicting positively correctly, FP represents the value of predicting negatively incorrectly, and FN stands for the value of predicting negatively while the truth is to predict positively. And so the checking accuracy rate would be the ratio of actual true positive cases among all predicted true cases, and checking completeness rate would be the proportion of predicted positive cases among all the true positive cases in reality. It is clear from the aforementioned calculations that checking accuracy rate and checking completeness rate are incompatible. As a result, both the checking accuracy rate and the checking completeness rate must reach a specific equilibrium level for the system to operate at its best.

(2) Mean Average Precision (MAP) and Average Precision (AP)

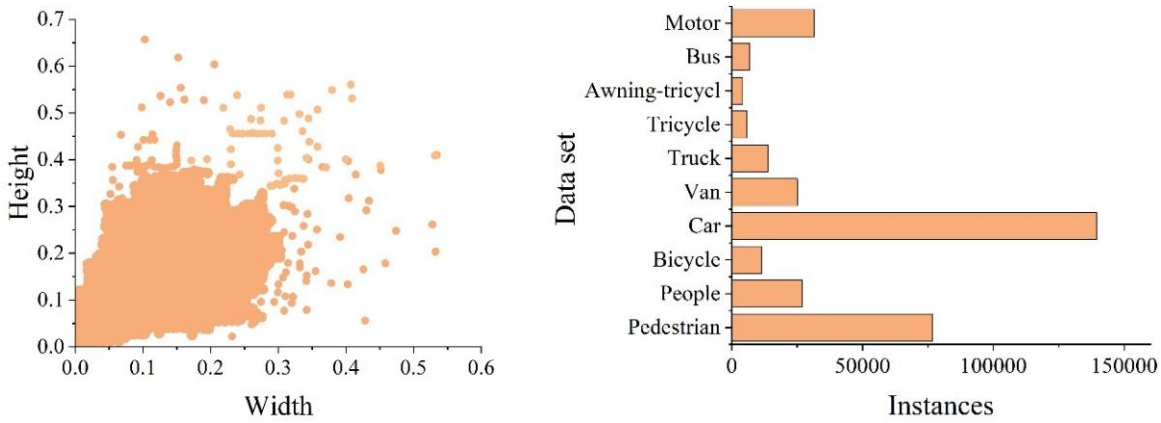
A popular target detection algorithm assessment statistic to assess the algorithm's detection effectiveness across many categories is Mean Average Precision (MAP). The precision-recall curve is used to compute Average Precision (AP). Precision reflects the algorithm's ability to correctly identify positive and negative samples, while recall measures how well the algorithm captures all positive samples. By adjusting the threshold levels, the accuracy and recall graph helps us understand the algorithm's overall performance. To accurately account for the algorithm's overall performance across several classes, the mean average accuracy value is calculated for each class and then averaged.

(3)mAP@.5 and mAP@.5:.95The intersection over union (IoU) threshold between the predicted bounding boxes and the ground truth bounding boxes in the mAP criterion is generally taken as 0.5, 0.75, 0.95, and so on. IoU helps assess the quality of the predicted bounding box; the closer the predicted bounding box is to the ground truth bounding box, the higher will be the IoU value, and vice versa. The mean average precision (mAP) @.5:95 signifies the mean average precision of the model for the IoU (Intersection over Union) range of 0.5 to 0.95 with an interval of 0.05 during object detection tasks. On the other hand, mAP@.5 refers to the mean average precision with an IoU of 0.5.

3.1.3 Analysis of the dataset's targets to be detected

In the VisDrone dataset, the distribution location, size and category of the targets to be detected are counted, and the details are shown in Fig. 3. In the dataset, the targets to be detected are mostly concentrated in the lower center of the image, and there are also some targets to be detected in the rest of the area. The number of small targets are all extremely large, and there are also some larger size targets to be detected in the dataset, with a larger target scale distribution. The number of samples in the categories of tricycle, awning-tricycle, bus and

bicycle is too small, and the sum of the number of samples in these four categories is less than 10% of the total number of samples.



