Knowledge Distillation via Hierarchical Matching for Small Object Detection

Yongchi Ma[1], Xiao Ma[1], Tianran Hao[1], Lisha Cui[1], Shaohui Jin[1], Pei Lv[1]

[1]Zhengzhou University

Abstract Knowledge distillation is often used for model compression and has achieved a great breakthrough in image classification, but there remains scope for improvement in object detection, especially for knowledge extraction of small objects. The main problem is the features of small objects are often polluted by background noise and not prominent due to down-sampling of Convolutional Neural Network (CNN), resulting in the insufficient refinement of small object features during distillation. In this paper, we propose a Hierarchical Matching Knowledge Distillation (HMKD) network that operates on P2 to P4 of feature pyramid network (FPN), aiming to intervene on small object features before affecting. We employ an encoder-decoder network to encapsulate low-resolution, highly semantic information, akin to eliciting insights from profound strata within a teacher network. Then it is matched with the high-resolution feature values of small objects from shallow layers as the key, during this period we use an attention mechanism to measure the relevance of the inquiry to the feature values. Also in the process of decoding, knowledge is distilled to the student. Experiments show that our method achieves excellent improvements in both one-stage and two-stage object detectors. Specifically, the proposed method based on Faster R-CNN achieves 41.7% mAP on COCO2017 (ResNet50 as the backbone), which is 3.8% higher than the baseline. In addition, our student outperforms even the teacher on VisDrone.

Keywords Knowledge distillation, Object detection, Small object detection, Machine learning

1 Introduction

In recent times, there has been a substantial surge in the field of computer vision research, with a particular upswing in the prominence accorded to the domain of small object detection. Previous researches usually apply more complex and larger network to improve the detection accuracy of small objects. However, their performance improvements are often accompanied by a large demand for computing power. Huge amount of mobile and edge computing devices are unable to provide powerful computing ability, thus preventing those complex models from being deployed on these devices. In addition to designing new architectures, quantization [1, 2, 3] and network pruning [4, 5, 6], knowledge distillation [7, 8, 9] has been an effective approach for lightweight models after it was proposed by Hinton [7] in 2015. It consists of a teacher (complex and large model) and a student (simple and small model). The student not only gains knowledge of the ground-truth in the dataset but also learns good lessons from the teacher to improve its generalization without changing the network structure.

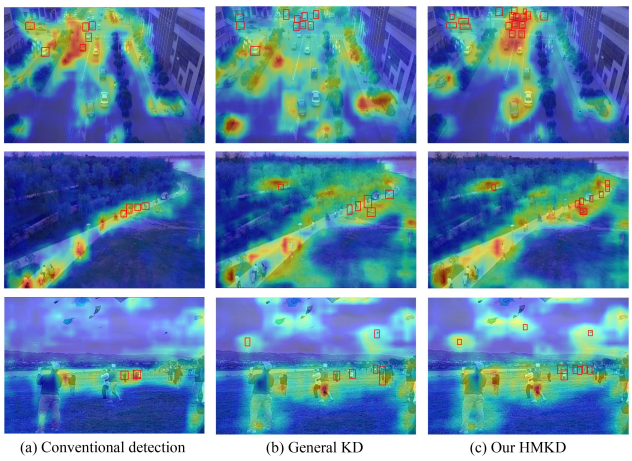

Fig. 1. The performance comparisons of small object detection with the heatmaps generated by Grad-CAM [10]. Small objects are marked by red boxes. It can be seen that our method is clearly more sensitive to small objects than other baselines.

Most knowledge distillation methods are based on object features [11, 12, 13, 14]. These researchers distill the suggested regions predicted by the Region Proposal Network (RPN) [15] or features extracted against

the Feature Pyramid Network (FPN) [16]. Researchers are trying their best to find out how to locate useful knowledge and explore what is useful knowledge. For example, Kang *et al.* [17] use conditional distillation network to locate the required knowledge and realize the transfer of instance knowledge.

Although these methods have achieved significant breakthroughs, they ignore the differences in the size of the objects themselves, which means that these objects can provide different degree of knowledge in the feature space. This leads to different difficulties in knowledge extraction. Therefore, objects of different scales should not be treated equally. Figure 1 shows the comparison of different methods for small object detection. The traditional knowledge distillation method is obviously insufficient for the recognition of small objects. The inherent reason why small object features are difficult to learn is that they may be diluted when the Convolutional Neural Network (CNN) is down-sampled. At the same time, small objects are easily disturbed by background noise, which makes them "more difficult" in knowledge distillation. When teaching student, teachers should pay more attention to those knowledge that is ambiguous or difficult to understand. Otherwise it can lead to knowledge omission or lack of understanding by student. These issues present a notable challenge in refining knowledge of small objects.

To enhance student's understanding of small object knowledge, we propose hierarchical matching for knowledge distillation (HMKD) as shown in Fig 2. It intends to enhance the student model's learning of small object features in the shallow high-resolution layers of the FPN, mainly focusing on the P2 to P4 layers. By applying hierarchical matching, a decoding network is introduced to discover and extract those difficult knowledge. Specifically, we first separate the foreground and background of the image to prevent small objects from being contaminated by the background during down-sampling. Then we encode the strong semantic information of small objects at low resolution as inquiries, and use the fine-grained feature value at high resolution as key values. Based on the experiments, we have observed that knowledge distillation for the foreground only is not the best choice, and the relationship between foreground and background needs to be considered. Therefore, we have designed an extra supplemental distillation module to teach student the background relationships as an additional knowledge. The contributions of this paper are as follows:

- We propose a novel Hierarchical Matching Knowledge Distillation (HMKD) framework to enhance small object distillation. In addition, supplemental distillation is introduced to complement the background information.
- In HMKD, we encode high-semantic information at low-resolution of FPN as inquiries, and represent fine-grained graph feature values at high-resolution as key-values.
- We perform experiments using the MS COCO2017 [18] and VisDrone [19] datasets on mainstream object detection frameworks. The results show that our approach results in a significant performance improvement of the model with strong generality.

