Adaptive Sparse Convolutional Networks with Global Context Enhancement for Faster Object Detection on Drone Images

Bowei Du[1,2†], **Yecheng Huang**[1,2†], **Jiaxin Chen**[2], **Di Huang**[1,2,3*]
[1] State Key Laboratory of Software Development Environment, Beihang University, Beijing, China
[2] School of Computer Science and Engineering, Beihang University, Beijing, China
[3] Hangzhou Innovation Institute, Beihang University, Hangzhou, China
{boweidu, ychuang, jiaxinchen, dhuang}@buaa.edu.cn

Abstract

Object detection on drone images with low-latency is an important but challenging task on the resource-constrained unmanned aerial vehicle (UAV) platform. This paper investigates optimizing the detection head based on the sparse convolution, which proves effective in balancing the accuracy and efficiency. Nevertheless, it suffers from inadequate integration of contextual information of tiny objects as well as clumsy control of the mask ratio in the presence of foreground with varying scales. To address the issues above, we propose a novel global context-enhanced adaptive sparse convolutional network (CEASC). It first develops a context-enhanced group normalization (CE-GN) layer, by replacing the statistics based on sparsely sampled features with the global contextual ones, and then designs an adaptive multi-layer masking strategy to generate optimal mask ratios at distinct scales for compact foreground coverage, promoting both the accuracy and efficiency. Extensive experimental results on two major benchmarks, i.e. VisDrone and UAVDT, demonstrate that CEASC remarkably reduces the GFLOPs and accelerates the inference procedure when plugging into the typical state-of-the-art detection frameworks (e.g. RetinaNet and GFL V1) with competitive performance. Code is available at https://github.com/Cuogeihong/CEASC.

1. Introduction

Recent progress of deep neural networks (*e.g.* CNNs and Transformers) has significantly boosted the performance of object detection on public benchmarks such as COCO [23]. By contrast, building detectors for unmanned aerial vehicle (UAV) platforms currently remains a challenging task. On the one hand, existing studies are keen on designing complicated models to reach high accuracies of tiny objects on high-resolution drone imagery, which are computationally consuming. On the other hand, the hardware equipped with UAVs is often resource-constrained, raising an urgent demand in lightweight deployed models for fast inference and low latency.

To deal with the dilemma of balancing the accuracy and efficiency, a number of efforts are made, mainly on general object detection, which basically concentrate on reducing

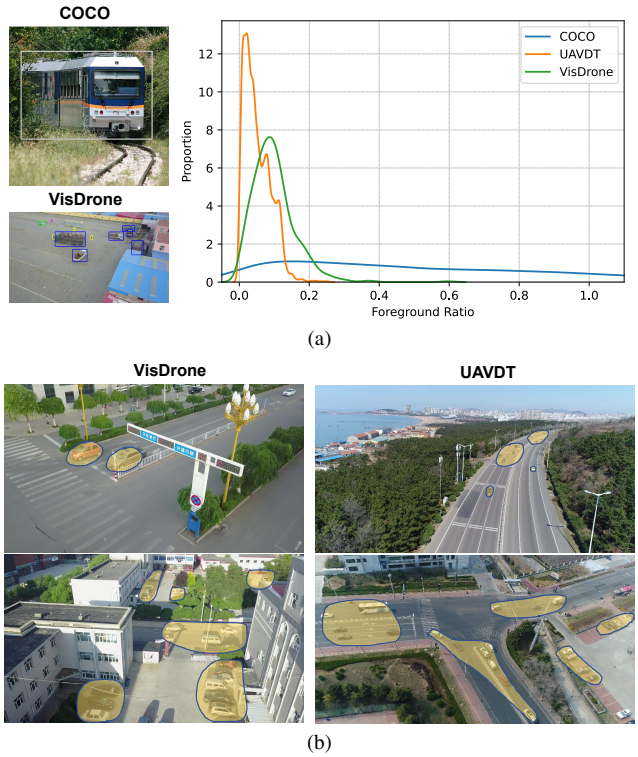


Figure 1. (a) Comparison of foreground proportions on the COCO and drone imagery databases; and (b) visualization of foregrounds (highlighted in yellow) on samples from VisDrone and UAVDT.

† indicates equal contribution.
∗ refers to the corresponding author.

the complexity of the backbone networks [2, 13, 47]. Despite some potential, these methods leave much room for improvement since they fail to take into account the heavy detection heads which are widely used by the state-of-the-art detectors [14, 21, 22, 46]. For instance, RetinaNet [22] taking ResNet18 [11] as backbone with 512 input channels adopts a detection head that occupies 82.3% of the overall GFLOPs. Recently, several methods have been presented to solve this problem, including network pruning [24, 45] and structure redesigning [1, 7], and prove effective in accelerating inference. However, the former is criticized by the sharp performance drop when computations are greatly decreased, evidenced by the attempt on detection for UAVs [45], and the latter is primarily optimized for low-resolution input (*e.g.* 640×640), making it not straightforward to adapt to high-resolution aerial images.

Sparse convolutions [6, 41] show another promising alternative, which limit computations by only operating convolutions on sparsely sampled regions or channels via learnable masks. While theoretically attractive, their results are highly dependent on the selection of meaningful areas, because the focal region of the learned mask in sparse convolutions is prone to locate within foreground. Regarding drone images, the vast majority of objects are of small scales (as shown in Fig. 1 (a)) and the scale of foreground areas varies along with flying altitudes and observing viewpoints (as shown in Fig. 1 (b)), and this issue becomes even more prominent. An inadequate mask ratio enlarges the focal part and more unnecessary computations are consumed on background, which tends to simultaneously deteriorate efficiency and accuracy. On the contrary, an exaggerated one shrinks the focal part and incurs the difficulty in fully covering foreground and crucial context, thus leading to performance degradation. DynamicHead [31] and QueryDet [42] indeed apply sparse convolutions to the detection head; unfortunately, their primary goal is to offset the increased computational cost when additional feature maps are jointly used for performance gain on general object detection. They both follow the traditional way in original sparse convolutions that set fixed mask ratios or focus on foreground only and are thus far from reaching the trade-off between accuracy and efficiency required by UAV detectors. Therefore, it is still an open question to leverage sparse convolutions to facilitate lightweight detection for UAVs.

