Lin WD, Deng YY, Gao Y *et al.* CAT: A simple yet effective cross-attention transformer for one-shot object detection. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 39(2): 460-471 Mar. 2024. DOI: 10.1007/s11390-024-1743-6

CAT: A Simple yet Effective Cross-Attention Transformer for One-Shot Object Detection

Wei-Dong Lin[1, 2] (林蔚东), Yu-Yan Deng[1, 2] (邓玉岩), Yang Gao[1, 2] (高　扬), Ning Wang[1, 2] (王　宁)
Ling-Qiao Liu[3] (刘凌峤), Lei Zhang[1, 2] (张　磊), *Member, CCF*
and Peng Wang[1, 2, *] (王　鹏), *Senior Member, CCF*

[1] *School of Computer Science, Northwestern Polytechnical University, Xi'an, 710000, China*

[2] *National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology, Northwestern Polytechnical University, Xi'an, 710000, China*

[3] *School of Computer Science, The University of Adelaide, Adelaide, SA 0115, Australia*

E-mail: weidong.lin@mail.nwpu.edu.cn; dengyuyan@mail.nwpu.edu.cn; gy7@mail.nwpu.edu.cn; ningw@mail.nwpu.edu.cn
lingqiao.liu@adelaide.edu.au; nwpuzhanglei@nwpu.edu.cn; peng.wang@nwpu.edu.cn

Received June 27, 2021; accepted January 18, 2024.

Abstract Given a query patch from a novel class, one-shot object detection aims to detect all instances of this class in a target image through the semantic similarity comparison. However, due to the extremely limited guidance in the novel class as well as the unseen appearance difference between the query and target instances, it is difficult to appropriately exploit their semantic similarity and generalize well. To mitigate this problem, we present a universal Cross-Attention Transformer (CAT) module for accurate and efficient semantic similarity comparison in one-shot object detection. The proposed CAT utilizes the transformer mechanism to comprehensively capture bi-directional correspondence between any paired pixels from the query and the target image, which empowers us to sufficiently exploit their semantic characteristics for accurate similarity comparison. In addition, the proposed CAT enables feature dimensionality compression for inference speedup without performance loss. Extensive experiments on three object detection datasets MS-COCO, PASCAL VOC and FSOD under the one-shot setting demonstrate the effectiveness and efficiency of our model, e.g., it surpasses CoAE, a major baseline in this task, by 1.0% in average precision (AP) on MS-COCO and runs nearly 2.5 times faster.

Keywords one-shot object detection, Transformer, attention mechanism

1 Introduction

Object detection is a fundamental task in computer vision domain, which aims to predict a bounding box with a category label for each instance of interest in the image[1]. Although deep convolutional neural networks (DCNNs) based object detection methods have achieved great success in recent years, their success heavily relies on a huge amount of annotated data, which is often difficult or even infeasible to collect in real applications due to the expensive annotation cost. Therefore, it is inevitable to cope with object detection for unseen classes with only a few annotated examples at the test phase.

In this study, we mainly focus on a challenging task in object detection, i.e., one-shot object detection. Given a novel class, there is only one query image with one annotated object, and a detector is then required to find all objects of the same category as the annotated object in a target image. Till now, some effective methods have been proposed, which mainly focus on building a two-stage paradigm[2]. Specifically, in the first stage, the features of the

Regular Paper

This work was supported by the National Science and Technology Major Project under Grant No. 2020AAA0106900, the National Natural Science Foundation of China under Grant Nos. U19B2307 and 61876152, the Shaanxi Provincial Key Research and Development Program of China under Grant No. 2021KWZ-03, and the Natural Science Basic Research Program of Shaanxi Province of China under Grant No. 2021JCW-03.

*Corresponding Author

query image and the target image are aggregated to exploit their semantic correspondence utilizing channel attention[3] or correlation filtering[4]. Then, a region proposal network is utilized to detect all candidate objects and real ones are ultimately located by a semantic similarity comparison based classifier. However, due to extremely limited guidance for the novel class (i.e., only one annotated sample) as well as the unseen appearance difference between the query object and the target one (e.g., that is often caused by the intra-class variation and different imaging endearments), these existing methods still fail to appropriately generalize well with pleasing performance.