2 Related Work

2.1 Knowledge Distillation

Recently, there have many works using knowledge distillation for object detection with good results. Kang *et al.* [17] propose a novel approach to extract knowledge by using instance search to transfer image features from the instructor to the student. Yang *et*

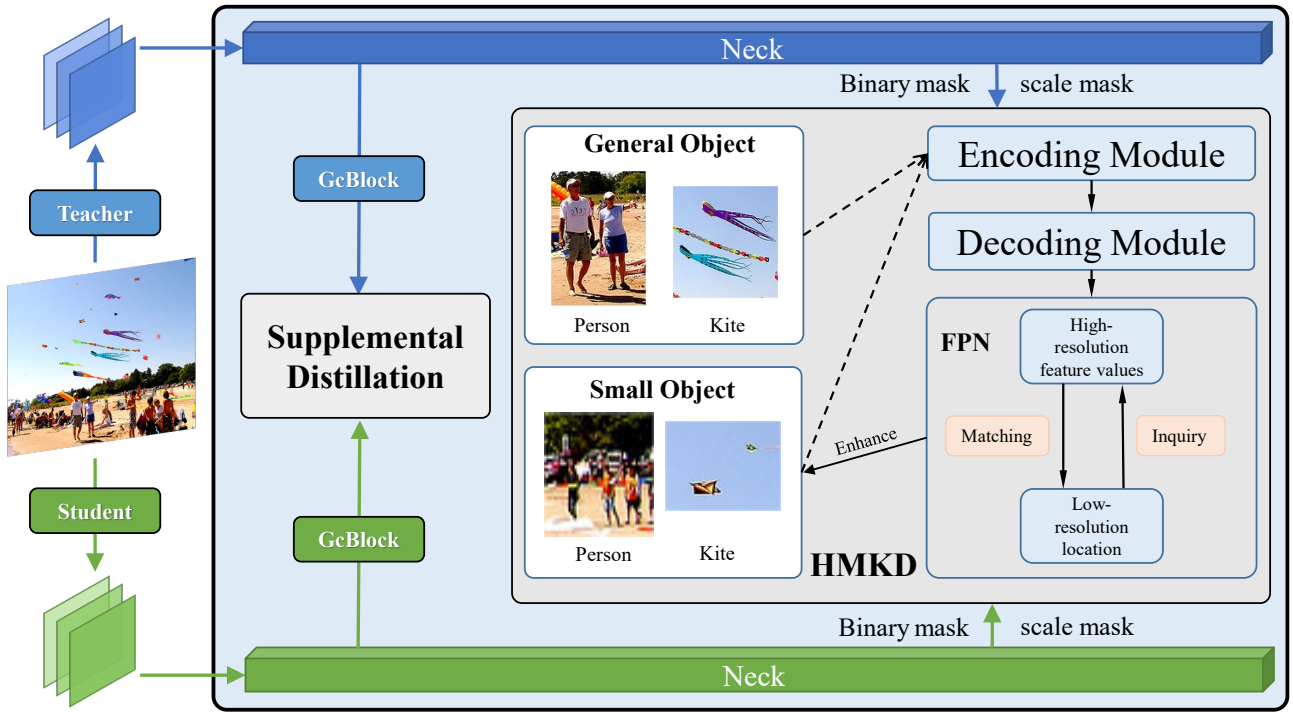

Fig.2. The pipeline of our method. Firstly, the foreground and background are separated to prevent the background information from excessively interfering with the extraction of small objects. Then HMKD is used to enhance the small object features, thus improving the detection ability of the student model. Meanwhile, GcBlock [20] is used to distill the background information.

al. [21] concern that uneven differences between feature map will negatively affect extraction, so they optimize the separation of background and foreground to force student to focus on the teacher's channels and pixels. Chen *et al.* [22] and Li *et al.* [23] propose prediction-based approaches to extract features of RPN regions and distill the foreground knowledge. Wang *et al.* [11] extract the specified regions of maximum IoU of anchor points and ground-truth. Guo *et al.* [12] propose a new loss function to solve the problem of imbalance between background and object. However, these are all heuristic-based methods that require rules to be designed in advance, which is inflexible. While Zhang *et al.* [13] incorporate the attention mechanism into the knowledge rectification approach and establish spatial channel attention to measure distillation. None of these methods pay attention to the detection of small objects. Objects of different sizes cannot be treated equally in the distillation process. Smaller objects are difficult to detect and the distillation process needs to be focused to attenuate the loss of information from small objects. Focused distillation is needed for smaller objects that are more difficult to detect, to prevent information loss of small objects during distillation.

To break through the above limitations, we propose a distillation method to enhance knowledge transfer for small objects, using hierarchical matching to enable student enhance understanding of difficult knowledge(small objects).