In this paper, we propose a novel plug-and-play detection head optimization approach to efficient object detection on drone images, namely global context-enhanced adaptive sparse convolution (CEASC). Concretely, we first develop a context-enhanced sparse convolution (CESC) to capture global information and enhance focal features, which consists of a residual structure with a context-enhanced group normalization (CE-GN) layer. Since CE-GN specifically preserves a set of holistic features and applies their statistics for normalization, it compensates the loss of context caused by sparse convolutions and stabilizes the distribution of foreground areas, thus bypassing the sharp drop on accuracy. We then propose an adaptive multi-layer masking (AMM) scheme, and it separately estimates an optimal mask ratio by minimizing an elaborately designed loss at distinct levels of feature pyramid networks (FPN), balancing the detection accuracy and efficiency. It is worth noting that CESC and AMM can be easily extended to various detectors, indicating that CEASC is generally applicable to existing state-of-the-art object detectors for acceleration on drone imagery.

The contribution of our work lies in three-fold:

1) We propose a novel detection head optimization approach based on sparse convolutions, *i.e.* CEASC, to efficient object detection for UAVs.

2) We introduce a context-enhanced sparse convolution layer and an adaptive multi-layer masking scheme to optimize the mask ratio, delivering an optimal balance between the detection accuracy and efficiency.

3) We extensively evaluate the proposed approach on two major public benchmarks of drone imagery by integrating CEASC to various state-of-the-art detectors (*e.g.* RetinaNet and GFL V1), significantly reducing their computational costs while maintaining competitive accuracies.

2. Related Work

2.1. General Object Detection

General object detection methods can be mainly divided into anchor-based detectors and anchor-free detectors depending on whether they use preset sliding windows or anchors to locate object proposals. In anchor-based detectors, the multi-stage detectors, including R-CNN [8], Faster-RCNN [29], Mask RCNN [10], first generate proposal regions and subsequently classify and localize target objects within them. On the contrary, classification and localization of objects can be directly conducted in the whole feature on the one-stage detectors such as RetinaNet [22] and GFL V1/V2 [20, 21], which treat anchors as final bounding box targets. As for the anchor-free ones (*e.g.* Centernet [5], FCOS [33] and FSAF [48]), the anchors that incur heavy computational burden are replaced by efficient alternatives such as centerness constraints or object heatmaps. Although gains are consistently delivered, it is not so straightforward to adapt such methods to the case on UAVs.

2.2. Object Detection on Aerial Images

For object detection on drone imagery, current studies usually follow a coarse-to-fine pipeline where a coarse detector is launched to locate large-scale instances and subregions that contain densely distributed small ones and a fine detector is further applied to those regions to find in-

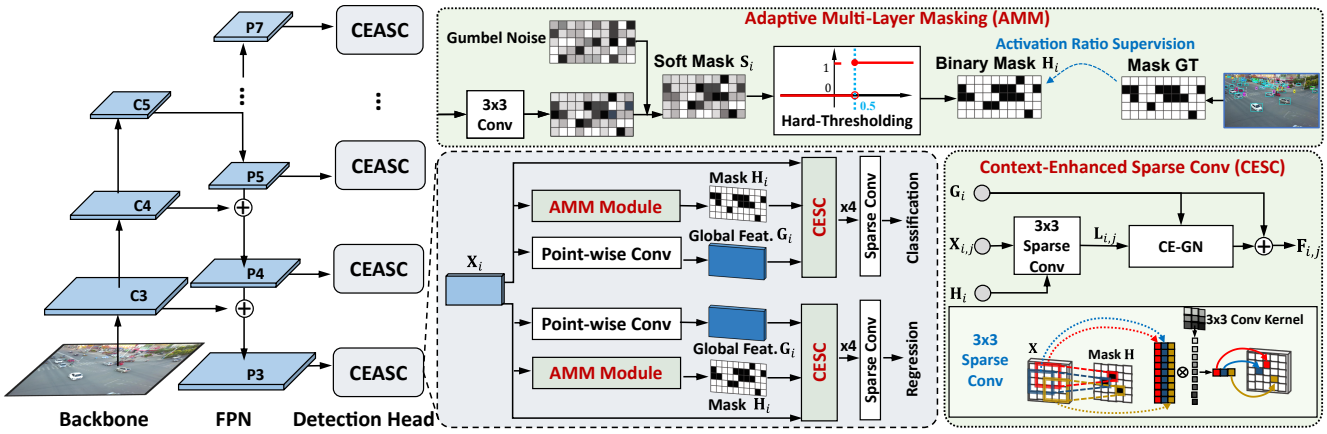


Figure 2. Framework of CEASC. Given a base detector such as GFL V1, CEASC replaces the detection head by context-enhanced sparse convolution (CESC) in each FPN layer, via generating a mask feature $\mathbf{H}_i$ and a global feature $\mathbf{G}_i$ for context enhancement. The mask ratio of $\mathbf{H}_i$ is automatically optimized by the adaptive multi-layer masking (AMM) scheme, promoting both the accuracy and efficiency.

stances of small sizes. For example, ClusDet [43] employs a scale estimation network (ScaleNet) for better fine detection; DMNet [19] optimizes region selection by conducting a density map guided connected crop generation; UFPMP-Det [14] merges sub-regions generated by a coarse detector into a unified image and designs the multi-proxy detection network to improve the detection accuracy of tiny objects; and Focus&Detect [17] makes use of the Gaussian mixture model to estimate focal regions and introduces incomplete box suppression to deal with overlapping focal areas. Despite of high accuracies achieved, these methods need to perform inference on one image for multiple times, which are not efficient, limiting their applications on the resource-constrained UAV platforms.