To mitigate this problem, we revisit the one-shot object detection problem and attempt to explore the accurate semantic correspondence between the query object and the target image for performance enhancement. Considering that the great appearance difference often conceals semantic correspondence between different objects into an unknown embedding space, we have to sufficiently exploit any detailed correspondence between two images. A direct way is to explore the relation between each sub-region from the query image and that in the target one. Following this idea, we propose a Cross-Attention Transformer (CAT) module and embed it into the two-stage detection paradigm for comprehensive exploration of the bidirectional correspondence between the target and query images. The proposed CAT module consists of two streams of interleaved Transformers[5]. Given the grid features generated from a Siamese feature extractor[3], the two-stream transformer is utilized to exploit the bi-directional correspondence between any paired sub-regions from the query and the target images through computing the cross-attention between them. As shown in Fig.1, the CAT module can sufficiently exploit the semantic characteristics of each image as well as their grid-level correspondences, which will be beneficial for accurate similarity comparison in the second stage. In addition, due to sufficient information captured by the CAT module, the dimensionality of the final feature representation of each object can be effectively compressed without performance loss. To verify the effectiveness of the proposed CAT, we compare it with the state-of-the-art on three standard one-shot object detection benchmarks and observe significant performance and efficiency improvement.

In summary, this study mainly contributes in the following three aspects.

(a) (b) (c) (d)

Fig.1. Visualization results of the intermediate feature maps. (a) Query images. (b) Target images. (c) Backbone images. (d) CAT outputs. We visualize the response maps of the input and output of our proposed CAT module in (c) and (d). By capturing the bidirectional correspondence between query and target images, our CAT module significantly refines the response map and pays more attention to the objects with the same category of query objects.

- We propose a CAT module which is able to sufficiently exploit the grid-level correspondence between the query and target image for accurate and efficient one-shot object detection. It is noticeable that the CAT module is an universal module which can be seamlessly plugged into other existing one-shot object detection frameworks.
- With the CAT module, we develop an effective one-shot detection network, which demonstrates the state-of-the-art accuracy on three standard benchmarks for one-shot object detection.
- By compressing the feature dimensions, the proposed model is capable of running nearly 2.5 times faster than the current state-of-the-art baseline CoAE[3] without accuracy degradation.

2 Related Work

In this section, we will briefly review two lines of researches related to this study.

2.1 Few-Shot Object Detection

The key for few-shot object detection is to establish a similarity metric that can be appropriately generalized to unseen classes with a few labeled examples (i.e., the query set). Efforts have been made recently from different perspectives, such as transfer

learning, metric learning and attention-based methods.

Specifically, for transfer learning, Chen *et al.*[6] presented the regularization techniques to relieve the over-fitting caused by directly transferring knowledge from a large auxiliary dataset to the novel classes. Kang *et al.*[7] developed a single-stage detector combined with a meta-model that re-weights the importance of features from the base model. For metric learning, Karlinsky *et al.*[8] introduced a distance metric based classifier into the Region of Interest (RoI) module in the detector, which maps the objects into the universal embedding space. The attention-based methods focus on modelling the correspondence between the target and the query. Hsieh *et al.*[3] designed a co-attention based model called CoAE which leverages the correlated features from the target and the query for better generalization performance. Fan *et al.*[4] introduced depth-wise convolution to get the attention feature map in the region proposal network (RPN) phase and proposed the multi-relation detector to model different relationships in the region-convolutional neural network (R-CNN) phase. Osokin *et al.*[9] firstly performed dense correlation matching based on local features and then conducted spatial alignment and bi-linear resampling to compute the detection score.

Our work lies on the third line of research, the attention-based methods. Different from previous work, our proposed CAT module empowers us to deeply exploit the grid-level bidirectional correspondence between the target and the query, using stacks of cross-attention transformer layers.

2.2 Visual Transformer

Witnessing that Transformer[5] becomes the de-facto standard in natural language processing (NLP)[10], recent literature commences introducing transformer-like networks into various computer vision tasks, including image recognition[11, 12], object detection[13, 14], segmentation[15], visual question answering (VQA)[16, 17], and point cloud[18]. The Vision Transformer (ViT)[11] directly feeds image patches into a transformer for image classification, which removes the need of any convolution operation. Yuan *et al.*[19] proposed a layer-wise tokens-to-tokens transformation to progressively structurize the image to tokens, which has better performance and lower overhead than ViT. Carion *et al.*[13] proposed DETR, a transformer encoder-decoder architecture that performs end-to-end object detection as set prediction. It does not rely on many manual components required by traditional detectors, such as non-maximum suppression and anchor selection. Ye *et al.*[15] proposed a cross-modal self-attention model to capture the long-range dependencies between language and visual features. LXMERT[16] and VL-BERT[17] are transformer-like visual-linguistic pretraining models that achieve superior performance on several vision-language tasks. To the best of our knowledge, our proposed model is the first attempt to employ Transformers for the task of one-shot object detection. Moreover, it relies on a two-stream cross-attention architecture, rather than the commonly-adopted self-attention mechanisms.

