IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 October 2023, accepted 8 November 2023, date of publication 14 November 2023,
date of current version 21 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3332815

RESEARCH ARTICLE

YOLOAL: Focusing on the Object Location for Detection on Drone Imagery

XINTING CHEN[1], WENZHU YANG[1,2], SHUANG ZENG[1], LEI GENG[1], AND YANYAN JIAO[1]
[1]School of Cyber Security and Computer, Hebei University, Baoding, Hebei 071002, China
[2]Hebei Machine Vision Engineering Research Center, Baoding, Hebei 071002, China

Corresponding author: Wenzhu Yang (wenzhuyang@hbu.edu.cn)

ABSTRACT Object detection in drone-captured scenarios, which can be considered as a task of detecting dense small objects, is still a challenge. Drones navigate at different altitudes, causing significant changes in the size of the detected objects and posing a challenge to the model. Additionally, it is necessary to improve the ability of the object detection model to rapidly detect small dense objects. To address these issues, we propose YOLOAL, a model that emphasizes the location information of the objects. It incorporates a new attention mechanism called the Convolution and Coordinate Attention Module (CCAM) into its design. This mechanism performs better than traditional ones in dense small object scenes because it adds coordinates that help identify attention regions in such scenarios. Furthermore, our model uses a new loss function combined with the Efficient IoU (EIoU) and Alpha-IoU methods that achieve better results than the traditional approaches. The proposed model achieved state-of-the-art performance on the VisDrone and DOTA datasets. YOLOAL reaches an AP50 (average accuracy when Intersection over Union threshold is 0.5) of 63.6% and an mAP (average of 10 IoU thresholds, ranging from 0.5 to 0.95) of 40.8% at a real-time speed of 0.27 seconds on the VisDrone dataset, and the mAP on the DOTA dataset even reaches 39% on an NVIDIA A4000.

INDEX TERMS Drone, small dense objects detection, attention mechanism, loss function.

I. INTRODUCTION

Object detection is a fundamental topic in the field of machine vision, which has been extensively researched for many years. Real-time and accurate object detection can provide effective support. In recent years, object detection has been widely used in, such as object tracking, scene understanding, and behavior recognition. With the emergence of deep learning techniques, object detection algorithms based on hand-crafted features that have become outdated. Alex proposed the famous convolutional neural network (CNN) AlexNet [1] in 2012, and since then, deep learning and CNN have gradually become popular and widely used. In general, recent deep learning models for object detection can be divided into two categories: one-stage detection models such as SSD [2] and YOLO [3], and two-stage detection models including R-CNN [4], Fast R-CNN [5], and Faster R-CNN [6].

Two-stage detection models have higher accuracy but lower real-time performance. This is because they extract regions of interest (ROI) from the images. Moreover, they perform bounding box (BBox) regression and classification within these ROIs. On the other hand, one-stage detection models do not require this step, as they perform localization and classification in a single stage. Consequently, the one-stage detection models have lower accuracy but higher detection speeds than two-stage detection models. Because real-time detection is important, YOLOv7, a one-stage object detection model, is chosen as the backbone for object detection.

In recent years, significant progress has been made in object detection research. However, small object detection for drone-captured images is still a difficult task in computer vision. Although numerous object detection algorithms have been proposed each year, deep learning has shown significant advancements. Nevertheless, existing object detection algorithms are primarily designed and trained for natural images, and their performance in dense small object scenarios still

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

requires improvement. We acknowledge that drone-captured images have unique characteristics compared with natural image datasets such as MS COCO [7]. First, the size of the photographed object varies significantly owing to the differences in the horizontal height of the drone. The same object may exist simultaneously as small, medium, and large objects when shot at different heights. This will undoubtedly affects the acquisition and identification of object features. Second, the sparsity of objects captured by drones varies significantly, and small objects are an important part of the dataset. For example, small objects make up more than 60% of VisDrone's training dataset.

Existing real-time object detection algorithms are based on YOLO [8], [9], [10] and FCOS [11], which rely on GPU for fast object detection. YOLO series algorithms are frequently utilized as real-time detection algorithms in one-stage detection models [8]. The detection speed of the YOLO series algorithms is fast, while their accuracy gradually increases with the improvement of the algorithms. The most recent update of the YOLO series is YOLOv8 [9]. However, the experiment showed that YOLOv7 [8] is less computationally intensive than it. Therefore, YOLOv7 [8] is selected as the backbone for detecting objects in drone capture scenarios. Our algorithm surpassed YOLOv8 [9] in terms of the accuracy.

To focus on the object location for detection on drone imagery, an object detection model, YOLOAL is proposed in this article. Our design focuses on obtaining the object location information to improve the accuracy and detection speed of object detection. First, an attention module is introduced to obtain the features of object in the channel and spatial dimensions. The attention mechanism allows the network to pay more attention to the object in the drone imagery and achieve good results. In addition, the degree of overlap, center point distance, and weight and height of the anchors between the object BBox and predicted BBox are considered in the model training process. Furthermore, we use Alpha-IoU to improve the BBox's regression accuracy. It adaptively up weights the loss and gradient of high IoU objects.

The contribution of this work is listed as follows:

(1) An improved object detection model, YOLOAL, is proposed for drone imagery detection. This model incorporates attention modules in the neck of the architecture and utilizes improved loss functions. ItÂ improves the ability to detect objects by emphasizing the location information.

(2) To address the challenge of detecting dense small objects, we introduce a new attention module called CCAM. This module leverages the coordinate information to accurately localize objects in high-density scenarios.

(3) To balance the difficulty of object detection and increase detection speed, we use EIoU as the loss function. Then, Alpha-IoU is induced to filter out a more accurate bounding box and increase the loss function flexibility.

II. RELATED WORK

A. OBJECT DETECTION

CNN-based object detection models can be categorized into one-stage object detection models and two-stage object detection models based on the presence of regions of interest (ROI) for classification. We can distinguish between an anchor-based object detection model and an anchor-free object detection model without an anchor box based on whether anchor boxes are generated. Although there are many distinguishing types, the CNN-based object detection model can be broadly divided into two parts: the backbone and prediction head. In addition, researchers in recent years have usually inserted some layers between the two parts, and people usually refer to this part as the neck of the detector. This is usually called the neck of the detector.

Backbone: The backbone can obtain the feature information of the images. Popular backbones include VGG [12], ResNet [13], EfficientNet [14], and CSPDarknet53 [15], which have demonstrated powerful feature acquisition capabilities.

Neck & Head: Once the backbone extracts the picture feature information, the neck reprocesses and rationally uses the feature maps at different stages. This enables a better capture and processing of these maps. The head is then responsible for determining both the category of the detected objects and their location.

B. SMALL OBJECT DETECTION

There are still many problems to be solved in the use of drones. Antonio Silva [16] use edge servers to merge a global map, including all simultaneous localization and mapping updates, to provide a consistent map that is effectively updated for the drone. Small object detection is one of the most difficult problems in drone imagery detection.

Small object detection methods are widely used in the fields of autonomous driving, drone capture, and video recognition. Although important, existing object detection algorithms still need improvement to achieve satisfactory results. Small object detection has to overcome the following difficulties: (1) small objects cover a small area, which makes it difficult to obtain information and features; (2) small objects are easily overlapped by medium and large objects; (3) small objects are susceptible to background interference.