2.3. Lightweight Models for Object Detection

Along with the advancement of deep learning, the complexity of object detection models has sharply increased, incurring heavy computational cost and slow inference speed. Several typical solutions are proposed in parallel to reduce computations for acceleration, including neural architecture search [32, 37], network pruning [24, 25], knowledge distillation [2, 44] and lightweight model design [28, 30]. Among them, lightweight model design is in the lead for detection on UAVs for its good potential in speed-accuracy trade-off.

Some methods focus on lightweight backbones, where MobileNet [12, 13, 30] and ShuffleNet [26, 47] are the representatives, which employ depth-wise separable convolutions and group convolutions, respectively. Some methods design lightweight detection heads, *e.g.* in the YOLO series, YOLO v6 [18] presents an efficient decoupled head, while YOLO v7 [36] plans re-parameterized convolutions.

Sparse CNN has recently emerged as a promising way to accelerate inference by generating pixel-wise sample masks for convolutions. In particular, [31, 42] have attempted to apply sparse convolutions to the detection head. [31] conducts a pixel-level combination of FPN features from different scales via spatial gates to reduce the computational cost. QueryDet [42] works on high-resolution images and utilizes the P_2 features from FPN to improve the accuracy on tiny objects, while a cascade sparse query structure is built and trained by the focal loss [22] for acceleration. Nevertheless, as these methods usually use a fixed mask ratio without capturing global context, they fail to handle severe fluctuations of foreground regions, leading to insufficiently optimized detection results on drone images. In contrast, our method adaptively adjusts the mask ratio with global feature captured to balance the efficiency and accuracy.

3. Method

As Fig. 2 shows, given a base detector, the entire CEASC network aims to optimize the detection head at different layers in FPN, by developing a context-enhanced sparse convolution (CESC), which integrates focal information with global context through a lightweight convolutional module as well as a context-enhanced group normalization (CE-GN) layer. An adaptive multi-layer masking (AMM) module is designed to enable the model adaptively generating masks with an adequate mask ratio, thus reaching a better balance in accuracy and efficiency.

The details of the components aforementioned are described in Sec. 3.1 and Sec. 3.2.

3.1. Context-Enhanced Sparse Convolution

3.1.1 Sparse Convolution

Most existing detectors on drone images work with dense detection heads, convolving on the whole feature maps. Al-

though fully exploring visual clues facilitates detecting tiny objects, the dense head requires much more computations, which is not applicable to the resource-constrained UAV platform. In the mean time, the foreground area only occupies a small part of a frame acquired by a drone as shown in Fig. 1, indicating that the dense head conducts a lot of computational operations on background, which contains much less useful information for object detection. This observation reveals the potential to accelerate the detection head by only computing on the foreground area.

Sparse convolution (SC) [6, 41] have recently been proposed, which learn to operate on foreground areas by employing a sparse mask and prove effective in speeding up the inference phase on a variety of vision tasks. Inspired by them, we construct our network based on SC.

Specifically, given a feature map $\mathbf{X}_i \in \mathbb{R}^{B\times C\times H\times W}$ from the i-th layer of FPN, SC adopts a mask network consisting of a shared kernel $\mathbf{W}_{mask} \in \mathbb{R}^{C\times 1\times 3\times 3}$, where B, C, H, W refers to the batch size, channel size, height and width, respectively. Convolving on $\mathbf{X}_i$ based on $\mathbf{W}_{mask}$ generates a soft feature $\mathbf{S}_i \in \mathbb{R}^{B\times 1\times H\times W}$, which is further turned to a mask matrix $\mathbf{H}_i \in \{0,1\}^{B\times 1\times H\times W}$ by using the Gumbel-Softmax trick [35] formulated as below:

$$\mathbf{H}_i = \begin{cases} \sigma\left(\frac{\mathbf{S}_i+g_1-g_2}{\tau}\right) > 0.5, & \text{For training} \\ \mathbf{S}_i > 0, & \text{For inference} \end{cases} \tag{1}$$

where $g_1, g_2 \in \mathbb{R}^{B\times 1\times H\times W}$ denote two random gumbel noises, σ refers to the sigmoid function, and τ is the corresponding temperature parameter in Gumbel-Softmax.

According to Eq. (1), only the area with the mask value 1 involves in convolutions during inference, thus reducing the overall computational cost. The sparsity of $\mathbf{H}_i$ is controlled by a mask ratio $r \in [0, 1]$, which is often set larger than 0.9 by hand in existing studies. Since the base detector (here we take GFL V1 as an example) has a classification head and a regression head in the detection framework, we separately introduce a mask network for each head considering that they often focus on different areas. Each detection head adopts four Convolution-GN-ReLU layers and a single convolution layer to make prediction, where we replace the conventional convolution layers with the SC ones.

3.1.2 Context Enhancement

As claimed in [44], contextual clues (*e.g.* background surrounding target objects) benefit object detection; however, SC performs convolutions only on foreground and abandons background with useful information, which probably undermines the overall accuracy, especially in the presence of tiny objects prevailing in drone images. To tackle with this problem, [40] attempts to recover surrounding context by interpolation, but it is not reliable as the focal and background areas exhibit large discrepancy. In this work, we propose a lightweight CESC module, jointly making use of focal information and global context for enhancement and simultaneously boosting the stability of subsequent computations. As displayed in Fig. 2, we apply a point-wise convolution to the feature map $\mathbf{X}_i$, generating the global contextual feature $\mathbf{G}_i$. Since only a few elements in $\mathbf{X}_i$ are processed by SC, $\mathbf{G}_i$ tends to become stable after multiple rounds of SC without taking much extra computational cost.