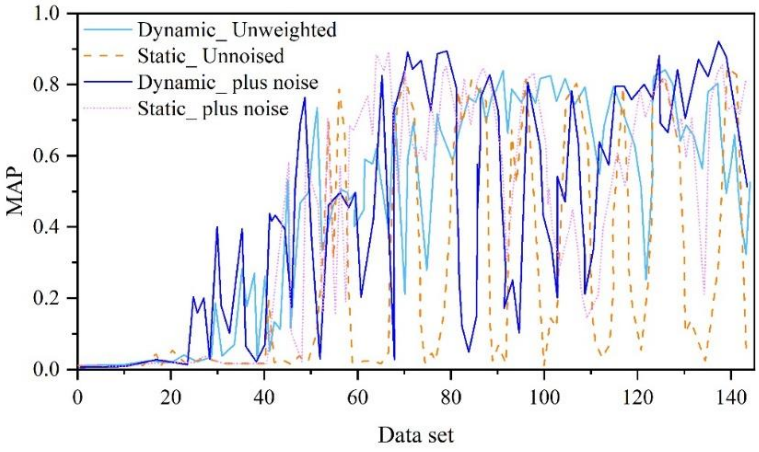
(a) Size dimension (b) Number of tests

Figure 3: Statistics on annotated boxes on the VisDrone training set

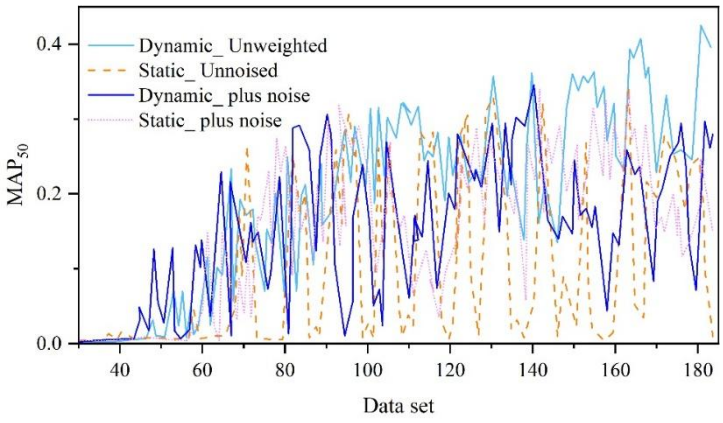
3.2 Data Expansion Based on Background Replacement and Target Noise Addition

3.2.1 Validation results of dynamic and static comparison datasets

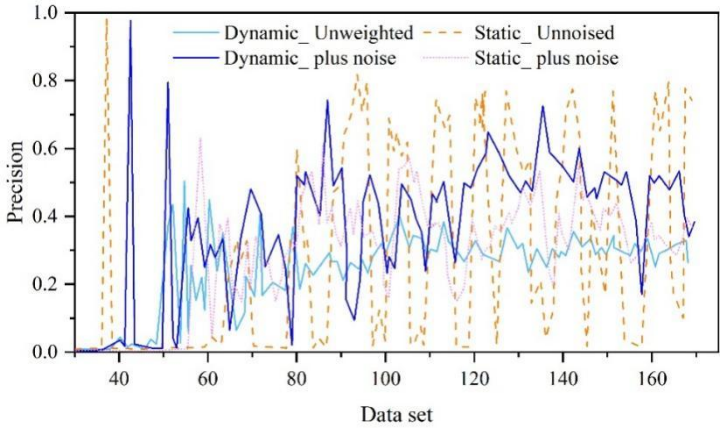
The outcomes from the static data set training and dynamic data set training are illustrated in Fig. 4. It can be noticed that the rate of convergence in the training process is quicker prior to adding noise, whereas the rate of convergence is significantly slower after adding noise since the object detection process brings many uncertainties into the system, resulting in a lower convergence rate during the training process. Also, the rate of convergence of the dynamic data set is relatively higher than that of the static data set.



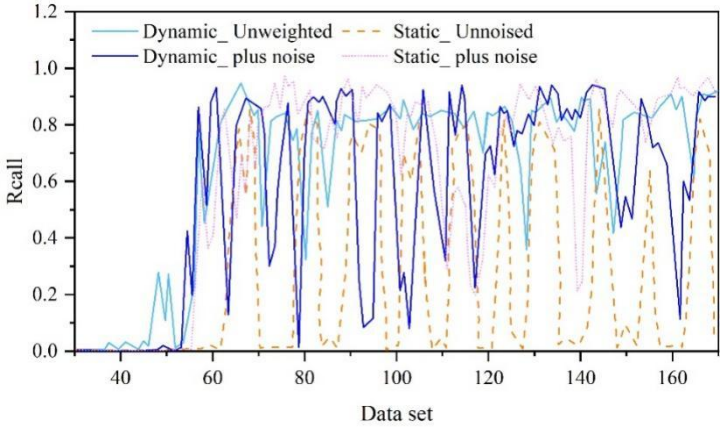
(a) MAP_0.5



(b) MAP_0.5:0.95



(c) Precision



(d) Recall

Figure 4: Training results before and after data set noise

The optimal results of training before and after target noise addition for static and dynamic datasets are shown in Table 2. It can be found that the AP of the static training set is significantly smaller than that of the dynamic dataset, but the value of AP_{50} is not much different, which proves that whether or not the target around the target is in motion mainly increases the position detection accuracy of the target. AP_{50} improves more after target noise addition, but the AP of