3 Our Approach

We formulate the one-shot object detection task as in CoAE[3]. Given a query image patch p with its class label, the one-shot detector aims to detect all object instances of the same class in a target image $\boldsymbol{I}$, where we assume that at least one instance exists in the target image. We denote the set of classes in the test phase (unseen classes) as C_0 while those in the training phase (seen classes) is C_1, and $C_0 \cap C_1 = \emptyset$. The model is trained with the annotated data of the seen classes, and generalized to detect unseen classes with a single query image.

3.1 Overall Architecture

As shown in Fig.2, our proposed method is composed of three parts, including the feature extractor (backbone), the cross-attention module CAT and the similarity-based detection head. At first, we adopt ResNet-50[20] to extract features from both the query image $\boldsymbol{I}_q \in \mathbb{R}^{3\times H_q \times W_q}$ and the target image $\boldsymbol{I}_t \in \mathbb{R}^{3\times H_t \times W_t}$. Note that the backbone parameters are shared between the query and the target images. What needs to be especially explained is that we only use the first three blocks of ResNet-50 to extract feature maps with high resolutions. For the ease of representation, we denote $\phi(\boldsymbol{I}_t) \in \mathbb{R}^{C\times H'_t \times W'_t}$ and $\phi(\boldsymbol{I}_q) \in \mathbb{R}^{C\times H'_q \times W'_q}$ as the feature maps of the target and the query images respectively, where ϕ represents the backbone, $C = 1\,024$, $H'_t = H_t/16$, $W'_t = W_t/16$, $H'_q = H_q/16$ and $W'_q = W_q/16$. After that, we use a 3×3 convolution and a 1×1 convolution to compress the number of channels of