As an important part of SC, we embed the global contextual information $\mathbf{G}_i$ into the SparseConvolution-GN-ReLU layers, which takes the feature map $\mathbf{X}_{i,j}$, the mask $\mathbf{H}_i$, and the global feature $\mathbf{G}_i$ as inputs, where j indicates the j-th SparseConvolution-GN-ReLU layer. Instead of using the activated elements to compute the statistics for group normalization as in conventional SC, we adopt the mean value and standard deviation of $\mathbf{G}_i$ for normalization, aiming to compensate the missing context. Supposing that $\mathbf{L}_{i,j}$ is the output feature map after applying SC on $\mathbf{X}_{i,j}$, the context-enhanced feature $\mathbf{F}_{i,j}$ is obtained by CE-GN as below

$$\mathbf{F}_{i,j} = w \times \frac{\mathbf{L}_{i,j} - mean[\mathbf{G}_i]}{std[\mathbf{G}_i]} + b \tag{2}$$

where $mean[\cdot]$ and $std[\cdot]$ denote the mean and standard deviation, respectively, and w and b are learnable parameters.

To further mitigate the information loss in SC and make the training process more stable, we additionally maintain the normal dense convolution besides the sparse one during training, generating a feature map $\mathbf{C}_{i,j}$ convolved on the full input feature map. We then employ $\mathbf{C}_{i,j}$ to enhance the sparse feature map $\mathbf{F}_{i,j}$ by optimizing the MSE loss as:

$$\mathcal{L}_{norm} = \frac{1}{4L}\sum_{i=1}^{L}\sum_{j=1}^{4} \|\mathbf{C}_{i,j} \times \mathbf{H}_i \; - \; \mathbf{F}_{i,j}\|^2, \tag{3}$$

where L is the amount of layers in FPN.

We finally adopt a residual structure before the activation layer by adding $\mathbf{G}_i$ to $\mathbf{F}_{i,j}$, *i.e.* $\mathbf{F}_{i,j} := \mathbf{F}_{i,j} + \mathbf{G}_i$, which strengthens context preservation. The complete architecture of the CESC module and the CE-GN layer are displayed in Fig. 2.

3.2. Adaptive Multi-layer Masking

Without any extra constraint, the sparse detector tends to generate the mask with a large activation ratio (or a small mask ratio) for a higher accuracy, thus increasing the overall computational cost. To deal with this issue, most existing attempts use a fixed activation ratio. However, as the foreground of aerial images exhibits severe fluctuations, a fixed ratio is prone to incur either significant increase in computation or decrease in accuracy due to insufficient coverage over foreground areas. For the trade-off between accuracy

Base Detector	Method	mAP	AP_{50}	AP_{75}	AR_1	AR_{10}	AR_{100}	AR_{500}	GFLOPs	FPS
GFL V1 [21]	Baseline	28.4	50.0	27.8	0.62	6.36	35.6	44.9	524.95	13.46
	Ours (CEASC)	**28.7**	**50.7**	**28.4**	**0.65**	**6.56**	35.6	**45.0**	**150.18**	**21.55**
RetinaNet [22]	Baseline	**21.8**	39.3	**21.1**	0.54	5.82	**29.1**	**35.3**	529.81	13.41
	Ours (CEASC)	21.6	**39.6**	20.6	**0.59**	5.82	28.9	34.7	**157.41**	**20.10**
Faster-RCNN [29]	Baseline	**24.8**	**43.6**	**25.0**	0.64	**5.97**	**33.0**	34.9	322.25	18.17
	Ours (CEASC)	24.6	43.4	24.7	0.64	5.91	32.8	**35.1**	**132.91**	**21.71**
FSAF [48]	Baseline	**26.3**	**50.3**	**23.7**	0.53	5.25	**32.5**	**43.5**	518.25	14.06
	Ours (CEASC)	25.0	48.9	22.0	**0.56**	**5.65**	31.1	41.5	**153.92**	**19.43**

Table 1. Comparison of AP/AR (%) and GFLOPs/FPS on VisDrone by using our approach with various base detectors.

and efficiency, we propose the AMM scheme to adaptively control the activation ratio (or reversely the mask ratio).

Specifically, AMM firstly estimates an optimal mask ratio based on the ground-truth label. By leveraging the label assignment technique, for the i-th FPN layer, we obtain the ground-truth classification results $\mathcal{C}_i \in \mathbb{R}^{h_i \times w_i \times c}$, where c represents the number of categories including the background; h_i and w_i indicate the height and width of the feature map, respectively. The optimal activation ratio $\mathcal{P}_i$ in the i-th FPN layer is estimated as

$$\mathcal{P}_i = \frac{Pos(\mathcal{C}_i)}{Numel(\mathcal{C}_i)}, \tag{4}$$

where $Pos(\mathcal{C}_i)$ and $Numel(\mathcal{C}_i)$ indicate the number of pixels belonging to the positive (foreground) instances and that of all pixels, respectively.

To guide the network adaptively generating a mask with an adequate mask ratio, we employ the following loss

$$\mathcal{L}_{amm} = \frac{1}{L}\sum_i \Big(\frac{Pos(\mathbf{H}_i)}{Numel(\mathbf{H}_i)} - \mathcal{P}_i\Big)^2, \tag{5}$$

where $\frac{Pos(\mathbf{H}_i)}{Numel(\mathbf{H}_i)}$ indicates the activation ratio of the mask $\mathbf{H}_i$. By minimizing $\mathcal{L}_{amm}$, $\mathbf{H}_i$ is forced to abide by the same activation ratio as the ground-truth foreground ratio $\mathcal{P}_i$, thus facilitating the generation of adequate mask ratios.