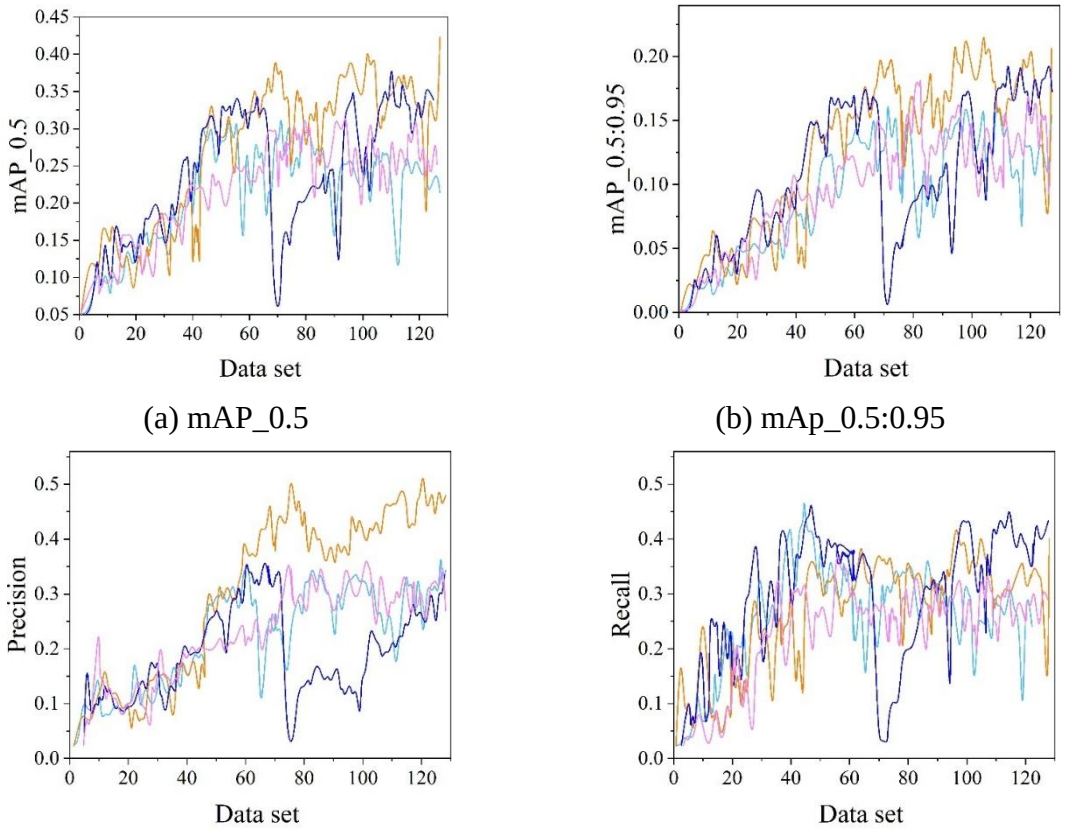
the dynamic dataset decreases, which indicates that target noise addition can improve the detection accuracy to a certain extent, but too much noise may not be conducive to the accurate localization of the target. The AP_{50} of the dynamic training set after noise addition is lower than the AP of the original dynamic dataset, which may be due to the larger background transformations in the static target, proving that the rich background is favorable for overcoming the negative samples and detecting the target more accurately.

Table 2: Dynamic and static data set target noise before and after verification data

Type	Static training set		Static training set	
	Target before noise	Target after noise	Target before noise	Target after noise
AP	37.6%	40.5%	49.7%	42%
AP_{50}	87.1%	94.7%	88.4%	91.5%
P	84.5%	82.5%	54.6%	66%
R	88.5%	97%	98.2%	99.5%

3.2.2 Validation results for small target detection dataset

Figure 5 displays the experimental findings on the tiny target detection dataset. The YOLOv5 training result is represented by the light blue line; the training result following target noise addition is represented by the dark blue line; the training result following background substitution is represented by the orange color; and the training result following both background substitution and target noise addition is represented by the pink line.



(a) mAP_0.5 (b) mAp_0.5:0.95

(c) Precision (d) Recall

Figure 5: Data augmentation experiment training results

The results of the data expansion experiments are shown in Table 3. It can be found that when not running the data expansion algorithm proposed in this paper or running only one data expansion algorithm, the variation of the training results is larger, while running two data expansion algorithms at the same time the training results are more stable, which proves that the data expansion algorithm has a certain effect of stabilizing the detection model.

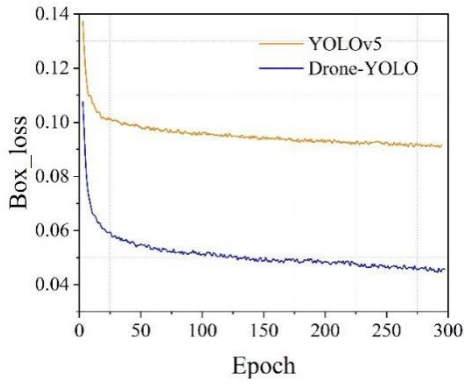
Table 3: Results of data expansion experiment

Background replacement	Targeting the noise	P	R	AP_{50}	AP
No	No	0.44	0.56	38.3%	21.3%
Yes	No	0.63	0.51	46.2%	26.9%
No	Yes	0.44	0.55	42.5%	22.2%
Yes	Yes	0.47	0.47	40.8%	24.6%

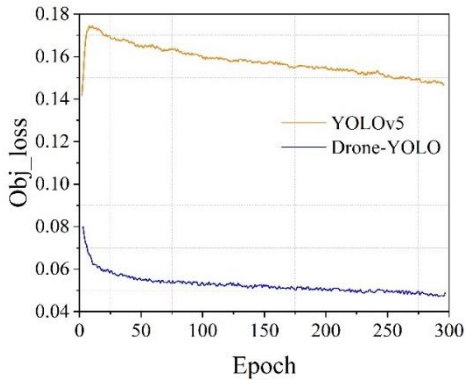
3.3 Performance Evaluation of VisDrone2019 Dataset

