IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 26 February 2024, accepted 13 March 2024, date of publication 18 March 2024, date of current version 25 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3378248

RESEARCH ARTICLE

Drone-TOOD: A Lightweight Task-Aligned Object Detection Algorithm for Vehicle Detection in UAV Images

KAITONG OU, CHAOJUN DONG, XIANKUN LIU, YIKUI ZHAI, (Senior Member, IEEE), YE LI, WANXIA HUANG, WENKANG QIU, YIZHI WANG, AND CHENGXUAN WANG

Facility of Intelligence Manufacture, Wuyi University, Jiangmen, Guangdong 529000, China

Corresponding author: Chaojun Dong (1697445576@qq.com)

This work was supported by the 2017 Provincial Science and Technology Plan Fund Project of Guangdong Province of China under Grant 2017A010101019.

ABSTRACT Vehicle object detection using UAV images is a crucial undertaking in urban traffic management and the advancement of autonomous driving technologies. Conventional networks fail to achieve accuary in detecting vehicle objects from a drone's perspective due to the significant variations in the size of the target items, unequal distribution of their positions in the image, and image degradation induced by the drone's movement.In order to surmount this challenge, this research suggests an enhanced TOOD object detection model named Drone-TOOD. The first proposal is to create a lightweight network skeleton called CSPRegNet by merging CSPblock with Regblock. Secondly, we incorporate Regblock into CSPPAFPN to enhance CSPRegPAFPN and incorporate EVCblock at the upsampling location of deep features to capture corner area details and minimize the degradation of feature information. In addition, a efficient task decomposition attention module is also proposed to enhance the interaction ability of positioning and classification tasks. This task decomposition module can highlight the characteristics of a specific task while retaining the characteristics of another task, thereby improving detection capabilities. Experiments conducted on the Drone Vision Challenge Benchmark (VisDrone) demonstrate that the enhanced model can obtain superior performance compared to TOOD. The average precision (mAP) achieved by our approach is 64%, surpassing TOOD by 7.9%. The frames per second (FPS) stayed constant at 27.2. Drone-TOOD demonstrates superior performance compared to other lightweight models on the VisDrone-2021 dataset. In order to demonstrate the robustness of our approach, we additionally performed ablation experiments and conducted tests on the UAV Detection and Tracking Dataset (UAVDT), resulting in an achieved mean average precision (mAP) of 64.6%. Furthermore, Drone-TOOD possesses a total of parameter approximately 21.7 M.

INDEX TERMS Vehicle detection, task align, real-time object detection, lightweight network.

I. INTRODUCTION

UAV technology is advancing rapidly and its use is increasing in military and civilian fields. UAV aerial photography is advantageous due to its lightweight, quick, cost-effective, and user-friendly nature, allowing for the easy gathering of high-precision aerial photos and minimizing expenses associated with materials and labor. Drone technology has increasingly become prevalent in different industries such as transportation, security, logistics, photography, surveying, and mapping, serving as a crucial tool for humans to understand the world and gather environmental information [1].

In the field of vehicle object detection, drone vehicle object detection technology has gradually become a new hot spot in research [2]. Due to the single shooting angle and geographical location of traditional surveillance cameras, it is difficult to obtain more comprehensive detection in complex scenes and severe occlusion situations. Unlike traditional

The associate editor coordinating the review of this manuscript and approving it for publication was Zhongyi Guo.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

surveillance cameras, Compared with surveillance cameras, the flexibility and wide shooting angle of drones can well make up for the shortcomings of traditional surveillance cameras. Currently, the task of UAV object detection technology is faced with the problems of limited resources on the airborne platform, rapid motion changes, and perspective specificity, which lead to image degradation, uneven object distribution, and real-time detection requirements. These problems Bringing major challenges to UAV vehicle object detection technology [3].

UAV vehicle object detection techniques based on deep learning can be separated into one-stage and two-stage categories in recent years due to the rapid growth of deep learning. The improved precision of the two-stage algorithm is one of its advantages. The object detection problem is divided into two steps by this kind of algorithm. First, it creates candidate areas, then uses those areas to do regression and classification. Its drawback is that the detection speed is comparatively slow because it produces a lot of candidate areas, which adds to the computation and time required. The R-CNN [4] series is a representation of the two-stage detector. As the first deep neural network model to use CNN for target detection, R-CNN is considered the founder of the R-CNN series. Selective search is used to extract the region proposals, and then a linear SVM classifier is used to forecast the locations of items within each region and determine the category of objects. Despite the significant advancements made by RCNN, its excessive amount of overlapping region proposals leads to an abundance of redundant feature calculations, causing a considerable decrease in detection speed. In response to the slow detection speed of R-CNN, K. He et al. proposed Spatial Pyramid Pooling Networks (SPPNet) [5]. Although the detection speed has improved, it is still a two-stage type during training. And SPPNet only fine-tunes the fully connected layer and ignores other layers, so there are still shortcomings. Fast-RCNN [6] subsequently proposed by R. Girshick et al. combines the advantages of R-CNN and SPPNet, but its detection speed is still limited by the generated candidate box area. The Faster-RCNN, introduced by S. Ren et al., improves upon the Fast-RCNN model by reducing the time needed to produce candidate box areas for target identification. This is achieved by the introduction of the Region Proposal Network (RPN). Nevertheless, there is computational redundancy in later calculations. Later, further researchers devised multiple enhancement techniques using Faster-RCNN [6] to address the aforementioned issues. Some of the strategies encompass Feature Pyramid Networks(FPN) [7], RFCN [8], Light head RCNN, and others. They expanded the components of the two-stage detector and improved its efficiency. Since the detection speed of the Two-stage detector is slow and it is difficult to achieve real-time detection tasks, the one-stage detector came into being. The advantages of the one-stage algorithm are fast speed and good real-time performance. This type of algorithm treats the target detection problem as a single regression problem and directly outputs the category and location information of the target. However, due to the simplified detection process, the disadvantage is that the accuracy is low and it is easy to miss detections and false detections. One-stage is represented by the YOLO series. YOLO was proposed by Redmon et al. [9]. It is the first one-stage detector in the deep learning era. This algorithm completely abandons the detection paradigm of proposal detection and verification, and instead applies a single neural network to the entire image, segmenting the image into multiple regions while predicting bounding boxes and probabilities for each region. Subsequently, Redmon and Farhadi implemented a sequence of enhancements to YOLO, resulting in the development of YOLOv2 [10] and YOLOv3 [11]. These versions not only increased the accuracy of object recognition but also preserved a high level of detection speed. Despite significant advancements in the accuracy of the YOLO series object detection system, its ability to accurately locate small targets remains subpar. The Single Shot MultiBox Detector (SSD) [12], proposed by Liu et al., is a one-stage target detector that incorporates multi-reference and multi-resolution detection technologies. This advancement significantly enhances the accuracy of the one-stage target detector, particularly for detecting small object target. The detection accuracy of the one-stage detector has been improved to a certain extent after continuous improvement, but it is still lower than that of the two-stage object detector. T.-Y. Lin et al. found the main reason that onestage detectors are less accurate than two-stage objects was the extreme foreground-background hierarchy imbalance encountered during dense detector training (extreme foreground-background class imbalance).For this reason, RetinaNet incorporates a novel loss function called Focal Loss [13] to address this issue. By modifying the usual cross-entropy loss function, the detector allocates greater emphasis to data that are challenging to categorize throughout the training phase. The utilization of Focal Loss enables a one-stage detector to attain precision that is on par with a two-stage detector, all while preserving an exceedingly rapid detection speed. Recently, several advanced one-stage detectors, including CenterNet [14], YOLOv4 [15], TOOD [16], and ObjectBox [17]have been developed thanks to the contributions of numerous scholars. These target detectors have progressively attained a level of performance that is comparable to two-stage target detectors. Hence, to strike a compromise between the speed and accuracy of object identification on UAVs with limited processing capacity, opting for a one-stage target detector is more effective than a Two-stage target detector.

The aforementioned conventional approaches exhibit subpar efficacy in detecting vehicle targets in UAV imagery. Consequently, in recent years, certain researchers have undertaken focused research and developed models on UAV perspective object detection technologies. The rapid development of large language models (LLM) and visual language models (VLM) in recent years has expanded the ability of

UAV to solve complex tasks, De Curt ò et al [18] proposed a zero sample drone scene literary text description application that combines LLM and VLM with the ability of drones to provide real-time vision and high-level data throughput. This technology can be further used to solve complex tasks. Zhong et al. [19] proposed a robust and reliable autonomous planning system for intelligent quadcopters to deal with the dangers of dynamic obstacles to drone flight. By combining tracking and trajectory prediction of dynamic obstacles, a more reliable real-time obstacle avoidance planning system can be achieved than existing methods. At the same time, this system, combined with LLM, makes human-machine interaction more feasible. Muzammul et al. [20] enhanced the detection of small-scale objects by integrating Slicing Aided Hyper Inference (SAHI). This integration not only improves the detection accuracy of the model but also opens up new ways for advanced image analysis in UAV applications. Yu et al. [21] proposed a multi-level micro vehicle detection framework (MTVD) for mid- and high-altitude drone images based on visual attention and spatiotemporal information, by using a segmentation network to extract road areas in the image and utilizing the attention mechanism. Improve the RSS algorithm with spatiotemporal information technology to suppress the impact of complex backgrounds on object detection and reduce false detections. Momin et al. [22] proposed a lightweight algorithm model that is feasible with limited computing resources. This algorithm model is based on YOLOv4-Tiny by using three prediction boxes and adding the second layer and the third layer to the backbone network. Three layers of output image resolution are used to increase the algorithm's detection accuracy of small targets in the data set. Shen et al. [23] designed corresponding anchor frames according to the size of vehicle targets in the drone's perspective, and used a branch structure to design a cost-effective stem block. Finally, a 1×1 volume was added to each stage block. To enhance small target feature extraction, this improved method is applied to Fast-RCNN, which effectively reduces detection time and improves detection accuracy. Luo et al. [24] introduced asymmetric convolution and an improved SPP module to the residual blocks of the upper, middle, and lower layers of the YOLOv5 backbone network to reduce the computational complexity while maintaining the original receptive field. Finally, in the Focus module, an attention mechanism module called ICEA is introduced to assist the network in emphasizing important features while suppressing irrelevant features. This improvement is beneficial to suppressing interference caused by complex backgrounds and achieving UAV vehicle target detection accuracy and speed. balance. On the basis of YOLOv5, Liu et al. [25] proposed a feature enhancement module called FCblock to solve the problem of a large number of small and dense targets and complex background interference in high-altitude photography. into adaptive weights, and then assign the weights to shallow feature maps to improve feature extraction of small targets, and then integrate FEBlock into spatial pyramid pooling (SPP) to generate enhanced spatial pyramid pooling. Secondly, the self-feature extended version (SCEP) is proposed to further improve the network's feature extraction capability, and finally a shallower detection layer is added to the large, medium and small detection layers to improve the network's detection ability of medium and small targets. Li et al. [26] proposed a lightweight rotating object detection algorithm. Aiming at the problem that traditional algorithms do not consider the diversity of vehicle scales in UAV images and cannot obtain rotation angle information, a method was introduced. A circular smooth label (CSL) angle classification method makes it suitable for detection scenarios based on rotating boxes, and the Cascaded Swin Transformer Block (STrB) is used to reduce the computational complexity in the feature fusion process in the backbone network, further enhancing semantic information and global perception capabilities of small objects. Finally a feature enhanced attention module (FEAM) is proposed to improve the utilization of detailed information through local feature self-supervision. Although the vehicle object detection task from the perspective of UAV has attracted the attention and research of many scholars and achieved many results, due to the problems of UAV angle specificity, limited airborne platform and uneven distribution of image samples, UAV Perspective vehicle object detection technology still requires further research to achieve a balance between higher detection accuracy and detection speed.