There are commonly used methods for detecting small objects such as improving the clarity of the image or introducing feature pyramids into the object detection network [2], [17]. DMNet [18] uses a density map-guided cropping strategy to remove areas without objects, balancing the impact of the background on object recognition. ClusDet [19] unifies object clustering and detection in an end-to-end framework by sequentially identifying clustered regions and detecting objects within those areas. TPH-YOLOv5 [20] uses a convolutional block attention module [21] (CBAM) in the model and significantly improves its accuracy. YOLO-ACN [22] uses the improved attention mechanism and CIoU to improve the detection accuracy of the MS COCO [7] dataset. These methods inspired us in this study.

Two-stage detection models are commonly utilized for detecting small objects because of their ability to achieve higher accuracy. However, our objective is to enhance both the detection accuracy and speed.

C. ATTENTION MECHANISM

The attention mechanism imitates the human visual system's ability to identify important areas in complex scenes. It has been widely used in machine vision tasks over the past few years and plays a crucial role in improving the object detection accuracy. One such attention module is SE [23], also known as squeeze and excitation. The ''squeeze'' step involves using global average pooling on the channel of obtained features and aggregating local information to obtain global information. The ''excitation'' step includes fully connected layers and activation functions that learn the nonlinear relationship between channels.

The SE attention module uses global average pooling to compress data, and it has been shown that the result using average pooling has an accuracy improvement (0.31%) compared with using max pooling in the ImageNet [24] dataset. Although some attention modules, such as SimAM [25], CBAM [21], and CCNet [26] use other methods for data processing, there are still many attention modules such as ECA [27] and coordinate [28], which are influenced by SE. They use the same data compression method as SE.

To enhance the detection accuracy of drone-captured images, CCAM is proposed in object feature processing. After the data are processed through the channel attention module, we use two one-dimensional coding processes and introduce the coordinate concept into the CCAM. Finally, our attention module achieves better results than recent attention modules.

D. LOSS FUNCTION

The loss function is a crucial component of the object detection model. Nevertheless, it is often overlooked. Although not as well known as the attention mechanism, researching and selecting an appropriate loss function will undoubtedly enhance the object detection ability.

Currently, the famous and commonly used loss functions include Intersection over Union [29] (IoU), Generalized IoU [30] (GIoU), Complete IoU [31] (CIoU), Distance-IoU [31] (DIoU), SCYLLA-IoU [32] (SIoU), Efficient IoU [33] (EIoU), etc. As the most classical loss function, IoU has the characteristics of scale invariance, symmetry, and triangular invariance. However, when the two bounding boxes do not intersect, the IoU value directly becomes zero, which cannot reflect the distance between the two bounding boxes. GIoU calculates the minimum closed area of the two bounding boxes and calculates the proportion of the closed area to which neither bounding box belongs before calculating IoU. Finally, it subtracts this proportion from the IoU to obtain the GIoU. However, when the two bounding boxes intersect, the convergence is slower in the horizontal and vertical directions. Considering the principle of IoU and the shortcomings of GIoU, DIoU improves the convergence speed by regressing the Euclidean distance between the centroids of two bounding boxes.

III. PROPOSED METHOD

A. BASE STRUCTURE

Designers must consider several factors when designing an efficient and stable object detection network. These include the number of layers of the architecture, number of parameters, amount of computation, and computational density. CSPVoVNet [34] considers the number of elements in the convolutional layer output tensors and analyzes gradient paths that allow the weights of different layers to learn more different features. After considering the problem of how to design efficient networks, ELAN [35] concludes that controlling the longest and shortest gradients is crucial. Ultimately, we select ELAN's stacked computational blocks as the base structure. The structure of ELAN is shown in Figure 1, and that of YOLOAL is shown in Figure 2.

B. CONVOLUTION AND COORDINATE ATTENTION MODULE

Our attention module comprises two components: a channel attention module and coordinate attention module. The feature map undergoes sequential processing through the channel and coordinate attention modules, resulting in the inference of separate attention maps along two dimensions. Despite its effectiveness, the CCAM is a lightweight module that does not impose a significant computational burden. Three CCAM modules are used on the neck of the detection model.

Further details on both components- the channel and coordinate attention modules - are provided below.

1) CHANNEL ATTENTION MODULE

The channel attention module focuses on ''what'' the object is, and channel attention maps can be generated by analyzing the relationship between image channels. The architecture of the channel attention module is shown in Figure 3. We compress the features of the object to make the computation more efficient and to reduce the number of parameters. Many attention mechanisms, such as SE [23], ECA [27] and coordinate [28], use average pooling to compress and aggregate the spatial information. However, this method weakens the boundaries between small objects and their backgrounds in drone-captured images. Instead, average pooling and max pooling are combined to compress the image information.

To generate the channel features, we compress the image information using global average pooling and max pooling. Subsequently, these two feature maps are introduced into a multilayer perceptron (MLP) network with hidden layers to

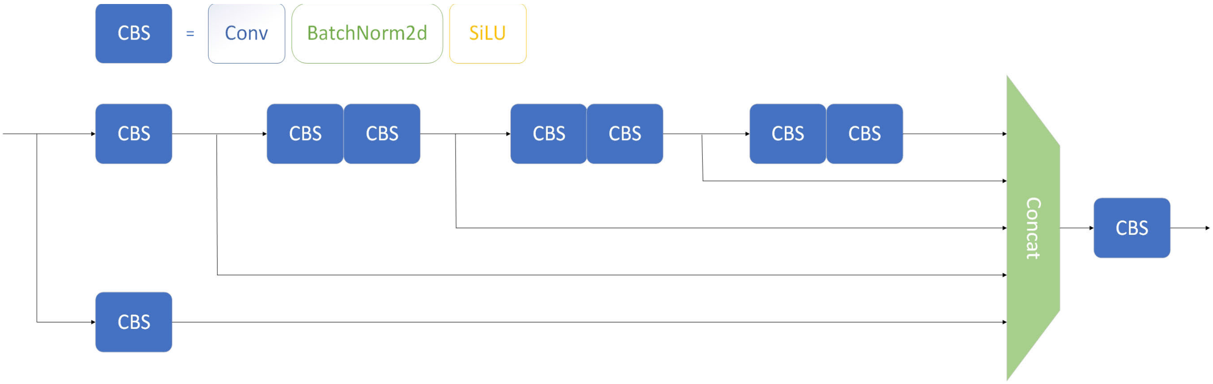

FIGURE 1. The structure of ELAN. CBS block is combined with convolution, batch normalization layer and SiLU activation function, which can enhance learning capability. What's more, ELAN optimizes the gradient length of the network by using the stack structure.