3.3.1 Convergence analysis

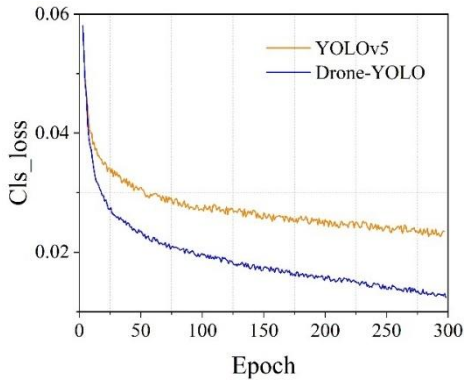
In comparison to the NWPU VHR-10 data set, the scale of the sample on the VisDrone data set is comparatively small, hence increasing the difficulty level of training.Comparison graphs for the convergence curves of YOLOv5 and Drone-YOLO on the VisDrone2019 Training Set and Validation Set are provided in Figure 6. In this figure, blue represents the convergence curve of YOLOv5, whereas the orange curve indicates Drone-YOLO.Confusion analysis of both the graphs provides the indication that the box loss, obj loss, and cls loss of Drone-YOLO are much less than that of YOLOv5.



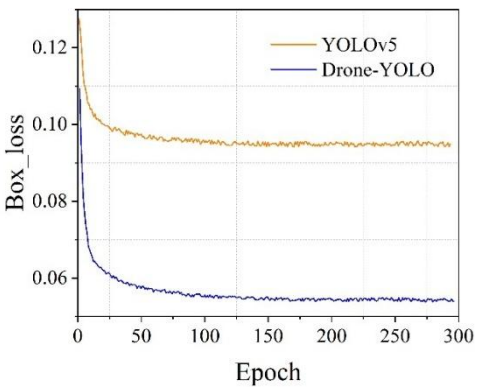
(a) Training set box_loss comparison



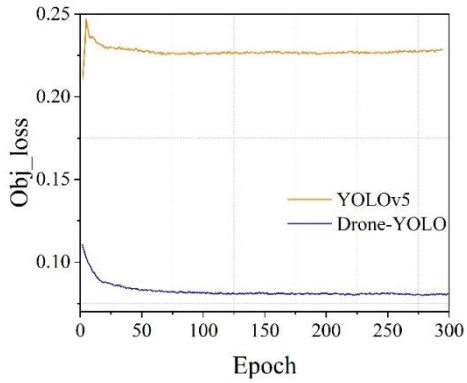
(b) Training obj_loss comparison



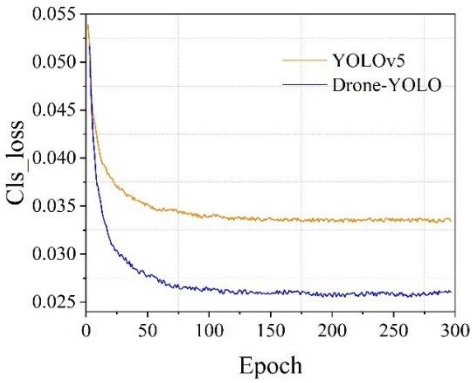
(c) Training set cls_loss comparison



(d) Validation set box loss comparison



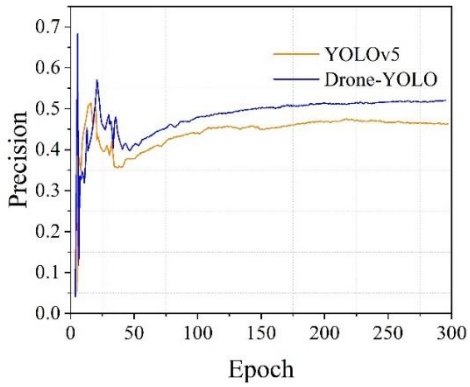
(e) Verification set obj_loss comparison



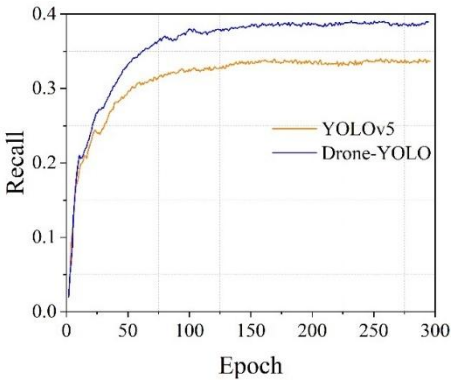
(f) Validation set cls_loss Comparison

Figure 6: Convergence curve on visdrone

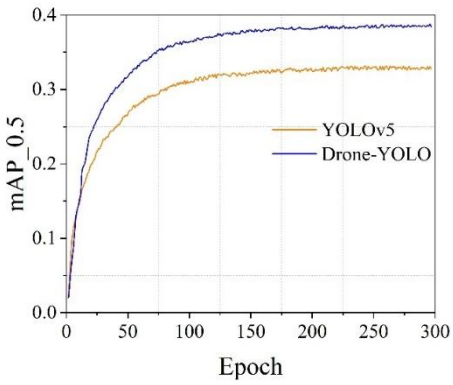
Figure 7 compares the evaluation metric curves of the two models. It can be observed that the precision of Drone-YOLO reached 51.5%, which is an improvement of 5.6% over the original model. Meanwhile, the improved recall rate increased from 34.2% to 39.6%, mAP@.5 improved from 34.5% to 39.8%, and mAP@.5:.95 also increased from 18.3% to 23.1%. It can be concluded that the improved detection model Drone-YOLO based on YOLOv5 has better convergence performance.