To address the issues of target detection in UAV photos in real-world circumstances, we initially break down the target detection algorithm into two tasks: classification and location. Prior target identification algorithms employed a coupled head structure to accomplish the tasks of categorization and location. This structure is susceptible to generating a significant quantity of parameters and processing resources, as well as overfitting. Consequently, in recent times, numerous academics have started utilizing decoupled head structures to accomplish classification and positioning duties while developing target detection algorithms. This structure can efficiently decrease the number of parameters and computations and improve its capacity to generalize and its robustness. However, this structure still has certain issues. Due to the interdependence of classification and positioning tasks, processing them individually can lead to misalignment and thus decrease the effectiveness of the detector. Feng C et al. proposed TOOD, which effectively solves the problem of task misalignment using decoupled head detectors by interactively processing classification and positioning tasks. Therefore, in order to solve the problem of different target sizes in the drone's perspective, uneven distribution of detection objects, misalignment of classification and positioning tasks, and difficulty in feature extraction, we further improved and proposed the Drone-TOOD based on TOOD. Our main contributions can be summarized as follows:

(1) Due to the large number of parameters and calculations in the TOOD model, it is not suitable for real-time

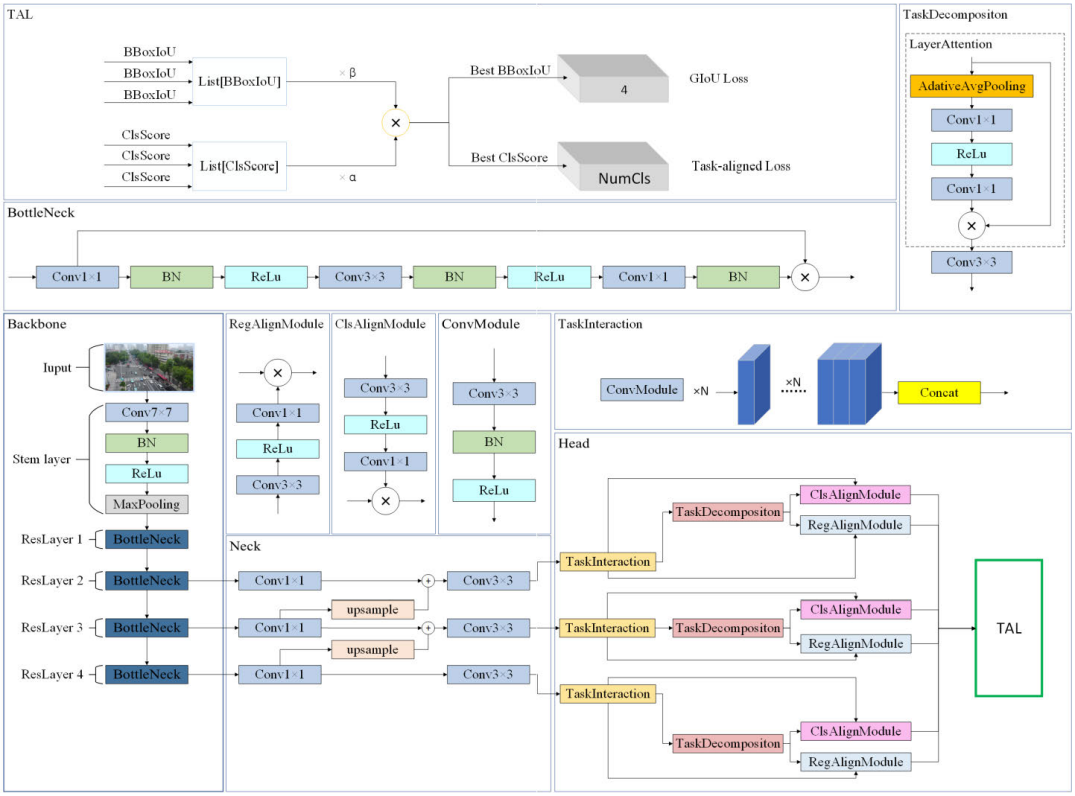

FIGURE 1. The structure of TOOD.

detection when the airborne platform is limited. Hence, this research employs the concepts of RegNet [27] and CSPNet [28] to revamp its primary network, resulting in a new network called CSPRegNet. This redesign aims to strengthen the backbone network's ability to extract features, decrease computing complexity, and enhance both detection accuracy and speed. CSPRegNet integrates the design space concept of RegNet with the cross-stage local network structure of CSPNet. The RegNet design space concept is utilized to determine the ideal network depth, network width, and network component modules. These parameters are then combined with the CSPNet network structure to enhance computational efficiency and improve detection performance, especially in scenarios with restricted computing resources.

(2) In order to address the issues of missed detection and false detection in unmanned aerial vehicle (UAV) imagery in dense vehicle scenarios, we employ the same building blocks as CSPRegNet and combine them with PAFPN [29] to obtain a more powerful and efficient feature extraction capability called CSPRegPAFPN. In addition, the principle of the Explicit Visual Center(EVC) [30] we introduced is to obtain explicit visual center information from the features in the deepest layer to adjust the shallow features, thereby improving the dense object detection capability. Therefore, we introduced EVC into CSPRegPAFPN to improve the detection of the detector in dense vehicle scenes.

(3) In the TOOD network structure, due to the interaction of single-designed task features, it is inevitable to introduce certain feature conflicts in the two tasks of classification and positioning. Therefore, the head network TAP uses the Layer Attention module to encourage the decomposition of task features, so that the classification and positioning tasks can pay more attention to the specific features of their own tasks after interacting with the features. However, the higher computational complexity of this module affects the detection efficiency. Therefore, we designed the Efficient Task Decomposition Attention (ETDA) module to replace it, which can effectively reduce the computational complexity and enhance task decomposition capabilities to improve detection efficiency.

Experiments show that the detection accuracy and speed of our method on VisDrone and UAVDT are effectively improved. The improvement of the TOOD model.

II. RESEARCH METHODS

This section describes the TOOD network structure, the design of CSPRegNet, the design of CSPRegPAFPN, the addition of EVCblock, and the design of ETD-Attention.

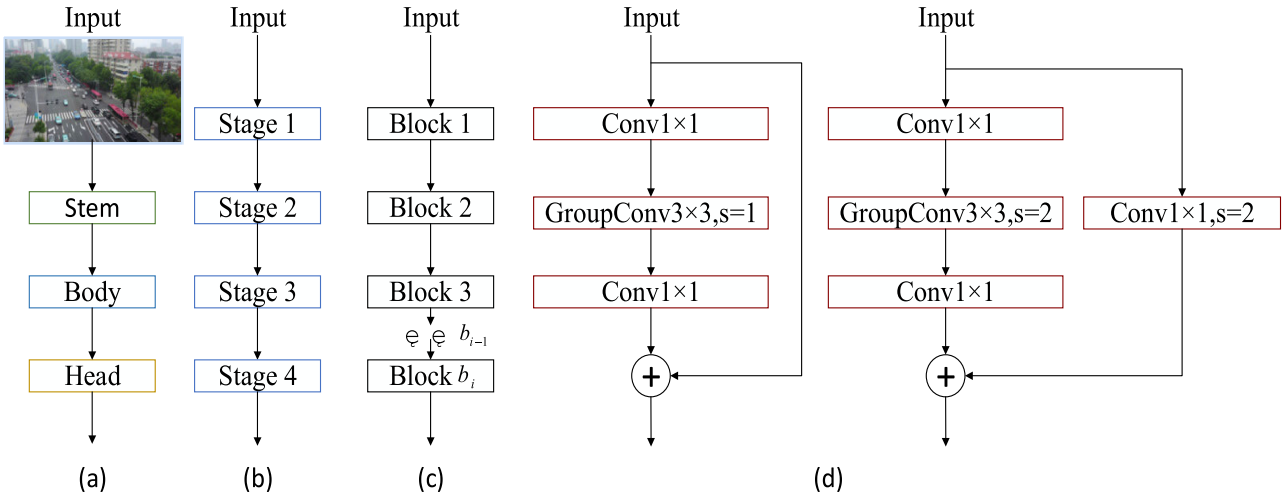

FIGURE 2. The space design structure of RegNet.

A. TOOD STRUCTURE

TOOD is a one-stage detection network that uses task alignment learning, task alignment indicators, and task alignment loss to solve the problem of inconsistent feature learning between classification tasks and positioning tasks in the existing decoupled head structure detection network.