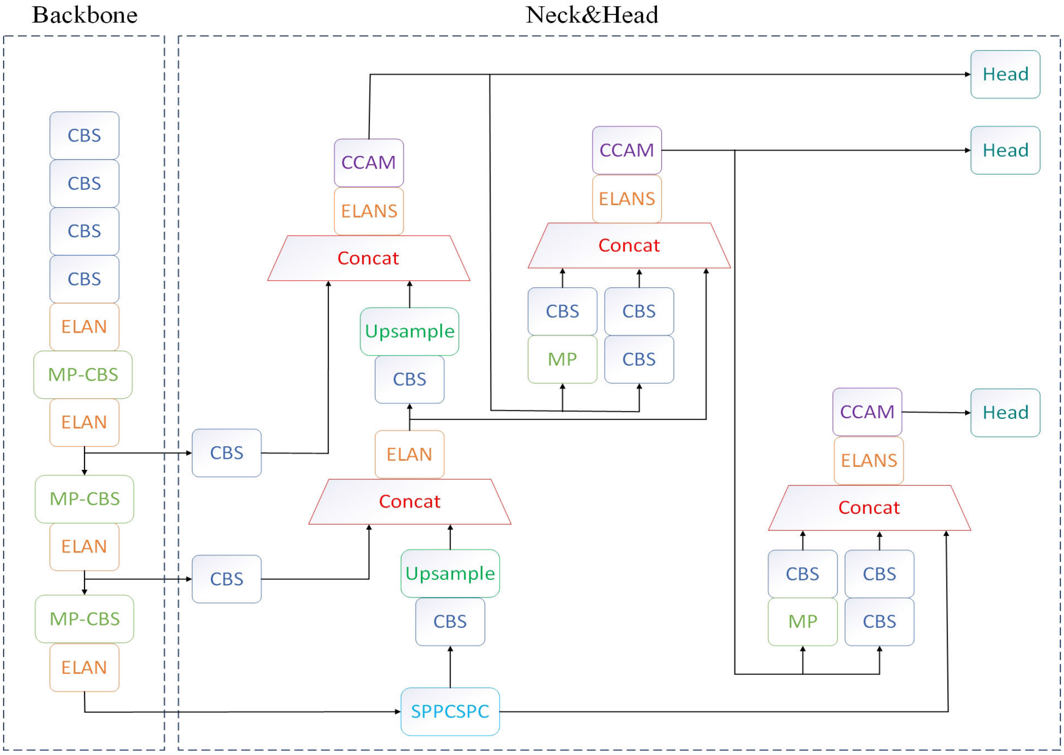

FIGURE 2. The architecture of the YOLOAL. "ELANS" lacks the final CBS structure compared to ELAN. "MP-CBS" uses max pooling before the CBS block.

generate a channel feature map. The hidden layer activation size of the MLP is set to (C/r*1*1), where C is the number of channels and r is the reduction ratio, which is fixed at 16. LeakyReLU [36] is used as the activation function in the MLP. The formula for LeakyReLU is as follows:

$$\text{LeakyReLU}(\mathrm{x}) = \begin{cases} \mathrm{x}, & \mathrm{x} > 0 \\ \alpha \mathrm{x}, & \mathrm{x} \leq 0 \end{cases} \tag{1}$$

where we set α to 0.01.

The LeakyReLU activation function adds a small linear component to the negative inputs, which helps to address the vanishing gradient problem. LeakyReLU ranges from negative infinity to positive infinity. Moreover, the additional computational cost for it is very small. The MLP processed feature information is activated by the sigmoid function after summation. The channel attention module is computed as follows:

$$\begin{aligned} \mathrm{F_c}(f) = \mathrm{S}(&\mathrm{MLP}(\mathrm{MaxPool}(f)) \\ &+ \mathrm{MLP}(\mathrm{AvgPool}(f))) \end{aligned} \tag{2}$$

f is the input feature, $\mathrm{F_c}(\cdot)$ denotes the feature map of channel attention. $\mathrm{S}(\cdot)$ is the sigmoid activation function. $\mathrm{MLP}(\cdot)$ means multilayer perceptron network. $\mathrm{MaxPool}(\cdot)$ and $\mathrm{AvgPool}(\cdot)$ are global max pooling and global average pooling, respectively.

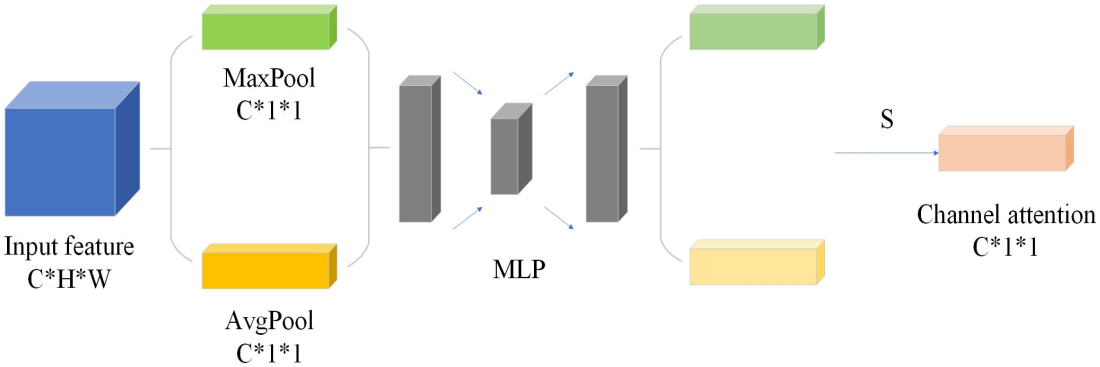

FIGURE 3. The overview of the channel attention module. "MaxPool" and "AvgPool" mean global max pooling and global average pooling. "S" means that the action involves summing up the processing results and then activating them with a sigmoid function.

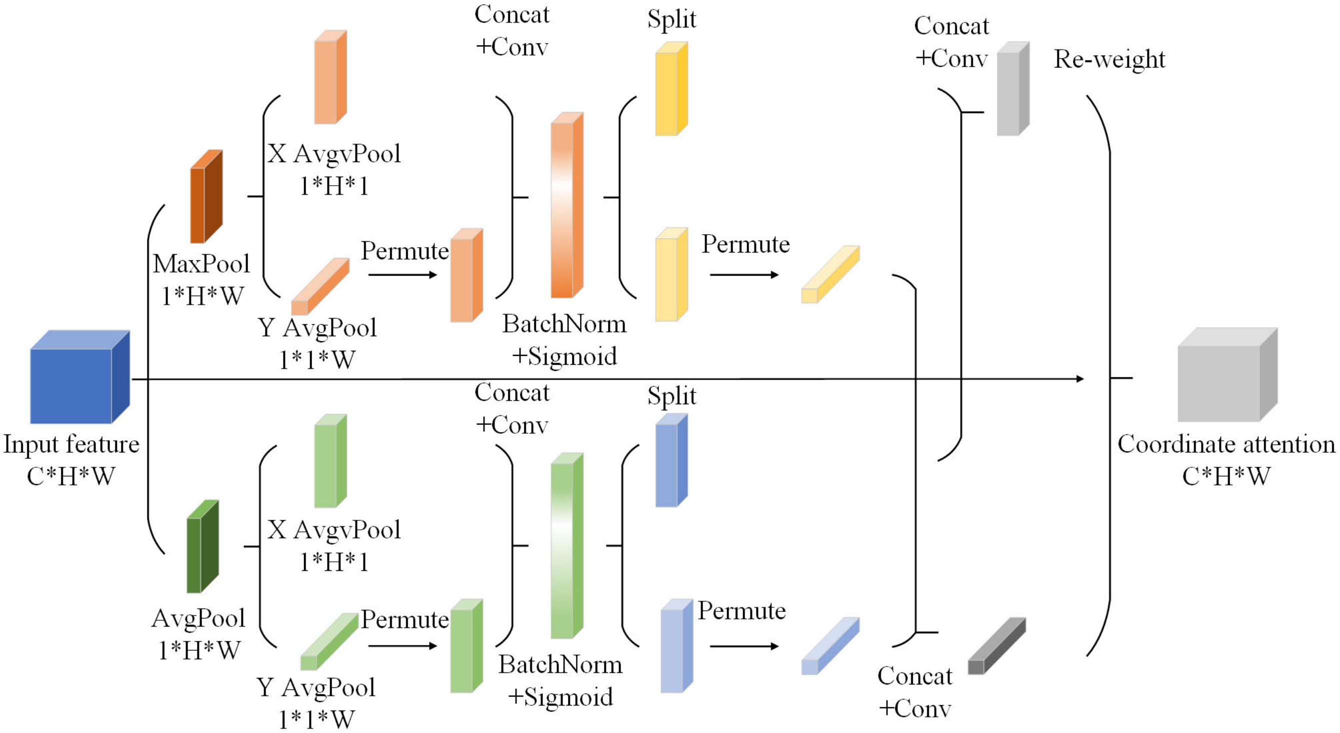

FIGURE 4. The overview of the spatial attention module. "MaxPool" and "AvgPool" mean max pooling and average pooling along the channel axis. "X AvgPool" and "Y AvgPool" mean 1D horizontal average pooling and 1D vertical average pooling, respectively.