2.2 Object Detection

CNN-based object detection frameworks can be classified as one-stage [24], two-stage [15] and anchor free [25]. Notably, classical one-stage detector is Reti-

naNet [24], where FocalLoss is applied to deal with the background and foreground imbalance. Its performance when combining with FPN is already comparable to two-stage detectors. There is also the YOLO [26] series of detectors, which are currently under development, have the ability to directly regress box coordinates and class probabilities from image pixels. The speed is also very favorable. A classical two-stage detector is the Faster R-CNN [15], which uses a Region Proposal Network (RPN) to efficiently generate proposal regions. FPN is also introduced to capture multi-scale feature maps through lateral connections. It comes with advantages in detection accuracy, but the detection speed is not as fast as the one-stage detectors. This anchor-based detection method uses anchor boxes with different aspect ratios to label objects, and then applies heads to classify and regress each anchor. Recently, the anchor free detection framework FCOS [25] is proposed to predict the coordinates of labels and candidate boxes using a fully convolutional network. Since they all require input features, our knowledge distillation approach can be applied on these detectors.

2.3 Small Object Detection

Small object detection is a more difficult part of object detection due to factors such as little semantic information and susceptibility to complex scene interference. Most of current researches make improvements to the performance of small object detection in terms of data augmentation [27, 28, 29], enhancing the resolution of input features [30, 31, 32], multi-scale information fusion [33, 34, 35], and contextual semantic information [36, 37, 38]. In recent years, scale regularization strategies such as SNIP [35] and SNIPER [39] are devised with the primary objective of mitigating the concern surrounding variations in object dimensions across images of disparate resolutions. Chen *et al.* [40] introduce a feedback-driven data provider, aiming at addressing the challenge of detecting small objects by balancing the detection loss. While the outlined technique presents a viable resolution for the challenge posed by diminutive object detection, regrettably, its integration into the knowledge distillation paradigm seems to have been inadvertently disregarded. In a similar vein, TridentNet [33] develops a parallel multi-branching approach that utilizes different perceptual domains to generate more accurate and discriminative features for small objects.

3 Method

3.1 Overview

In contemporary object detection methodologies that hinge on detector architecture, the extraction of features pertaining to diminutive entities frequently entails recourse to high-resolution feature maps. However, student may have weaker capabilities in extracting such features compared to its teacher. To address this issue, we propose an algorithm that enhances the student's ability to learn features of small objects. Specifically, we strengthen the extraction of high-resolution features in the shallow stages of the FPN. We adopt a hierarchical matching approach for knowledge transfer, taking high-level semantics of small objects at low-resolution as inquiries and fine-grained graph features proposed by teachers at high-resolution layers as key-values. Finally, the student's knowledge is updated by an attention-weighted feature extraction loss.

3.2 Hierarchical Matching Knowledge Distillation

3.2.1 Foreground and background separation

In this section, our approach HMKD is elaborated, which is an enhanced small object knowledge transfer algorithm based on hierarchical matching. Carion *et*

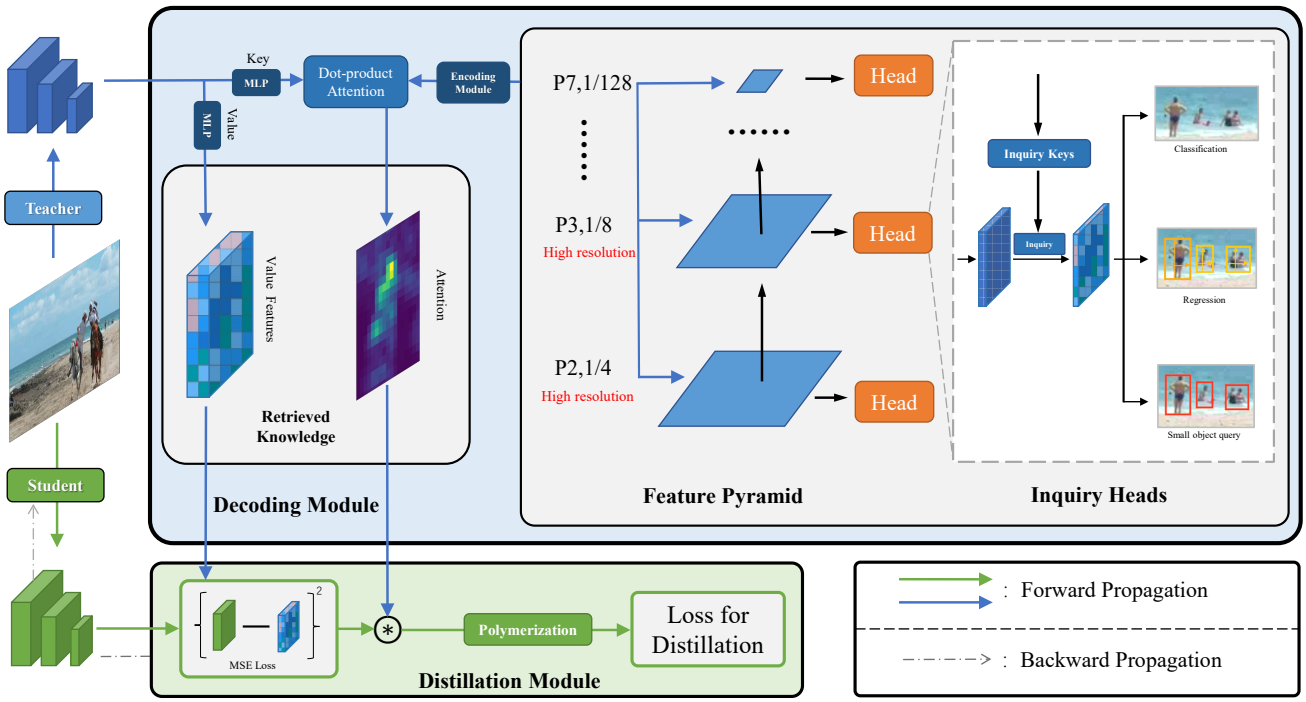

Fig.3. Illustration of HMKD in detail. The high semantic information of small objects at low resolution is encoded as a inquiry, and the fine-grained feature values at high resolution are used as key values. In this way, the detection ability of the student for small objects gradually converges to that of the teacher.