By adding the conventional detection loss $\mathcal{L}_{det}$, we formulate the overall training loss as follows:

$$\mathcal{L} = \mathcal{L}_{det} + \alpha \times \mathcal{L}_{norm} + \beta \times \mathcal{L}_{amm}, \tag{6}$$

where α, β are hyper-parameters balancing the importance of $\mathcal{L}_{norm}$ and $\mathcal{L}_{amm}$.

4. Experiments

We evaluate the effectiveness of CEASC by comparing it to the state-of-the-art lightweight approaches and conducting comprehensive ablation studies.

4.1. Datasets and Metrics

We adopt two major benchmarks for evaluation in drone-based object detection, *i.e*. VisDrone [49] and UAVDT [4]. VisDrone consists of 7,019 high-resolution (2,000×1,500) aerial images belonging to 10 categories. Following previous work [42,43], we use 6,471 images for training and 548 images for testing. UAVDT contains 23,258 training images and 15,069 testing images with a resolution of 1,024×540 from 3 classes.

We employ the mean Average Precision (mAP), Average Precision (AP) and Average Recall (AR) as the evaluation metrics on accuracy, as well as GFLOPs and FPS as the ones on efficiency.

4.2. Implementation Details

We implement our network based on PyTorch [27] and MMDetection [3]. On VisDrone, all models are trained for 15 epochs with the SGD optimizer, and the learning rate is initialized as 0.01 with a linear warm-up and decreased by 10 times after 11 and 14 epochs. On UAVDT, we train models for 6 epochs with an initial learning rate at 0.01, decreased by 10 times after 4 and 5 epochs. The trade-off hyper-parameters α and β in Eq. (6) are set to 1 and 10, respectively, and the temperature parameter τ in Gumbel Softmax is fixed as 1. We make use of GFL V1 as the base detector and ResNet18 as the backbone with 512 feature channels by default. The input image sizes are set to 1,333×800 and 1,024×540 on VisDrone and UAVDT, respectively. All experiments are conducted on two NVIDIA RTX 2080Ti GPUs, except that the inference speed is test on a single RTX 2080Ti GPU.

4.3. Evaluation on Different Detectors

It is worth noting that the proposed CEASC network is plug-and-play. To validate its effect in a wide range of base detectors, we report the performance by combining CEASC with four prevailing base detectors: GFL V1 [21], RetinaNet [22], Faster-RCNN [29] and FSAF [48]. As shown in Table 1, by integrating CEASC, the GFLOPs of all the base detectors are reduced by at least 60%, and the FPS is pro-

CESC	AMM	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
		28.4	50.0	27.8	524.95	13.46
✓		28.6	50.6	28.2	158.23	19.26
✓	✓	**28.7**	**50.7**	**28.4**	**150.18**	**21.55**

Table 2. Ablation on CESC and AMM with GFL V1 as the base detector on VisDrone.

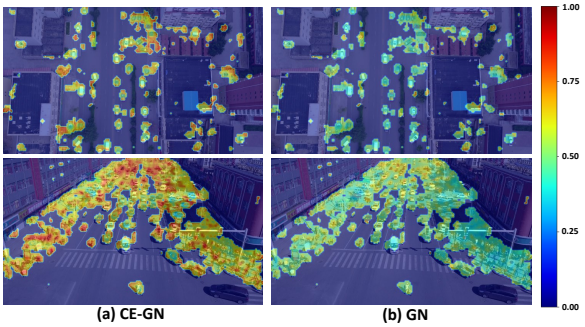


Figure 3. Visualization on correlation between features generated by dense convolutions and those by sparse convolutions using distinct normalization schemes on VisDrone, (a) and (b) use CE-GN and GN on sparse convolutions, respectively.

moted by 20%∼60% with slight fluctuations in mAP, indicating its effectiveness and generalizability in accelerating detectors without sacrificing their accuracies.

4.4. Ablation Study

We validate the main components of CEASC, where we also adopt GFL V1 as the base detector in all the ablation studies.

4.4.1 On CESC and AMM

As Table 2 reports, by employing the CESC component, the base detector saves about 70% of GFLOPs and runs 1.43 times faster without any drop in accuracy, since SC reduces the complexity and the CE-GN layer together with the residual structure compensates the loss of context. By adopting the dynamic mask ratio to obtain a compact foreground coverage, the AMM component further increases the accuracy and accelerates the inference speed by 11.9% while saving 5.1% of GFLOPs. Note that the training process of GFL V1 becomes extremely unstable when directly applying SC without CESC, and we thus do not provide the result by individually evaluating AMM on GFL V1.

4.4.2 On Detailed Designs in CESC

We separately evaluate the effect of the residual structure (Res. for short), CE-GN and the normalization loss $\mathcal{L}_{norm}$ in Eq. (3) on the performance of CESC. Recall that directly applying SC to GFL V1 makes the training process unstable. As summarized in Table 3, when employing the residual structure, GFL V1 with SC turns to be stable and requires much less GFLOPs, but the mAP sharply drops due to the loss of context. By adding the context information via CE-GN, the accuracy is significantly promoted with a slight increase in GFLOPs. $\mathcal{L}_{norm}$ further boosts the accuracy and efficiency, since it implicitly strengthens the sparsity of the features.