2) COORDINATE ATTENTION MODULE

The coordinate attention module focuses on "where" the object is, and spatial attention maps can be generated by analyzing the coordinate relationship. The architecture of the coordinate attention module is shown in Figure 4. We place it serially after the channel attention module in tandem, and it can accurately identifies both the class and location of the objects.

To obtain object location information, coordinates are introduced into the module to capture long-distance relationships and information interactions in space using precise location data. We obtain two 2D maps that are compressed by max pooling and average pooling along the channel axis. By processing the input feature to reduce the influence of channel feature, the spatial weight can be calculated better. Specifically, we perform 1D average pooling in both the horizontal and vertical directions and then combine them for batch normalization and standardized processing. After sigmoid activation, the feature maps are merged and normalized according to the horizontal and vertical directions, yielding the corresponding coordinate weights. Finally, we obtain a coordinate attention feature map by multiplying these weights with the input feature.

The attention module considers the shortcomings of the single feature compression method, using max pooling and average pooling in parallel. In contrast to the traditional compression method for generating a single feature vector, the processed feature map aggregates features along two spatial directions to produce a pair of feature maps. Such processing allows the attention blocks to spatially capture long-range interactions and retain precise location information oriented along one another. By combining the feature information that has undergone max pooling and average pooling, noise can be avoided and the loss of feature information can be reduced.

We have attempted a variety of methods to combine the channel attention module with the coordinate attention module. The experiment proved that the best accuracy can be achieved by placing the coordinate attention module behind the channel attention module. With the attention mechanism introduced in the detection model, the network can enhance the ability of feature expression in a specific region without increasing the computational cost.

C. IOU LOSS FUNCTION

In recent YOLO series object detection models such as YOLOv5 [10], YOLOv7 [8], and YOLOv8 [9], CIoU [31] is used as the regression loss function. This increases the loss of width and height based on DIoU to make the predicted BBox closer to the real BBox. There is some ambiguity because the width and height loss of the CIoU are relative values, not definite values. Additionally, the CIoU cannot increase or decrease both width and height simultaneously, which slows down the loss convergence. To address these issues, EIoU [33] minimizes the difference between the width and height of the object BBox and predicted BBox, which results in faster convergence and better localization results.

The loss of EIoU, which is defined as follows:

$$L_{\mathrm{EIoU}} = L_{\mathrm{IoU}} + L_{\mathrm{dis}} + L_{\mathrm{asp}} \tag{3}$$

$$L_{\mathrm{IoU}} = 1 - IOU = 1 - \frac{\left|B \cap B_{gt}\right|}{\left|B \cup B_{gt}\right|} \tag{4}$$

L_{IoU} is the IoU loss, existing predicted BBox B and object BBox B_{gt}. It is used to calculate the similarity of the two boxes.

$$L_{\mathrm{dis}} = \frac{\rho^2\left(b, b^{gt}\right)}{(w^c)^2 + (h^c)^2} \tag{5}$$

L_{dis} is the distance loss, it shows the distance between the two boxes. b and b^{gt} represent the center points of B and B_{gt}. $\rho(x, y)$ is the Euclidean Distance between points x and y. The minimum enclosing box of B and B_{gt} is C, w^c and h^c are the width and height, respectively.

$$L_{\mathrm{asp}} = \frac{\rho^2\left(w, w^{gt}\right)}{(w^c)^2} + \frac{\rho^2\left(h, h^{gt}\right)}{(h^c)^2} \tag{6}$$

L_{asp} is the aspect loss, it can minimize the difference of the B's and B_{gt}'s width and height. w and w^{gt} represent the widths of the two bounding boxes, and h and h^{gt} represent the heights of B and B_{gt}.

Alpha-IoU [37] is used to increase the accuracy of the loss function. This method accelerates the gradient of IoU objects for high IoU objects, which facilitates the later training phase when alpha>1. Simultaneously, the effect on low IoU objects is minimal. Unlike using only traditional loss functions, this approach enhances robustness and helps stabilize model training in the early stages when the gradient is large.

The loss of the new loss function is as follows:

$$L_{\mathrm{IoU}-\alpha} = L_{\mathrm{IoU}} + L_{\mathrm{dis}-\alpha} + L_{\mathrm{asp}-\alpha} \tag{7}$$

$$L_{\mathrm{dis}-\alpha} = \left(\frac{\rho^2\left(b, b^{gt}\right)}{(w^c)^2 + (h^c)^2}\right)^{\alpha} \tag{8}$$

$$L_{\mathrm{asp}-\alpha} = \left(\frac{\rho^2\left(w, w^{gt}\right)}{(w^c)^2}\right)^{\alpha} + \left(\frac{\rho^2\left(h, h^{gt}\right)}{(h^c)^2}\right)^{\alpha} \tag{9}$$

$L_{\mathrm{dis}-\alpha}$ and $L_{\mathrm{asp}-\alpha}$ represent bringing alpha into the calculation of the corresponding loss function.

IV. EXPERIMENTAL ANALYSIS

A. DATASET

To verify the validity of the proposed method, we apply it to two publicly accessible datasets: VisDrone [38] and DOTA [39].

(1)VisDrone: The VisDrone [38] dataset, which includes 6471 training images, 548 validation images, and 3190 test images. The test set is divided into a test-dev with 1610 images and a test-challenge with 1580 images. As labels are not provided for the test-challenge set, we only test our models on the validation and test-dev datasets. There are 10 types of objects in this dataset, which are obtained using drones. Images were obtained from 14 different cities in China. They are very different in terms of the environment (urban and rural areas), objects (such as pedestrians, vehicles, and bicycles), and density (sparse and crowded scenes), with the difference and representativeness of drone-captured scenarios.

(2)DOTA: The DOTA [39] dataset consists of 2806 aerial images collected from multiple sensors/platforms, such as Google Earth, across several cities. These aerial images have varying pixel sizes ranging from 800*800 to 4000*4000, contain objects of different scales, orientations, and shapes. Following ClusDet [19] and AdaZoom [40], we select images of objects such as planes, vehicles, ships, and helicopters.

B. EXPERIMENTAL DETAILS

We conducted our experiments on a single NVIDIA A4000, implementing the object detection network using PyTorch 1.8.1 and Ubuntu 18.04. To evaluate the evaluation method in MS COCO [7], mAP (average of all 10 IoU thresholds, ranging from 0.5 to 0.95) and AP50 (average accuracy when IoU threshold is 0.5) are used to indicate object detection accuracy. Giga floating point of operations (GFLOPs) is utilized to assess the computational cost, and frames per second (FPS) can show the operational efficiency of these models.

Some results from the referenced literature were obtained from the validation dataset, and we also observed a strong correlation between the results of object detection experiments in the validation dataset and the test dataset. To ensure the consistency of the data results, we used the validation and test-dev dataset results for the comparison and experiments.