al. [41] apply the idea of encoder and decoder to object detection in DETR [41]. Kang *et al.* [17] also apply it to knowledge distillation. Inspired by this, we notice the difficulty in transferring information about small objects in knowledge distillation. HMKD utilizes the idea of encoder and decoder. It uses the high semantic information of small objects at low resolution as inquiries, and the fine-grained graph feature values presented by teacher in the high-resolution layer are used as keys. Finally, student is updated through feature distillation losses based on attention weighting, whose goal is to make these difficult knowledge available to student.

Since background noise may have an effect on small object features, we decide to remove it as much as possible at first. This allows the model to focus on the pure distillation of the foreground. We separate the background and foreground using the binary mask M.

$$M_{x,y} = \begin{cases} 1, & if(x,y) \in g \\ 0, & otherwise \end{cases} \tag{1}$$

where g indicates the ground-truth area of objects, x is the horizontal coordinate of the feature map, and y is the vertical coordinate. If (x, y) matches ground-truth, then $M_{x,y} = 1$, otherwise it is 0. The proportions of the background and the object are different in different images. Large-scale objects cause more losses because they take up more pixels. So we treat each different goal equally in order not to affect the extraction of small goals. The proportional mask K is to balance the loss in separation:

$$K_{x,y} = \begin{cases} \frac{1}{H_g W_g}, & if(x,y) \in g \\ \frac{1}{N_{bg}}, & Otherwise \end{cases} \tag{2}$$

$$N_{bg} = \sum_{x=1}^{H} \sum_{y=1}^{W} (1 - M_{x,y}) \tag{3}$$

where H_g and W_g denote the height and width of the boxes respectively. If a pixel is part of more than one object, K takes the minimum value.

3.2.2 Distillation Strategy

The one-stage RetinaNet classifies the objects by using a single FPN module, and the two-stage Faster R-CNN uses two detection heads for localization. If the input image size is $H \times W$, feature size of FPN is $P = P_l \in R^{H \times W \times C}$, where l indicates the level of the pyramid. We add a new head module (Inquiry Head) to predict the approximate location of the small object, which works in parallel with the original head module. Inquiry head receives feature maps as input with stride 2^l and output $MAP_l = R^{H \times W}$ where $MAP_l^{x',y'}$ denotes the probability that the network (x', y') contains a small object. If the area occupied by the object is less than 32×32, then it is considered as a small object. We encode the distance between its central point and other positions on the feature map as the object mapping of the inquiry head. The distance is also set to be less than s_l to 1 and 0. Then we train inquiry head using FocalLoss and select the position with a prediction result higher than the critical value t as the inquiry.

Next, student needs to learn the more difficult parts of the knowledge, and we suggest to focus on delivering small object information between the teacher and the student, which can be seen in Fig 3. I_i^S and I_i^T correspond to the i-th information.

$$L_{small-distill} = \sum_{i=1}^{N} \sum_{x=1}^{H} \sum_{y=1}^{W} M_{x,y} K_{x,y} L_s(I_{i(l-1)}^S, I_{i(l-1)}^T) \tag{4}$$

The knowledge of the teacher denotes as T and the knowledge of the condition z_i can be expressed as $I_i^T = D(T, z_i)$, I_i^S similarly, and where D is the decoding module. Since currently commonly used detectors contain Feature Pyramid Network (FPN), we denote the multi-scale features as:

$$T = \{Z_\gamma \in R^{C \times H_\gamma \times W_\gamma}\}_{\gamma \in E} \tag{5}$$

where E denotes the spatial resolution and C denotes the channel dimension. We acquire $X^T \in R^{U \times C}$ from the connections at different scales, where $U = \sum_{\gamma \in E} H_\gamma W_\gamma$ is the sum of the number of pixels at multi scales. After we get the information of the small object, we need to annotate it, which is denoted by $A = \{a_i\}_{i=1}^{Q}$, where Q is the number of objects and a_i is the annotation of the information, including category and size information.

To generate learnable embedding of localization knowledge for each small object, we need to annotate the inquiry feature vector. This inquiry feature vector b_i specifies the conditions for collecting the required knowledge:

$$b_i = F_b(\delta(a_i)), b_i \in R^C \tag{6}$$

where $\delta(*)$ is the encoding function and F_b is the multi-layer perception network (MLP). This is represented by the dot product attention [42] of N_m heads in terms of inquiry key attention. Where each head j corresponds to three linear layers (F_j^k, F_j^q, F_j^v) for the construction of keys, inquiries and values respectively. The eigenvalue I_i^T is calculated by representing X^T and the location embedding $V = R^{U \times C}$ projection teacher.

