VistrongerDet: Stronger Visual Information for Object Detection in VisDrone Images

Junfeng Wan[1], Binyu Zhang[1], Yanyun Zhao[1,2], Yunhao Du[1], and Zhihang Tong[1]

[1]Beijing University of Posts and Telecommunications
[2]Beijing Key Laboratory of Network System and Network Culture, China,
{wanjunfeng, zhangbinyu, zyy, dyh_bupt, tongzh}@bupt.edu.cn

Abstract

*Existing methods are especially difficult to detect objects accurately in videos and images captured by UAV. In the work, we carefully analyze the characteristics of VisDrone DET 2021 dataset, and the main reasons for the low detection performance are tiny objects, wide scale span, long-tail distribution, confusion of similar classes. To mitigate the adverse influences caused thereby, we propose a novel detector named **VistrongerDet**, which possesses **Stronger V**isual **I**nformation. Our framework integrates the novel components of FPN level, ROI level and head level enhancements. Benefitted from the overall enhancements, VistrongerDet significantly improves the detection performance. Without bells and whistles, VistrongerDet is pluggable which can be used in any FPN-based two-stage detectors. It achieves **1.23** points and **1.15** points higher Average Precision (AP) than Faster R-CNN and Cascade R-CNN on VisDrone-DET test-dev set.*

1. Introduction

Drones, or general UAVs, equipped with cameras have been fast deployed to a wide range of applications, including agricultural, aerial photography, fast delivery, and surveillance. Consequently, automatic understanding of visual data collected from these platforms become highly demanding, which brings computer vision to drones more and more closely [9]. Object detection is the most basic and important task which crucially restricts the performance of high level visual tasks such as tracking, activity recognition and other automatic understanding of visual data, and has been a research focus in computer vision.

The convolution neural network (CNN) has high been appreciated in computer vision since its more representable features than handcrafted features. A number of detectors [14, 32, 23, 35, 42] based on CNN have been presented one after another, and have achieved excellent successes on public datasets, e.g., MS COCO [24] and PASCAL VOC [13]. Although object detection methods based on deep learning have made great progress, there are still open problems such as small objects, occlusion, and generalization that need to be solved. In particular, it is especially difficult to detect objects accurately in videos and images captured by UAV, e.g., VisDrone [44] and UAVDT [11], which suffers from a large number of tiny objects, wide scale distribution, serious long-tail distribution and difficulty for distinguishing similar categories caused by aerial shooting. The object detection on VisDrone DET 2021 dataset is a challenging visual task with such difficulties as shown in Figure 1. In this paper, we propose several effective strategies to solve the difficult problems in VisDrone DET 2021. And we achieved good competition ranking

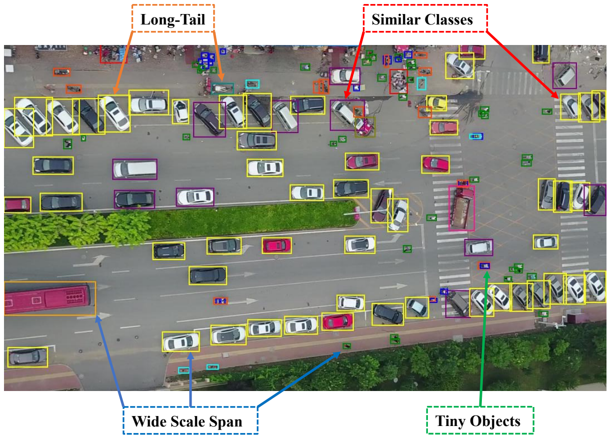

Figure 1. Challenges of object detection in the VisDrone 2021 dataset. Objects of the same category are labeled with the same color. Several difficulties are marked in the figure.

Size	$<200^2$	$200^2\sim400^2$	$>400^2$	$<32^2$	$32^2\sim96^2$	$>96^2$
number	487887	2035	42	306262	159999	23703

Table 1. The statistics data of all object bounding boxes in VisDrone2021 train&val set. The first row represents the object size range and the second row shows the number of objects falling within each range.

in this evaluation task by integrating our strategies into the high-performance detectors.

In VisDrone DET task, tiny objects which size is less than 32 pixels and wide scale distribution of objects are two problems that must be faced. The pyramidal feature network (FPN) [22] proposed by Lin et al. could obtain more convincing semantic representation and be widely applied for object detection. However the general detectors based on FPN have poor detection performance for tiny objects in VisDrone DET dataset, and FPN features are not sensitive to tiny objects, even the bottom level. Furthermore, wide scale distribution of objects in VisDrone dataset leads to the imbalance of instances in each layer of FPN, and also influences the model performance. Inspired by [15], we adopt effective fusion factors to avoid the impact caused by the imbalance of instances in each layer of FPN. Meanwhile, in order to make FPN features of objects more expressive, we introduce mask supervision on each layer during the model training. We refer to these two strategies as FPN level enhancement.