Training phase: The training batch size is 2. There are differences in the resolutions of different images on the VisDrone and DOTA datasets. To facilitate the training phase, we compressed and normalized the image pixels of different sizes and set them to 1024 and 1536, which were consistent with the input sizes of the detector. The initial learning rate was set to 0.01, and the final OneCycleLR learning rate was the same during the model training on both datasets. Because the VisDrone dataset was relatively small, we only trained the model for 100 epochs, whereas the DOTA dataset training iterations were set to 200. Moreover, mosaic and mix-up data augmentation methods were used in the training phase.

Test phase: The input sizes of the detectors are kept consistent with those of the training phase, that is, both 1024 and 1536. The threshold values of the standard non-maximum suppression (NMS) were set to 0.65 across all datasets tested.

C. EFFECT OF CCAM

Attention mechanisms are crucial in machine vision tasks, and researchers have been looking for ways to improve them over the years. SE [23] is still widely used as a well known attention module. However, SE attention only considers compressing channel information and ignores the importance of positional information, which is important for obtaining object structures in vision tasks. Later work, such as CBAM [21], attempts to exploit positional information by reducing the tensor's channel dimensionality and using convolution to compute spatial attention. But, convolution can only capture local relationships and cannot obtain the overall information relationship for vision tasks. Coordinate attention [28] was proposed by Hou et al. In contrast to transforming the feature tensor into a feature vector after 2D average pooling, the coordinate attention mechanism aggregates the features in two directions. This helps to preserve the spatial position correlation of the objects. Coordinate attention obtains direction and position sensitive information to capture object feature information, but its processing of channel information is too simple and still needs to be optimized and improved. SimAM [25] optimizes the energy function to calculate the importance of each neuron. To infer the 3D attention weights for the feature maps, we derived a fast analytical solution for the energy function without increasing the original network parameters. We attempted to add different attention modules to the baseline, and the experimental results are shown in Table 1.

It can be seen from the experimental results for VisDrone. By adding the attention module to the neck of the object detection model, we can get an improvement in the object detection accuracy, and different attention algorithms achieve different improvements. As shown in Table 1, CCAM achieves 36.1% and 59.4% for mAP and AP50, respectively, which are better than those of other attention modules. Compared to the baseline, CCAM was approximately 0.8% higher on AP50 and 0.6% higher on mAP. CCAM1 and CCAM2 are two different arrangements of the attention modules. In CCAM1, the coordinate attention module was placed before the channel attention module, whereas in CCAM2, the channel attention module was placed parallel to the coordinate attention module. Sequential generation in attention maps is more effective than parallel generation for inferring accurate attention maps. Additionally, the order of these modules can affect detection efficiency and accuracy. Therefore, the selection of an appropriate order can further improve overall accuracy.

TABLE 1. Comparison of performance for different attention modules on the VisDrone validation dataset with an image size of 1024.

Methods	AP50(%)	mAP(%)
Baseline	58.6	35.5
+SE [23]	59.3	36.0
+CBAM [21]	59.1	36.0
+Coordinate attention [28]	59.3	35.9
+SimAM [25]	58.9	35.7
+CCAM1(Ours)	58.8	35.7
+CCAM2(Ours)	58.7	35.5
+CCAM(Ours)	59.4	36.1

D. EFFECT OF LOSS FUNCTION

The method for calculating the loss function is constantly being improved. Our goal is to use a more suitable loss function for object detection algorithms in drone-captured images, which will help address the issue of imbalanced difficulty levels in object detections. Currently, the most commonly used loss function in the YOLO series object detection algorithm is CIoU [31]. However, it suffers from the problem that the ratio of the prediction BBox width and height is inversely proportional, and both cannot be increased or decreased at the same time. By replacing CIoU with EIoU as the loss function of the object detection model, higher detection accuracy can be obtained in the VisDrone dataset. Compared with CIoU, EIoU yields approximately 0.9% higher on AP50(59.5) and 0.9% higher on mAP(36.2).

In addition, Alpha-IoU [37] is combined with CIoU and EIoU, and the object detection accuracy obtained for different alpha values are different. The experimental results are shown in Figure 5 and Figure 6.

When the alpha value is set to 1, it is equivalent to using the EIoU and CIoU as the loss function for object detection. The results show that when the alpha value increases, the mAP obtained is higher than when using EIoU and CIoU alone. Specifically, an alpha value of 2 yielded the highest result with mAP at 36.7% and AP50 at 59.6%, which were 0.5% and 0.1% higher, respectively, than those obtained using EIoU alone on VisDrone. If the value of alpha continues to increase beyond this point, both the mAP and AP50 gradually decrease. When the alpha was 7, there was a more obvious downward trend. The results of EIOU and Alpha 2 are the highest for VisDrone and DOTA. Regardless of the value of the alpha setting, EIoU obtains better object detection accuracy than CIoU. We believe that when we set the alpha to

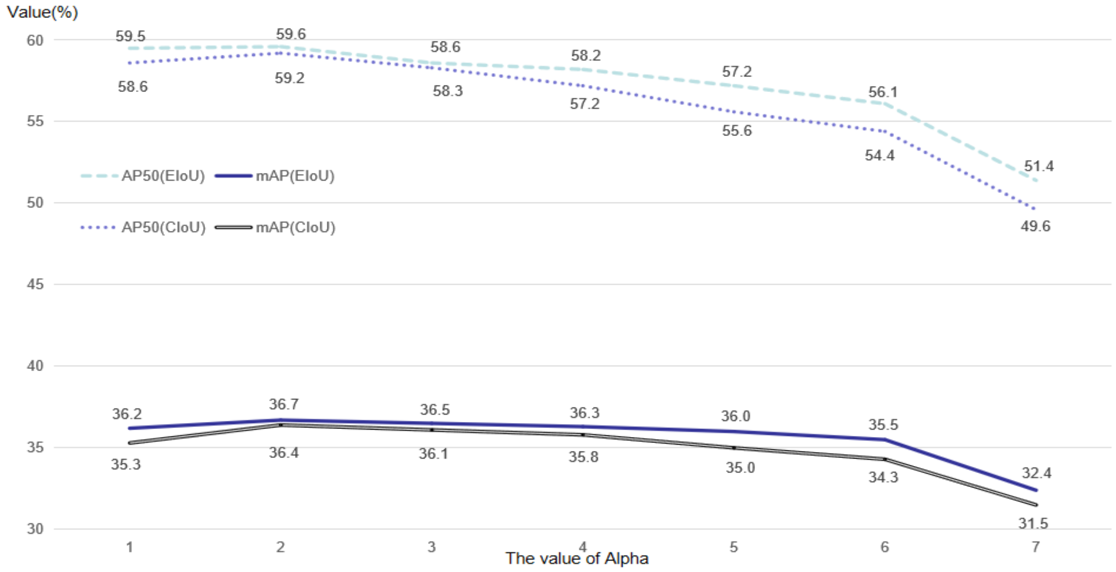

FIGURE 5. The AP50 and mAP of the experiment using Alpha-IoU with CIoU and EIoU on VisDrone. We can see that the object detection accuracy is highest when EIoU is used, and the value of alpha is 2.