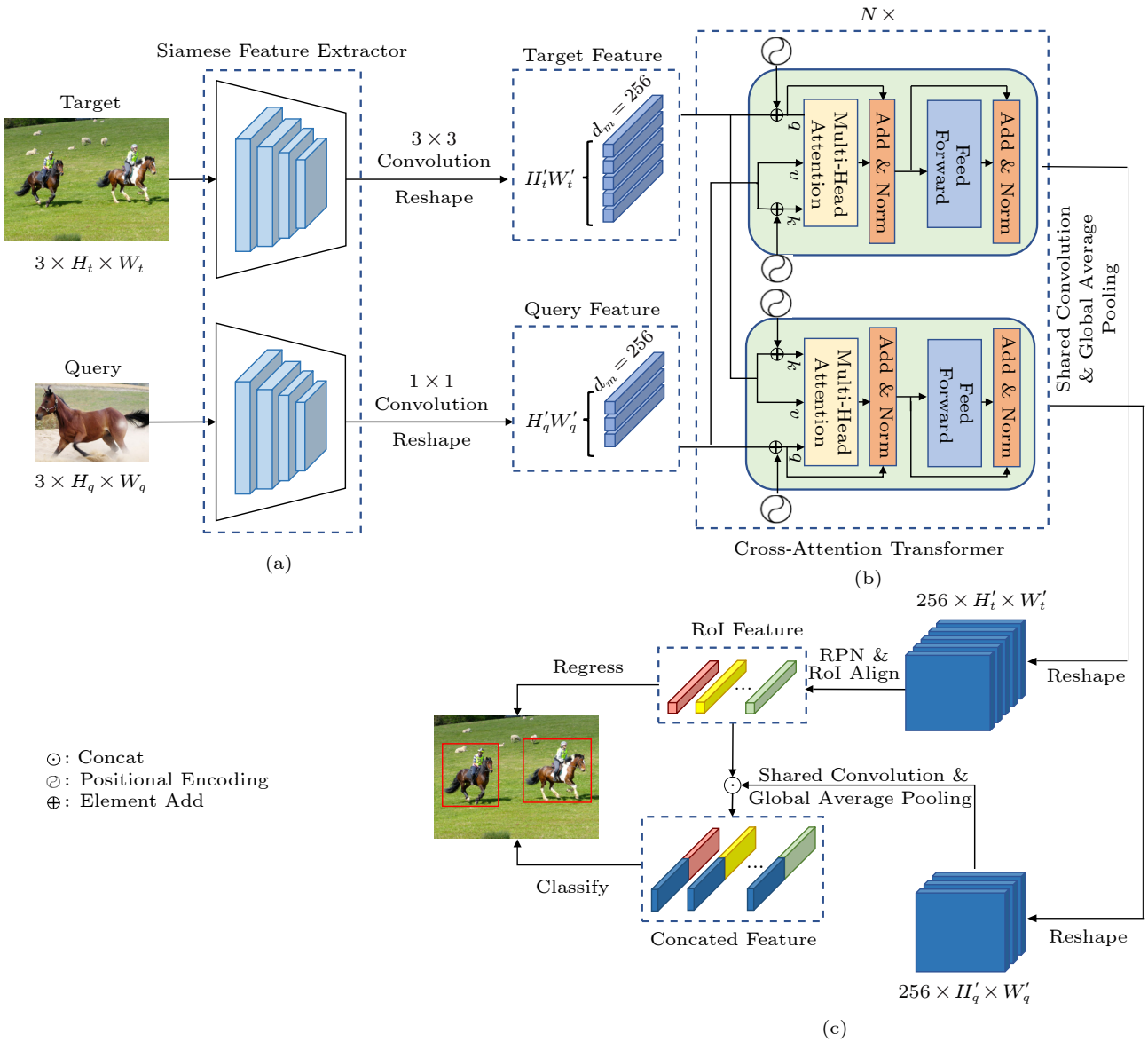


Fig.2. Overall architecture of the proposed module for one-shot object detection. Our detector is composed of three parts. The first part is (a) a shared ResNet-50[20] backbone used to extract features of both the target and query images. And the following part is (b) our Cross-Attention Transformer (CAT) module that fuses the features from backbone and enhances the features of the regions which may be the same category as query in the target image, while the last part is (c) the detection head with a regular RPN head and an R-CNN head like Faster R-CNN[2].

$\phi(\boldsymbol{I}_t), \phi(\boldsymbol{I}_q)$, respectively from 1 024 to $d_m = 256$. Both features are flattened in the spatial dimension and further deeply aggregated by the CAT module with the cross-attention mechanism as defined in the following formula:

$$(\boldsymbol{F}_t, \boldsymbol{F}_q) = CAT(\phi(\boldsymbol{I}_t)', \phi(\boldsymbol{I}_q)'),$$

where $\phi(\boldsymbol{I}_t)' \in \mathbb{R}^{d_m \times H'_t W'_t}$, $\phi(\boldsymbol{I}_q)' \in \mathbb{R}^{d_m \times H'_q W'_q}$ are the input sequences, $\boldsymbol{F}_t \in \mathbb{R}^{d_m \times H'_t \times W'_t}$, $\boldsymbol{F}_q \in \mathbb{R}^{d_m \times H'_q \times W'_q}$ are the output feature maps after cross-attention, and CAT represents the operation of our proposed Cross-Attention Transformer.

In the end, the RPN-based head takes as input the aggregated target features and generates proposals for further classification and regression. The features of proposals $\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_n$ extracted from $\boldsymbol{F}_t$ by RoI align are fed into a regressor to obtain refined bounding boxes.

$$bbox_i = \Phi_r(\psi(\boldsymbol{F}_t, \boldsymbol{p}_i)),$$

where Φ_r represents the regressor and ψ represents the operation of RoI align. For similarity-based classification, we first apply global average pooling on the RoI features and the aggregated query features $\boldsymbol{F}_q$,

and then concatenate them as the input of classifier Φ_c. The classification results $P(bbox_i), i = 1, 2, \ldots, n$ can be formulated as:

$$P(bbox_i) = \Phi_c(Concat(GAP(\psi(\boldsymbol{F}_t, \boldsymbol{p}_i)), GAP(\boldsymbol{F}_q))),$$

where $Concat$ represents the operation of concatenate, and GAP represents the operation of global average pooling.

3.2 Cross-Attention Transformer Module

The Cross-Attention Transformer (CAT) model is the key component of our proposed framework. Based on the Transformer architecture, it models the bidirectional correspondences between grids of the target and query images and performs dual feature aggregation for both the target and the query.

The basic building block of Transformer is the "Scaled Dot-Product Attention" defined as follows:

$$Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V},$$

where $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ represent queries, keys and values, respectively. d_k is the dimension of keys.