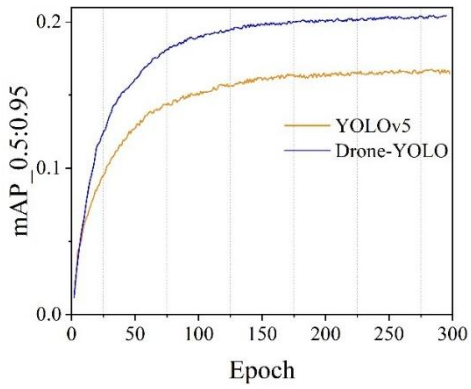
(a) Precision comparison



(b) Recall comparison



(c) mAP@.5 Comparison



(d) mAP@.5:.95 Comparison

Figure 7: Performance metrics on visdrone

3.3.2 Comparison of classification accuracy

Figures 8 and 9 display the confusion matrices produced by YOLOv5 and Drone-YOLO on the VisDrone dataset, respectively. These two figures demonstrate that:

(1) Through optimization, the classification accuracy of Drone-YOLO is improved in all categories in the training results. Among all the categories, the accuracy of BUS is most significantly improved by 12%, CAR by 15%, and MOTOR by 10%.

(2) One of the crucial elements impacting the detection output of the initial YOLOv5 is the occurrence of many detections which have not been made. By implementing the optimized model named Drone-YOLO, there is a reduction in the likelihood of categorizing all the classes into the background. In order from the least to the greatest amount of improvement, the minimum amount of reduction in detection was achieved in detecting the following: PEOPLE, BICYCLE, and PEDESTRIAN, while the others showed an improvement of at least 11% and the maximum for TRICYCLE.

(3) The number of samples of different categories in VisDrone2019 varies greatly, which leads to a large difference in their classification accuracies, the car category with the largest number of samples has an accuracy of up to 83%, followed by pedestrian, bus, and motor, with accuracies of 47%, 45%, and 44%, and the lowest accuracy is in the lowest number of samples in the awning- tricycle category with the lowest precision of only 10%.

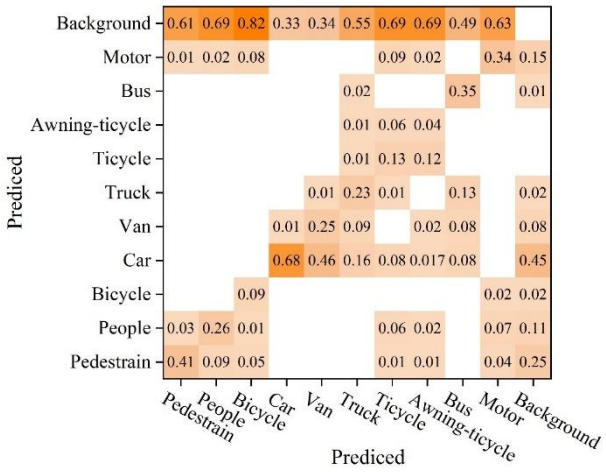


Figure 8: Confusion matrix generated by YOLOv5 at visdrone2019

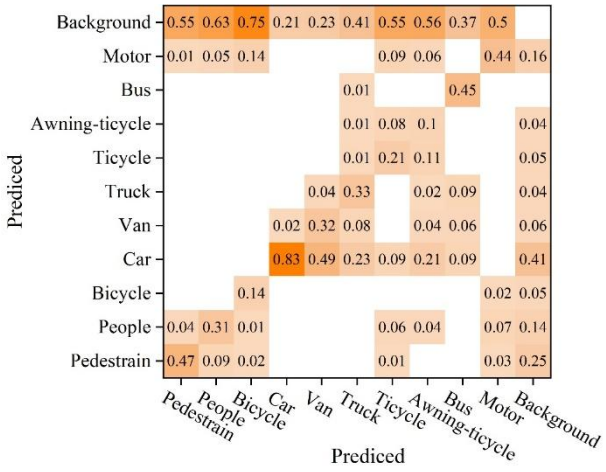


Figure 9: Confusion matrix generated by Drone-YOLO at visdrone2019

3.3.3 Comparison experiments

In the first place, we analyze the performance of Drone-YOLO compared with the YOLOv5 baseline method, with the results provided in Table 4 based on test data of Visdrone dataset. As the above performance analysis table shows, it is evident that Drone-YOLO enhances the key metric mAP@0.5 by 8.9%, which proves that the Drone-YOLO framework designed in this chapter effectively solves the problem of low detection accuracy of YOLOv5 model on aerial imagery. Meanwhile, the mAP@0.5:0.95 metric enhanced 4.8%, the precision P improved 3.9%, the recall R was raised by 6.6%, but the computational complexity and parameter amount rose up correspondingly, causing some drop in detection speed. All in all, HPD-YOLO satisfies the research needs for precision detection models presented in this chapter.