$$I_{i(l-1)}^T = F_j^k(X^T + F_{pe}(V)), F_j^k \in R^{U \times c} \tag{7}$$

$$O_{j(l-1)}^T = F_j^v(X^T), O_j^T \in R^{U \times c} \tag{8}$$

$$b_{ij(l-1)} = F_j^q(b_i), b_{ij} \in R^c \tag{9}$$

$$m_{ij} = softmax(\frac{I_i^T b_{ij}}{\sqrt{c}}), m_{ij} \in R^U \tag{10}$$

where F_{pe} denotes the linear projection on the location embedding. The value features $O_{j(l-1)}^T$ and the inquiry $b_{ij(l-1)}$ are projected by linear mapping on P_{l-1} to the subspace with channel $c = C/N_m$, F_j^v and F_j^q respectively. The perceptual attention mask m_{ij} for the

j-th head of the i-th information is obtained by the normalized dot product of $I^T_{i(l-1)}$ and $b_{ij(l-1)}$. In summary, the inquiries along the key and value features describe the correlation between results and the small object information. We gather $I^T_{i(l-1)} = (m_{ij}, O^T_{j(l-1)})^{N_m}_{j=1}$ as the localization information extracted for small objects from T, which encodes the knowledge corresponding to the i-th information.

It should be noted that there are fundamental differences between our method and Querydet [43]. The inquiry mechanisms of the two are different. Our approach focuses on enhancing the transfer of knowledge for small objects from the teacher to the student during the knowledge distillation process.

3.2.3 *Supplemental Distillation*

Next is the supplemental distillation module, where we use Gcblock [20] to extract background knowledge. In the previous sections, we separate the background and distill the foreground knowledge first, which ignore the relationship between foreground and background. Therefore, we utilize this module to supplement missing knowledge regarding the overall relationship between objects and backgrounds, and transfer it from the teacher to the student. The loss function about background is as follows:

$$L_{backgroud} = \alpha \cdot \sum f'\left(|F^T - F^S|\right)^2$$
$$f'(F) = F + C_{v2} ReLU(LN(C_{v1} \sum_{j=1}^{N_p} \frac{e^{C_k F_j}}{\sum_{m=1}^{N_p} e^{C_k F_M}} F_j)) \tag{11}$$

where C_k, C_{v1} and C_{v2} are the corresponding convolution layers, LN is used for normalization, N_p is the sum of pixels and α is the balance factor.

3.3 Distillation Loss

Finally, we present the final knowledge distillation formula. The attention mask is used as a measure of the correlation between the features and each information.

$$L_{small-distill} = L_{others} + \frac{1}{N_m N_s} \sum_{j=1}^{N_m} \sum_{i=1}^{N_s} < m_{ij}, L_{MSE}(I^S_{i(l-1)}, I^T_{i(l-1)}) > \tag{12}$$

where N_s is the number of information, and $L_{MSE}(I^S_{i(l-1)}, I^T_{i(l-1)})$ is the mean squared error of pixels in the hidden dimension which stabilizes the normalized feature. $< -, - >$ is the Dirac notation of the inner product, and L_{others} is the feature distillation of the other dimensional object information. Combined with the supervised learning loss $L_{detection}$ and the formula is summarized as follows:

$$L_{total} = L_{detection} + L_{backgroud} + \beta L_{small-distill} \tag{13}$$

where β is the hyper-parameter. As described above, we divide the small object knowledge into key knowledge taught to student by the teacher while distilling knowledge of the object based on the hierarchical matching. We use the low-resolution high semantic information of the small object in the neck stage as a inquiry and the high-resolution fine-grained feature map feature values as keys to enhance student's learning of the small object knowledge.

4 Experiments

4.1 Datasets

MS COCO2017 dataset. Our main experiments are implemented on COCO [18], which contains 80 object classes. The training set is 120k images and the validation set is 5k images. All the following results are evaluated on this validation set. Detectors with different performances are evaluated using average precision.

VisDrone dataset. We also conduct experiments on the VisDrone [19] dataset, which is collected by the AISKYEYE team at Tianjin University and contains 11 categories of drone-captured images. The training

Method	Faster R-CNN				RetinaNet			
	mAP	AP_S	AP_M	AP_L	mAP	AP_S	AP_M	AP_L
ResNet101 (Teacher).3×	42.0	25.2	45.6	54.6	40.4	24.0	44.3	52.2
ResNet50 (Student).1×	37.9	22.4	41.1	49.1	37.4	23.1	41.6	48.3
+FitNet [9]	39.3	22.7	42.3	51.7	38.2	23.1	41.6	48.8
+FGFI [11]	39.3	22.5	42.3	52.2	38.6	21.4	42.5	51.5
+ICD [17]	40.9	24.5	44.2	53.5	40.7	24.2	45.0	52.7
+FGD [21]	40.5	22.6	44.7	53.2	39.7	22.0	43.7	53.6
+TinyKD [44]	33.1	15.8	36.2	45.1	-	-	-	-
+**Ours**	**41.7 (+3.8)**	**24.8(+2.4)**	44.9	54.2	**40.7 (+3.3)**	**24.6 (+1.5)**	44.3	52.1

Table 1. Our methods are implemented on Faster R-CNN and RetinaNet, respectively, and compared with others' methods on MS COCO dataset. The teacher model is ResNet101 (3×) and the student uses ResNet50 (1×).