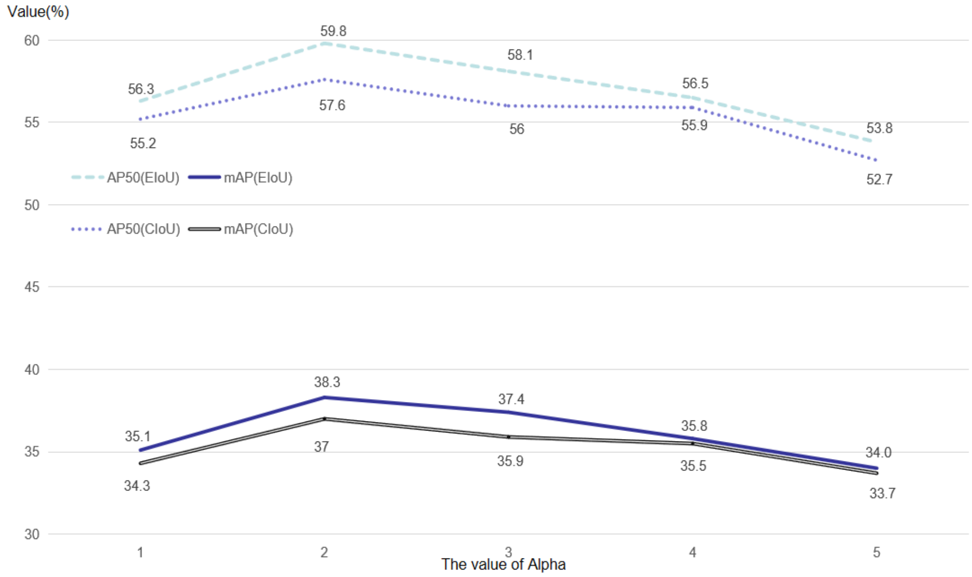

FIGURE 6. The AP50 and mAP of the experiment using Alpha-IoU with CIoU and EIoU on DOTA.

a finer size, such as at 0.1 intervals, we can achieve a higher detection accuracy.

E. COMPARISONS WITH THE STATE-OF-THE-ART

We detected the VisDrone dataset using the YOLO series object detection algorithm, and the detection results are listed in Table 2.

From the data in Table 2, when training conditions are consistent, the training accuracies of YOLOv7 [8] and YOLOv8 [9] are much higher than that of YOLOv5. In terms of performance on the VisDrone test-dev dataset, YOLOAL's AP50 was 49.6 and mAP was 29.1, both higher than YOLOv7 and YOLOv8. Through GFLOPs, we can see that the computational load of YOLOAL is only slightly higher than that of YOLOv7, but much smaller than that of either YOLOv5 or YOLOv8.

TABLE 2. Comparison of performances on VisDrone test-dev for different YOLO series models.

Methods	Score threshold	AP50(%)	mAP(%)	GFLOPs
YOLOv5 [10]	0.5	42.3	25.6	203.9
YOLOv7 [8]	0.5	48.5	27.9	188.2
YOLOv8 [9]	0.5	47	28.3	257.4
YOLOAL(Ours)	0.65	49.5	29	188.3
YOLOAL(Ours)	0.5	49.6	29.1	188.3

Furthermore, we find that increasing the score threshold leads to a decrease in detection accuracy. For instance, setting a score threshold of 0.65 for YOLOAL results in a

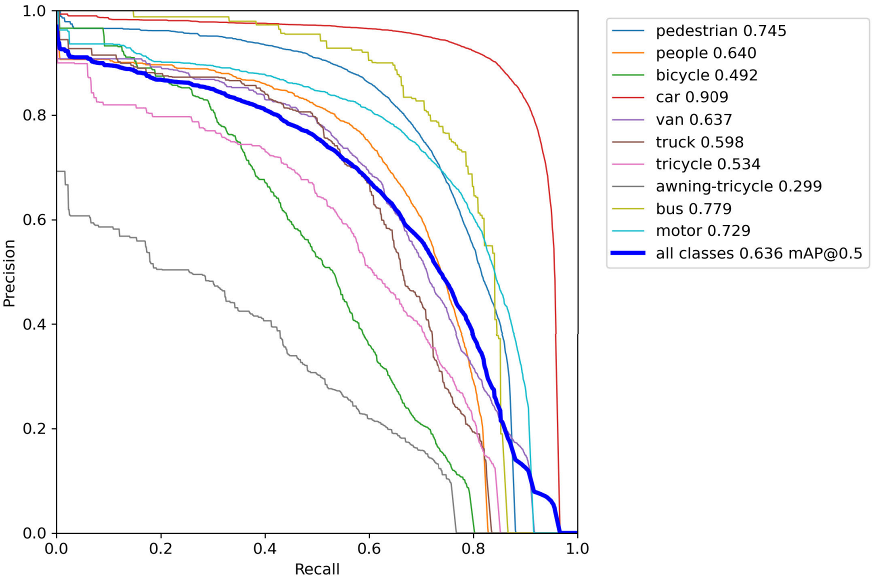

FIGURE 7. The precision-recall curve of YOLOAL on the VisDrone dataset. Precision is the proportion of the positive data in data that is predicted to be positive, and recall is the proportion of the positive data in data that can be predicted correctly.

TABLE 3. Comparison of performances on VisDrone validation for different detection models. The size of the images is 1536. Results for state-of-the-art (SOTA) are taken from the publications. We also report the inference time per image.

Methods	AP50(%)	mAP(%)	s/img(GPU)
CRENet [41]	54.3	33.7	0.901
DMNet [18]	47.6	28.2	-
CascadeNet [42]	58.02	30.12	-
MPFPN [43]	54.38	29.05	-
ClusDet [19]	56.2	32.4	0.773
SAIC-FPN [44]	62.97	35.69	2.568
MSCC-YOLOv5 [45]	-	36.1	-
DSDE-Net [46]	34.8	21.1	-
ROSD [47]	58.21	34.57	-
YOLOAL(Ours)	63.6	40.8	0.27

slight reduction (0.1) in AP50 and mAP compared to using a threshold value of 0.5. Although the test results were different, the ability of the model to perform object detection remained unchanged. To ensure the rigor of the experiment, the score threshold for the subsequent experiments is set to 0.65.

To evaluate the performance of the proposed model on VisDrone, we compared our approach with the state-of-the-art approaches. The detailed experimental results are presented in Table 3.

CRENet [41] avoids anchor settings and cluster regions overlapping by using coarse-level preview detection. ClusDet [19] trains regionally generated networks using supervised learning based on pseudo-generative annotations. SAIC-FPN [44] uses scale-adaptive image cropping to detect small objects with a higher resolution. These two-stage object detection algorithms improve the accuracy of object detection, but the processing also slows down the processing speed. YOLOAL gets 63.6 in AP50 and 40.8 in mAP, outperforming other methods in terms of accuracy while maintaining a faster detection speed compared to other methods. The precision-recall curve for YOLOAL is shown in Figure 7.

TABLE 4. Comparison of performances on DOTA for different detection models. Results for SOTA are taken from the publications and the size of images is 1024.

Methods	AP50(%)	mAP(%)	s/img(GPU)
Faster R-CNN [6]	54.5	32.3	-
ClusDet [19]	47.1	31.4	-
AdaZoom [40]	63.5	37.8	0.599
YOLOAL (Ours)	61.2	39	0.10

YOLOAL also achieved good results on DOTA. Detailed experimental results are presented in Table 4.

AdaZoom [40] constructs a reinforcement learning framework to focus on region generation, where the scales and aspect ratios of the generated regions are adaptive to the scales and distribution of objects inside. This treatment helps to improve the performance of the object detection model and achieves good results in the AP50. At the same time, this complicates object detection calculations and increases the time required for detection. In comparison, YOLOAL achieved a better result of 39% in mAP and a 1.2% improvement compared to AdaZoom. We used a one-stage object detection model to surpass two-stage object detection models in terms of accuracy. At the same time, the model detection speed was much faster than that under the same conditions. The results validate that the proposed YOLOAL algorithm is efficient for object detection in drone-captured scenarios.