Table 4: Result comparison between Drone-YOLO and YOLOv5

Method	Map_0.5(%)	Map_0.5:0.95(%)	P(%)	R(%)	FLOPs/G	Params/M	FPS
YOLOv5	33.6	19.8	47.3	36.1	29.6	12.5	158.4
Drone-YOLO	42.5	24.6	51.2	42.7	55.9	16.11	96.2

Using the testing dataset of VisDrone, a comparison analysis is carried out between the proposed Drone-YOLO model and certain detection models with a different structure from the proposed one together with different versions of YOLO. The models that are part of the comparison analysis include YOLOv5, TPH-YOLOv5, YOLOv10m, RT-DETR-R18, MVT, and YOLOv11. Table 5 displays the comparison study's findings.

The table shows that the Drone-YOLO network model suggested in this study has a mAP0.5 of 53.7%, which is superior than the Drone-YOLO model in all indices and has a notable advantage over the most recent YOLOv11s. Even compared with YOLOv10 with m-architecture and RT-DETR-R18 based on DETR architecture, Drone-YOLO still achieves better results with smaller number of parameters and computation, and although slightly lower than MVT with Transformer architecture, the number of parameters of Drone-YOLO is significantly lower, which indicates more efficient computation.

Table 5: Performance comparison of different algorithms

Models	mAP_0.5(%)	mAP_0.5:0.95(%)	FLOPs/G	Params/M
YOLOv5	39.4	22.4	16.7	8.2
TPH-YOLOv5	47.5	29.6	146.2	63.5
MVT	51.8	30.4	-	58.8
YOLOv10m	45.3	28.5	59.1	17.4
YOLOv11s	42.5	27.3	23.7	10.8
RT-DETR-R18	48.1	29.5	59.0	20.7
Drone-YOLO	53.7	31.4	55.8	16.3

3.4 Visualization of VisDrone2019 dataset detection results

Various scenes were selected to compare the detection results of the algorithms before and after improvements. The first set of shooting scenes are park scenes, while the image contents mainly include small targets. From the results of the detections, it is observed that the detection model proposed Drone-YOLO, which is based on the YOLOv5, can greatly improve the accuracy of detecting small targets in comparison to the original YOLOv5 model. It shows that the improved model can perform better in terms of stability and precision for detecting small targets;

the second and third shooting scenes are nighttime scenes with poor illumination conditions. Under this condition, the target might lose some detail information due to lack of sufficient lighting. Thus, the requirement of extracting features of the model becomes relatively higher. From the results of the detections, it is noted that the improved YOLOv5 model can achieve better results when detecting the targets under dark-light conditions. Its detection capability is far superior to that of the original model in terms of detection rate and confidence degree. In the fourth group, the shooting environment is a high altitude lane environment, the image resolution is low and some of the vehicles in the image are blocked by trees, the original YOLOv5 model misses several targets on the top side of the lane due to blockage and low target resolution, while the Drone-YOLO model successfully detects these targets and the confidence level is significantly higher than that of the original model in general.

From the results of the experiment, the Drone-YOLO model shows superiority to the initial model not only in the detection of small objects but also in detecting objects at night as well as in complex scenes, which indicates its comprehensive superiority and application value through comparative analysis.

4 Conclusion

In order to address the defects of the current target detection model in UAVs which has poor detection effects on small targets, and to solve the issue of missed and incorrect detections, we propose an improved model based on the YOLOv5 detection algorithm called Drone-YOLO. From the experimental results conducted on the UAV VisDrone dataset, we can see that the accuracy comparison has been carried out according to different types of objects, and the results indicate that there have been improvements made to the precision of detecting any type when comparing to YOLOv5; furthermore, this paper’s Drone-YOLO detection model has exceeded other detection models on small targets. With multiple performance metrics improved, the model presented in this paper is better than others in terms of small targets detection and has increased by 8.9% mAP_0.5 on UAV VisDrone dataset.

Future research will continue to focus on optimizing the model's network structure and reducing the model's size as much as feasible without compromising its accuracy. In the meantime, additional UAVs and scenarios might be added to the dataset, improving its platform for further research.

About the Author

Jingping Guo (born December 1979), a Han Chinese woman from Jilin City, Jilin Province, holds a bachelor's degree and a master's degree. She is a lecturer specializing in big data applications, drone applications, and drone countermeasures.

References

[1] Rahman, M. F. F., Fan, S., Zhang, Y., & Chen, L. (2021). A comparative study on application of unmanned aerial vehicle systems in agriculture. Agric.

[2] Li, Y., Liu, M., & Jiang, D. (2022). Application of unmanned aerial vehicles in logistics: a literature review. Sustainability, 14(21), 14473.

[3] Manfreda, S., McCabe, M. F., Miller, P. E., Lucas, R., Pajuelo Madrigal, V., Mallinis,

G., ... & Toth, B. (2018). On the use of unmanned aerial systems for environmental monitoring. Remote sensing, 10(4), 641.

[4] Hamissi, A., & Dhraief, A. (2023). A survey on the unmanned aircraft system traffic management. ACM Computing Surveys, 56(3), 1-37.

[5] Lyu, M., Zhao, Y., Huang, C., & Huang, H. (2023). Unmanned aerial vehicles for search and rescue: A survey. Remote Sensing, 15(13), 3266.

[6] Lim, Y., Ramasamy, S., Gardi, A., Kistan, T., & Sabatini, R. (2018). Cognitive human-machine interfaces and interactions for unmanned aircraft. Journal of Intelligent & Robotic Systems, 91(3), 755-774.

