

# ViT-YOLO:Transformer-Based YOLO for Object Detection

Zixiao Zhang, Xiaoqiang Lu, Guojin Cao, Yuting Yang, Licheng Jiao, Fang Liu  
School of Artificial Intelligence, Xidian University  
Xi'an, Shaanxi Province, 710071, China

{zhangzx1999, xqlu, 20181214133, Ytyang\_1}@stu.xidian.edu.cn lchjiao@mail.xidian.edu.cn

## Abstract

*Drone captured images have overwhelming characteristics including dramatic scale variance, complicated background filled with distractors, and flexible viewpoints, which pose enormous challenges for general object detectors based on common convolutional networks. Recently, the design of vision backbone architectures that use self-attention is an exciting topic. In this work, an improved backbone MHSA-Darknet is designed to retain sufficient global context information and extract more differentiated features for object detection via multi-head self-attention. Regarding the path-aggregation neck, we present a simple yet highly effective weighted bi-directional feature pyramid network (BiFPN) for effectively cross-scale feature fusion. In addition, other techniques including time-test augmentation (TTA) and wighted boxes fusion (WBF) help to achieve better accuracy and robustness. Our experiments demonstrate that ViT-YOLO significantly outperforms the state-of-the-art detectors and achieve one of the top results in VisDrone-DET 2021 challenge (39.41 mAP for test-challenge data set and 41 mAP for the test-dev data set).*

## 1. Introduction

The goal of object detection is to predict a set of bounding boxes and category labels for each object of interest. Recently, with the advent of Unmanned Aerial Vehicles (UAV), drones equipped with cameras have been fast deploys to a wide range of applications, which include agriculture, aerial photography, fast delivery, surveillance, etc. Hence, automatic and effective object detection plays an important role in scene parsing on UAV platforms.

However, as shown in Figure 1, drone captured images have overwhelming characteristic including dramatic scale variance, complicated background filled with distractors, and flexible viewpoints, which pose enormous challenges for general object detectors based on common convolutional networks.

Convolutional neural network (CNN) have achieved

great breakthrough in various kinds of fields in computer vision [8]. Generally, modern detectors employ pure convolution network to extract features. Classical image classification networks (e.g., VGG [29], ResNet [10] are used as the backbones for state-of-the-art detectors Faster RCNN [27] and RetinaNet [17], etc.) As for YOLO series detectors [24], they apply a newfangled residual network the Darknet which is much more efficient for performing feature extraction.

Nowadays, transformers [32] have become the dominant model in natural language processing owing to their ability to learn complex dependencies between input sequences via self-attention. We also observe that the recently introduced vision transformers [6] achieve competitive results on benchmark classification tasks by treating an image as a sequence of patches. For drone captured images with large-scale and complex scene, to improve the semantic discriminability and alleviate category confusion, collecting and associating scene information from a large neighborhood can be useful in learning relationships across objects. But for convolutional networks, the locality of the convolution operation limits its capacity for capturing global context information. In contrast, transformers are capable of globally focusing on dependencies between image feature patches and retain sufficient spatial information for object detection via multi-head self-attention.

In addition, to handle the problem of viewpoint changing in aerial images, the object detector should have enhanced domain adaptation capability and dynamic receptive fields. The study in the literature [22] has indicated that vision transformers are much more highly robust to severe occlusions, perturbations and domain shifts, compared to CNNs. Therefore, an intuitive way for enhancing the detection performance is to embed the transformer layer into the purely convolutional backbone to bring more context information and learn more distinguishable feature representations.

On the other hand, objects from the drone captured images vary a lot in sizes while the feature map from a single layer of the convolutional neural network has limited capacity of representation, so it is crucial to effectively represent



Figure 1. The challenges in the UAV vision.

and process multi-scale features. A classical method is to combine low-level and high-level features through a summation or concatenation operation, but simply summing up or concatenating without distinction may cause feature mismatch and performance degradation. Our key insight is to introduce learnable weights to learn the importance of different input features, while repeatedly applying top-down and bottom-up multi-scale feature fusion.

In this paper, we mainly follow the one-stage detector design and propose a hybrid detector called ViT-YOLO. The framework integrates the CSP-Darknet [1] and multi-head self-attention [32] for feature extraction. In addition, the architecture interfaces with BiFPN [31] for effectively combining the features at different scales. Subsequently, the YOLOv3 coupled head [26] is employed for final bounding boxes classification and regression tasks. Furthermore, We implement effective strategies including Test-Time Augmentation (TTA) and Weighted Boxes Fusion (WBF) [30] to improve detection performance. Our experiment demonstrates that the improved network architecture significantly outperforms the existing state-of-the-art detectors on the VisDrone2019 test-challenge dataset with mAP 39.41.

In summary, the main contributions of this paper are:

- We introduce a multi-head self-attention block in the original backbone CSP-Darknet to bring more context information and learn more distinguishable feature representations.
- We present a simple yet highly effective weighted bi-directional feature pyramid network (BiFPN) for effectively cross-scale feature fusion.
- We implement effective strategies including Test-Time Augmentation (TTA) and Weighted Boxes Fusion (WBF) to achieve competitive performance on VisDrone2019 benchmark dataset.