TABLE 5. Comparison of performances on VisDrone validation for each category. The data obtained were the average of multiple tests. The FPS can represent the speed of object detection, with higher values indicating that the model can detect more images per second. Some categories were represented by the first three letters.

Methods	all(%)	ped(%)	peo(%)	bic(%)	car(%)	van(%)	tru(%)	tri(%)	awn(%)	bus(%)	mot(%)	FPS
Model 1	35.5	35	24.8	19.9	64	41.5	38.5	28.4	15.8	53.4	33.6	19.5
Model 2	36.1	35.3	25.9	20.4	64.1	42.3	38.9	29.5	16.7	54.4	33.8	19.2
Model 3	36.7	35.3	25.4	20	64.7	43.8	40	30.1	17	56.3	34.1	19.8
Model 4	37	35.5	25.7	20.6	64.9	44.1	40	30.1	17.2	57.3	34.5	19.3
Model 5	39.5	39.2	28.1	24.2	66.8	46.2	42.9	32.9	19.4	58.2	37.5	11.4
Model 6	40.8	39.4	28.6	25.6	67.6	48.1	44.5	33.9	21	61.2	38.6	9.4

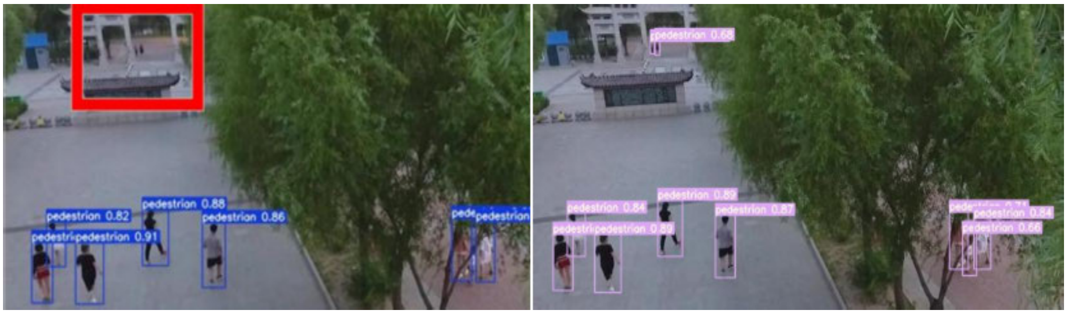

FIGURE 8. These are the visualization results of object detection. The image on the left is from YOLOv7, while the one on the right is from YOLOAL. Pedestrians in the distance and children obscured by branches were not detected by YOLOv7.

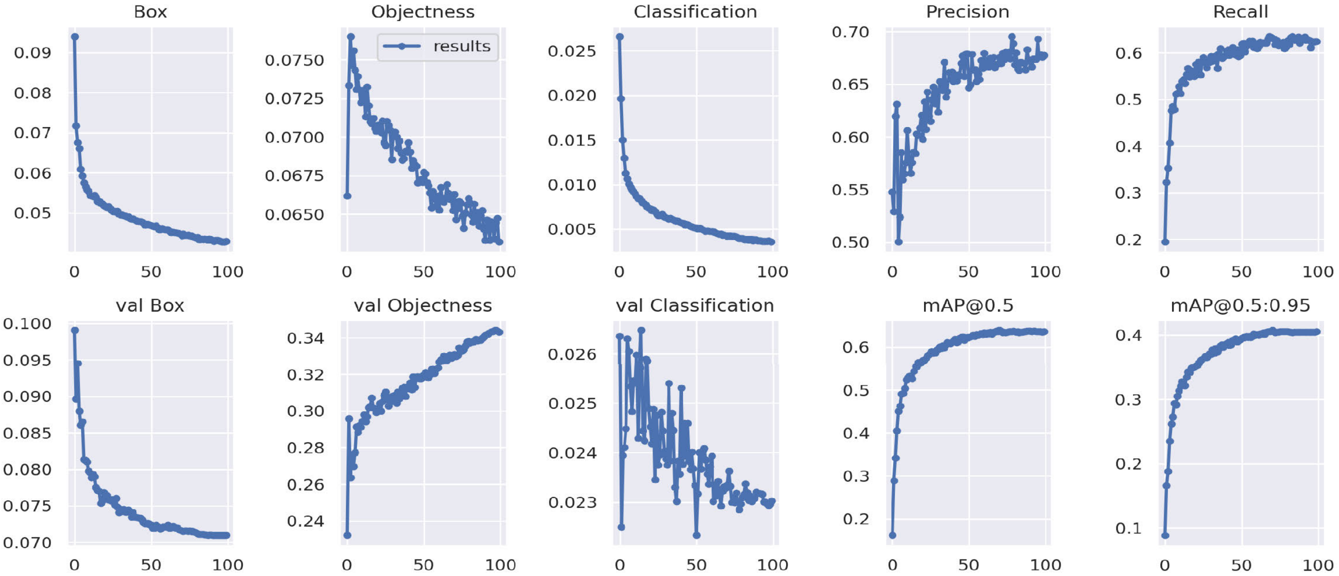

FIGURE 9. Graphs of training results for the VisDrone dataset.

F. ABLATION EXPERIMENTS

We list the mAP of six different models for each class and compare them with each other in Table 5. To ensure the uniqueness and comparability of the experimental results, we kept the pre-trained weights and training epochs consistent across all the models. However, we varied the model architecture and image size during training. The size of the training image was set as the same as the detection size. In this way, we obtained the experimental results with reference values.

Table 5 shows the mAP of different models in different categories of pedestrian, people, bicycle, car, van, trunk, tricycle, awning-tricycle, bus, and motor on the VisDrone validation dataset. During the testing phase, the results for batches 1 and 32, and found no significant difference in object detection accuracy between these two methods. When experimenting with batch 32, the FPS obtained was slightly higher than that of batch 1, but the difference was negligible. We used the batch size 1 throughout our experiments to maintain consistency.

(1)Model 1 uses the input image size of 1024 and uses the baseline model to train. (2) Model 2 uses the input image size of 1024 and uses the baseline model with CCAM to train. (3) Model 3 uses the input image size of 1024 and uses the baseline model with EIoU and Alpha 2 to train. (4) Model 4 uses the input image size of 1024 and uses the baseline model with CCAM, EIoU, and Alpha 2 to train. (5) Model 5 uses the input image size of 1536 and uses the baseline model to train. (6) Model 6 uses the input image size of 1536 and uses the baseline model with CCAM, EIoU, and Alpha 2 to train.

From the comparison of the data in Table 5, we found that the CCAM can improve the accuracy of object detection and increase the number of model layers from 362 to 404, while slightly reducing the FPS. Although there was a slight drop in speed, using CCAM resulted in a noteworthy improvement of 0.6% in accuracy when the image size was set to 1024. Therefore, we believe that it is worth implementing.

Compared with CCAM, the improved loss function achieves even higher detection accuracy improvement without changing the number of layers in the object detection model and slightly improves the FPS.

After combining the two methods, we observed a substantial boost in object detection accuracy, but also experienced a tiny decrease in FPS. When we increased the image size of the training and test phases from 1024 to 1536, the object detection accuracy significantly improved. When the image size was 1536, our model obtained an mAP score of 40.8%, which is a 1.3% improvement compared to the baseline. Whether the image size is either 1024 or 1536 pixels, our model performs better than the baseline and consistently delivers better results across various types of objects.

Figure 8 shows the comparison results of YOLOv7 and YOLOAL. The object confidence threshold is 0.5.