Vaswani *et al.*[5] thought Multi-Head Attention mechanism can be further employed to jointly attend to information from different representation subspaces:

$$MH(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = Concat(head_1, \ldots, head_{\mathrm{M}})\boldsymbol{W}^O,$$
$$head_i = Attention(\boldsymbol{Q}\boldsymbol{W}_i^Q, \boldsymbol{K}\boldsymbol{W}_i^K, \boldsymbol{V}\boldsymbol{W}_i^V),$$

where MH represents the operation of multi-head attention, $\boldsymbol{W}_i^Q \in \mathbb{R}^{d_m \times d'}$, $\boldsymbol{W}_i^K \in \mathbb{R}^{d_m \times d'}$, $\boldsymbol{W}_i^V \in \mathbb{R}^{d_m \times d'}$ are the matrices to compute the so-called query, key and value embeddings respectively, and $\boldsymbol{W}^O \in \mathbb{R}^{Md' \times d_m}$ is the projection matrix. In our work, we set $d' = d_m/M$, $d_m = 256$ and $M = 8$.

After the multi-head attention operation, the output is sent into a feed-forward network (FFN) module composed of two linear transformations with ReLU activation, defined as:

$$FFN(\boldsymbol{x}) = \max(\boldsymbol{0}, \boldsymbol{x}\boldsymbol{W}_1 + \boldsymbol{b}_1)\boldsymbol{W}_2 + \boldsymbol{b}_2,$$

where $\boldsymbol{W}_1, \boldsymbol{W}_2$ and $\boldsymbol{b}_1, \boldsymbol{b}_2$ are the weight matrices and basis vectors respectively.

Carion *et al.*[13] proposed a Transformer-like model (DETR) for general object detection and obtained competing performance. Although we also employ Transformer in this work, there are still significant differences between DETR and our model. Firstly, the challenges faced by the two models are different. As a general object detector, DETR focuses on the discrimination between the foreground and the background, and accurate bounding box regression. On the contrary, the difficulty of one shot detection is mainly on similarity-based comparison, rather than proposal generation[21]. Through experiments, we found that in many cases, one-shot detection models can produce accurate bounding boxes of salient objects but fail to assign correct class labels. To resolve their individual challenges, DETR and our model choose different model architectures. DETR is built upon self-attention that explores long-range dependencies between pixels of a single input image. In contrast, our model relies on a two-stream architecture which performs cross-attention (query-to-target and target-to-query) to exploit the similarity between sub-regions of the query and the target images.

To be more specific, $\boldsymbol{X}_t \in \mathbb{R}^{N_t \times d_m}$ and $\boldsymbol{X}_q \in \mathbb{R}^{N_q \times d_m}$ represent the input sequences that are the flattened feature maps of the target and the query images respectively, as shown in Fig.2. Note that $N_t = H_t' \times W_t'$ and $N_q = H_q' \times W_q'$ are the lengths of the sequences, respectively. Following Carion *et al.*[13], we use the *sine* function to generate spatial position encoding for input sequences $\boldsymbol{X}_t$ and $\boldsymbol{X}_q$. In one stream of CAT, we let $\boldsymbol{Q} = \boldsymbol{X}_t$ and $\boldsymbol{K} = \boldsymbol{V} = \boldsymbol{X}_q$, and obtain the aggregated target features. This procedure can be summarized as:

$$Y_t = Norm(\widetilde{\boldsymbol{X}}_t + FFN(\widetilde{\boldsymbol{X}}_t)),$$

$$\widetilde{\boldsymbol{X}}_t = Norm(\boldsymbol{X}_t + \boldsymbol{P}_t + MH(\boldsymbol{X}_t + \boldsymbol{P}_t, \boldsymbol{X}_q + \boldsymbol{P}_q, \boldsymbol{X}_q)),$$

where $Norm$ represents the operation of layer normalization, MH represents the operation of multi-head attention, and $\boldsymbol{P}_t \in \mathbb{R}^{N_t \times d_m}, \boldsymbol{P}_q \in \mathbb{R}^{N_q \times d_m}$ are the spatial position encodings corresponding to $\boldsymbol{X}_t$ and $\boldsymbol{X}_q$, respectively. In another stream, we set $\boldsymbol{Q} = \boldsymbol{X}_q$ and $\boldsymbol{K} = \boldsymbol{V} = \boldsymbol{X}_t$ and generate $\boldsymbol{Y}_q$, the aggregated query features. The above whole computation can be viewed as one layer of our proposed CAT, and the outputs of one layer will be the inputs of the next layer. In our work, we set the number of layers $N = 4$.