## 2. Related work

**General object detection:** With the development of deep learning [14], various object detection algorithms have been proposed. Existing object detectors are mostly categorized by whether they have a region-of-interest proposal

step (two-stage [27, 7, 9, 3]) or not (one-stage [25, 24, 17, 28]). Recently, following the one-stage detector design, YOLO series [24, 25, 1, 26] have attracted substantial attention due to their efficiency and simplicity. They extract the most advanced detection technologies available at the time (e.g., the SPP module [11] for YOLOv3 [26], Mish activation [21] for YOLOv4 [1]) and optimize the implementation for best practice. Hence, we selected YOLOv4 as our baseline model.

**Vision Transformer:** Nowadays, The Transformer [32] model has become the preferred solution for a wide range of natural language processing (NLP) tasks, showing impressive progress in machine translation [15], question answering [5], text classification [23], document summarization [33], and more. Part of this success comes from the Transformer’s ability to learn complex dependencies between input sequences via self-attention. The Vision Transformer (ViT) [6] demonstrated for the first time that a transformer architecture can be directly applied to images as well, by treating an image as a sequence of patches, which performs comparably to state-of-the-art convolutional networks on image recognition tasks. DETR [4] is notable in that it is the first approach to successfully utilize transformers for the object detection task. Specifically, DETR added a transformer encoder and decoder on top of a standard CNN model (e.g., ResNet-50/101), and uses a set-matching loss function.

**Multi-scale feature fusion:** One of the main difficulties in object detection is to effectively represent and process multi-scale features. Earlier detectors often directly perform predictions based on the pyramidal feature hierarchy extracted from backbone networks [2, 28, 20]. As one of the pioneering works, feature pyramid network (FPN) [16] proposes a top-down pathway to combine multi-scale features. Following this idea, PANet [19] adds an extra bottom-up path aggregation network on top of FPN. The recent detector EfficientDet [31] propose BiFPN to introduce learnable weights to learn the importance of different input features.

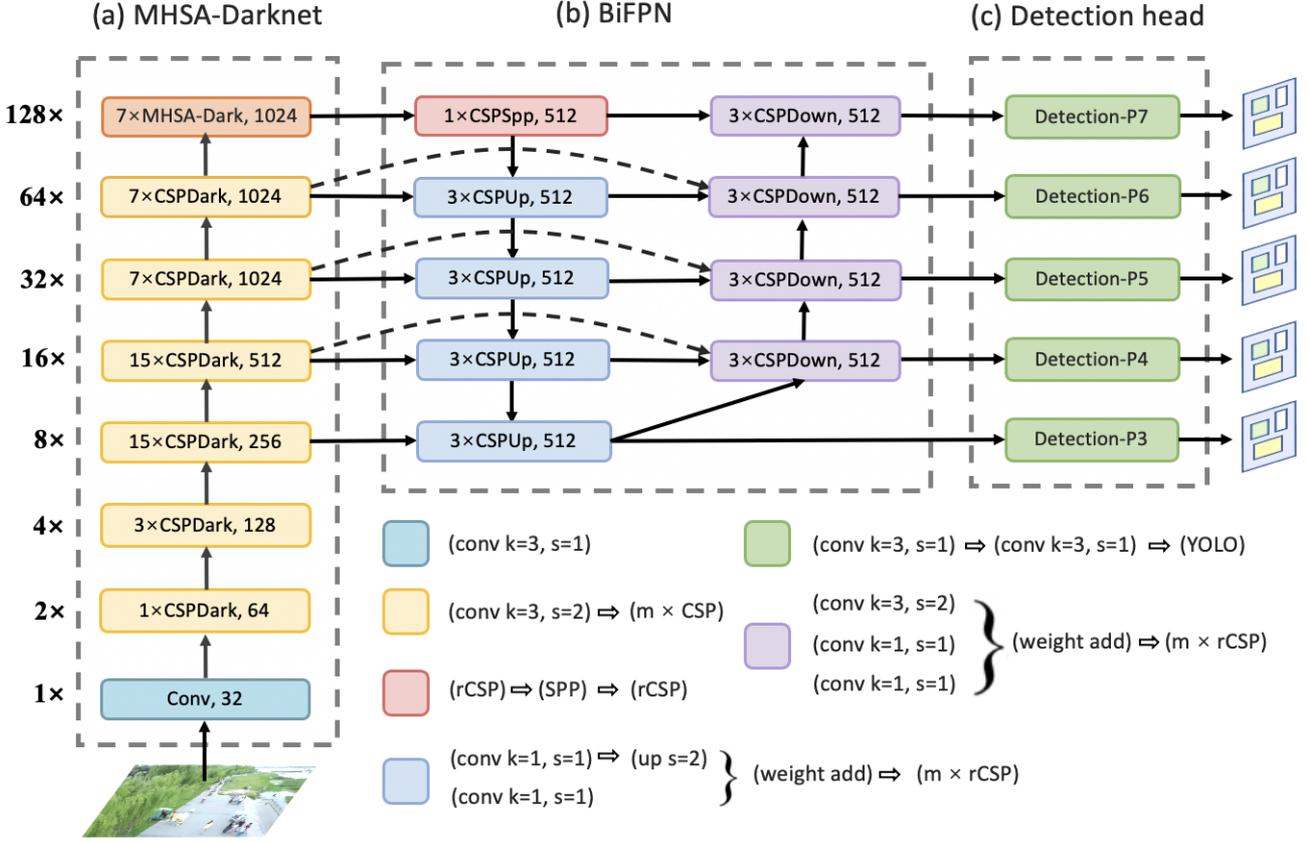


Figure 2. The whole network structure. An input image will be input to the backbone (a) MHSADarknet, which integrates transformer layer into CSP Darknet, where the MHSADark block and the CSPDark block are described in Figure 3. The feature maps further refine with (b) BiFPN, which aggregates features from different backbone levels for different detector levels. Finally, (c) YOLO detection head is employed to predict boxes at 5 different scales.