We further evaluate the performance of CE-GN by comparing it to the counterparts including the one without using normalization as in QueryDet [42], GroupNorm (GN) [39] as in DynamicHead [31], BatchNorm (BN) [16] and InstanceNorm (IN) [34]. We also report the results by the original GFL V1 detector denoted as 'Dense Conv.'. As displayed in Table 4, CE-GN substantially promotes the accuracy of the model without normalization by 2.6%. In comparison to the other normalization schemes, CE-GN achieves the best accuracy, 0.7%, 2,6% and 0.8% higher than GN, BN and IN, respectively. It is worth noting that CE-GN performs the best in efficiency in regards of GFLOPs and FPS as well. To

SC	Res.	CE-GN	$\mathcal{L}_{norm}$	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
				28.4	50.0	27.8	524.95	13.46
✓	✓			26.1	47.2	25.3	151.66	21.51
✓	✓	✓		28.5	50.5	28.3	155.90	19.91
✓	✓	✓	✓	**28.7**	**50.7**	**28.4**	**150.18**	**21.55**

Table 3. Ablation on detailed designs in CESC with GFL V1 on VisDrone.

Method	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
Dense Conv.	28.4	50.0	27.8	524.95	13.46
w/o Normalization	26.1	47.2	25.3	151.66	21.51
GN [39]	28.0	49.9	27.7	154.49	18.82
BN [16]	26.1	47.0	25.4	150.81	19.55
IN [34]	27.9	49.7	27.6	160.91	19.30
CE-GN (**Ours**)	**28.7**	**50.7**	**28.4**	**150.18**	**21.55**

Table 4. Ablation on CE-GN with GFL V1 on VisDrone.

Method	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
3×3 convolution	28.5	50.1	28.1	262.38	17.12
GhostModule [9]	28.3	50.1	27.8	194.66	19.22
CBAM [38]	28.4	50.3	27.8	**148.08**	16.20
Criss-Cross Attn. [15]	28.4	50.3	27.8	159.27	15.40
Point-wise (**Ours**)	**28.7**	**50.7**	**28.4**	150.18	**21.55**

Table 5. Comparison of distinct methods to encode global context with GFL V1 on VisDrone.

Method	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
Global	28.4	50.2	28.1	162.53	19.84
Layer-wise	**28.7**	**50.7**	**28.4**	**150.18**	**21.55**

Table 6. Comparison of estimating the mask ratio in different ways by AMM on VisDrone.

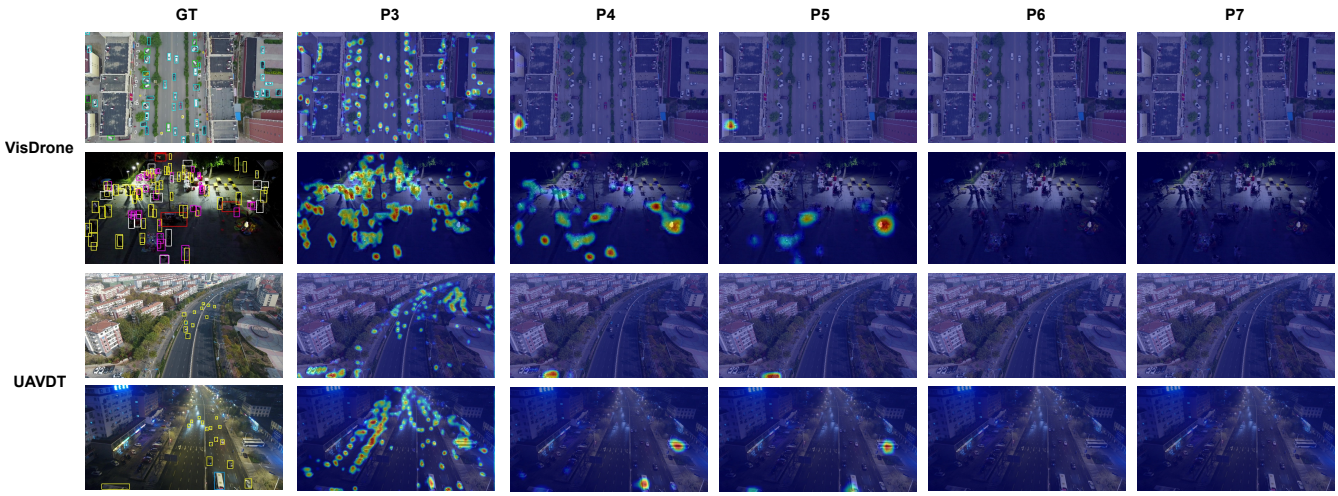
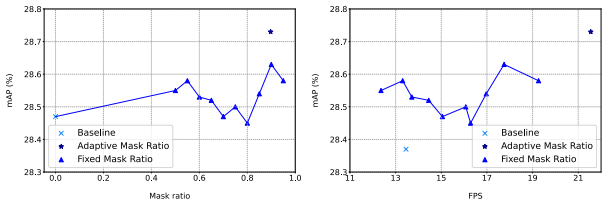
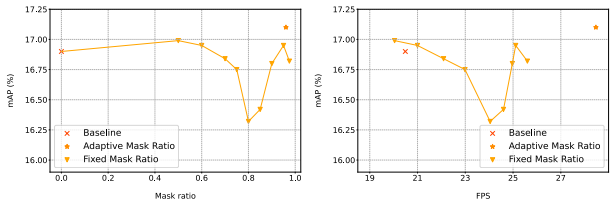


Figure 4. Visualization of the dynamic masks estimated by AMM for different layers (from 'P3' to 'P7') in FPN of GFL V1. Highlighted areas are activated for computation.



(a) VisDrone



(b) UAVDT

Figure 5. Comparison of the fixed mask ratio and the dynamic one estimated by AMM.

highlight the advantages of CE-GN, we visualize the cosine similarities between the features generated by dense convolutions and sparse convolutions, where CE-GN and GN are separately utilized to normalize SC. As Fig. 3 illustrates, the features using CE-GN exhibit higher correlations than those using GN, showing the superiority of CE-GN in enhancing global context for SC.