The outputs of the CAT module are then reshaped to new feature maps $\boldsymbol{F}_t$ and $\boldsymbol{F}_q$ that share the same sizes as the origin feature maps, where $\boldsymbol{F}_t$ is fed into the subsequent RPN and $\boldsymbol{F}_q$ is used in similarity-based classification.

4 Experiments

Our experiments are conducted on MS-COCO[22]

(short for COCO), PASCAL VOC[23] and the recently released FSOD[4] dataset. In Subsection 4.1, we first introduce implementation details. Then we carry out ablation study and comparison with SOTA in Subsections 4.2 and 4.3 respectively.

4.1 Implementation Details

Training Details. Our network is trained with stochastic gradient descent (SGD) over four NVIDIA RTX-2080Ti GPUs for 10 epochs with the initial learning rate being 0.01 and a mini-batch of 16 images. The learning rate is reduced by a factor of 10 at epoch 5 and 9, respectively. Weight decay and momentum are set to 0.000 1 and 0.9, respectively. As in CoAE[3], the backbone ResNet-50 model is pretrained on a reduced training set of ImageNet in which all the MS-COCO classes are removed to ensure that our model does "foresee" any unseen class. The target images are resized to have their shorter side being 600 and their longer side less than or equal to 1 000, and the query image patches are resized to a fixed size of 128×128. We built our model on MMDetection[24], which is a general object detection framework based on PyTorch. Based on spatial-wise and channel-wise co-attention, CoAE[3] achieves the best performance over existing approaches and serves as a major baseline in our paper. For strictly fair comparison, we re-implemented the CoAE model on the same MMDetection framework, and achieved significantly better results than the original author-provided version on all the three evaluated datasets. The reason may be better training strategies in MMDetection, such as multiply data augmentations and optimized pipeline.

Inference Details. The same evaluation strategy as CoAE is applied for fair comparison. Specifically, we firstly shuffle the query image patches of that class with a random seed of target image ID, and then sample the first five query image patches. We run our evaluations on these patches and take the average of these results as the stable statistics for evaluation.

4.2 Ablation Study

Since CAT is the key component of our model, in this subsection we mainly explore the effect of this module with different hyper-parameters. For easy illustration, our ablation experiments are conducted on COCO split 1 which will be discussed in Subsection 4.3.

Transformer Structure. Our CAT module consists of a stack of two-stream transformer layers, each of stream performing target-to-query or query-to-target attention and generating the corresponding target or query features. To better demonstrate the effectiveness of our two-stream transformer layers, we conduct several ablative experiments as shown in Table 1. For a clear comparison, we first remove the entire transformer layers and report the performance on both unseen and seen classes. One can see that the model incurs 8.0% and 14.2% AP (average precision) drops on unseen and seen classes respectively, which demonstrates the attention mechanism is critical for modeling the relation between the query and the target. It is worth noticing that "No Attention" in Table 1 has lower inference speed (FPS) than our proposed CAT; we hypothesize that it is mostly due to the difference in channel dimensions: original Faster R-CNN has 1 024 channel dimensions; however, we use 256 for accuracy-efficiency trade-offs (see Table 2).

Table 1. Results of Different Dimensions of Feature Embedding

Method	FPS	Unseen		Seen	
		AP	AP50	AP	AP50
No attention	6.0	8.5	15.0	17.1	28.7
CAT (one stream)	18.4	15.2	25.3	30.3	48.6
CAT (two streams)	16.3	**16.5**	**27.1**	31.3	50.5
Self attention	14.7	16.2	26.5	**32.6**	**52.2**

Note: "Two streams" represents our model that performs both query-to-target and target-to-query attentions, while "one stream" represents a model that executes the query-to-target side. The bold number is the maximum value of the same column.

Table 2. Results of Different Dimensions of Embeddings

d_m	Params ($\times 10^6$)	FPS	Unseen		Seen	
			AP	AP50	AP	AP50
128	12.74	20.8	14.6	26.0	29.0	49.6
256	19.10	16.3	16.5	27.1	31.3	50.5
512	37.67	9.5	**16.6**	**27.3**	32.1	51.6
1 024	110.53	4.9	15.7	25.8	**32.7**	**52.4**

Note: Params represents the number of parameters of the model with different dimension of embeddings.