### 3. Proposed Method

The proposed network architecture is a hybrid model ViT-YOLO that uses both convolution and self-attention, which is mainly based on the YOLOv4-P7 [1]. The structure of ViT-YOLO is presented in Figure 2, which is divided into 3 parts. For the first part, we use MHSADarknet as the backbone which integrates multi-head self-attention into original CSP-Darknet to extract more differentiated features. The details of MHSADarknet is described in Section 3.1. The second processing component BiFPN in substitution for PANet aims to aggregate features from different backbone levels for different detector levels, which is discussed in Section 3.2. For the third part, the general YOLO detection heads are employed for predicting boxes at 5 different scales.

Furthermore, other effective techniques are adopted to achieve better accuracy and robustness, including Test-Time Augmentation (TTA) and Weighted Boxes Fusion (WBF). TTA is an application of data augmentation to the test dataset. And WBF utilizes confidence scores of all pro-

posed bounding boxes to construct the averaged boxes so that it works better when used for model fusion.

#### 3.1. MHSADarknet

For drone captured images with large scale and complex scene, to improve the semantic discriminability and alleviate category confusion, collecting and associating scene information from a large neighborhood can be useful in learning relationships across objects. But for convolutional networks, the locality of the convolution operation limits its capacity for capturing global context information. In contrast, transformers are capable of globally focusing on dependencies between image feature patches and retain sufficient spatial information for object detection via multi-head self-attention. On the other hand, viewpoint variation is one of the biggest challenges in images captured from drones, which requests that the detector should have an enhanced domain adaptation capability and dynamic receptive fields. The study in the literature [22] has indicated that vision transformers are much more highly robust to

Stage	output	CSP Darknet		MHSA Darknet	
P0	/2	$3 \times 3, 32$ $3 \times 3, 64, \text{stride} = 2$		$3 \times 3, 32$ $3 \times 3, 64, \text{stride} = 2$	
P1	/4	$1 \times 1, 32$ $\begin{bmatrix} 1 \times 1, 16 \\ 3 \times 3, 32 \end{bmatrix} \times 1$ $1 \times 1, 32$	$1 \times 1, 32$	$1 \times 1, 32$ $\begin{bmatrix} 1 \times 1, 16 \\ 3 \times 3, 32 \end{bmatrix} \times n$ $1 \times 1, 32$	$1 \times 1, 32$
		$\text{concat}, 64$ $3 \times 3, 128, \text{stride} = 2$		$\text{concat}, 64$ $3 \times 3, 128, \text{stride} = 2$	
P2	/8	$1 \times 1, 64$ $\begin{bmatrix} 1 \times 1, 32 \\ 3 \times 3, 64 \end{bmatrix} \times 3$ $1 \times 1, 64$	$1 \times 1, 64$	$1 \times 1, 64$ $\begin{bmatrix} 1 \times 1, 32 \\ 3 \times 3, 64 \end{bmatrix} \times 3$ $1 \times 1, 64$	$1 \times 1, 64$
		$\text{concat}, 128$ $3 \times 3, 256, \text{stride} = 2$		$\text{concat}, 128$ $3 \times 3, 256, \text{stride} = 2$	
P3	/16	$1 \times 1, 128$ $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 128 \end{bmatrix} \times 15$ $1 \times 1, 128$	$1 \times 1, 128$	$1 \times 1, 128$ $\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 128 \end{bmatrix} \times 3$ $1 \times 1, 128$	$1 \times 1, 128$
		$\text{concat}, 256$ $3 \times 3, 512, \text{stride} = 2$		$\text{concat}, 256$ $3 \times 3, 512, \text{stride} = 2$	
P4	/32	$1 \times 1, 256$ $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 256 \end{bmatrix} \times 15$ $1 \times 1, 256$	$1 \times 1, 256$	$1 \times 1, 256$ $\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 256 \end{bmatrix} \times 15$ $1 \times 1, 256$	$1 \times 1, 256$
		$\text{concat}, 512$ $3 \times 3, 1024, \text{stride} = 2$		$\text{concat}, 512$ $3 \times 3, 1024, \text{stride} = 2$	
P5	/64	$1 \times 1, 512$ $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 512 \end{bmatrix} \times 7$ $1 \times 1, 512$	$1 \times 1, 512$	$1 \times 1, 512$ $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 512 \end{bmatrix} \times 7$ $1 \times 1, 512$	$1 \times 1, 512$
		$\text{concat}, 1024$ $3 \times 3, 1024, \text{stride} = 2$		$\text{concat}, 1024$ $3 \times 3, 1024, \text{stride} = 2$	
P6	/128	$1 \times 1, 512$ $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 512 \end{bmatrix} \times 7$ $1 \times 1, 512$	$1 \times 1, 512$	$1 \times 1, 512$ $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 512 \end{bmatrix} \times 7$ $1 \times 1, 512$	$1 \times 1, 512$
		$\text{concat}, 1024$ $3 \times 3, 1024, \text{stride} = 2$		$\text{concat}, 1024$ $3 \times 3, 1024, \text{stride} = 2$	
P7	/128	$1 \times 1, 512$ $\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 512 \end{bmatrix} \times 7$ $1 \times 1, 512$	$1 \times 1, 512$	$1 \times 1, 512$ $[MHSA, 512] \times 7$ $1 \times 1, 512$	$1 \times 1, 512$
		$\text{concat}, 1024$		$\text{concat}, 1024$	