Figure 9 shows the changes in the evaluation metrics during training. The Box is the mean of the loss function, and a smaller value indicates that the difference between the true BBox and prediction BBox is smaller. The values of Precision and Recall increase as the number of epochs increases, which means that the proposed method produces better model parameters when optimizing the neural network. mAP@0.5 is AP50, mAP@0.5:0.95 is mAP. These two values represent the performance of the model for object detection. The higher the value, the better the performance.

V. CONCLUSION

Inspired by the convolution block attention module and loss function. In this paper, a one-stage detection model, YOLOAL, is proposed by developing a lightweight network. To address the challenge of identifying small objects in drone-captured images, we design a new attention module, CCAM, which enhances recognition ability by introducing a coordinate mechanism into the module. To address the issue that the width and height of CIoU cannot converge simultaneously, we use the EIoU loss function combined with Alpha-IoU, which improves the detection speed and accuracy. The VisDrone and DOTA datasets are used to train, validate, and test the model. The detection accuracy is 5.11% higher than that of the classic two-stage object detection algorithm SAIC-FPN and the speed is 9 times faster. Through the experiment, we can see that our YOLOAL performs well in drone-captured scenarios.

To further improve YOLOAL, we will improve CCAM and explore loss function enhancement methods. What's more, we will try to use stronger backbone to improve feature acquisition capabilities.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1–9.

[2] 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.

[3] 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.

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

[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[7] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollr, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Computer Vision—ECCV*. Zurich, Switzerland: Springer, 2014, pp. 740–755.

[8] 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.

[9] G. Jocher, A. Chaurasia, and J. Qiu. *YOLO By Ultralytics (Version 8.0.0)*. Accessed: 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[10] G. Jocher. *YOLOv5 by Ultralytics (Version 7.0)*. Accessed: 2022. [Online]. Available: https://github.com/ultralytics/yolov5

[11] 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.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[13] 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.

[14] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[15] C.-Y. Wang, H.-Y. Mark 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.

[16] A. Silva, M. Basso, P. Mendes, D. Rosário, E. Cerqueira, B. J. G. Praciano, J. P. J. da Costa, and E. P. de Freitas, "A map building and sharing framework for multiple UAV systems," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2022, pp. 1333–1342.

[17] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Computer Vision—ECCV*. Amsterdam, The Netherlands: Springer, 2016, pp. 354–370.

[18] C. Li, T. Yang, S. Zhu, C. Chen, and S. Guan, "Density map guided object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 737–746.

[19] F. Yang, H. Fan, P. Chu, E. Blasch, and H. Ling, "Clustered object detection in aerial images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8310–8319.

[20] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 2778–2788.

[21] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.

[22] Y. Li, S. Li, H. Du, L. Chen, D. Zhang, and Y. Li, "YOLO-ACN: Focusing on small target and occluded object detection," *IEEE Access*, vol. 8, pp. 227288–227303, 2020.

[23] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[24] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[25] L. Yang, R. Y. Zhang, L. Li, and X. Xie, "SimAM: A simple, parameter-free attention module for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 11863–11874.

[26] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612.

[27] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11531–11539.

[28] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13708–13717.

[29] M. A. Rahman and Y. Wang, "Optimizing intersection-over-union in deep neural networks for image segmentation," in *Proc. Int. Symp. Vis. Comput.*, 2016, pp. 234–244.

[30] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 658–666.

[31] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: Faster and better learning for bounding box regression," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 7, pp. 12993–13000.

[32] Z. Gevorgyan, "SIoU loss: More powerful learning for bounding box regression," 2022, *arXiv:2205.12740*.

[33] Y.-F. Zhang, W. Ren, Z. Zhang, Z. Jia, L. Wang, and T. Tan, "Focal and efficient IoU loss for accurate bounding box regression," *Neurocomputing*, vol. 506, pp. 146–157, Sep. 2022.

[34] C.-Y. Wang, A. Bochkovskiy, and H. M. Liao, "Scaled-YOLOv4: Scaling cross stage partial network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13024–13033.

[35] C.-Y. Wang, H.-Y. Mark Liao, and I.-H. Yeh, "Designing network design strategies through gradient path analysis," 2022, *arXiv:2211.04800*.

[36] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," 2015, *arXiv:1505.00853*.

[37] J. He, S. Erfani, X. Ma, J. Bailey, Y. Chi, and X. S. Hua, "Alpha-IoU: A family of power intersection over union losses for bounding box regression," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 20230–20242.

[38] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7380–7399, Nov. 2022.

[39] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3974–3983.

[40] J. Xu, Y. Li, and S. Wang, "AdaZoom: Adaptive zoom network for multi-scale object detection in large scenes," 2021, *arXiv:2106.10409*.

[41] Y. Wang, Y. Yang, and X. Zhao, "Object detection using clustering algorithm adaptive searching regions in aerial images," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 651–664.

[42] X. Zhang, E. Izquierdo, and K. Chandramouli, "Dense and small object detection in UAV vision based on cascade network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 118–126.

[43] Y. Liu, F. Yang, and P. Hu, "Small-object detection in UAV-captured images via multi-branch parallel feature pyramid networks," *IEEE Access*, vol. 8, pp. 145740–145750, 2020.

[44] J. Zhou, C.-M. Vong, Q. Liu, and Z. Wang, "Scale adaptive image cropping for UAV object detection," *Neurocomputing*, vol. 366, pp. 305–313, Nov. 2019.

[45] C. Zhao, Y. Song, X. Yang, Y. Zhou, and J. Yang, "Target detection based on multi-scale feature fusion and cross-channel interactive attention mechanism," *J. Phys., Conf. Ser.*, vol. 2562, no. 1, Aug. 2023, Art. no. 012046.

[46] H. Qin, Y. Wu, F. Dong, and S. Sun, "Dense sampling and detail enhancement network: Improved small object detection based on dense sampling and detail enhancement," *IET Comput. Vis.*, vol. 16, no. 4, pp. 307–316, Jun. 2022.

[47] J. C. Lee, J. Yoo, Y. Kim, S. Moon, and J. H. Ko, "Robust detection of small and dense objects in images from autonomous aerial vehicles," *Electron. Lett.*, vol. 57, no. 16, pp. 611–613, Aug. 2021.

XINTING CHEN received the B.E. degree from the Wuhan University of Technology, Wuhan, China, in 2017. He is currently pursuing the master's degree with the School of Cyber Security and Computer, Hebei University. His advisor is Prof. Wenzhu Yang. His research interest includes small object detection in images.

WENZHU YANG received the Ph.D. degree in electronic engineering from China Agricultural University, Beijing, China, in 2010. He is currently a Professor with the School of Cyber Security and Computer, Hebei University. His research interests include video analysis, machine learning, and image processing. He is a member of the IFIP and the Agricultural Engineering Society, an Editorial Board Member of the *International Journal of Psychology* (IPA), and a reviewer of several international journals. Over the past years, he has (co-)chaired several National Natural Science Foundation of China.

SHUANG ZENG received the B.E. degree in computer science and technology from the Wuhan University of Science and Technology, Wuhan, China, in 2021. She is currently pursuing the master's degree with the School of Cyber Security and Computer, Hebei University. Her current research interest includes object detection in computer vision.

LEI GENG received the B.E. degree from Hebei University, Hebei, China, in 2021, where she is currently pursuing the master's degree with the School of Cyber Security and Computer. She worked on problems in computer vision. Her research interest includes action recognition.

YANYAN JIAO received the B.E. degree from Zhengzhou University, Zhengzhou, China, in 2021, where she worked on software engineering and deep learning. She is currently pursuing the master's degree with the School of Cyber Security and Computer, Hebei University. Her research interest includes video understanding.