The TOOD model architecture consists of four main components: input, backbone network, neck network, and head network. The network structure of the TOOD is illustrated in Figure 1.

B. CONCEPTS AND PRINCIPLES OF CSPREGNET DESIGN

The backbone network uses ResNet-50 [31] for feature extraction, the neck network uses FPN for feature fusion, and the head uses TAP to output the target prediction. The detection method begins by feeding the image into ResNet-50, which produces four feature maps of varying sizes. Next, the feature maps generated by the second, third, and fourth layers are chosen and fed into the FPN to combine their features and improve the network's capacity to recognize objects at different scales. Prior to being inputted into TAP, the Task Interaction module is utilized for the purpose of task feature interaction and decomposition, subsequently followed by TAP for prediction. Since the task features of classification and positioning tasks have different focuses, TAP uses the Layer Attention hierarchical attention mechanism to dynamically calculate the task features specific to positioning and classification to decompose the tasks to avoid introducing conflicts between the two task features. Finally, task alignment learning (TAL) is used to further guide the head network for task alignment. The detector's task alignment method can achieve good accuracy in UAV vehicle object detection. Nevertheless, the ResNet-50 and Layer Attention modules suffer from reduced efficiency and hinder real-time performance due to their numerous parameters and extensive calculations. Additionally, although the FPN neck network can transmit multi-scale information, it fails to adequately address the issue of incomplete information transmission caused by the significant variation in target object scales from the drone's perspective. Hence, the TOOD detector, which employed for the purpose of detecting targets in drone vehicles, still required additional enhancements. The backbone network is an essential element of the whole object detector. The task of this component is to extract features from the input image by performing convolutional operations, which also help in abstracting the input image. TOOD utilizes ResNet-50 as its backbone network architecture. Despite its impressive feature extraction capabilities, the detector's performance in detection can be enhanced by utilizing a less sophisticated backbone network, owing to the significant number of network parameters and computing demands. We got inspiration from RegNet,CSPNet,and YOLOX [32] and designed the CSPRegNet backbone network. RegNet followed a progressive design method through design space design to find the optimal network model and discovered some general guidelines for network design. First, the initial neural network model was abstracted into three parts: stem, body, and head, as shown in Figure 2 (a). The stem is the input layer used to process different types of input data. The body is the main body of the network and is also called the backbone layer. The backbone layer is generally divided into four stages as shown in Figure 2(b). Each stage is composed of multiple blocks, as shown in Figure 2(c), block is generally the level at which convolution operations are performed, and its structure and parameters have no restrictions, as shown in Figure 2(d).

The head is the output layer, and the output layer structure is adjusted according to different task types and content.RegNet first starts optimizing from the body part. In each stage, the block has four hyperparameters, which are the number of layers of the block d_i, the network width w_i, the bottleneck rate, and the number of groups of grouped

convolutions. According to the range of these four hyperparameters, a large number of experiments were conducted to obtain four basic guidelines for model design. First, introduce a shared bottleneck rate; second, use shared group convolutions in all stages; third, the width of the stage increases as the stage increases; the depth of the stage increases as the stage increases. And increase (excluding the last stage). Continuing in-depth optimization based on the above basic guidelines, we found that the width under each block can be approximately fitted to a linear relationship, but the design space limits all blocks in each stage to use the same width, so the width needs to be quantified. First, the block width is linearly parameterized through the following formula:

$$\mathrm{u}_j = w_0 + w_a \cdot j, 0 \leq j < d \tag{1}$$

This parameterization has four parameters, namely network depth, bottleneck rate j, initial width $\boldsymbol{w}_0$,and straight line slope w_a Then the quantization factor S_j is calculated by the following formula:

$$u_j = w_0 \cdot w_m^{S_j} \tag{2}$$

Round the calculated quantization factor S_j and express it as $\lceil S_j \rfloor$, and recalculate the network width through the following formula:

$$u_j = w_0 \cdot w_m^{\lceil S_j \rfloor} \tag{3}$$

Then put blocks with the same network width into the same stage, so that the width of the i-th stage can be expressed as:

$$w_j = w_0 \cdot w_m^i \tag{4}$$

The number of blocks d_i counts the number of blocks with the same width, and the width w_j in each stage can be expressed as:

$$w_j = w_9 \cdot w_m^i \tag{5}$$

$$d_i = \sum_j 1[\lceil S_j \rfloor = i] \tag{6}$$

In this way, the quantification of width is achieved, where w_0, w_a and $w_{\mathbf{m}}$ can be determined by performing a network search based on the network depth. On the premise that the number of stages is four, we specify the network structure through six parameters d_i, w_0, w_a, w_m, bottleneck rate b and group width g, and generate the width and depth of the block by formulas (2)-(4), thereby obtaining a network structure with better design space.

Therefore, RegNet effectively reduces computational complexity and enhances feature extraction capabilities by optimizing the design of the neural network structure, thereby improving detection speed and accuracy. However, due to the small vehicle targets in UAV images, higher resolution is often required for small object detection and more detailed features for accurate detection, although RegNet can adapt to small targets by increasing the resolution, it will also increase the computational complexity, so we need to further optimize based on RegNet. In order to enhance the feature extraction capabilities of small targets, we combine the design space idea of RegNet with CSPNet.

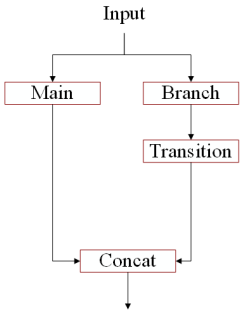

FIGURE 3. The structure of CSP.

The core idea of CSPNet is to divide the feature map into two parts, one part is called the main path (Main), and the other part is called the branch (branch). Its structure is shown in Figure 3.

The main path is responsible for extracting global and abstract features, while the branch path is responsible for extracting local and detailed features. This separated design enables CSPNet to better capture features of different scales and levels, thereby better capturing detailed information about small targets.Many detectors improve feature extraction capabilities and computational efficiency by introducing CSP connections and CSP structures. Therefore, we combined the design space idea of RegNet with the Cross Stage Partial Network structure, obtained CSPRegNet and replaced it with the backbone network of TOOD. The CSRegNet structure is shown in Figure 4.CSPRegNet is divided into a stem layer and four stages. The initial three stages are outfitted with a ConvModule and a CSPRegBottleNeck. In the final stage, the SPPFBottleNeck module is incorporated to enhance the network's receptive field and its capacity to detect tiny objects. The ConvModule class comprises a Conv2d layer, along with Batch Normalize and SiLu [33] activation functions. In CSPRegBottleNeck, we put RegNet-block (Regblock) at the branch position, and perform feature splicing after the feature map is processed by main and branch. Then the spliced features will go through the channel attention mechanism module we introduced, which consisting of an adaptive average pooling consists of a conv and a hardsigmoid, allowing all channels to learn from each other the features learned in main and branch, and finally output the feature map after passing through a ConvModule.CSPRegNet takes full advantage of the advantages of CSPNet and RegNet and can bring powerful feature extraction capabilities and low computing costs.

C. CSPREGPAFPN STRUCTURE OF DRONE-TOOD

Due to the wide area specificity of UAV and the large differences in vehicle target sizes, it is necessary to fuse the feature maps incoming from the backbone network so that the network can obtain multi-scale fusion information and

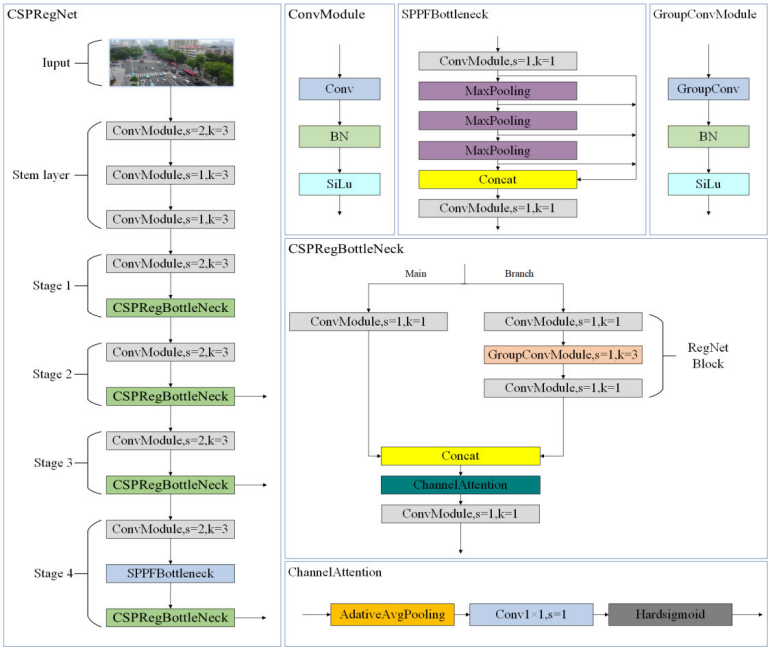

FIGURE 4. The structure of CSPRegNet.

improve the model's adaptability to targets of different sizes. FPN establishes connections for feature maps at different levels using a top-up and bottom-down approach to fuse deep features with shallow features, thereby extracting and utilizing multi-scale features. However, when FPN has large-scale differences, it is difficult to align and integrate the features between the bottom layer and the top layer. The top-down structure prevents the bottom layer features from affecting the top layer features. Therefore, we use the PAN structure to introduce a path aggregation module to solve the above problems.

By adding a bottom-down path to the top-down path of FPN, PAN not only shortens the information propagation path but also utilizes the positioning information of underlying features. Not only that, we use CSPRegBottleNeck in PAN, whose structure is the same as that in CSPRegNet, to enhance network feature fusion capabilities and improve detection performance. We named it RegPAFPN, and its network structure is shown in Figure 5.

D. EVC STRCTURE OF DRONE-TOOD

The feature pyramid is widely used by many object detectors due to its excellent performance. However, many current works only focus on inter-layer feature interaction and ignore intra-layer features. In scenes where small-sized objects are too densely spaced between objects in UAV images, although effective feature interaction can be performed through feature pyramids to obtain rich feature representations or the Vision Transformer method can be used to obtain global context and long-term dependencies these methods suffer from limited contextual information, complex calculations, and neglect of corner areas.