In FPN structure, the feature of each ROI is obtained by only pooling on the features of one level. However two ROIs with similar size may be assigned to different levels by the procedure. Then such pooling algorithm may produce sub-optimal detections since there is a strong correlation between levels, especially adjacent levels. Therefore, we propose an Adjacent ROI Fusion (ARF) module to fuse ROI features from adjacent levels by parameterizing the procedure of ROI pooling. We refer to this processing as ROI level enhancement.

The third problem to be solved is the long-tail distribution in the VisDrone DET dataset. The ideas of many methods for solving long-tail distribution problem come from long-tail classification. Buda M et al. balanced the differences between long-tail categories and head categories with more samples for tail classes during training [2]. Cui Y et al. settled this problem by assigning large weights for tail categories during training [7]. However, these methods can only solve some problems, and they may lead to over-fitting, even cause optimization difficulty. We exploit a Dual Sampler and Head Network (DSHNet) [39] to handle head and tail classes separately. In addition, there exist similar categories in the VisDrone dataset, such as pedestrian and people, which are difficult to distinguish for aerial shot images. We cleverly add two supervisors to the classification head: multi-label prediction and grouping softmax, thereby indirectly avoid modifying the structure of original detection network. We term these strategies HEAD level enhancement.

In light of these challenges and the characteristic of VisDrone dataset, we propose a novel detector named **VistrongerDet**, which possesses **Stronger** **V**isual **I**nformation. Our framework integrates the novel components of FPN level, ROI level and head level enhancements.

In summary, the main contributions of this paper are as follows:

1) We propose a novel detector (VistrongerDet), enhanced from FPN level, ROI level, and head level respectively.

2) Our improvement method is pluggable, and can be used in any FPN-based two-stage detectors such as [32, 4, 30, 29, 16].

3) We achieve significant improvement compared to the benchmark provided by the VisDrone Challenge. Finally, our model is ranked 5th in VisDrone-DET2021 challenge [9].

2. Related Work

In the section, an overview of related work is presented in response to the proposed research work, which mainly includes the following three aspects: general object detection, object detection in UAV images and Long-tail object detection.

General Object Detection. The current popular object detection frameworks mainly divided into anchor-free and anchor-based. Anchor-free approaches focus on detecting objects by locating and regressing key points. CornerNet detects an object as a pair of key points—the top-left corner and bottom-right corner of the bounding box [18]. Whereafter, grouping the corners based on the distances to get the final detection results. CenterNet represents objects by a single at their bounding box center, and regresses to the corresponding size for each object according to the center point [42]. ExtremeNet detects four extreme points (topmost, left-most, bottom-most, right-most) of an object [43]. Anchor-based approaches can then be subdivided into one-stage and two-stage detectors. SSD [26] and YOLO [31] are commonly used one-stage detectors, the mainly advantage is fast but without high accuracy. RetinaNet proposes a focal loss to solve the problem of imbalance between positive and negative samples and difficult and easy samples [23].

Compared to one-stage methods, two-stage detectors add Region Proposal Network (RPN) [32] to predict the rough position, then perform classification and location correction predictions on these proposals. Cascade R-CNN uses cascade structure to further refine the previous results to obtain higher quality detection results [4].

Object Detection In UAV Images. Compared with ground images, object detection in UAV images is more challenging. There exists a lot of tiny objects in the images shot by UAV, such as the size of the object less than 32 pixels. Wang et al. propose a Receptive Field Expansion Block (RFEB) to increase the receptive field size, and a Spatial-Refinement Module (SRM) to repair the spatial details of multi-scale objects in images [37]. DPNet [12] introduces the global context module (GC) [5] and deformable convolution (DC) [8] into the backbone network. DroneEye2020 [10] uses Recurisve Feature Pyramid (RFP) for neck, and additionally uses Switchable Atrous Convolution (SAC) for better performance [30]. Many approaches [41, 19, 28] generate a set of sub-images based on cropping methods, which can increase the size of objects and enlarge the datasets. The above methods only indirectly avoid the trouble caused by tiny objects, and there are no specific algorithms or structures proposed.

Long-Tail Object Detection. Another serious challenge is long-tail distribution problem in UAV datasets. A few classes in VisDrone such as car, pedestrian and person account for more than 70%, while other classes have very few numbers, e.g., tricycle, awning-tricycle and bus. Resampling [2, 3] is a common method, which balances the differences between the long-tail categories and the head categories with more samples for the tail categories during training. Assigning large weights for the tail categories is another kind methods of for processing long-tail categories during training [7]. Although the above methods can solve some problems, they may also lead to over-fitting, even cause optimization difficulty. Forest R-CNN clusters fine-grained classes into coarser parent classes, and constructs a tree structure to learn the relationship between subcategories through its parent category [38]. According to the number of each category, Li Y et al. divide the similar number of categories into a group and performs cross-entropy loss supervision separately in the group [20]. These two methods help to alleviate the extreme imbalance problem, but will introduce errors in the parent classes or change the original softmax structure.