[7] Kang, X., Song, B., Guo, J., Qin, Z., & Yu, F. R. (2022). Task-oriented image transmission for scene classification in unmanned aerial systems. IEEE Transactions on Communications, 70(8), 5181-5192.

[8] Cano, E., Horton, R., Liljegren, C., & Bulanon, D. M. (2017). Comparison of small unmanned aerial vehicles performance using image processing. Journal of Imaging, 3(1), 4.

[9] Yu, H., Li, G., Zhang, W., Huang, Q., Du, D., Tian, Q., & Sebe, N. (2020). The unmanned aerial vehicle benchmark: Object detection, tracking and baseline. International Journal of Computer Vision, 128(5), 1141-1159.

[10] Ramachandran, A., & Sangaiah, A. K. (2021). A review on object detection in unmanned aerial vehicle surveillance. International Journal of Cognitive Computing in Engineering.

[11] Laghari, A. A., Jumani, A. K., Laghari, R. A., Li, H., Karim, S., & Khan, A. A. (2024). Unmanned aerial vehicles advances in object detection and communication security review. Cognitive Robotics, 4, 128-141.

[12] Abba, S., Bizi, A. M., Lee, J. A., Bakouri, S., & Crespo, M. L. (2024). Real-time object detection, tracking, and monitoring framework for security surveillance systems. Heliyon, 10(15).

[13] Xiao, P., Wang, C., Zhu, L., Xu, W., Jin, Y., & Zhu, R. (2024). An Efficient and Accurate Quality Inspection Model for Steel Scraps Based on Dense Small-Target Detection. Processes, 12(8), 1700.

[14] Iftikhar, S., Asim, M., Zhang, Z., Muthanna, A., Chen, J., El-Affendi, M., ... & Abd El-Latif, A. A. (2023). Target detection and recognition for traffic congestion in smart cities using deep learning-enabled UAVs: A review and analysis. Applied sciences, 13(6), 3995.

[15] Chen, H., Min, B. W., & Zhang, H. (2024). A study on a target detection model for autonomous driving tasks. IET Image Processing, 18(12), 3447-3459.

[16] Wang, S., Mamelak, A. N., Adolphs, R., & Rutishauser, U. (2018). Encoding of target detection during visual search by single neurons in the human brain. Current Biology, 28(13), 2058-2069.

[17] Wang, H., Peng, J., & Yue, S. (2018). A directionally selective small target motion detecting visual neural network in cluttered backgrounds. IEEE transactions on cybernetics, 50(4), 1541-1555.

[18] Gao, F., Wang, C., & Li, C. (2020). A combined object detection method with application to pedestrian detection. IEEE Access, 8, 194457-194465.

[19] Jurevičius, R., Goranin, N., Janulevičius, J., Nugaras, J., Suzdalev, I., & Lapusinskij, A. (2019). Method for real time face recognition application in unmanned aerial vehicles. Aviation, 23(2), 65-70.

[20] Cai, H., Song, Z., Xu, J., Xiong, Z., & Xie, Y. (2022). CUDM: a combined UAV detection model based on video abnormal behavior. Sensors, 22(23), 9469.

[21] Zhang, D., Watson, R., Dobie, G., MacLeod, C., Khan, A., & Pierce, G. (2020). Quantifying impacts on remote photogrammetric inspection using unmanned aerial vehicles. Engineering Structures, 209, 109940.

[22] Munir, A., Siddiqui, A. J., & Anwar, S. (2024). Investigation of uav detection in images with complex backgrounds and rainy artifacts. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 221-230).

[23] Gao, Y., Hou, R., Gao, Q., & Hou, Y. (2021). A fast and accurate few-shot detector for objects with fewer pixels in drone image. Electronics, 10(7), 783.

[24] Jones, L. R., Elmore, J. A., Krishnan, B. S., Samiappan, S., Evans, K. O., Pfeiffer, M. B., ... & Iglay, R. B. (2023). Controllable factors affecting accuracy and precision of human identification of animals from drone imagery. Ecosphere, 14(9), e4657.

[25] Nam, D., & Yeom, S. (2020). Moving vehicle detection and drone velocity estimation with a moving drone. International Journal of Fuzzy Logic and Intelligent Systems, 20(1), 43-51.

[26] Lanča, L., Mališa, M., Jakac, K., & Ivić, S. (2025). Optimal Flight Speed and Height Parameters for Computer Vision Detection in UAV Search. Drones, 9(9), 595.

[27] Zhao, J., Zhang, X., Gao, C., Qiu, X., Tian, Y., Zhu, Y., & Cao, W. (2019). Rapid mosaicking of unmanned aerial vehicle (UAV) images for crop growth monitoring using the SIFT algorithm. Remote Sensing, 11(10), 1226.

[28] Chu, H., Zhang, D., Shao, Y., Chang, Z., Guo, Y., & Zhang, N. (2018, November). Using HOG descriptors and UAV for crop pest monitoring. In 2018 Chinese Automation Congress (CAC) (pp. 1516-1519). IEEE.