There is limited contextual information, complex calculations, and neglect of corner areas. In order to solve the above problems, we introduce EVC, which captures global long-distance dependencies and local corner areas of the image by using lightweight MLP [34] and a learnable visual center mechanism (LVC) in parallel, as shown in Figure 6 Show.Lightweight MLP is mainly composed of residual blocks based on depth convolution and residual blocks based on channel MLP. Specifically, in the lightweight MLP processing flow, the feature map first undergoes feature smoothing processing through a stem layer to obtain the feature map X_{in} after passed to the deep convolution residual module, in which the deep convolution residual module uses group normalization and depth separable convolution to process the feature map and uses channel scaling operations and drop path operations to improve the generalization of features and robustness capabilities, followed by residual connections. The above process can be expressed as:

$$\tilde{X}_{in} = DConv(GN(X_{in})) + X_{in} \tag{7}$$

After obtaining $\tilde{X}_{in}$ from the depth-based convolutional residual module, it is then passed to the channel MLP-based residual block for group normalization and channel MLP processing. At the same time, the channel scaling operation is used again on the processed feature map and DropPath operations to enhance feature generalization and robustness capabilities, followed by residual connection to finally obtain feature maps with global long-distance dependencies. The above process can be expressed as:

$$MLP(X_{in}) = CMLP(GN(\tilde{X}_{in})) + \tilde{X}_{in} \tag{8}$$

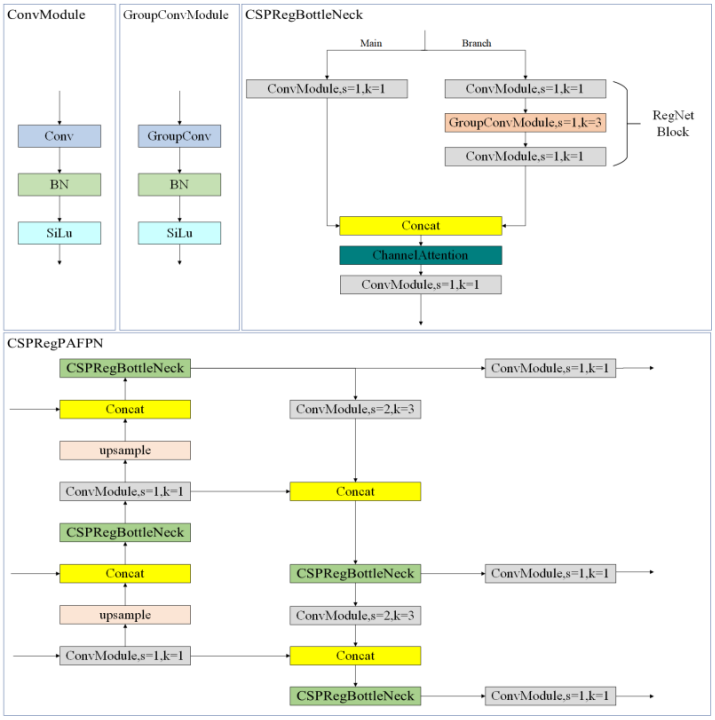

FIGURE 5. The structure of CSPRegPAFPN.

LVC is a coder with an inherent dictionary, which consists of an inherent codebook and a set of scaling factors. In the LVC processing flow, the feature map X_{in} that has been smoothed by the stem layer is first passed into a set of convolutional layers for combined encoding, and then the encoded features are processed by the CBR block and passed into the codebook. A set of scaling factors s is used in the codebook to sequentially map X_i and b_k to the corresponding position information. The information for the k-th codeword can be calculated using the following formula:

$$e_k = \sum_{i=1}^{N} \frac{e^{-S_k||\tilde{X}_i - \mathrm{b}_k||^2}}{\sum_{j=1}^{K} e^{-S_k||\tilde{X}_i - \mathrm{b}_k||^2}} (\tilde{X}_i - \mathrm{b}_k) \tag{9}$$

Then use ϕ to fuse all e_k to obtain the full information about the K codes of the entire image. The specific formula is as follows:

$$e = \sum_{k=1}^{K} \phi(e_k) \tag{10}$$

Then the output of the codebook is passed into a fully connected layer and a 1×1 convolution layer to predict the prominent key class features to obtain the part and corner region features of the scale factor coefficient, and then perform channel multiplication and residual connection with the input features X_{in} to obtain the feature with feature mapping of local corner region information. Finally, the two feature maps processed by lightweight MLP and LVC are connected channel by channel to obtain global long-range dependencies and feature maps that retain local corner area information of the input image as much as possible to complete intra-layer feature adjustment. Since the deep features of the top layer contain global information, we insert the EVC module at the upsampling point in the top-down path in RegPAFPN. The EVC module uses the acquired EVC features to adjust the shallow features by capturing the image's local corner area information and global long-range dependencies. In reality, top-level features are gradually fused by top-down path continuous upsampling; however, this approach will gradually dilute the EVC feature information in the process. Therefore, we adopt a cross-level feature fusion method to directly upsample the EVC features to shallow features and then use 1×1 convolution to reduce the dimension to 256 channels after splicing along the channels, thereby realizing the top-level features to adjust the shallow features across levels so that each layer of the feature pyramid obtains Obtain global but differentiated feature representations., thereby improving the network's feature extraction capabilities in dense objects and thus improving detection accuracy.

E. ETDA STRCTURE OF DRONE-TOOD

In the TAP structure of the TOOD network, Layer-Attention is used to decompose the interactive features of the positioning and classification tasks. Firstly, the task interaction features obtained by stacking the feature maps of each layer passed in from the neck network can be expressed as:

$$\mathrm{X}_{\mathrm{k}}^{\mathrm{inter}} = \begin{cases} \delta(conv_k(X^{FPN})), k=1 \\ \delta(conv_k(X_{k-1}^{\mathrm{inter}})), k>1 \end{cases}, \forall k \in \{1, 2, \ldots\ldots, N\} \tag{11}$$

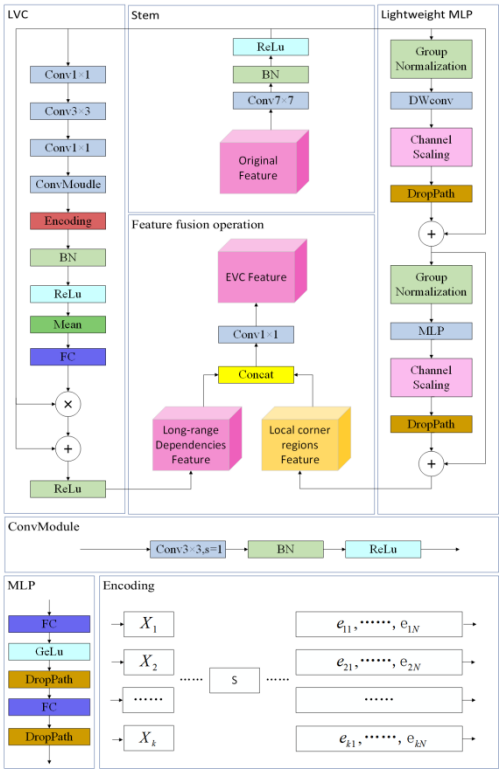

FIGURE 6. The structure of EVCblock.

Among them, $conv_k$ and σ are the convolution layer and activation function, X^{FPN} is the feature map passed in from FPN, and then X_k^{inter} is processed by adaptive average pooling and then passed to two fully connected layers to calculate the specific task feature weights. The process It can be expressed as:

$$w = \sigma(fc_2(\delta(fc_1(x^{\text{inter}})))) \tag{12}$$

Finally, the specific task feature weight and the feature interaction feature map are multiplied channel by channel to obtain the specific task interaction feature map. The process can be expressed as:

$$\mathrm{X}_{\mathrm{k}}^{task} = w_k \cdot \mathrm{X}_{\mathrm{k}}^{inter}, \forall k \in \{1, 2, \ldots\ldots, N\} \tag{13}$$

Although the above method can effectively interact with positioning and classification task information and retain task-specific features, due to the complex calculations in Layer Attention and excessive dimensionality reduction operations, information loss affects detection efficiency and performance. Therefore, we designed ETDA to use To enhance specific task decomposition capabilities and improve detection efficiency, the structure of ETDA is shown in Figure 8.Since the interaction features incoming from the neck network are mixed with classification and positioning information and there is a lot of redundant information in the features, this method first extracts some features from the task interaction feature map for convolution operations and then decomposes them according to specific tasks. The

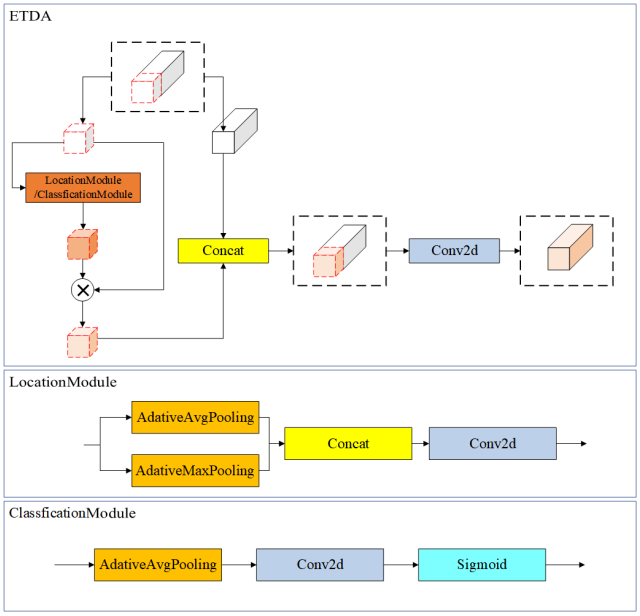

FIGURE 7. The structure of ETDA.

requirement is to obtain the feature weight of a specific task. After splicing the feature map and performing a convolution operation, the feature information of a specific task in the task interaction feature map can be highlighted while still retaining the feature information of another task, thereby task alignment capabilities and improving detection efficiency. The process can be expressed as:

$$\mathrm{X}_{\mathrm{k}}^{partial} = \mathrm{X}_{\mathrm{k}}^{inter} \times \omega \tag{14}$$

Partial channel feature map $X_k^{partial}$ is obtained through factor ω and used for subsequent adaptive task feature weight extraction. The adaptive task feature weight module extracts specific weights from the incoming partial channel feature maps according to specific task requirements to obtain positioning task weight w_L and classification task weight w_c. The specific formula is as follows:

$$w_L = Conv(concat(Avg(X^{partial}), Max(X^{Partial})) \tag{15}$$

$$w_C = \sigma(Conv(Avg(X^{partial})) \tag{16}$$

Finally, specific task weights are assigned to some channel feature maps and spliced with the remaining channel feature maps. After a simple convolution, the feature map X_k^{task} obtains the positioning and classification task weights, thereby enhancing task alignment capabilities and improving detection accuracy. The specific formula is as follows:

$$X_k^{task} = Conv(concat(w_{L/C} \otimes X_k^{partial}, X_k^{residue})) \tag{17}$$

III. EXPERIMENTAL RESULTS AND ANALYSIS

In this work, all experiments in this article were conducted on the Window 10 system. In the hardware configuration, our CPU uses I9-11900K and the GPU uses RTX 3090 (24GB).