Table 1. Architecture of MHSA-Darknet: The only difference from the original CSPDarknet is MHSA-Dark block in substitution for the CSPDark block in P7 ( Figure 3). For an input resolution of  $1024 \times 1024$ , the MHSA layers of P7 operate on  $8 \times 8$ .

severe occlusions, perturbations and domain shifts, compared to CNNs. In order to improve the transferability of the learned features and meanwhile to capture long-distance context information, we propose the MHSA-Darknet backbone to extract features for the detector. MHSA-Darknet

by design is simple: embed the Multi-Head Self-attention (MHSA) layers into the top CSPDark block to implement global (all2all) self-attention over a 2D feature map. The MHSA-Darknet architecture is described in Table 1 and the MHSA layer is presented in Figure 4. A CSP-Darknet in

YOLOv4-P7 typically has 7 stages (or block groups) commonly referred to as  $[P1, P2, P3, P4, P5, P6, P7]$  with strides  $[2, 4, 8, 16, 32, 64, 128]$  relative to the input image, respectively. Stacks  $[P1, P2, P3, P4, P5, P6, P7]$  consist of multiple CSPDark blocks with Cross Stage Partial (CSP) connections. (i.e. CSP-Darknet in YOLOv4-P7 has  $[1, 3, 15, 15, 7, 7, 7]$  CSPDark blocks).

Notably, when the network is relatively shallow and the feature map is relatively larger, the transformer layer is used prematurely to enforce regression boundaries which can lose some meaningful context information. Hence, in the MHSA-Darknet, the transformer layer is only applied on the  $P7$ , rather than  $P3, P4, P5$ , and  $P6$ . Besides, considering that self-attention when performed globally across  $n$  entities requires  $O(n^2d)$  memory and computation[32], we believe that the simplest setting that adheres to the above factors would be to incorporate self-attention at the lowest resolution feature maps in the backbone, i.e., the CSPDark blocks in the  $P7$  stack. The  $P7$  stack in the Darknet backbone typically uses 7 CSP bottleneck blocks with one spatial  $1 \times 1$  convolution and one spatial  $3 \times 3$  convolution in each. Replacing them with MHSA layers forms the basis of the MHSA-Darknet architecture.

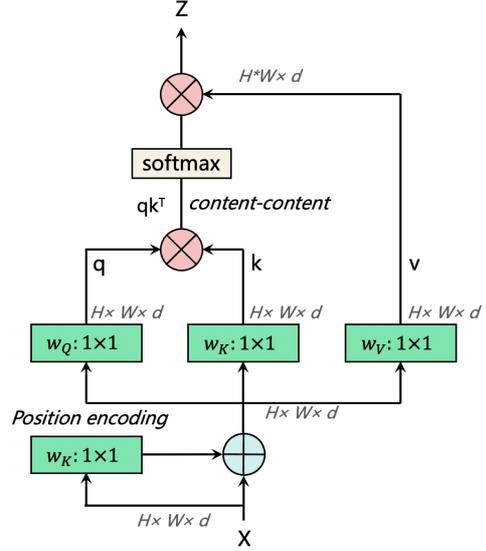


Figure 4. Multi-Head Self-Attention (MHSA) layer used in the MHSA-Dark block. While we use 4 heads, we do not show them on the figure for simplicity. A standard learnable 1D position embeddings with a linear layer are employed to retain positional information. The attention logits are  $qk^T$  where  $q, k$  represent query, key.  $\oplus$  and  $\otimes$  represent element wise sum and matrix multiplication respectively, while  $1 \times 1$  represents a pointwise convolution.

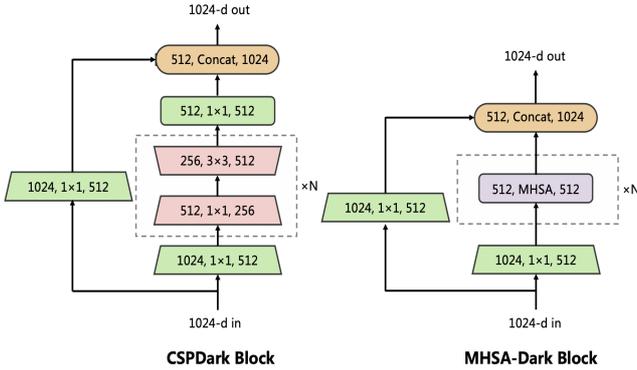


Figure 3. Left: A CSPDark Block Right: A MHSA-Dark Block. The only difference is in  $P7$  where the Multi-Head Self-attention (MHSA) with  $n$  layers replace  $n$  CSP Bottlenecks, each of which consist of one  $3 \times 3$  spatial convolution and one  $1 \times 1$  spatial convolution. The structure of self-attention layer is described in Figure 4.