Then we compare the two-stream architecture with a one-stream transformer that only performs query-to-target attention and generates aggregated target features. The results show that the one-stream model incurs 1.3% and 1.0% AP drops on unseen and seen classes respectively, demonstrating the importance of bidirectional feature aggregation. We also find that our two-stream cross-attention method slightly outperforms self-attention with less memory usage and a faster inference speed.

Number of CAT Layers. We investigate the performance of CAT with different numbers of layers. As shown in Table 3, we test the results of CAT with the number of layers ranging from 3 to 6. The CAT with

Table 3. Ablation Study of CAT on the COCO Split 1

Layer	Unseen		Seen	
	AP	AP50	AP	AP50
CAT (3 layers)	15.8	25.9	31.2	50.0
CAT (4 layers)	**16.5**	**27.1**	31.3	50.5
CAT (5 layers)	16.5	27.1	**32.1**	**51.6**
CAT (6 layers)	16.3	27.1	31.8	51.3

four layers achieves the best performance on unseen classes, while on seen classes the best AP is obtained with the number of layers of 5. It can be found that increasing the number of layers does not always improve performance, which may be caused by the overfitting on seen classes. Note that even using only three layers, our model still outperforms CoAE, demonstrating the superiority of the proposed method. In the remaining experiments, we set the number of layers to 4 by default.

Dimension of Feature Embeddings. Table 2 shows the results with different values of d_m on COCO split 1. We also report their number of parameters and inference speed (FPS). From the results, we can find that reducing d_m to 128 will significantly decrease AP by 2 points on unseen classes. The APs with $d_m = 256$ and $d_m = 512$ are close to each other, but setting d_m to 512 will significantly increase the model size and slow down the inference speed. The results with $d_m = 1\,024$ show an overfitting on seen classes. To strike the balance between the accuracy and speed, we set d_m to 256 in the following experiments.

4.3 Comparison with State-of-the-Arts

MS-COCO. Following Hsieh *et al.*[3], we divide the 80 classes of the COCO dataset[22] into four groups, alternately taking three groups as seen classes and one group as unseen classes. We use the "train 2017" (118 000 images) split for training and minival (5 000 images) split for evaluation. We compare our method with SiamMask[25] and CoAE in Tables 4 and 5. Besides the authors' release of the CoAE model (denoted as CoAE in Tables 4 and 5), we also re-implement this model in the MMDetection framework (denoted as CoAE (Reimp)) for strictly fair comparison. Note that CoAE (Reimp) is trained with the same strategies as our model and achieves better results than the original version, and thus it serves as a strong and fair baseline. Tables 4 and 5 show the comparison on unseen and seen classes, respectively. Compared with the re-implemented CoAE model, our model achieves 1.0% and 0.9% improvements on the average AP and AP50 respectively. As for seen classes, our model also achieves better performance that outperforms CoAE (Reimp) by 0.9% AP point on average.

PASCAL VOC. As for VOC[23], we divide the 20 classes into 16 seen classes and 4 unseen classes, where the choice of seen classes and unseen classes is consistent with that of CoAE. Note that our model is trained on the union set of VOC2007 train&val sets and VOC2012 train and validate sets, while is evaluated on VOC2007. We evaluate the average precision of each category, and calculate the mean average precision (mAP) of seen classes and unseen classes, respectively. Tables 6 and 7 show the comparison with CoAE and other several baselines[26–28], whose evaluation settings are consistent with ours. Our model outperforms the re-implemented CoAE by 1.3% mAP points on unseen classes and performs slightly worse

Table 4. Results (Average Precision) on the Unseen Classes of the COCO Dataset

Method	Split 1		Split 2		Split 3		Split 4		Average	
	AP	AP50	AP	AP50	AP	AP50	AP	AP50	AP	AP50
SiamMask[25]	–	15.3	–	17.6	–	17.4	–	17.0	–	16.8
CoAE	11.8	23.2	12.2	23.7	9.3	20.3	9.4	20.4	10.7	21.9
CoAE (Reimp)	15.1	25.7	15.3	25.4	11.0	21.0	12.5	**21.7**	13.5	23.5
Ours	**16.5**	**27.1**	**16.6**	**26.6**	**12.4**	**22.5**	**12.6**	21.4	**14.5**	**24.4**

Note: We set the results of CoAE as our baseline. For fair comparisons, we re-implement CoAE on our code framework and report its results.