We do not use pre-training in the training strategy, and we use AdamW [35] as the training optimizer. The initial learning rate is 0.002, the momentum weight is 0.0002, the weight attenuation coefficient is 0.0001, the number of training batches is 16, the total training is 300 epochs, and there is cosine annealing [36]The learning rate strategy intervenes at 150 epochs.

A. DATASET

All experiments in this article use two public data sets, namely VisDrone-2021 [37] and UAVDT [38]. VisDrone-2021 is a dataset for drone vision benchmark evaluation, that was jointly developed by the Institute of Automation, the Chinese Academy of Sciences (CASIA), and the Chinese University of Hong Kong (CUHK).The pictures in this data set contain most daily scenes, with rich shooting environments, various shooting heights and angles, and complete annotation information. They are of great value and can provide more credible verification for the method proposed in this article. In the object detection challenge, the data set contains 8629 static aerial images, including 6471 training set images, 548 verification sets, and 1610 test sets. These images contain ten categories, namely Car, Truck, Van, Bus, Bicycle, Awning Tricycles, Tricycles, Pedestrians and People. In this article, we only perform detection analysis on four-wheeled motor vehicles, so we remove images that only contain non-four-wheeled motor vehicle categories from the data set to obtain the VisDrone-2021 subset. The VisDrone-2021 subset contains 7982 static aerial images, of which the training set contains 5930 images, the verification set contains 518 images, and the test set contains 1534 images. The detection categories are Car, Truck, Van and Bus. The image resolutions of the VisDrone-2021 subset are distributed from 960×540 to 2000×1500, and our experiments in this dataset unified the input images to 1280×1280.

UAVDT is a vehicle traffic data set based on drone shots. It consists of frame images captured from 100 video sequences. These video sequences were shot by drones in multiple locations, covering common scenes in urban areas. Including squares, main roads, toll booths, highways, intersections, etc.,we selected 8228 static images from it, of which 5760 images were used as the training set, 823 images were used as the training set, and 1645 images were used as the test set. This data The set of images contains three categories, namely Car, Truck and Bus, and the image resolution is 1024×540. Our experiments in this dataset unified the input images into 640×640. Since this data set is mainly used for target tracking tasks and is not specifically designed for target detection, its authoritativeness in vehicle object detection research on actual targets is not as good as VisDrone-2021. Therefore, this data set serves as a supplementary experimental data set for this article. Demonstrate the effectiveness of our proposed method. In addition, we uniformly use Rezise, RandomCrop, and RandomFlip to enhance the input image data to improve model robustness and generalization. The

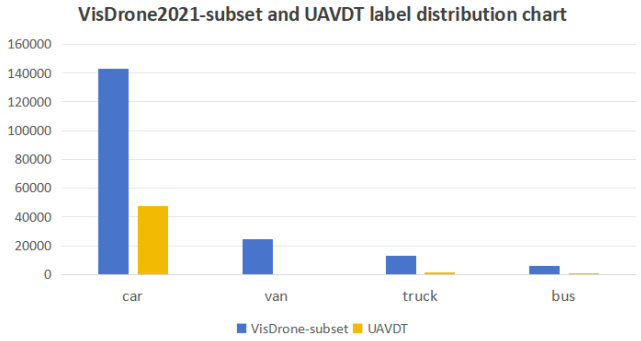

FIGURE 8. Dataset label distribution chart.

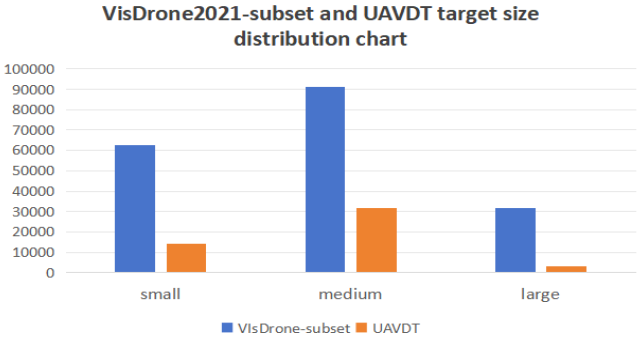

FIGURE 9. Dataset label distribution chart.

distribution of the number of category labels and the distribution of target sizes in the VisDrone-2021 subset and UAVDT are shown in Figure 8 and Figure 9, which truly reflect the overview of actual UAV traffic application scenarios.

Evaluation indiators:

In this experiment, we select the commonly used evaluation indicators in object detection tasks, average precision (AP), average precision average precision mean(mAP). as a metric for our model evaluation. These evaluation metrics are defined as follows:

$$AP = \int_0^1 P(R)\, dR = \sum_{k=1}^{N} P(k)\Delta R(k) \tag{18}$$

$$mAP = \frac{\sum_{i=0}^{n} AP(i)}{n} \tag{19}$$

$$P = \frac{TP}{TP + FP} \tag{20}$$

At the same time, this experiment considers the complexity of the model and the detection efficiency to evaluate the real-time performance of the model. The complexity of the model can be evaluated by the size of the model, the number of parameters, and FLOPs. The detection efficiency of the model can be evaluated by the frames per second(FPS).

B. BACKBONE NETWORK COMPARISON EXPERIMENT

In order to prove the effectiveness of our proposed CSPRegNET, this article conducts comparative experiments

TABLE 1. The backbone network comparison experiment.

Backbone Network	mAP0.5	mAP0.5:0.95	Params	FPS	FLOPs
ResNet-50	56.1%	38.8%	34.1M	21.4	185G
RegNet-400M	53.6%	34.6%	12.5M	26.8	109G
RegNet-800M	56.1%	37.5%	14.4M	26.4	117G
RegNet-1.6G	56.3%	37.7%	16.2M	23.9	133G
RegNet-3.2G	58.5%	39.4%	22.3M	19.8	162G
CSPResNet-50	62.2%	43.4%	28.7M	27.7	113G
CSPRegNet-800M	62.5%	43.8%	12.8M	34.3	85G

between different backbones in the VisDrone-2021 data set. We selected ResNet-50, RegNet-400M, RegNet-800M, RegNet-1.6G, RegNet3.2G, and CSPResNet50, the above networks are derived from our improved baseline network. The comparative experiments are shown in Table 1.Although ResNet-50 can achieve higher accuracy, due to the high computational complexity of its model, it is not suitable for situations where the airborne platform is limited. Compared with ResNet-50, RegNet can achieve better accuracy with lower model complexity as the network depth and width deepen, such as RegNet-800M. With the same accuracy, RegNet-800M has fewer parameters. Compared with ResNet-50, the number of parameters has been reduced by 19.7M and the detection speed has been increased by 5FPS. Therefore, the network structure designed through space is more suitable for UAV image vehicle detection tasks than deeper networks. In order to balance speed and accuracy, we selected RegNet-800M as the benchmark for our subsequent experiments.

In the vehicle object detection task from the drone perspective, the difficulty in extracting small target features has always been a problem for today's target detectors. In order to verify the effectiveness of the cross-stage local network structure in improving small object detection capabilities, we separate the CSPNet structure into a Fusion comparison with ResNet-50 and RegNet-800M.And the CSPRegNet-800M we proposed is still 0.3% more accurate than the CSPResNet-50 with its low parameter quantity and low complexity. Therefore, the CSPRegNet-800M proposed in this article obtains the optimal network depth, network width, and network component modules through the RegNet design space idea, and then combines the cross-stage local network structure to achieve efficient calculation and more efficient calculations under the condition of limited computing resources. Its well detection performance achieves a balance between accuracy and real-time performance. The visual training process of the skeleton network comparison experiment is shown in Figure 10.

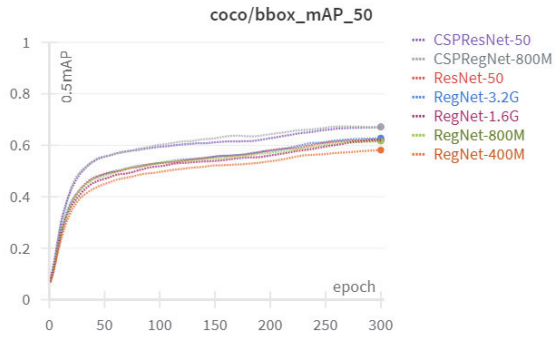

FIGURE 10. Backbone Network comparison experiment visualization.

C. ABLATION EXPERIMENT

The ablation experiments in this article were all conducted on VisDrone-2021 and UAVDT based on CSPRegNet-800M as the skeleton network to fairly verify the effectiveness of the improvement of each part of the module.