[29] Zhang, Y., Wu, C., Zhang, T., Liu, Y., & Zheng, Y. (2023). Self-attention guidance and multiscale feature fusion-based UAV image object detection. IEEE Geoscience and Remote Sensing Letters, 20, 1-5.

[30] Tian, G., Liu, J., & Yang, W. (2021). A dual neural network for object detection in UAV images. Neurocomputing, 443, 292-301.

[31] Radovic, M., Adarkwa, O., & Wang, Q. (2017). Object recognition in aerial images using convolutional neural networks. Journal of Imaging, 3(2), 21.

[32] Lu, W., Lan, C., Niu, C., Liu, W., Lyu, L., Shi, Q., & Wang, S. (2023). A CNN-transformer hybrid model based on CSWin transformer for UAV image object detection. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 16, 1211-1231.

[33] Avola, D., Cinque, L., Diko, A., Fagioli, A., Foresti, G. L., Mecca, A., ... & Piciarelli, C. (2021). MS-Faster R-CNN: Multi-stream backbone for improved Faster R-CNN object detection and aerial tracking from UAV images. Remote Sensing, 13(9), 1670.

[34] Jawaharlalnehru, A., Sambandham, T., Sekar, V., Ravikumar, D., Loganathan, V., Kannadasan, R., ... & Alzamil, Z. S. (2022). Target object detection from Unmanned Aerial Vehicle (UAV) images based on improved YOLO algorithm. Electronics, 11(15), 2343.

[35] Liu, X., & Zhang, Z. (2021). A Vision-Based Target Detection, Tracking, and Positioning Algorithm for Unmanned Aerial Vehicle. Wireless Communications and Mobile Computing, 2021(1), 5565589.

[36] Ren, X., Zhang, Z., Yang, F., & Cheng, W. (2024, November). An Unmanned Aerial Vehicle Image Object Detection Algorithm Based on Improved YOLOv8. In 2024 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML) (pp. 1166-1172). IEEE.

[37] Lu, P., Ding, Y., & Wang, C. (2021). Multi-small target detection and tracking based on improved YOLO and SIFT for drones. International Journal of Innovative Computing, Information and Control, 17(1), 205-224.

[38] Zhao, H., Zhang, H., & Zhao, Y. (2023). Yolov7-sea: Object detection of maritime uav images based on improved yolov7. In Proceedings of the IEEE/CVF winter conference on applications of computer vision (pp. 233-238).

[39] Zhang, J., Wan, G., Jiang, M., Lu, G., Tao, X., & Huang, Z. (2023). Small object detection in UAV image based on improved YOLOv5. Systems Science & Control Engineering, 11(1), 2247082.

[40] Wang, X., He, N., Hong, C., Wang, Q., & Chen, M. (2023). Improved YOLOX-X based UAV aerial photography object detection algorithm. Image and Vision Computing, 135, 104697.

[41] Lyu, Y., Zhang, T., Li, X., Liu, A., & Shi, G. (2025). LightUAV-YOLO: a lightweight object detection model for unmanned aerial vehicle image. J. Supercomput., 81(1), 105.

[42] Kaleem, Z. (2024). Lightweight and computationally efficient YOLO for rogue UAV detection in complex backgrounds. IEEE Transactions on Aerospace and Electronic Systems.

[43] Wang, X., He, N., Hong, C., Sun, F., Han, W., & Wang, Q. (2023). Yolo-erf: lightweight object detector for uav aerial images. Multimedia Systems, 29(6), 3329-3339.

[44] Zhao, X., Xia, Y., Zhang, W., Zheng, C., & Zhang, Z. (2023). YOLO-ViT-based method for unmanned aerial vehicle infrared vehicle target detection. Remote Sensing, 15(15), 3778.

[45] Zhao, L., Liang, G., Hu, Y., Xi, Y., Ning, F., & He, Z. (2024). YOLO-RLDW: An algorithm for object detection in aerial images under complex backgrounds. IEEE Access, 12, 128677-128693.

[46] Koay, H. V., Chuah, J. H., Chow, C. O., Chang, Y. L., & Yong, K. K. (2021). YOLO-RTUAV: Towards real-time vehicle detection through aerial images with low-cost edge devices. Remote Sensing, 13(21), 4196.

[47] Qiu, Q., & Lau, D. (2023). Real-time detection of cracks in tiled sidewalks using YOLO-based method applied to unmanned aerial vehicle (UAV) images. Automation in Construction, 147, 104745.

[48] Weng, S., Wang, H., Wang, J., Xu, C., & Zhang, E. (2025). YOLO-SRMX: A Lightweight Model for Real-Time Object Detection on Unmanned Aerial Vehicles. Remote Sensing, 17(13), 2313.

[49] Afrah Almansoori, Mostafa Al Emran & Khaled Shaalan. (2025). Determinants of Users' Cybersecurity Behavior in the Metaverse: A Deep Learning-Based Hybrid SEM-ANN Approach. International Journal of Human–Computer Interaction,41(17),11116-11133.

[50] Ahmed Chantoufi,Aziz Derouich,Said Mahfoud,Najib El Ouanjli,Abderrahman El Idrissi,Shimaa A. Hussien & Mohamed I. Mosaad. (2025). Enhancing direct torque control of doubly fed induction motor in electric vehicle using artificial neural networks. Scientific Reports,15(1),32094-32094.