Detector	Setting	Type	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Mask R-CNN	ResNet101 (Teacher).3×	BBox	42.9	63.3	46.8	26.4	46.6	56.1
	ResNet50 (Student).1×		38.6	59.5	42.1	22.5	42.0	49.9
	Ours		**41.0 (+2.4)**	61.5	45.0	**25.2 (+2.7)**	44.2	53.1
	ResNet101 (Teacher).3×	Mask	38.6	60.4	41.3	19.5	41.3	55.3
	ResNet50 (Student).1×		35.2	56.3	37.5	17.2	37.2	50.3
	Ours		**37.2 (+2.0)**	58.6	40.1	**19.3 (+2.1)**	40.0	53.2
FCOS	ResNet101 (Teacher).3×	BBox	43.2	62.4	46.8	26.1	46.2	52.8
	ResNet50 (Student).1×		38.6	57.4	41.4	22.3	42.5	49.8
	Ours		**43.6 (+5.0)**	62.3	47.3	**27.4 (+5.1)**	47.5	55.6

Table 2. Our approach is also evaluated on Mask R-CNN and FCOS. The results with bounding boxes (BBox) or instance masks (Mask) are reported, respectively.

Detector	Setting	mAP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster R-CNN	ResNet101 (Teacher).3×	42.0	62.5	45.9	25.2	45.6	54.6
	VoVNetV2-19 (Student).1×	32.0	51.4	34.0	18.4	34.4	40.8
	Ours	**36.4 (+4.4)**	56.8	39.1	**21.6 (+3.2)**	39.1	46.3
Faster R-CNN	VoVNetV2-57 (Teacher).3×	43.3	64.3	47.0	27.5	46.7	55.3
	ResNet50 (Student).1×	37.9	58.8	41.1	22.4	41.1	49.1
	Ours	**40.8 (+2.9)**	61.4	44.7	**24.2 (+1.8)**	44.2	53.3
Faster R-CNN	ResNet101 (Teacher) .3×	42.0	62.5	45.9	25.2	45.4	54.6
	MobileNetV2(Student) .1×	26.4	45.0	27.2	14.9	28.6	33.2
	Ours	**29.8 (+3.4)**	47.7	31.9	**16.7 (+1.8)**	32.0	38.8
RetinaNet	ResNet101 (Teacher) .3×	40.4	60.3	43.2	24.0	44.3	52.2
	MobileNetV2 (Student) .1×	20.4	33.3	21.6	10.7	22.1	26.1
	Ours	**23.4 (+3.0)**	37.2	24.7	**13.4 (+2.7)**	25.0	29.2

Table 3. Results with different backbone networks. We replace the student and the teacher model on Faster R-CNN with the more efficient VoVNetV2, respectively. And we also consider the mobile network and use MobileNetV2 for our experiments.

set includes 6,471 images and 10 object classes, which contains mostly small objects.

4.2 Implementation Details

Our experiments are all set up in the widely used detectron2 [45] and AdelaiDet [46], using pytorch [47] for the study. All programs are run on a single NVIDIA RTX3060 and our batch size sets to 2. We follow the criterion in detectron2 where 1× scheduler denotes 9k training sessions. For optimizing the transformer decoder during knowledge distillation, we use the AdamW optimizer [48] as the decoder. The MLP uses the regular settings [41, 42]. Other hyper-parameters are set with reference to DETR [41], where the learning rate and weight decay are set to 0.001. We set the hidden dimension of the decoder and all MLPs to 256, and the decoder has 8 heads in parallel. In the supplemental distillation module, the background distillation hyper-parameters $\alpha = 1 \times 10^{-3}$.

4.3 Main Results

Results in MS COCO. Our method has good generality and can be easily applied to various detection frameworks. We start with experiments on the popular detectors, as shown in Table 1. The pre-trained models in the experiments come from the official release of detectron2, the teacher model uses the ResNet101 backbone network trained on 3×, and the student uses the ResNet50 backbone network trained on 1×. It can be observed that we compare with several more advanced methods listed so far, and it is not difficult to find that the results of our method have an advantage. In Faster R-CNN, the mAP value of our method improves by 3.8 over the baseline, while the AP_S (less than 32× 32) is higher by 2.4. This fully demonstrates that our method can lead to an improvement in the detection of small objects. The reason for the poor performance of the TinyKD [44] experiment may be that it is an exclusive design for tiny person detection, and it is not suitable for general scenarios of the COCO dataset. This also reflects the more versatile nature of our method. As shown in Table 2, we also apply our method to instance segmentation [49] and FCOS [25], the experimental results demonstrate its effectiveness in this task as well.

To confirm the effectiveness and generalization of our approach, we replace ResNet with more efficient VoVNetV2 [50]. For the lightweight network, we try MobileNetV2 [51], which is proposed by Google in 2018.

Heterogeneous Distillation. To confirm the considerable generality of our method, we replace the teacher network under Faster R-CNN and student network for our experiments. VoVNetV2 is an efficient backbone network proposed by Lee *et al.* [50] in 2019 for real-time object detection and can fully exploit the computational efficiency of GPU, so we choose VoVNetV2 to replace ResNet for our experiments. The pre-trained models we used are from the official release, with VoVNetV2-57 (3×) for the teacher model and VoVNetV2-19 (1×) for the student, other settings remain the same as before. As shown in Table 3, the experimental results demonstrate that our method is capable of conducting heterogeneous network distillation. It is worth exploring that the detection ability of the student does not exceed the original experimental results after replacing it with a powerful teacher model. This itself may be related to the difference between different network structures.