According to Table 2, in order to prove the effectiveness of the spatially designed convolution module of the cross-stage local network in PAFPN, we designed an ablation experiment of the neck network to verify that the spatially designed convolution module of the cross-stage local network is used in PAFPN. The enhanced effect on feature fusion. Experimental results show that PAFPN not only shortens the information propagation path by adding a bottom-down path to the FPN top-down path, but also utilizes the positioning information of the underlying features, thereby increasing the feature fusion capability and improving the accuracy by 0.1%. In contrast, CSPRegPAFPN adds a spatially designed convolution module of the cross-stage local network to PAFPN, thus improving the multi-scale feature extraction capability. Due to the increase in the number and complexity of its network model parameters, the detection speed of the neck network structure has slightly decreased, but its detection accuracy has improved by 0.4%.

According to Table 3, this topic introduces the EVC module to solve the problem that many neck networks currently focus on inter-layer features and ignore intra-layer features when performing feature fusion. This often results in limited context information, complex calculations, and corner areas. Neglected problem. In order to ensure the effectiveness of the module in the experiment, the module was inserted at the upsampling operation of the top-up path of the neck

TABLE 2. The neck network comparison experiment.

CSPRegNet800M	PAFPN	CSPRegPAFPN	mAP0.5	mAP0.5:0.95	Params	FPS	FLOPs
√	√		62.6%	42.7%	15.2M	33.6	87G
√		√	63.0%	43.5%	16.2M	30.8	90.7G

TABLE 3. The EVC in different neck networks comparison experiment.

FPN+EVC	PAFPN+EVC	CSPRegPAFPN+EVC	mAP0.5	mAP0.5:0.95	Params	FPS	FLOPs
√			63.2%	42.7%	16.7M	28	100G
	√		63.1%	43.5%	19.0M	27.4	106G
		√	63.3%	43.5%	21.2M	25.6	110G

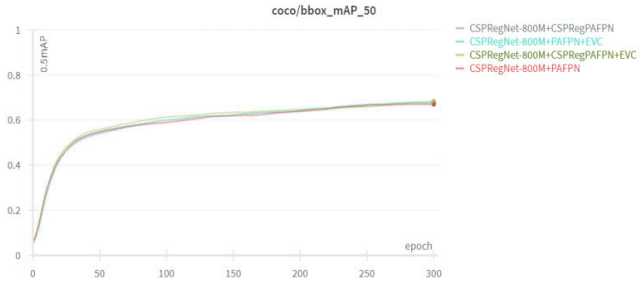

FIGURE 11. Neck Network and EVC comparison experiment visualization.

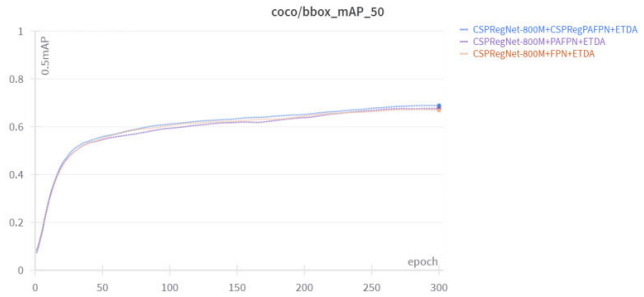

FIGURE 12. ETDA comparison experiment visualization.

network for experiments. The experimental results show that the upsampled feature map is processed by the EVC module to obtain corner area information and global information adjustment and is improved to varying degrees in different neck networks. The EVC module achieves an accuracy of 0.3% based on CSPRegPAFPN. The visual training process of its neck network comparison experiment is shown in Figure 12. In dense scenes, the EVC module can enhance the network's attention to the global and corners of dense targets. At the same time, this module also brings a certain amount of complexity and parameters to the network but still maintaining the balance between precision and speed.

Before the neck network is passed to the head network, the fusion features output by the neck network will interact with task features and then be passed to the head network for task feature decomposition. Since the incoming features of the neck network are sufficiently saturated, the processing of interactive features should pay more attention to the respective specificities of classification and localization task features in the task decomposition process. The visual training process of its ETDA comparison experiment is shown in Figure 12. This paper was designed by ETDA to improve the task decomposition process. In ETDA, we conducted multiple experiments on the factor w of the partial channel feature map. The experiment found that when the factor w is too large, the specific task features affected are too large. When performing convolutional fusion later on, it will weaken the original interactive feature effect, reduce detection accuracy, and increase computational complexity, resulting in a decrease in detection speed. When factor w is too small, the specific task features affected are too small, making it difficult to decompose task interaction features into specific tasks, resulting in task conflict issues and a decrease in detection accuracy. In the end, we selected 0.25 as the optimal value for our factor w and applied it to subsequent experiments. The experimental results are shown in Table 4. The classification and positioning tasks can highlight their task feature information in the interactive features according to their own task feature requirements while still retaining the feature information of another task. The experimental results are shown in Table 5.

In order to visualize the effectiveness of our proposed module, we generate a heat map as shown in Figure 14. We observe that the model using EVC in the neck network can pay more attention to the contour information of the target object than the model without EVC in dense scenes. This enables the model to accurately focus on the target object. At the same time, the ETDA we proposed pays more attention to the target object than the Layer Attention used by the baseline model because we use ETDA to enhance the interactivity of the model's classification and positioning tasks, thereby improving The model's attention to the target object improves detection accuracy, and its heat map is shown in Figure 15.

In addition, in order to further verify that the proposed method makes the network robust, we conducted ablation experiments on the proposed method in the UAVDT data

TABLE 4. The ETDA in different factor comparison experiment.

ω	mAP0.5	mAP0.5:0.95	FPS
0.75	62.3%	42.4%	31.2
0.5	62.6%	42.7%	32.3
0.25	62.8%	42.9%	34.7
0.125	62.5%	42.7%	35.2
0.0625	61.5%	41.7%	35.5

TABLE 5. The ETDA in different neck networks comparison experiment.

FPN+ETDA	PAFPN+ETDA	CSPRegPAFPN+ETDA	mAP0.5	mAP0.5:0.95	Params	FPS	FLOPs
√			62.8%	42.9%	12.4M	34.7	84.9G
	√		63.3%	43.3%	14.6M	34.2	86.4G
		√	63.4%	43.5%	15.8M	31.3	90.4G

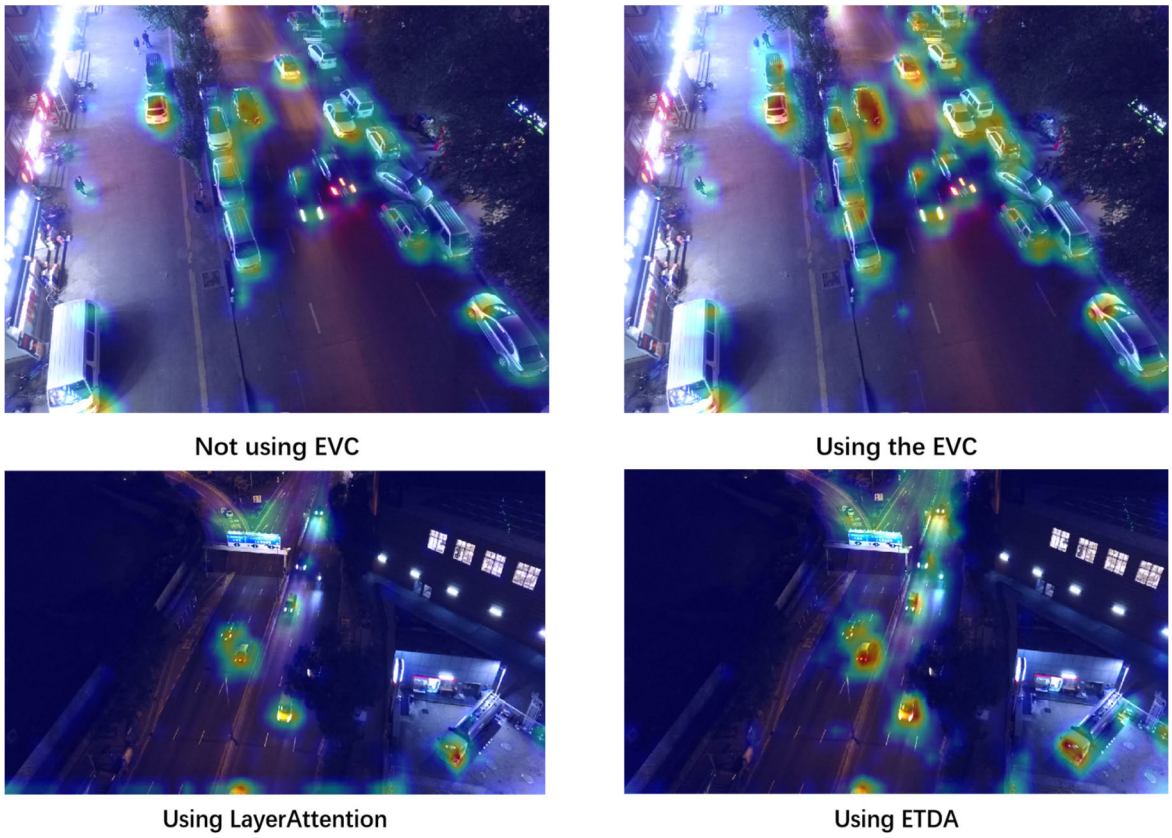

FIGURE 13. EVC and ETDA usage comparison heat map.

set. The experimental results are shown in Table 6. Using CSPRegNet-800M as the baseline network and CSPRegPAFPN as the neck network, the performance compared to the baseline network is improved by 4.2%. There are many target objects of different scales in the drone images. CSPRegPAFPN has many targets of different scales in the drone images. It shows excellent performance on problems where target objects make it difficult to extract features, confirming the effectiveness of CSPRegPAFPN. At the same time, adding the EVC module to the upsampling position in the CSPRegPAFPN bottom-up path improves the network's ability to extract dense target features, improving 4.3% compared to the baseline network performance. As the experiment in the VisDrone-2021 data set, this

TABLE 6. Ablation experiments of different modules in UAVDT.

CSPRegNet-800M	CSPReg PAFPN	EVC	ETDA	mAP0.5	mAP0.5:0.95	Params	FPS	FLOPs
√				59.9%	42.5%	12.4M	39.3	20.3G
√	√			64%	45.7%	15.8M	35.6	21.6G
√	√	√		64.1%	46.7%	21.2M	34	25.6G
√	√		√	64.3%	43.4%	16.2M	37.2	20.6G
√	√	√	√	64.6%	47%	21.6M	36.6	25.2G

TABLE 7. Performance of different models on VisDrone dataset.