3. Methodology

We aim to maximize the detection performance in drone images by our enhancing strategies on FPN level, ROI level and HEAD level in order to alleviate the degradation caused by tiny objects, wide scale span, long-tail distribution, confusion of similar classes. The overall method framework is based on Cascade R-CNN [4] as shown in Figure 2. The processing flow is as: (1) The **BACKBONE** stage performs to extract the features of the input images, generate feature maps, and lay the foundation for the subsequent stages. (2) In **FPN** stage, the fusion from deep layers to shallow layers employs three different factors. Furthermore, the mask head and fusion module on each layer of FPN make feature extraction paying more attention to object regions, specially to tiny objects. And these constitute the FPN level enhancement. (3) In **ROI** stage, we perform ROI pooling procedure based on feature maps fused in the previous stage. For ROI pooling, the features of the current ROI layer and its adjacent layers are specially integrated, and the internal spatial attention mechanism of ROI is also utilized. We name such strategies ROI level enhancement. (4) In **HEAD** stage, that is our HEAD level enhancement, we take different branches to process the head categories and tail categories separately. Group-Softmax Classification and Multi-Label Classification are especially used to solve classification of similar categories. All components will be detailed in the following sections.

3.1. FPN level enhancement

We explore two strategies to execute FPN-level enhancement in VisDrone 2021 data set. Firstly, the fusion factor as [15] is adopted to solve the problem of wide range of object scale distribution. Secondly, the mask of the object region is added in the training phase to improve the detection of tiny objects.

Wide range of object scale distribution is one intractable issue in the VisDrone dataset. Table 1 is the statistics on the absolute size of objects in train&val sets of VisDrone 2021, while the size of objects varies from 1^2 to 400^2 pixels and object sizes are unevenly distributed at different scales. Such characteristic of object distribution may lead some layers of FPN to have much fewer training samples than others. In the original FPN [22], all fusion factors from deep to shallow layers are the same as 1. In this way, imbalance of instances in each layer of FPN would obstruct the update efficiency of network parameters when the gradient back propagates. Inspired by [15], we describe the couple degree of adjacent layers in FPN with different fusion factors. We calculate the number of training samples N_{P_i} on each layer with IOU matching algorithm; then three different fusion factors a_i^{i+1} are got as [15]:

$$a_i^{i+1} = \begin{cases} N_{P_{i+1}}/N_{P_i}, & \text{i <0} \\ (N_{P_{i+1}} + N_{P_{i+2}})/N_{P_i}, & i \geq 0, \end{cases} \qquad (1)$$

where i represents the level of pyramid. Therefore, we obtain the fusion factors with the sample distribution of different scales in the training data set, which can adjust the fusion of different layer features adaptively and optimize the network parameters more effectively.

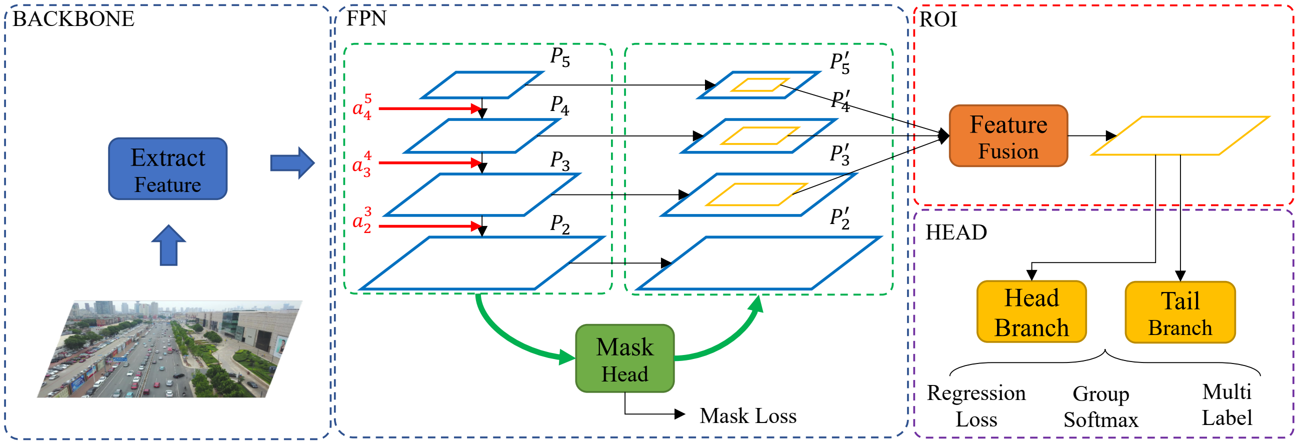

Figure 2. Overall pipeline of VistrongerDet. BACKBONE extracts the feature of input images, it can be ResNet [17], ResNeSt [40], Swing Transformer [27] and so on. The brief structure of three enhancements including FPN, ROI and HEAD has been illustrated on the figure.