Mobile Network. The main purpose of the lightweight network is to deploy the model on devices with low computing power, and we mostly use mobile network for deployment. So we use MobileNetV2, a classic lightweight network, for our experiment. We use ResNet101 (3×) for the teacher and MobileNetV2 (1×) for the student. The teacher model uses the of-

Method	RetinaNet			
	mAP	AP_S	AP_M	AP_L
R101 (Tea).3×	23.8	14.0	36.3	58.0
R50 (Stu).1×	20.6	11.5	31.9	55.0
+ICD [17]	23.6	13.9	36.1	56.5
+FGD [21]	22.9	11.8	37.3	57.5
+Ours	**24.0 (+3.4)**	**14.2 (+2.7)**	36.4	56.6

Table 4. Results on VisDrone. This set of comparison experiments shows that our method still performs well on the small object dataset. Both teacher and student are trained by ourselves. ICD [17] and FGD [21]. R101(Tea) means ResNet101(Teacher) and R50(Stu) means ResNet50(Student).

ficial published model in detectron2, and the student model is obtained by our training. Our experimental results are shown in Table 3, which shows the comparison among students under different frameworks.

Results in VisDrone. The experimental results are obtained on the validation set and are shown in Table 4. The results indicate that our proposed method is indeed effective for enhancing the features of small objects and has better detection capabilities compared to the baseline and other methods. It is worth noting that the detection accuracy of our student even surpasses that of the teacher. The hyper-parameters are set the same as in the COCO for the experiments.

4.4 Ablation Studies

Effect of supplemental distillation. Our purpose of separating the foreground from the background is to avoid the effect of background noise on small objects. However, the background information also carries some important information, which contributes to the model learning more knowledge. Therefore, we introduce a supplemental distillation module to teach the background knowledge to the student. This can make the student's knowledge framework more comprehensive. To validate the effectiveness of the supplemental distillation module, we conducted a set of experiments as shown in Table 5. The results demonstrate a noticeable decrease in experimental performance when the supplemental distillation module is removed. Therefore, its inclusion in the HMKD framework is highly necessary.

Separate+Supplement	mAP	AP_S	AP_M	AP_L
×	40.6	24.3	43.9	53.1
✓	41.7	24.8	44.9	54.2

Table 5. Ablation study of the supplemental distillation module.

Number of heads in the decoder. The setting of heads in the decoder is an important influence factor on the detection performance. Heads balance the number of dimensions and spaces in the subspace. After our experiments, the best number is again kept around 8 as shown in Table 6, which is the same as the original number of probes.

Heads	mAP	AP_S	AP_M	AP_L
1	38.4	23.0	41.6	49.9
4	40.0	23.8	44.4	51.3
8	40.7	24.6	44.3	52.1
16	39.8	23.5	43.9	50.9

Table 6. The effect of different amounts of heads.

Model Analysis. The computation power of a model is typically measured in Giga-Floating Point Operations per Second (GFLOPS), which indicates the amount of computational work that can be performed per second. As the Fig 4 shows, the MobileNet model in Faster R-CNN network is not very effective in light-

weighting. Its model parameters are the same as those of ResNet50 and its computational power is lower than ResNet50. This suggests that the MobileNet model may not be the most effective option for light-weighting in Faster R-CNN network. In contrast, the RetinaNet network shows a more pronounced change in the computational power with a lighter model. This also corroborates that the one-stage model may be more suitable for deployment on edge devices.

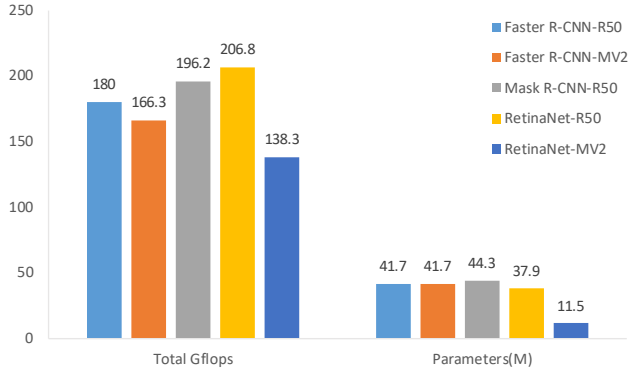

Fig.4. Gflops and Parameters with different methods.

5 Conclusion

We note that small object knowledge is not easily transferred to students during knowledge distillation, which is a challenging task. Therefore, we design Hierarchical Matching Knowledge Distillation (HMKD) to enhance students' knowledge learning of small objects. We encode high-semantic information at low-resolution of FPN as inquiries, and represent fine-grained graph feature values at high-resolution as key-values. Through extensive experiments, our method effectively enhances the student's understanding of small objects detection capability and is suitable for mainstream object detectors or instance segmentation models. The training time increases due to the additional augmentation design for small objects and this is also where we will optimize in the next step. At the same time, we will investigate differences in knowledge extraction or learning effectiveness between teachers and students.

References

[1] Hubara I, Courbariaux M, Soudry D, El-Yaniv R, Bengio Y. Binarized neural networks. *Advances in neural information processing systems*, 2016, 29.

[2] Hubara I, Courbariaux M, Soudry D, El-Yaniv R, Bengio Y. Quantized neural networks: Training neural networks with low precision weights and activations. *The Journal of Machine Learning Research*, 2017, 18(1):6869–6898.

[3] Rastegari M, Ordonez V, Redmon J, Farhadi A. Xnor-net: Imagenet classification using binary convolutional neural networks. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV*, 2016, pp. 525–542.

[4] Han S, Pool J, Tran J, Dally W. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 2015, 28.

[5] Li H, Kadav A, Durdanovic I, Samet H, Graf H P. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[6] He Y, Zhang X, Sun J. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1389–1397.

[7] Hinton G E, Vinyals O, Dean J. Distilling the knowledge in a neural network. *arXiv: Machine Learning*, 2015.

[8] Ji M, Heo B, Park S. Show, attend and distill: Knowledge distillation via attention-based feature

matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2021, pp. 7945–7952.