Models	mAP0.5	mAP0.5:0.95	Params	FPS	Inference time	FLOPs	Memory consumption
Faster-RCNN	55.3%	34.2%	41.3M	24.4	40.9ms	191G	572MB
Cascade-RCNN[39]	56.7%	39.2%	69.3M	21.3	46.9ms	219G	681MB
CenterNet	49.4%	-	32.3M	33	30.3ms	184G	254MB
FCOS[40]	50.5%	-	32.2M	28.5	35ms	184G	387MB
EfficientDet-b0[41]	50.1%	-	3.8M	33.3	30ms	13.8G	488MB
ATSS[42]	51.6%	-	32.3M	28.6	34.9ms	189G	387MB
PPYOLOEPlus-m[43]	41.2%	26.2%	22.7M	23.4	42.7ms	99.8G	263MB
YOLOv5-s	57.6%	38.3%	7M	40	25ms	31.7G	233MB
YOLOv5-m	61.9%	42.6%	20.8M	27.6	36.2ms	96.5G	249MB
YOLOX-s	61.8%	40.7%	8.9M	51	19.6ms	53.2G	263MB
ObjectBox	57.3%	36.7%	86.1M	17	58.8ms	203.7G	582MB
YOLOv6-s[44]	62.7%	43%	17.2M	31.5	31.7ms	87.53G	365MB
YOLOv7-tiny[45]	49.7%	33%	26.2M	46	21.7ms	26.25G	242MB
YOLOv7-l	59.1%	39.2%	36.5M	22	45.4ms	207G	534MB
YOLOv8-nano	57%	38.5%	2.3M	77.1	12.9ms	15.2G	52MB
YOLOv8-s	63.4%	43.1%	8.6M	35	28.5ms	52.7G	140MB
DINO[46]	59.4%	39.3%	47.5M	12.9	77.5ms	252G	550MB
TOOD	56.1%	38.8%	34.1M	21.4	46.7ms	185G	495MB
Ours-Drone-TOOD	64.0%	43.5%	21.7M	27.2	36.7ms	106G	286MB

module brings certain The model complexity and number of parameters increase, but the balance between accuracy and speed is still maintained, confirming the effectiveness of this module in improving network performance in vehicle detection tasks in UAV images. Finally, we verified the effectiveness of our proposed ETDA in improving network performance on UAVDT. Experiments show that ETDA enhances specific task features in the task interaction feature map, while maintaining the feature information of another task, improving task alignment capabilities, and increasing detector performance. The ETDA implemented in the CSPRegPAFPN improves performance by 0.2% with a marginal increase of 0.4 million parameters. The decrease in model complexity improves FPS and confirms the module's effectiveness in improving the accuracy of the detection network.

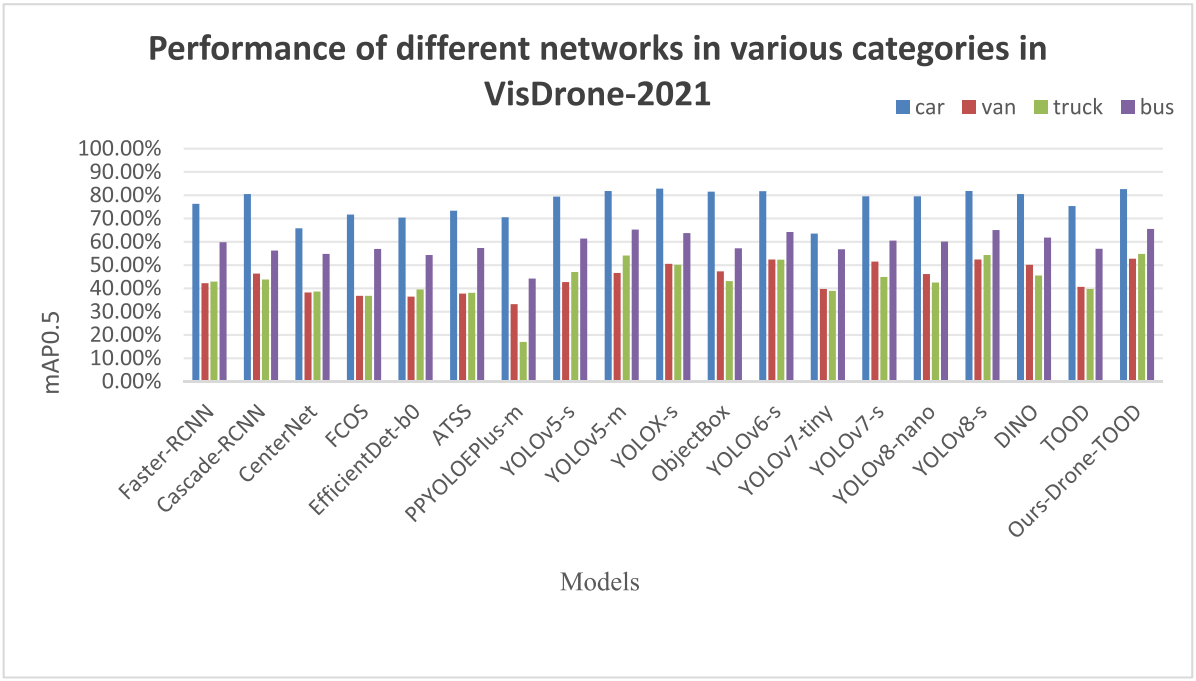

FIGURE 14. Performance of different networks in various categories in VisDrone-2021.

FIGURE 15. Comparison of environment with occlusion and uneven illumination object detection result.

D. COMPARATIVE EXPERIMENT

This article compares Drone-TOOD with the YOLO target detector in recent years, other lightweight detectors, and classic target detectors, mainly to evaluate the speed and detection accuracy of the algorithm. This comparison experiment uses a network model similar in size to the network model proposed in this article to ensure the fairness of this comparison experiment. The experimental results are shown in Table 7. We chose VisDrone-2021 as the comparative experimental data set and adopted consistent data augmentation methods, image input size, and learning strategies. At the same time, we also compared it with some of the current excellent object detectors. Even if their model sizes are larger than the detectors proposed in this article, it can better prove that our detector can match the detection performance of these high-parameter detectors under the premise of being lightweight. The performance remains on par or even higher, highlighting the advantages of the detector proposed in this paper.

In Table 7, we can see that the algorithm performance of the YOLOv5 series is relatively low. Although its number of parameters is smaller than Drone-TOOD, its speed and model complexity are still higher than ours, and its detection performance is still lower than our proposed Drone-TOOD. In addition, although YOLOv6s is slightly higher than the YOLOv5 series detectors in low complexity and situations, it is not ideal for small object detection, and due to the use of decoupling heads, the classification and positioning tasks cannot be aligned, so the overall detection performance Still lower than Drone-TOOD, where we used the task alignment method. Compared with the previous YOLO series,YOLOv7-tiny and YOLOv8-nano perform better in small object detection capabilities, and the detection speed is

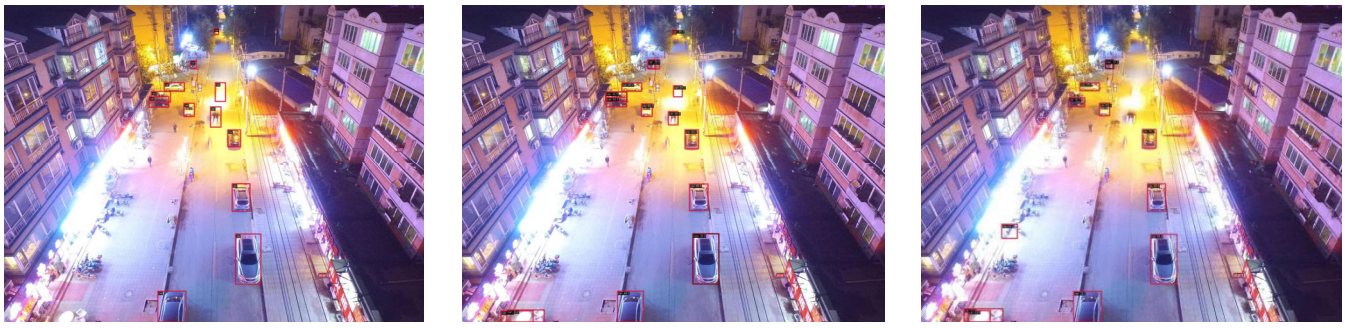

FIGURE 16. Comparison of environment with poor lighting conditions and low contrast at night object detection result.

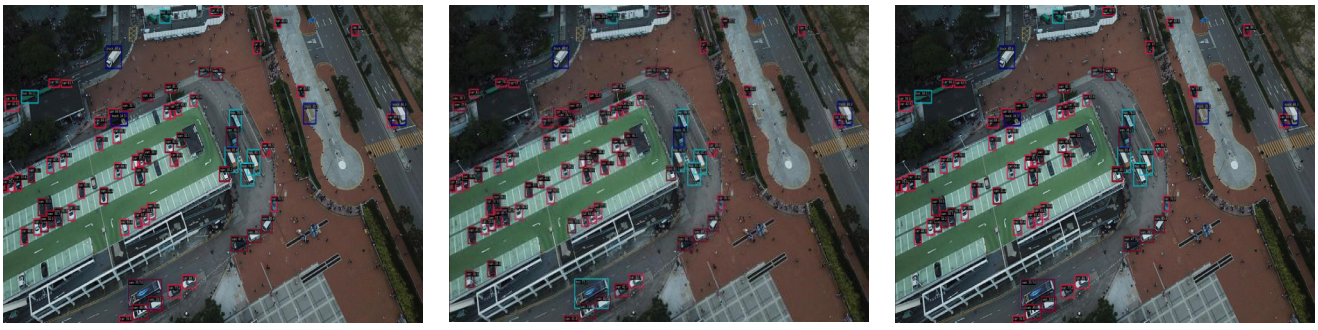

FIGURE 17. Comparison of argets are densely distributed object detection result.

faster. Compared with Drone-TOOD, although the small network models YOLOv7-tiny and YOLOv8-nano have faster detection speeds, their detection performance is still worse than that of Drone-TOOD. At the same time, we also compared it with non-YOLO series equivalent network models such as PPYOLOE Plus-m, EfficientDet, and ObjectBox. Experiments show that our improved model is better than other equivalent models, and the performance of all networks in each category is shown in Figure 14 shown. Since the number of vans and buses is small and appearance features are not fixed, the performance of all networks is lower in this category, Therefore, the performance of all networks in this category is lower, and our improved model outperforms other networks in all categories, proving the effectiveness of our proposed method in the task of vehicle detection in UAV images.