To handle 2D images, we flatten the spatial dimensions of the 2D feature map  $x \in \mathbb{R}^{H \times W \times d}$  into a sequence  $x_p \in \mathbb{R}^{(H \times W) \times d}$ , where  $(H, W)$  is the resolution of the original feature map,  $d$  is the number of channels, and  $H * W$  serves as the effective input sequence length for the transformer layer. In order to make the attention operation position aware, Transformer based architectures typically make use of a position encoding[32]. We use standard learnable 1D position embeddings with a linear layer to retain positional information. The MHSA layer is presented in Figure 4.

### 3.2. BiFPN

Objects from the drone captured images vary a lot in sizes while the feature map from a single layer of the convolutional neural network has limited capacity of representation, so it's crucial to effectively represent and process multi-scale features. Conventional top-down FPN [16] is inherently limited by the one-way information flow. To address this issue, PANet [19] adds an extra bottom-up path aggregation network, as shown in Figure 5(a). Cross-scale connections are further studied in [13, 12, 34]. In this work, the simple yet highly effective weighted bidirectional feature pyramid network (BiFPN), as shown in Figure 5(b), implements two optimizations for cross-scale connections. First, BiFPN adds an extra edge from original input to output node if they are at the same level, in order to fuse more features without adding much cost.

Second, while combining low-level and high-level features, BiFPN introduces learnable weights to learn the importance of different input features instead of simply summing up or concatenating, which may cause feature mismatch and performance degradation. Formally, given a list of multi-scale features  $\vec{P}^{in} = (P_{l_1}^{in}, P_{l_2}^{in}, \dots)$ , where  $P_{l_i}^{in}$  presents the feature at level  $l_i$ . The list of intermediate feature on the pathway is represented as  $\vec{P}^{td} = (P_{l_1}^{td}, P_{l_2}^{td}, \dots)$ . Our goal is to find a transformation  $f$  that can effectively aggregate different features and output a list of new features:  $\vec{P}^{out} = f(\vec{P}^{in})$ . Figure 5(a) shows the conven-

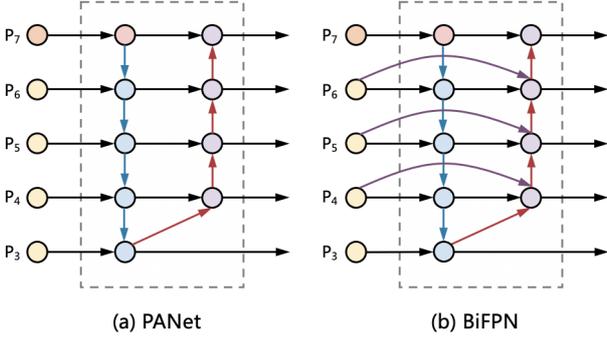


Figure 5. Feature network design (a) PANet adds an additional bottom-up pathway on top of FPN. (b) BiFPN implements two optimizations for cross-scale connections.

tional top-down and bottom-up PANet [19]. It takes level 3 – 7 input features  $\bar{P}^{in} = (P_3^{in}, \dots, P_7^{in})$ , where  $P_i^{in}$  represents a feature level with resolution of  $1/2^i$  of input images. For instance, if input resolution is  $1024 \times 1024$ , then  $P_3^{in}$  represents feature level 3 ( $1024/2^3 = 128$ ) with resolution  $128 \times 128$ , while  $P_7^{in}$  represents feature level 7 with resolution  $8 \times 8$ . The conventional PANet aggregates multi-scale features in a simple summing manner, as layer 6 for example:

$$\begin{aligned} P_6^{td} &= Conv(P_6^{in} + Resize(P_7^{td})) \\ P_6^{out} &= Conv(P_6^{td} + Resize(P_5^{out})) \end{aligned} \quad (1)$$

where, *Resize* is usually a upsampling or downsampling operation for resolution matching, and *Conv* is usually a convolutional operation for feature processing.

While the BiFPN integrates both the bidirectional cross-scale connections and the fast normalized fusion with learnable weights. As a concrete example, here we describe the two fused features at level 6 for BiFPN shown in Figure 2(b):

$$\begin{aligned} P_6^{td} &= Conv\left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot Resize(P_7^{in})}{w_1 + w_2 + \varepsilon}\right) \\ P_6^{out} &= Conv\left(\frac{w'_1 \cdot P_6^{in} + w'_2 \cdot Resize(P_6^{td}) + w'_3 \cdot Resize(P_5^{out})}{w_1 + w_2 + w_3 + \varepsilon}\right) \end{aligned} \quad (2)$$

where,  $P_6^{td}$  is the intermediate feature at level 6 on the top-down pathway, and  $P_6^{out}$  is the output feature at level 6 on the bottom-up pathway. All other features are constructed in a similar manner. Notably, the difference from the original BiFPN proposed in EfficientDet [31] is that SPP additional module is employed in the path-aggregation neck to enhance the intermediate feature and that Cross Stage Partial (CSP) connections are used in place of simple convolution for feature processing.