There exists a lot of tiny objects in the images of VisDrone dataset. According to MS COCO's classification method, Table 1 also shows the number of object scales within the three levels of small, medium and large, and their majority are tiny objects which size is less than 32. Tiny objects would become smaller after down sampling, and maybe disappear from the feature maps of FPN easily. In order to strengthen the sensitivity of feature maps to tiny objects, we introduce mask supervision on each layer during the model training phase.

We generate a heatmap $Y \in [0,1]^{H \times W \times C}$ as label, indicating the foreground and background with ground-truth bounding boxes annotations. All pixels in the bounding boxes of the heatmap Y are set to 1 and the rest pixels are set to 0. In training phase of mask supervision, for a sample image and detection box shown in Figure 3, feature maps are first extracted by FPN network; then we use convolution operation to gradually reduce the number of channels to 1 without changing the resolution of feature maps. The training objective is a pixel-wise MSELoss. The foreground mask supervision training could make the feature extraction more attention to object regions.

Further, mask supervision on each layer could strengthen the sensitivity of feature maps to tiny objects, and the mask branches $\{M_2, M_3, M_4, M_5, M_6\}$ of 5 layers have learned features different from original FPN $\{P_2, P_3, P_4, P_5, P_6\}$. The network model would get the features that have both advantages through fusing M_i and P_i branches, where $i \in [2,6]$. Therefore, we design a novel module named Spatial Attention Fusion (**SAF**) to adaptively combine the both features. The structure of SAF is illustrated in Figure 4 and the features extracted by SAF training fashion would pay more attention to tiny object instructed by foreground mask. In practice, to reduce the number of model parameters, the SAF module utilizes the intermediate results of the heatmaps process as shown in Figure 3.

3.2. ROI level enhancement

In FPN structure, one ground-truth bounding box will only be arranged for training on a certain level by IOU matching algorithm. In this way, the feature of each ROI is obtained by pooling on the features of on one level. However, empirically, there is a relationship among the different levels.

PANet utilizes the maximum of ROI adapted by linking all feature levels of ROI to enhance features [25]. The max operation just employs the local feature of strong response and ignores the features of other location. AugFPN [16] proposes Soft ROI Selection (SRS) to generate the final ROI features based on the adaptive weights from features at all the pyramid levels. Both methods utilize correlation among different layer features to guide the feature expression of the current layer ROIs.

Actually, two ROIs with similar size may be assigned to adjacent layers during training. The correlation of between the top and the bottom layer is not strong with this one-to-one training strategy. Furthermore, tiny objects can only appear in the bottom layer of FPN, and the information at the top is no longer instructive. If the network rigidly learns the relationship between all layers, it would decrease the generalization performance and convergence speed. On the contrary, the correlation of adjacent levels features is the greatest, which would contain more surrounding and detailed informations. Therefore, we propose an Adjacent ROI Fusion (**ARF**) module to fuse ROI features from adjacent levels by parameterizing the procedure of ROI pooling.

Specifically, we first pool features from adjacent levels for each ROI. Each ROI feature R_{jn} will be augmented to three ROI features $\{R_{in}, R_{jn}, R_{kn}\}$, and R_{in} and R_{kn} are taken from the corresponding positions above and below the

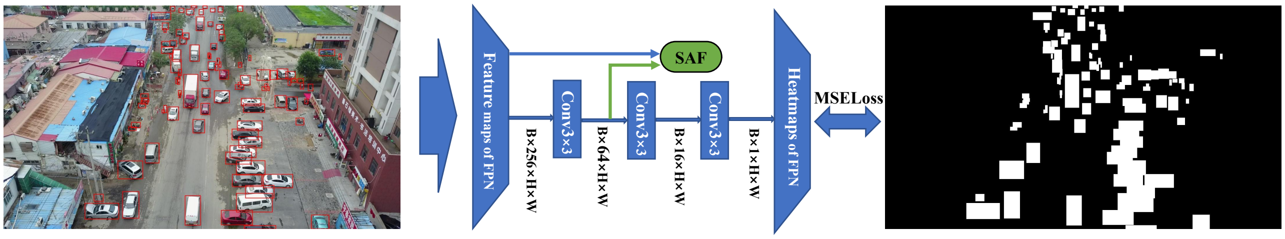

Figure 3. The detailed process of mask supervision on each layer of FPN. The padding and stride of all 3×3 convolutions are 1.