E. ALIDATION OF PREDICTION EFFECT

In the VisDrone data set, we selected images in three situations for detection. The first case is shown in Figure 13. In an environment with occlusion and uneven illumination, the original model has occlusions for the vehicle position and uneven illumination, resulting in missed detections and false detections. In the above situation, the detector Only 30% to 40% accuracy is obtained. Our improved model has strong robustness to the above situations due to the strong feature extraction ability of CSPRegNet, which establishes a stable relationship between patterns and features during training. When an object is occluded, the model can infer the possible features and shapes of the occluded part based on the visible features around it. Therefore, our model can accurately detect target vehicles with occlusion and uneven lighting in the image. This indicates that our model has strong local feature acquisition ability.

The second scenario is shown in Figure 16. In images with poor lighting conditions and low contrast at night, due to image degradation, the original model is prone to missing the detection of vehicle targets, and erroneous detection may occur under poor lighting conditions. Our model has a powerful CSPRegPAFPN for multi-scale fusion and ETDAttention module for task alignment, which enables the model to learn the feature representation of vehicles under image degradation and strong lighting conditions and accurately classify and locate vehicles.

The third scenario is shown in Figure 17. The targets are densely distributed in the image. When the targets are densely arranged, the original model does not capture corner area feature information and global context information, making it difficult for the target detection network to accurately distinguish the boundaries and positions of each target. Therefore,

the detection effect in dense target situations is poor, and a large number of false detections and missed detections occur. Our improved model introduces the EVC module to capture corner area feature information and global contextual feature information. In dense vehicle situations, it can capture corner area feature information and establish contextual feature information with global information. At the same time, the ETDAttention enables the model to accurately classify and locate vehicle targets, greatly reducing detector missed and false detections in this situation.

IV. CONCLUSION

In view of the image degradation, target occlusion, and dense target distribution that exist in the UAV image object detection task, this paper proposes an improved TOOD model and names it Drone-TOOD for the UAV image vehicle object detection task. Based on the innovative combination of RegNet spatial design ideas and CSPNet cross-stage local network structure, a new backbone network named CSPRegNet is proposed, which has powerful feature extraction capabilities and reduces computational complexity and model parameters. At the same time, a new PAFPN is launched based on the CSPRegBlock structure proposed in CSPRegNet and PAFPN, which enhances multi-scale feature fusion and solves the problem of different target sizes in drone images. In addition, we also introduced the EVC module to capture corner area feature information and global upper feature information in deep features to adjust shallow feature information, effectively improving the performance of the detector in dense target situations. Finally, we replaced the task decomposition module in the head network of the original model TOOD with the more efficient and lightweight ETDAttention. This module not only avoids the loss of feature information caused by over-dimensionality reduction in the original decomposition module but also adapts to the task-specific It can obtain the characteristics of a specific task while retaining the characteristics of another task, thereby enhancing task alignment ability and improving detection performance.

Through our experiments, we found that the proposed CSPRegNet has higher detection accuracy and less computational complexity than RegNet, ResNet, and CSPResNet. At the same time, we further proposed CSPRegPAFPN, which has better feature fusion capabilities than the current mainstream neck network. Finally, we The proposed ETDAttention enhances the task alignment capability, enables the positioning and classification tasks to be effectively decomposed, and each has task specificity without losing the other's information. It also effectively reduces the computational complexity and improves the detection network performance. In general, Drone-TOOD has greatly improved compared to TOOD on the VisDrone-2021 and UAVDT data sets and is also the best compared to other advanced detectors in the UAV image vehicle detection task.

This work has been verified on VisDrone-2021 and UAVDT data and achieved good results. Due to the limitations of the UAV airborne platform, the detector still needs to be lightweight to reduce the computational complexity and the number of parameters while ensuring stable performance. At the same time, there are few severe weather scenes in the data set used in this article, and the detector may fail in some severe weather scenarios. The performance is poor, so our future research direction is to further reduce the weight of the target detector and improve the detection performance under severe weather conditions.

REFERENCES

[1] X. Wu, W. Li, D. Hong, R. Tao, and Q. Du, "Deep learning for unmanned aerial vehicle-based object detection and tracking: A survey," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 1, pp. 91–124, Mar. 2022.

[2] U. Seidaliyeva, L. Ilipbayeva, K. Taissariyeva, N. Smailov, and E. T. Matson, "Advances and challenges in drone detection and classification techniques: A state-of-the-art review," *Sensors*, vol. 24, no. 1, p. 125, Dec. 2023.

[3] B. Jiang, R. Qu, Y. Li, and C. Li, "Object detection in UAV imagery based on deep learning: Review," *Acta Aeronaut. Astronaut. Sin*, vol. 42, Jan. 2021, Art. no. 524519.

[4] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[6] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[7] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.

[8] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[10] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6517–6525.

[11] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[12] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Computer Vision—ECCV*, Amsterdam, The Netherlands. Springer, 2016, pp. 21–37.

[13] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.

[14] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "CenterNet: Keypoint triplets for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6568–6577.

[15] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*.

[16] C. Feng, Y. Zhong, Y. Gao, M. R. Scott, and W. Huang, "TOOD: Task-aligned one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3490–3499.

[17] M. Zand, A. Etemad, and M. Greenspan, "ObjectBox: From centers to boxes for anchor-free object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 390–406.

[18] J. De Curtò, I. De Zarza, and C. T. Calafate, "Semantic scene understanding with large language models on unmanned aerial vehicles," *Drones*, vol. 7, no. 2, p. 114, Feb. 2023.

[19] J. Zhong, M. Li, Y. Chen, Z. Wei, F. Yang, and H. Shen, "A safer vision-based autonomous planning system for quadrotor uavs with dynamic obstacle trajectory prediction and its application with llms," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, Jan. 2024, pp. 920–929.

[20] M. Muzammul, A. Algarni, Y. Y. Ghadi, and M. Assam, "Enhancing UAV aerial image analysis: Integrating advanced SAHI techniques with real-time detection models on the VisDrone dataset," *IEEE Access*, vol. 12, pp. 21621–21633, 2024.
[21] R. Yu, H. Li, Y. Jiang, B. Zhang, and Y. Wang, "Tiny vehicle detection for mid-to-high altitude UAV images based on visual attention and spatial–temporal information," *Sensors*, vol. 22, no. 6, p. 2354, Mar. 2022.
[22] M. A. Momin, M. H. Junos, A. S. M. Khairuddin, and M. S. A. Talip, "Lightweight CNN model: Automated vehicle detection in aerial images," *Signal, Image Video Process.*, vol. 17, no. 4, pp. 1209–1217, Jun. 2023.
[23] J. Shen, N. Liu, and H. Sun, "Vehicle detection in aerial images based on lightweight deep convolutional network," *IET Image Process.*, vol. 15, no. 2, pp. 479–491, Feb. 2021.
[24] X. Luo, Y. Wu, and F. Wang, "Target detection method of UAV aerial imagery based on improved YOLOv5," *Remote Sens.*, vol. 14, no. 19, p. 5063, Oct. 2022.
[25] Z. Liu, X. Gao, Y. Wan, J. Wang, and H. Lyu, "An improved YOLOv5 method for small object detection in UAV capture scenes," *IEEE Access*, vol. 11, pp. 14365–14374, 2023.
[26] Z. Li, C. Pang, C. Dong, and X. Zeng, "R-YOLOv5: A lightweight rotational object detection algorithm for real-time detection of vehicles in dense scenes," *IEEE Access*, vol. 11, pp. 61546–61559, 2023.
[27] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10425–10433.
[28] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A new backbone that can enhance learning capability of CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 1571–1580.
[29] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8759–8768.
[30] Y. Quan, D. Zhang, L. Zhang, and J. Tang, "Centralized feature pyramid for object detection," *IEEE Trans. Image Process.*, vol. 32, pp. 4341–4354, 2023.
[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[32] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOx: Exceeding YOLO series in 2021," 2021, *arXiv:2107.08430*.
[33] P. Ramachandran, B. Zoph, and Q. V. Le, "Swish: A self-gated activation function," 2017, *arXiv:1710.05941*.
[34] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, and M. Lucic, "MLP-mixer: An all-MLP architecture for vision," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 24261–24272.
[35] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.
[36] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," 2016, *arXiv:1608.03983*.
[37] P. Zhu, L. Wen, X. Bian, H. Ling, and Q. Hu, "Vision meets drones: A challenge," 2018, *arXiv:1804.07437*.
[38] D. Du, Y. Qi, H. Yu, Y. Yang, K. Duan, G. Li, W. Zhang, Q. Huang, and Q. Tian, "The unmanned aerial vehicle benchmark: Object detection and tracking," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 370–386.
[39] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6154–6162.
[40] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9626–9635.
[41] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10778–10787.
[42] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9756–9765.
[43] S. Xu, X. Wang, W. Lv, Q. Chang, C. Cui, K. Deng, G. Wang, Q. Dang, S. Wei, and Y. Du, "PP-YOLOE: An evolved version of YOLO," 2022, *arXiv:2203.16250*.
[44] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie, Y. Li, B. Zhang, and Y. Liang, "YOLOv6: A single-stage object detection framework for industrial applications," 2022, *arXiv:2209.02976*.
[45] C.-Y. Wang, A. Bochkovskiy, and H.-Y.-M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 7464–7475.
[46] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, "DINO: DETR with improved denoising anchor boxes for end-to-end object detection," 2022, *arXiv:2203.03605*.