## 4. Experiments

### 4.1. Datasets

VisDrone2019-Det benchmark dataset [35] consists of 10209 static images captured by drone platforms in unconstrained challenging scenes, including 6741 images in the training subset, 548 in the validation subset, 1610 in the test-dev subset, and 1580 in the test-challenge subset. Drone captured images have overwhelming characteristic including dramatic scale variance, complicated background filled with distractors, and flexible viewpoints. Images are manually labeled with bounding boxes and ten predefined classes (i.e., pedestrian, person, car, van, bus, truck, motor, bicycle, awning-tricycle, and tricycle). All models in this paper are trained on training set and evaluated on the test-dev set.

### 4.2. Evaluations Metrics

Similar to the evaluation protocol in MS COCO benchmark [18], we adopt metrics including *AP*, *AP50*, *AP75*, *AR1*, *AR10*, *AR100* and *AR500* to evaluate detectors by penalizing missing detections and false alarms. Specifically, *AP* is the average of all 10 intersection over union (IoU) thresholds in the range [0.50 : 0.95] with uniform step size 0.05 of all categories, which is used as the primary metric for ranking.

### 4.3. Implementation Details

In VisDrone-DET2021 challenge, we choose MHSA-Darknet backbone, BiFPN path-aggregation neck, and YOLOv3 (anchor based) head as the architecture of ViT-YOLO. Our model uses SGD as optimizer, with a weight decay of 0.0005 and momentum of 0.937 as default. In the initial training of the model, we first perform 3 epoch warm-up training. During the warm-up process, the momentum of the optimizer SGD is set to 0.8, and one-dimensional linear interpolation is used to update the learning rate of each iteration. After warm-up training, the cosine annealing function is used to attenuate the learning rate, where the initial learning rate is 0.02, and the minimum learning rate is  $0.2 \cdot 0.01$ . Finally, we train the model for 300 epochs.

### 4.4. Experimental Results

Table 2 shows our results evaluated on the VisDrone2019 test-dev dataset. Among those state-of-the-art methods, all YOLO family algorithms hold obviously outstanding performance. As a modified version of the superior model YOLOv4-P7, our proposed architecture shown in Figure 2 achieves competitive performance with *mAP* 38.5 without TTA and multi-model fusion, which is even 3.07 higher than the baseline YOLOv4-P7.

Method	AP	AP50	AP75	AR1	AR10	AR100	AR500
CornerNet	23.43	41.18	25.02	0.45	4.24	33.05	34.23
Light-RCNN	22.08	39.56	23.24	0.32	3.64	31.19	32.06
FPN	22.06	39.57	22.50	0.29	3.50	30.64	31.61
Cascade R-CNN	21.80	37.84	22.56	0.28	3.55	29.15	30.09
DetNet	20.07	37.54	21.26	0.26	2.84	29.06	30.45
RefineDet	19.89	37.27	20.18	0.24	2.76	28.82	29.41
RetinaNet	18.94	31.67	20.25	0.14	0.68	7.31	27.59
YOLOv5x6	32.19	55.33	33.06	2.24	13.31	41.74	46.45
YOLOv4-P7	35.43	58.94	36.43	2.35	14.66	44.28	49.61
<b>ours</b>	<b>38.5 (+3.07)</b>	<b>63.15</b>	<b>40.48</b>	<b>2.33</b>	<b>14.93</b>	<b>48.04</b>	<b>55.47</b>

Table 2. The Comparisons between the results of baseline methods and ViT-YOLO on VisDrone2019 test-dev dataset

Method	pedestrian	person	bicycle	car	van	trunk	tricycle	awning-tricycle	bus	motor
YOLOv4-P7 (Baseline)	24.19	13.72	15.68	56.73	37.66	44.87	22.78	19.74	53.74	24.98
+MHSA-Darknet	26.12	15.08	16.79	58.15	39.35	46.41	25.23	22.88	55.22	27.44
+BiFPN	26.86	16.01	17.22	58.74	41.11	47.63	26.98	24.21	56.69	28.12
+TTA	27.34	16.56	17.96	58.99	42.39	48.78	27.09	24.74	57.97	28.39
+Multi-model Fusion	28.43	17.18	18.64	60.93	44.07	51.54	28.87	26.32	59.63	29.95

Table 3. The Comparison between the results of ten categories after subsequent operations on VisDrone2019 test-dev dataset.

Method	AP	AP50	AP75
YOLOv4-P7 (Baseline)	35.43	58.94	36.43
+ MHSA-Darknet	37.56 (+2.1)	61.46	38.34
+ BiFPN	38.5 (+1.3)	63.15	40.48
+ TTA	39.32 (+0.8)	64.19	42.1
+ Multi-model Fusion	41 (+1.7)	65.89	43.14

Table 4. The Ablation study on the test-dev set.