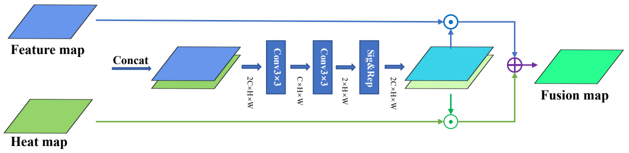

Figure 4. The detailed process of Spatial Attention Fusion (SAF) module. The padding and stride of all 3×3 convolutions are 1. Sig&Rep means Sigmoid and Repeat operation.

current ROI layer if j is index of middle layers of FPN. Then we balance the three ROI features by simply averaging as

$$\overline{R_n} = \frac{1}{3}(R_{in} + R_{jn} + R_{kn}), \tag{2}$$

where i,j and k represent the level of pyramid and n represents the id of ROI. The values of index i, j, and k are taken as:

$$\begin{cases} i = j, k = j + 1, & j = 2 \\ i = j - 1, k = j, & j = 6 \\ i = j - 1, k = j + 1, & 2 < j < 6 \end{cases} \tag{3}$$

In order to enhance the feature sensitivity, we utilize the contextual information of the current ROI and adopt the following function to represent the feature of current ROI:

$$R_n^{'} = Norm(R_{jn} + Drop(MHA(\overline{R_n}))), \tag{4}$$

where $Norm(\cdot)$ indicates Layer Normalization (LN) [1], $Drop(\cdot)$ denotes Dropout [34], and $MHA(\cdot)$ represents Multi-Head Attention [36]. This module makes full use of interlayer correlation and spatial self-attention of objects to enhance feature representation and improve network detection performance.

3.3. HEAD level enhancement

In VisDrone dataset, imbalanced class distribution is also a serious problem depressing network performance. A few classes such as car, pedestrian and person account for more than 70%, while other classes have very few samples, e.g., tricycle, awning-tricycle and bus. Inspired by [39], we exploit two branches to process head classes and tail classes separately. The head category branch uses more head category samples to train, while vice versa. In inference phase, the detection results of the two branches are merged to complement each other.

In addition, there exists similar categories in the VisDrone dataset, e.g., pedestrian and people, which are difficult to distinguish for classifiers since they have many similarities. Just like human perception, our solution first judges an object is a person rather than a vehicle with the outline features; then discriminate whether it is pedestrian or people according to the detailed features, which is more challenging relatively. Therefore we explore multi-label classification and group-softmax classification to achieve our ideas.

Multi-Label Classification. The most straightforward way to discriminate the fine-grained categories is to predict the parent categories firstly, and then predict the child categories based on parent predictions. However such method will damage the classifying structure of networks, and cause the loss of correlation between sub-categories, such as pedestrian and motor as shown in Figure 5(a). Forest R-CNN [38] clusters fine-grained classes into coarser parent classes, and constructs a tree structure to learn the relationship among sub-categories through its parent classes as shown in Figure 5(b). Although this method can strengthen the similar relationship, the prediction deviation of parent classes would affect the prediction of sub-categories. Instead, we exploit a different method that adds several parent classes to the classifier, e.g., c_1 represents the parent of pedestrian and people, c_2 represents the parent of bicycle and motor and so on as shown in Figure 5(c). During training phase, we use Binary Cross Entropy (BCE) to supervise multi-label classification. Then we remove the parent classes and just adopt the predicted results of the sub-categories in inference phase. In this way, we not only maintain the correlation of subclasses so as to get their common features, but also avoid increasing the prediction error from a superclass into its subclasses.

Group-Softmax Classification. Multi-label classification help classifiers to learn the commonality rather than repulsion among similar sub-categories. However a classifier is often difficult to discriminate similar sub-categories, such as pedestrian and people. According to the number of each category, [20] divides the similar number of categories into a group and performs softmax classification separately in the group. Inspired by the grouping idea, we propose a group-softmax classification by cleverly grouping the similar sub-categories into a group and separately executing softmax classification in each group so as to solve misclassification of similar classes. It is worth noting that the antagonistic categories of a category are all the remaining categories in dataset, and the classifier does not know which categories are easy to misclassify if the softmax calculation is performed on all categories.

Actually, group-softmax classification and multi-label classification are opposite, which "push" and "pull" the features among sub-categories respectively. Therefore, these two classifications cannot be applied on the same fully connected layer. So we employ two fully connected layers to get two sets of nodes and severally perform multi-label classification and group-softmax classification as shown in Figure 6, which mainly helps to extract better feature representation of the shared layer for similar classes distinguishing. The total classification loss is as follows.

$$L_{cls} = L_m(p^{'}, g) + \lambda \cdot L_g(p^{''}, g), \tag{5}$$

where L_m and L_g are objective functions corresponding to multi-label and group-softmax respectively. Predictions of two fully connected layers are denoted as $p^{'}$ and $p^{''}$. And g represents targets, the weight λ is used to balance between two supervisions. We set $\lambda = 0.1$ in all our experiments unless specified otherwise. Through the above two head level enhancement modules, we can classify similar categories in VisDrone dataset well and improve network model performance.