[9] Romero A, Ballas N, Kahou S E, Chassang A, Bengio Y. Fitnets: Hints for thin deep nets. In *ICLR*, 2015.

[10] Selvaraju R R, Cogswell M, Das A, Vedantam R, Parikh D, Batra D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[11] Wang T, Yuan L, Zhang X, Feng J. Distilling object detectors with fine-grained feature imitation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4933–4942.

[12] Guo J, Han K, Wang Y, Wu H, Chen X, Xu C, Xu C. Distilling object detectors via decoupled features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2154–2164.

[13] Zhang L, Ma K. Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors. In *International Conference on Learning Representations*, 2021.

[14] Heo B, Kim J, Yun S, Park H, Kwak N, Choi J Y. A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1921–1930.

[15] Ren S, He K, Girshick R, Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 2015, 28.

[16] Lin T Y, Dollar P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. *IEEE Computer Society*, 2017.

[17] Kang Z, Zhang P, Zhang X, Sun J, Zheng N. Instance-conditional knowledge distillation for object detection. *Advances in Neural Information Processing Systems*, 2021, 34:16468–16480.

[18] Lin T Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick C L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 2014, pp. 740–755.

[19] Du D, Zhu P, Wen L, Bian X, Lin H, Hu Q, Peng T, Zheng J, Wang X, Zhang Y 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*, 2019, pp. 0–0.

[20] Cao Y, Xu J, Lin S, Wei F, Hu H. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, 2019, pp. 0–0.

[21] Yang Z, Li Z, Jiang X, Gong Y, Yuan Z, Zhao D, Yuan C. Focal and global knowledge distillation for detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4643–4652.

[22] Chen G, Choi W, Yu X, Han T, Chandraker M. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 2017, 30.

[23] Li Q, Jin S, Yan J. Mimicking very efficient network for object detection. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 2017, pp. 6356–6364.

[24] Lin T Y, Goyal P, Girshick R, He K, Dollár P. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[25] Tian Z, Shen C, Chen H, He T. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9627–9636.

[26] Ge Z, Liu S, Wang F, Li Z, Sun J. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021.

[27] Kisantal M, Wojna Z, Murawski J, Naruniec J, Cho K. Augmentation for small object detection. *arXiv preprint arXiv:1902.07296*, 2019.

[28] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C Y, Berg A C. Ssd: Single shot multibox detector. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, 2016, pp. 21–37.

[29] Zoph B, Cubuk E D, Ghiasi G, Lin T Y, Shlens J, Le Q V. Learning data augmentation strategies for object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, 2020, pp. 566–583.

[30] Cai Z, Fan Q, Feris R S, Vasconcelos N. A unified multi-scale deep convolutional neural network for fast object detection. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, 2016, pp. 354–370.

[31] Fu C Y, Liu W, Ranga A, Tyagi A, Berg A C. Dssd: Deconvolutional single shot detector. *arXiv preprint arXiv:1701.06659*, 2017.

[32] Kong T, Yao A, Chen Y, Sun F. Hypernet: Towards accurate region proposal generation and joint object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 845–853.

[33] Li Y, Chen Y, Wang N, Zhang Z. Scale-aware trident networks for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6054–6063.

[34] Lin T Y, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[35] Singh B, Davis L S. An analysis of scale invariance in object detection snip. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3578–3587.

[36] Cao C, Wang B, Zhang W, Zeng X, Yan X, Feng Z, Liu Y, Wu Z. An improved faster r-cnn for small object detection. *Ieee Access*, 2019, 7:106838–106846.

[37] Chen L C, Papandreou G, Kokkinos I, Murphy K, Yuille A L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions*

on pattern analysis and machine intelligence, 2017, 40(4):834–848.

[38] Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[39] Singh B, Najibi M, Davis L S. Sniper: Efficient multi-scale training. *Advances in neural information processing systems*, 2018, 31.

[40] Chen Y, Zhang P, Li Z, Zhang X, Meng G. Feedback-driven data provider for object detection. arxiv. *Preprint]. Available online at: https://arxiv. org/abs/2004.12432 (accessed January 10, 2021).[Google Scholar]*, 2020.

[41] Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, 2020, pp. 213–229.

[42] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser Ł, Polosukhin I. Attention is all you need. *Advances in neural information processing systems*, 2017, 30.

[43] Yang C, Huang Z, Wang Y C, Huang Z, Wang N. Querydet: Cascaded sparse query for accelerating high-resolution small object detection. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, 2022, pp. 13668–13677.

[44] Liu H, Liu Q, Liu Y, Liang Y, Zhao G. Exploring effective knowledge distillation for tiny object detection. In *2023 IEEE International Conference on Image Processing (ICIP)*, 2023, pp. 770–774.

[45] Wu Y, Kirillov A, Massa F, Lo W Y, Girshick R. Detectron2. 2019.

[46] Tian Z, Chen H, Wang X, Liu Y, Shen C. Adelaidet: A toolbox for instance-level recognition tasks, 2019.

[47] Yang J, Lu J, Batra D, Parikh D. A faster pytorch implementation of faster r-cnn. *GitHub repository*, 2017.

[48] Loshchilov I, Hutter F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[49] He K, Gkioxari G, Dollár P, Girshick R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[50] Lee Y, Hwang J w, Lee S, Bae Y, Park J. An energy and gpu-computation efficient backbone network for real-time object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019, pp. 0–0.

[51] Sandler M, Howard A, Zhu M, Zhmoginov A, Chen L C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.