To encode global context, we utilize the point-wise convolution, and make comparison to existing techniques including the plain 3×3 convolution, GhostModule [9], and several attention-based methods such as CBAM [38] and Criss-Cross Attention [15]. As summarized in Table 5, the point-wise convolution outperforms the counterparts in detection accuracy. Meanwhile, it reaches the lowest GFLOPs in the convolution-based approaches and achieves the highest FPS among all the methods, clearly demonstrating its advantage in balancing the accuracy and efficiency.

Method	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
with P3	26.9	48.6	26.3	**143.03**	**27.78**
with P3-P4	28.5	50.5	28.1	149.09	24.60
with P3-P5	**28.7**	**50.7**	**28.4**	150.01	21.79
Ours (with P3-P7)	**28.7**	**50.7**	**28.4**	150.18	21.55

Table 7. Ablation on FPN with GFL-V1 on VisDrone.

4.4.3 On Detailed Analysis of AMM

We compare the AMM module with a fixed mask ratio ranging from 0.50 to 0.95 on VisDrone and from 0.50 to 0.975 on UAVDT, respectively. As Fig. 5 shows, more features are involved in convolution when reducing the mask ratio, resulting in higher computational cost and lower FPS. In the mean time, we can see that the detection accuracy is sensitive to the mask ratio, which is not consistently improved as the ratio increases. Moreover, the optimal fixed mask ratio varies on different datasets, *e.g.* 0.9 on VisDrone and 0.95 on UAVDT in regards of mAP. In contrast, AMM adaptively determines an appropriate mask ratio, with which the base detector reaches the best accuracy and the highest inference speed, demonstrating its necessity.

Note that AMM separately computes the mask ratio for different layers in a "Layer-wise" way. We compare it to a "Global" version, which estimates a global mask ratio for all layers. As demonstrated in Table 6, the "Layer-wise" method clearly performs better than the "Global" one in terms of mAP and FPS. The reason lies in that the optimal

Base Detector	Method	Backbone	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
GFL V1 [21]	Baseline	ResNet18	28.4	50.0	27.8	524.95	13.46
	MobileNet V2 [30]	MobileNet V2	28.5	50.2	28.1	491.47	13.63
	ShuffleNet V2 [26]	ShuffleNet V2	26.2	46.6	25.7	488.94	13.92
	Ours (CEASC)	ResNet18	**28.7**	**50.7**	**28.4**	**150.18**	**21.55**
RetinaNet [22]	Baseline	ResNet50	20.2	**36.9**	19.5	586.77	10.27
	QueryDet [42]	ResNet50	19.6	35.7	19.0	-	10.65
	QueryDet-CSQ [42]	ResNet50	19.3	35.0	18.9	-	11.71
	Ours (CEASC)	ResNet50	**20.8**	35.0	**27.7**	**201.96**	**14.27**

Table 8. Comparison of mAP/AP (%) and GFLOPs/FPS with the state-of-the-art approaches on VisDrone. '-' indicates that the result is not reported or not publicly available.

Method	mAP	AP_{50}	AP_{75}	GFLOPs	FPS
Baseline	16.9	29.5	**17.9**	271.66	20.49
Ours (CEASC)	**17.1**	**30.9**	17.8	**64.12**	**28.47**

Table 9. Comparison of mAP/AP (%) and GFLOPs/FPS with GFL V1 on UAVDT.

mask ratio varies in different layers of FPN as displayed in Fig. 4, and the "Layer-wise" method estimates the mask ratio more precisely than the "Global" one, thus promoting both the accuracy and efficiency. We also evaluate its effect at different FPN layers in Table 7. With less FPN layers, GFLOPs and FPS are improved. Abandoning P6-P7 does not affect much as they are less informative. Removing P4 incurs a sharp drop in mAP, indicating that P4 is crucial, which is consistent with the visualization.

4.5. Comparison to SOTA

We compare our network with the state-of-the-art ones: 1) the lightweight methods including MobileNet V2 [13] and ShuffleNet V2 [47]; 2) the detection head optimization methods for drone imagery including QueryDet [42] and its acceleration part QueryDet-CSQ [42]. Since GFL V1 [21] with ResNet18 as the backbone is widely used and proves effective in drone-based object detection, we select it as the base detector, and denote the original version as the 'Baseline' method. We also report the result by using RetinaNet [22] with ResNet50 as the backbone, since it is used as the base detector in QueryDet and QueryDet-CSQ. Note that the same data augmentation technique used in QueryDet is adopted in our implementation for fair comparison.

As summarized in Table 8, CEASC remarkably reduces the GFLOPs of the base detectors (GFL V1 and RetinaNet), reaching a slightly higher mAP in the mean time. For instance, CEASC decreases the GFLOPS of the Baseline GFL V1 by 71.4% and achieves 60% speedup in terms of FPS during inference, with a 0.3% improvement in mAP. Since the lightweight models, *i.e*. MobileNet V2 and ShuffleNet V2, quest for efficiency by simplifying the network structures, their mAPs are lower than ours. Moreover, they apply dense detection heads, thus requiring much more GFLOPs. Though QueryDet-CSQ considers to optimize the detection head by the CSQ module with sparse convolutions, it only concentrates on small objects and ignores the loss of contextual information. Besides, QueryDet introduces an extra heavy query head to promote performance, which inevitably incurs more computational cost. In contrast, CEASC newly develops the context-enhanced sparse convolution module and designs an adaptive multi-layer masking scheme, thus clearly outperforming QueryDet and QueryDet-CSQ, both in accuracy and efficiency.

We also evaluate CEASC on UAVDT. As reported in Table 9, our method reduces the GFLOPs by 76.3% and boosts the inference speed by 38.9% with a gain of 0.2% in mAP, compared with the Baseline.