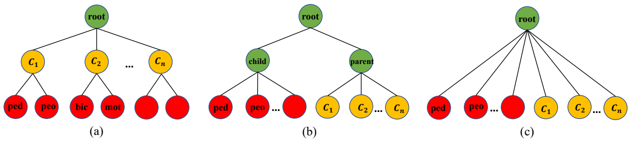

Figure 5. Several forms of classifiers. (a) is straightforward way to predict hierarchically. (b) is divided into two branches to predict the parent classes and sub-categories [38]. (c) is to directly predict all classes and use multi-label classification.

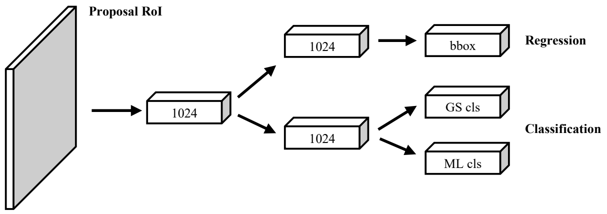

Figure 6. The head architecture of VistrongerDet. The figure shows the branch of the head categories. The branch of the tail categories is similar to it. GS cls means that discriminates similar categories with Group-Softmax Classification. ML cls means that classifies similar categories from other categories with Multi-Label Classification.

4. Experiments

We demonstrate the effectiveness of our proposed framework (VistrongerDet) on the VisDrone-DET [44] dataset.

4.1. Datasets

VisDrone-DET [44] is an object detection dataset with drone perspective. In Visdrone-Det, there are 6471 images in the training set, 548 images in the validation set and 1580 images in the test-challenge set, all labeled with 11 categories. This dataset is very challenging for object detection tasks. First, the scale of objects varies with the flying height and most of the objects are very small (less than 32 pixels). Second, it has different viewpoints, which leads to large gaps among objects even they belong to the same category. Third, the dataset is labeled with fine-grained category, e.g. the people with standing and walking poses are labeled as pedestrian and with other poses are labeled as people.

4.2. Implementation details

Our VistrongerDet is implemented on the MMdetection [6] toolbox. In pursuit of better Average Precision (AP), we choose the Faster R-CNN [32] and Cascade R-CNN [4] with Feature Pyramid Network (FPN) [22] as the baseline detection networks. If there is no special statement, we choose the ResNet-50 [17] as the backbone. As the same as DSHNet [39], pedestrian, people and car are considered as head classes, while other categories are regraded as tail classes.

Ignore region. In the VisDrone-DET [44] dataset, there are 11 classes including pedestrian, people, bicycle, car, van, truck, tricycle, awning-tricycle, bus, motor and others. Our goal is to predict the top ten categories. Therefore, we ignore both others and ignore region. Specifically, we calculate the IOU between the training sample and ignore region. Then we avoid training samples with IOU greater than 0.5.

Training phase. In order to save the memory usage and increase the input resolution, we divide original training images by 2×2 and horizontally flip all patches. The input resolution is 1600×1050 and batch size is set to 2 on 8 GPUs. There are 12 epochs totally and initial learning rate is set to 0.02, which then decreased by 10 and 100 times at the 9th and the 12th epoch, respectively. Anchor size is set to 4, and aspect ratio is set to (0.5, 1.0, 2.0). To expand the dataset, we load the pre-training model parameters on MS COCO [24].

Testing phase. In order to maintain consistency with the training configuration, the input size is set to 3200×2100 without cropping. The maximum number of objects in an image is set to 500. In VisDrone-test-challenge, we use also Test Time Augmentaion (TTA) to perform random modifications to the test images. In addition, we use weighted

Method	TTA	AP	AP50	AP75	AR1	AR10	AR100	AR500
CornerNet* [18]	-	17.41	34.12	15.78	0.39	3.32	24.37	26.11
Light-RCNN* [21]	-	16.53	32.78	15.13	0.35	3.16	23.09	25.07
FPN* [22]	-	16.51	32.20	14.91	0.33	3.03	20.72	24.93
Cascade* [4]	-	16.09	31.91	15.01	0.28	2.79	21.37	28.43
FAS	-	30.72	54.36	30.81	0.39	3.01	32.13	43.72
FAS	✓	31.64	55.86	32.17	0.39	3.00	32.49	44.9
FAS+Vistronger	✓	32.87	57.25	33.06	0.36	2.27	7.38	50.29
CAS	-	31.68	53.38	32.8	0.41	**3.36**	32.54	43.76
CAS	✓	32.70	55.08	33.87	**0.41**	3.35	**34.83**	44.88
CAS+Vistronger	✓	**33.85**	**57.27**	**34.81**	0.39	2.17	8.10	**51.12**

Table 2. Comparisons with other methods on VisDrone-DET test-dev set. * indicates that the baseline algorithm submitted by the committee. TTA means using Test Time Augmentaion during testing.