## 4.5. Ablation Experiments

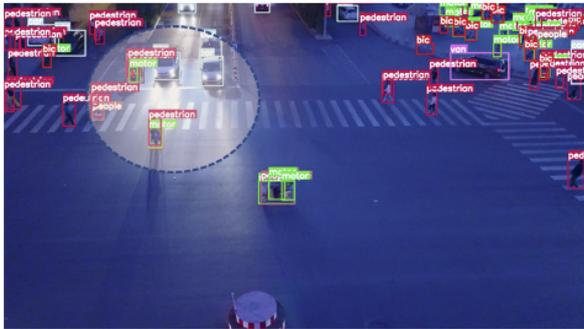
In this section, we perform a thorough ablation study on the VisDrone2019 test-dev subset to analyze our algorithm, which is shown in Table 4. On the basis of the baseline model YOLOv4-P7, we propose MHSA-Darknet as the backbone which embed multi-head self-attention into original CSP-Darknet, and adopt the simple yet highly effective weighted BiFPN path-aggregation neck in place of PANet. Other tricks including Test Time Augmentation (TTA) and Weighted Boxes Fusion (WBF) also helps to improve the detection performance. Table 4 shows the comparison between the results of ten categories.

### 4.5.1 MHSA-Darknet

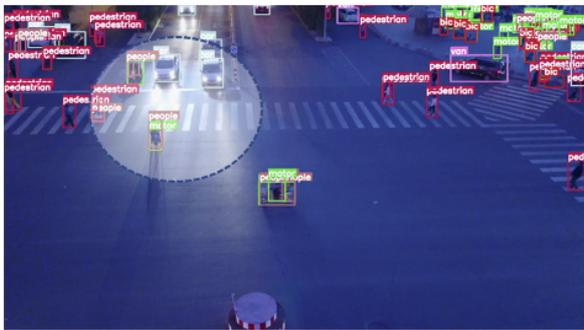
The evaluation of the proposed MHSA-Darknet is shown in Table 4 and Table 3. After integrating multi-head self-attention into the original CSP-Darknet, the total mAP of

results is significantly boosted from 35.43 To 37.56. And we see that there is a significant AP boost, particularly for small objects: 24.19 to 26.12 overall for pedestrians, 13.72 to 15.08 overall for persons, 22.78 to 25.23 overall for tri-cycle, 24.98 to 27.44 overall for motors. These results suggest that self-attention has a big effect in detecting small objects which is considered to be an important and hard problem for deploying object systems in the real world. For CNN-based deep networks, the high-level feature maps are of fairly low spatial resolution, so that lack of the unbroken information to localize the large objects accurately or recognize the small objects. While small objects always show the co-occurrences of certain classes in images, which to some extent explains the reason why the transformer-based model which focuses on global context information is helpful for accurately detecting small objects.

On the other hand, from the comparison between prediction results from the original baseline and our transformer-based model in Figure 6, we observe that our transformer-based model successfully recognize the people on motors while the baseline model wrongly classify them into pedestrians. It suggests that the improved backbone MHSA-Darknet is capable of extracting more differentiated features for object detection via multi-head self-attention and demonstrates a stronger semantic discriminability to alleviate category confusion.



YOLOv4-P7



YOLOv4-P7(MHSA)

Figure 6. Visualized detection results of YOLOv4-P7 and YOLOv4-P7(+MHSA) on an arbitrary image from Visdrone dataset. While not very different, YOLOv4-P7(+MHSA) is able to create precise localizations as well as detect small objects better. A notable example is exactly recognizing the people on the motor, instead of wrongly classifying them into pedestrians.

#### 4.5.2 Test-Time Augmentation

In VisDrone-DET2021 challenge, we use Test-time augmentation (TTA) to improve the performance of our method, which is an application of data augmentation to the test dataset. Specifically, we create multiple augmented copies of each image in the test set, having the model make a prediction for each, then returning an ensemble of those predictions. Copies of samples in the test dataset are created with some image manipulation techniques performed, such as zooms, flips, shifts, and more. As shown in Table 3, after TTA operation, the performance of every class improved to a large extent.

#### 4.5.3 Multi-model Fusion

Single detection models only select the boxes and can not produce averaged localization of predictions combined from various models effectively. Ensembles of models are widely used in applications that do not require real-time inference. Combining predictions from different models gen-

eralizes better and usually yields more accurate results compared to a single model.

In VisDrone-DET2021 challenge, we adopt an effective method for combining predictions of object detection models: Weighted Boxes Fusion (WBF). WBF utilizes confidence scores of all proposed bounding boxes to construct the averaged boxes so that it works better when used for the models’ ensemble. We used several models trained on visdrone dataset, including the YOLOv5 models, YOLOv4 models and our ViT-YOLO model to predict boxes. Then, we use the ensemble method WBF to obtain the final predictions. In Table 4, the combined predictions for the test-dev data set yields superior performance with mAP 41.