5. Conclusion

We propose a novel plug-and-play detection head optimization approach, namely CEASC, to object detection on drone imagery. It develops the CESC module with CE-GN, which substantially compensates the loss of global context and stabilizes the distribution of foreground. Furthermore, it designs the AMM module to adaptively adjust the mask ratio for distinct foreground areas. Extensive experimental results achieved on VisDrone and UAVDT demonstrate that CEASC remarkably accelerates the inference speed of various base detectors with competitive accuracies.

Acknowledgment

This work is partly supported by the National Key R&D Program of China (2021ZD0110503), the National Natural Science Foundation of China (62022011 and 62202034), the Research Program of State Key Laboratory of Software Development Environment (SKLSDE-2021ZX-04), and the Fundamental Research Funds for the Central Universities.

References

[1] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 2

[2] Guobin Chen, Wongun Choi, Xiang Yu, Tony Han, and Manmohan Chandraker. Learning efficient object detection models with knowledge distillation. In *NeurIPS*, 2017. 2, 3

[3] 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. 5

[4] 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 *ECCV*, 2018. 5

[5] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *ICCV*, 2019. 2

[6] Michael Figurnov, Maxwell D Collins, Yukun Zhu, Li Zhang, Jonathan Huang, Dmitry Vetrov, and Ruslan Salakhutdinov. Spatially adaptive computation time for residual networks. In *CVPR*, 2017. 2, 4

[7] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 2

[8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[9] Kai Han, Yunhe Wang, Qi Tian, Jianyuan Guo, Chunjing Xu, and Chang Xu. Ghostnet: More features from cheap operations. In *CVPR*, 2020. 6, 7

[10] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 2

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[12] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *ICCV*, 2019. 3

[13] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 2, 3, 8

[14] Yecheng Huang, Jiaxin Chen, and Di Huang. Ufpmp-det: Toward accurate and efficient object detection on drone imagery. In *AAAI*, 2022. 2, 3

[15] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *ICCV*, 2019. 6, 7

[16] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 6

[17] Onur Can Koyun, Reyhan Kevser Keser, İbrahim Batuhan Akkaya, and Behçet Uğur Töreyin. Focus-and-detect: A small object detection framework for aerial images. *SPIC*, 104:116675, 2022. 3

[18] Chuyi Li, Lulu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, Liang Li, Zaidan Ke, Qingyuan Li, Meng Cheng, Weiqiang Nie, et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*, 2022. 3

[19] Changlin Li, Taojiannan Yang, Sijie Zhu, Chen Chen, and Shanyue Guan. Density map guided object detection in aerial images. In *CVPR Workshops*, 2020. 3

[20] Xiang Li, Wenhai Wang, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In *CVPR*, 2021. 2

[21] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. In *NeurIPS*, 2020. 2, 5, 8

[22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 2, 3, 5, 8

[23] 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 *ECCV*, 2014. 1

[24] Liyang Liu, Shilong Zhang, Zhanghui Kuang, Aojun Zhou, Jing-Hao Xue, Xinjiang Wang, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Group fisher pruning for practical network compression. In *ICML*, 2021. 2, 3

[25] Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *ICCV*, 2017. 3

[26] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *ECCV*, 2018. 3, 8

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 5

[28] Zheng Qin, Zeming Li, Zhaoning Zhang, Yiping Bao, Gang Yu, Yuxing Peng, and Jian Sun. Thundernet: Towards real-time generic object detection on mobile devices. In *ICCV*, 2019. 3

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 2, 5

[30] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018. 3, 8

[31] Lin Song, Yanwei Li, Zhengkai Jiang, Zeming Li, Hongbin Sun, Jian Sun, and Nanning Zheng. Fine-grained dynamic head for object detection. In *NeurIPS*, 2020. 2, 3, 6

[32] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, 2019. 3

[33] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *ICCV*, 2019. 2

[34] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 6

[35] Thomas Verelst and Tinne Tuytelaars. Dynamic convolutions: Exploiting spatial sparsity for faster inference. In *CVPR*, 2020. 4

[36] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *arXiv preprint arXiv:2207.02696*, 2022. 3

[37] Ning Wang, Yang Gao, Hao Chen, Peng Wang, Zhi Tian, Chunhua Shen, and Yanning Zhang. Nas-fcos: Fast neural architecture search for object detection. In *CVPR*, 2020. 3

[38] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018. 6, 7

[39] Yuxin Wu and Kaiming He. Group normalization. In *ECCV*, 2018. 6

[40] Zhenda Xie, Zheng Zhang, Xizhou Zhu, Gao Huang, and Stephen Lin. Spatially adaptive inference with stochastic feature sampling and interpolation. In *ECCV*, 2020. 4

[41] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 2, 4

[42] Chenhongyi Yang, Zehao Huang, and Naiyan Wang. Querydet: Cascaded sparse query for accelerating high-resolution small object detection. In *CVPR*, 2022. 2, 3, 5, 6, 8

[43] Fan Yang, Heng Fan, Peng Chu, Erik Blasch, and Haibin Ling. Clustered object detection in aerial images. In *ICCV*, 2019. 3, 5

[44] Zhendong Yang, Zhe Li, Xiaohu Jiang, Yuan Gong, Zehuan Yuan, Danpei Zhao, and Chun Yuan. Focal and global knowledge distillation for detectors. In *CVPR*, 2022. 3, 4

[45] Pengyi Zhang, Yunxin Zhong, and Xiaoqiong Li. Slimyolov3: Narrower, faster and better for real-time uav applications. In *ICCV Workshops*, 2019. 2

[46] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *CVPR*, 2020. 2

[47] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *CVPR*, 2018. 2, 3, 8

[48] Chenchen Zhu, Yihui He, and Marios Savvides. Feature selective anchor-free module for single-shot object detection. In *CVPR*, 2019. 2, 5

[49] Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Ling, and Qinghua Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018. 5