method	AP	ped	person	bicycle	car	van	trunk	tricycle	awn.	bus	motor
CornerNet* [18]	17.41	20.43	6.55	4.56	40.94	20.23	20.54	14.03	9.25	24.39	12.10
Light-RCNN* [21]	16.53	17.02	4.83	5.73	32.39	22.12	18.39	16.63	11.91	29.02	11.93
FPN* [22]	16.51	15.69	5.02	4.93	38.47	20.82	18.82	15.03	10.84	26.72	12.83
Cascade* [4]	16.09	16.28	6.16	4.18	37.29	20.38	17.11	14.48	12.37	24.31	14.85
FAS	31.64	22.21	13.73	13.28	53.00	35.49	31.00	18.47	16.57	45.00	22.70
FAS+Vistronger	32.87	22.37	14.06	14.33	54.96	38.12	31.43	18.56	17.19	46.96	22.99
CAS	32.7	22.97	13.61	13.20	54.36	35.33	34.10	**19.07**	17.20	46.61	22.97
CAS+Vistronger	**33.85**	**23.15**	**14.28**	**14.35**	**55.80**	**38.20**	**34.23**	18.86	**18.33**	**48.60**	**23.48**

Table 3. The results of each class on VisDrone-DET test-dev set. * indicates that the baseline algorithm submitted by the committee.

boxes fusion (WBF) to fuse many results of difficult model instead of non-maximum suppression (NMS) [33].

4.3. Experimental results

In this section, we evaluate VistrongerDet on VisDrone-DET test-dev set and compare with other methods. In order to reflect the scalability of Vistronger, we use two representative detectors as the baseline, including Faster R-CNN [32] and Cascade R-CNN [4]. For convenience, **FAS** and **CAS** are used to replace respectively.

Table 2 reports all experimental results. In the training phase, we use the above training strategies. Surprisingly, our baseline is much higher than the result submitted by the committee. While maintaining the same parameter settings, we add method of VistrongerDet to the baseline, and achieved **1.23%** and **1.15%** in AP improvements respectively. TTA is also a common method of object detection, and achieved **0.92%** and **1.02%** in AP improvements respectively.

Moreover, in almost all the cases, AP of each class is improved compared with baseline as shown in Table 3. This demonstrates that VistrongerDet plays an important role in solving tiny objects and long-tail distribution problems. For example, our VistrongerDet boosts the APs of tiny classes bicycle and people by about **1.15%** and **0.67%** respectively. For tail categories, such as awning-tricycle and bus, VistrongerDet also has a significant improvement, **1.13%** and **2.01%** respectively.

4.4. Ablation study

To validate the contributions of FPN, ROI and HEAD level to the improvement of detection performance respectively, we carry out ablation experiments on VisDrone dataset with Cascade R-CNN [4].

As shown in Table 4, we gradually add modules at each level to the baseline to prove that these modules are not conflicting. The first row shows the performance of the baseline. From the second to the last row, AP/AP50 gradually increased to **33.85/57.27** from 32.70/55.08.

Effect of Mask. In order to verify the effectiveness of mask supervision and SAF module respectively, we conduct two experiments as shown in Table 5(a). Firstly, we only use mask supervision on FPN. The performance is on slightly better than the baseline (32.89/55.46 vs. 32.70/55.08), indicating that it is slightly helpful to simply perform mask supervision on features of FPN. So the further idea is naturally whether the fusion of the features of mask and FPN would have a more improvement. Secondly, we use SAF to adaptively combine both context features. The performance of AP has improved better than baseline (**0.33%** vs. 0.19%).

Effect of ARF. To show that ARF is better than SRS [16]

Method	Factor	Mask	ARF	DSH	GS	ML	AP/AP50
CAS	-	-	-	-	-	-	32.7/55.08
	✓	-	-	-	-	-	32.9/55.37
	✓	✓	-	-	-	-	32.89/55.71
	✓	✓	✓	-	-	-	33.16/55.85
	✓	✓	✓	✓	-	-	33.31/55.98
	✓	✓	✓	✓	✓	-	33.47/56.18
	✓	✓	✓	✓	✓	✓	**33.85/57.27**

Table 4. The detection performance on VisDrone-DET test-dev set. "✓" means this method is used. "-" means this method is not used.

Method	CAS	(a)		(b)		(c)		
		+Mask	+Mask&SAF	+SRS	+ARF	+GS	+ML	+GS&ML
AP/AP50	32.70/55.08	32.89/55.46	**33.03/55.65**	32.41/54.94	**32.81/55.2**	32.85/55.21	33.0/55.95	**33.4/56.67**

Table 5. The ablation study of Mask, ARF and Head on VisDrone-DET test-dev set.

in fusing ROI features, we conduct two experiments as shown in Table 5(b). The performance of SRS is slightly lower than the baseline (32.41 vs. 32.70). Our proposed ARF module fuses ROI features from adjacent levels, and achieves an AP/AP50 gain of **0.11/0.12**. This result adequately shows the correlation of ROI features from adjacent levels is strong.