## 5. Conclusion

Drone captured images have overwhelming characteristics including dramatic scale variance, complicated background filled with distractors, and flexible viewpoints, which pose enormous challenges for general object detectors based on common convolutional networks. In this work, the improved backbone MHSA-Darknet dramatically enhances the detection performance, especially for small objects, and meanwhile exhibits a stronger semantic discriminability to alleviate category confusion. The ViT encoder’s ability to globally attend to the entire image is a possible explanation for this enhancement. Regarding the path-aggregation neck, we adopt the BiFPN to integrate both the bidirectional cross-scale connections and perform the fast normalized fusion with learnable weights, which help the model to achieve a more competitive performance. Furthermore, other techniques including time-test augmentation and multi-model fusion also improve the accuracy and robustness. Based on them, our algorithm significantly outperforms the existing state-of-the-art detectors on the VisDrone test-challenge dataset with mAP 39.41 in the ICCV VisDrone2021 Object Detection Challenge.

## References

- [1] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv:2004.10934 [cs, eess]*, Apr. 2020. arXiv: 2004.10934.
- [2] Zhaowei Cai, Quanfu Fan, Rogerio S. Feris, and Nuno Vasconcelos. A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection. *arXiv:1607.07155 [cs]*, July 2016. arXiv: 1607.07155.
- [3] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving Into High Quality Object Detection. pages 6154–6162, 2018.
- [4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. *arXiv:2005.12872 [cs]*, May 2020. arXiv: 2005.12872.

- [5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*, May 2019. arXiv: 1810.04805.
- [6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929 [cs]*, June 2021. arXiv: 2010.11929.
- [7] Ross Girshick. Fast R-CNN. pages 1440–1448, 2015.
- [8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *arXiv:1311.2524 [cs]*, Oct. 2014. arXiv: 1311.2524.
- [9] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. pages 2961–2969, 2017.
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*, Dec. 2015. arXiv: 1512.03385.
- [11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, Sept. 2015. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [12] Seung-Wook Kim, Hyong-Keun Kook, Jee-Young Sun, Mun-Cheon Kang, and Sung-Jea Ko. Parallel Feature Pyramid Network for Object Detection. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, volume 11209, pages 239–256. Springer International Publishing, Cham, 2018. Series Title: Lecture Notes in Computer Science.
- [13] Tao Kong, Fuchun Sun, Wenbing Huang, and Huaping Liu. Deep Feature Pyramid Reconfiguration for Object Detection. *arXiv:1808.07993 [cs]*, Aug. 2018. arXiv: 1808.07993.
- [14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015.
- [15] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *arXiv:1910.13461 [cs, stat]*, Oct. 2019. arXiv: 1910.13461.
- [16] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature Pyramid Networks for Object Detection. *arXiv:1612.03144 [cs]*, Apr. 2017. arXiv: 1612.03144.
- [17] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal Loss for Dense Object Detection. pages 2980–2988, 2017.
- [18] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. *arXiv:1405.0312 [cs]*, Feb. 2015. arXiv: 1405.0312.
- [19] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path Aggregation Network for Instance Segmentation. Mar. 2018.
- [20] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single Shot MultiBox Detector. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, pages 21–37, Cham, 2016. Springer International Publishing.
- [21] Diganta Misra. Mish: A Self Regularized Non-Monotonic Neural Activation Function. page 13.
- [22] Muzammal Naseer, Kanchana Ranasinghe, Salman Khan, Munawar Hayat, Fahad Shabbaz Khan, and Ming-Hsuan Yang. Intriguing Properties of Vision Transformers. *arXiv:2105.10497 [cs]*, June 2021. arXiv: 2105.10497.
- [23] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. page 67.
- [24] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. *arXiv:1506.02640 [cs]*, May 2016. arXiv: 1506.02640.
- [25] Joseph Redmon and Ali Farhadi. YOLO9000: Better, Faster, Stronger. pages 7263–7271, 2017.
- [26] Joseph Redmon and Ali Farhadi. YOLOv3: An Incremental Improvement. *arXiv:1804.02767 [cs]*, Apr. 2018. arXiv: 1804.02767.
- [27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, June 2017. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [28] Pierre Sermanet, David Eigen, Xiang Zhang, Michael Mathieu, Rob Fergus, and Yann LeCun. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *arXiv:1312.6229 [cs]*, Feb. 2014. arXiv: 1312.6229.
- [29] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv:1409.1556 [cs]*, Apr. 2015. arXiv: 1409.1556.
- [30] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, 107:104117, Mar. 2021. arXiv: 1910.13302.
- [31] Mingxing Tan, Ruoming Pang, and Quoc V. Le. EfficientDet: Scalable and Efficient Object Detection. *arXiv:1911.09070 [cs, eess]*, July 2020. arXiv: 1911.09070.
- [32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *arXiv:1706.03762 [cs]*, Dec. 2017. arXiv: 1706.03762.
- [33] Xingxing Zhang, Furu Wei, and Ming Zhou. hiBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization. *arXiv:1905.06566 [cs]*, May 2019. arXiv: 1905.06566.

- [34] Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, and Haibin Ling. M2Det: A Single-Shot Object Detector based on Multi-Level Feature Pyramid Network. *arXiv:1811.04533 [cs]*, Jan. 2019. arXiv: 1811.04533.
- [35] Pengfei Zhu, Longyin Wen, Dawei Du, Xiao Bian, Qinghua Hu, and Haibin Ling. Vision Meets Drones: Past, Present and Future. *arXiv:2001.06303 [cs]*, July 2020. arXiv: 2001.06303.