Effect of Head. The above describes group-softmax classification and multi-label classification are opposite. Therefore, it is necessary to verify the influence between them. Table 5(c) shows all results. The second and third row show that both of them can enhance the performance of the network by **0.15%** and **0.30%** respectively. The last row shows that we use the two classifications separately and supervise the respective fully connected layer to ensure that the shared fully connected layer learns the advantages of both.

5. Conclusion

In this paper, we analyze some problems that need to be solved urgently in the VisDrone dataset, e.g., tiny objects, large scale span, long-tail distribution, confusion of similar classes. In response to these issues, we propose a VistrongerDet, which possesses stronger visual information. The whole framework is mainly enhanced from three levels, including FPN, ROI and HEAD. Extensive experiments demonstrate the effectiveness of our method. In the future, we would conduct experiments on some general object detection datasets to verify the scalability of our method, e.g., MS COCO [24] and PASCAL VOC [13].

6. Acknowledgements

This work is supported by Chinese National Natural Science Foundation under Grants (62076033, U1931202).

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 5

[2] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018. 2, 3

[3] Jonathon Byrd and Zachary Lipton. What is the effect of importance weighting in deep learning? In *International Conference on Machine Learning*, pages 872–881. PMLR, 2019. 3

[4] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018. 2, 3, 6, 7

[5] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[6] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 6

[7] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277, 2019. 2, 3

[8] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 3

[9] Dawei Du et al. visdrone. http://aiskyeye.com/. 1, 2

[10] Dawei Du et al. Visdrone-det2020: The vision meets drone object detection in image challenge results. In *European Conference on Computer Vision*, 2020. 3

[11] Dawei Du, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 370–386, 2018. 1

[12] Dawei Du, Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Lin, Qinghua Hu, Tao Peng, Jiayu Zheng, Xinyao Wang, Yue Zhang, et al. Visdrone-det2019: The vision meets drone object detection in image challenge results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[13] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 1, 8

[14] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 1

[15] Yuqi Gong, Xuehui Yu, Yao Ding, Xiaoke Peng, Jian Zhao, and Zhenjun Han. Effective fusion factor in fpn for tiny object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1160–1168, 2021. 2, 3

[16] Chaoxu Guo, Bin Fan, Qian Zhang, Shiming Xiang, and Chunhong Pan. Augfpn: Improving multi-scale feature learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12595–12604, 2020. 2, 4, 7

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 6

[18] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European conference on computer vision (ECCV)*, pages 734–750, 2018. 2, 7

[19] Changlin Li, Taojiannan Yang, Sijie Zhu, Chen Chen, and Shanyue Guan. Density map guided object detection in aerial images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 190–191, 2020. 3

[20] Yu Li, Tao Wang, Bingyi Kang, Sheng Tang, Chunfeng Wang, Jintao Li, and Jiashi Feng. Overcoming classifier imbalance for long-tail object detection with balanced group softmax. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10991–11000, 2020. 3, 5

[21] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. Light-head r-cnn: In defense of two-stage object detector. *arXiv preprint arXiv:1711.07264*, 2017. 7

[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 2, 3, 6, 7

[23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 1, 2

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1, 6, 8

[25] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8759–8768, 2018. 4

[26] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 2

[27] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021. 4

[28] F Ozge Unel, Burak O Ozkalayci, and Cevahir Cigla. The power of tiling for small object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 3

[29] Jiangmiao Pang, Kai Chen, Jianping Shi, Huajun Feng, Wanli Ouyang, and Dahua Lin. Libra r-cnn: Towards balanced learning for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 821–830, 2019. 2

[30] Siyuan Qiao, Liang-Chieh Chen, and Alan Yuille. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10213–10224, 2021. 2, 3

[31] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 2

[32] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015. 1, 2, 3, 6, 7

[33] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: ensembling boxes for object detection models. *arXiv e-prints*, pages arXiv–1910, 2019. 7

[34] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 5

[35] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 1

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia

Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 5

[37] Haoran Wang, Zexin Wang, Meixia Jia, Aijin Li, Tuo Feng, Wenhua Zhang, and Licheng Jiao. Spatial attention for multi-scale feature refinement for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[38] Jialian Wu, Liangchen Song, Tiancai Wang, Qian Zhang, and Junsong Yuan. Forest r-cnn: Large-vocabulary long-tailed object detection and instance segmentation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1570–1578, 2020. 3, 5, 6

[39] Weiping Yu, Taojiannan Yang, and Chen Chen. Towards resolving the challenge of long-tail distribution in uav images for object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3258–3267, 2021. 2, 5, 6

[40] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020. 4

[41] Xindi Zhang, Ebroul Izquierdo, and Krishna Chandramouli. Dense and small object detection in uav vision based on cascade network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 3

[42] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 1, 2

[43] Xingyi Zhou, Jiacheng Zhuo, and Philipp Krahenbuhl. Bottom-up object detection by grouping extreme and center points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 850–859, 2019. 2

[44] Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Ling, and Qinghua Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018. 1, 6