Table 5. Results (Average Precision) on the Seen Classes of the COCO Dataset

Method	Split 1		Split 2		Split 3		Split 4		Average	
	AP	AP50	AP	AP50	AP	AP50	AP	AP50	AP	AP50
SiamMask[25]	–	38.9	–	37.1	–	37.8	–	36.6	–	37.6
CoAE	22.4	42.2	21.3	40.2	21.6	39.9	22.0	41.3	21.8	40.9
CoAE (Reimp)	31.2	**51.3**	27.3	45.3	27.7	45.0	28.8	47.3	28.8	47.2
Ours	**31.3**	50.5	**28.8**	**46.1**	**28.9**	**45.3**	**29.6**	**47.5**	**29.7**	**47.3**

Table 6. Results (Average Precision) on the Seen Classes of the VOC Dataset

Method	Seen Class																mAP
	Plant	Sofa	TV	Car	Bottle	Boat	Chair	Person	Bus	Train	Horse	Bike	Dog	Bird	Mbike	Table	
SiamFC	3.2	22.8	5.0	16.7	0.5	8.1	1.2	4.2	22.2	22.6	35.4	14.2	25.8	11.7	19.7	27.8	15.1
SiamRPN	1.9	15.7	4.5	12.8	1.0	1.1	6.1	8.7	7.9	6.9	17.4	17.8	20.5	7.2	18.5	5.1	9.6
CompNet	28.4	41.5	65.0	66.4	37.1	49.8	16.2	31.7	69.7	73.1	75.6	71.6	61.4	52.3	63.4	39.8	52.7
CoAE	30.0	54.9	64.1	66.7	40.1	54.1	14.7	60.9	77.5	78.3	77.9	73.2	80.5	70.8	**72.4**	**46.2**	60.1
CoAE (Reimp)	**47.3**	61.8	**72.1**	83.0	**56.6**	63.1	40.4	**80.3**	**81.3**	80.6	79.6	77.1	83.2	75.0	69.4	45.5	**68.5**
Ours	44.2	**65.5**	67.1	**83.9**	54.2	**66.8**	**45.6**	79.5	76.8	**82.3**	**81.4**	**78.5**	**84.0**	**76.7**	71.0	33.9	68.2

Note: We compare our model with several previous methods and our baseline model CoAE.

Table 7. Results (Average Precision) on the Unseen Classes of the VOC Dataset

Method	Unseen Class				mAP
	Cow	Sheep	Cat	Aero	
SiamFC	6.8	2.28	31.6	12.4	13.3
SiamRPN	15.9	15.70	21.7	3.5	14.2
CompNet	75.3	60.00	47.9	25.3	52.1
CoAE	83.9	67.10	75.6	46.2	68.2
CoAE (Reimp)	**84.8**	75.60	**83.7**	**57.8**	**75.5**

(0.3% mAP) on seen classes, which presents a stronger generalization ability from seen classes to unseen classes.

FSOD. The FSOD dataset[4] is specifically designed for few-shot object detection. It contains 1 000 categories, with 800 for training and 200 for test. We test the performance of our model and our re-implemented CoAE model on this dataset, with the same one-shot setting. Table 8 shows that our model outperforms CoAE by 1.7% in AP and 2.5% in AP75 on novel classes.

Table 8. Results (Average Precision) on the Unseen Classes of the FSOD Dataset

Method	AP	AP50	AP75
CoAE (Reimp)	40.3	63.8	41.7
Ours	**42.0**	**64.0**	**44.2**

Inference Speed. Note that our model achieves superior accuracy with a much smaller dimension of features ($d_m = 256$) than that of the previous SOTA CoAE (1 024). On the other hand, the dot-product attention adopted by Transformer is more parallelizable and space-efficient. These two characteristics lead to a much faster inference speed: on an NVIDIA RTX-2080Ti GPU, our model achieves 16.3 FPS, while the speed of CoAE is only 5.9 FPS that is nearly 2.5 times slower than ours.

4.4 Visualization of CAT Layers

For intuitively understanding our model, we visualize the intermediate feature maps according to the intensity of response. As shown in Fig.3, the first and the second columns represent the query and the target images respectively, and the remaining columns correspond to the visualization of different CAT layers. Without incorporating any query information, the backbone outputs endow higher responses (higher values on the heatmap) on salient objects or features. With the increase of layers and deeper aggregation of query information, the CAT outputs gradually focus on the objects of the same category as the query. The

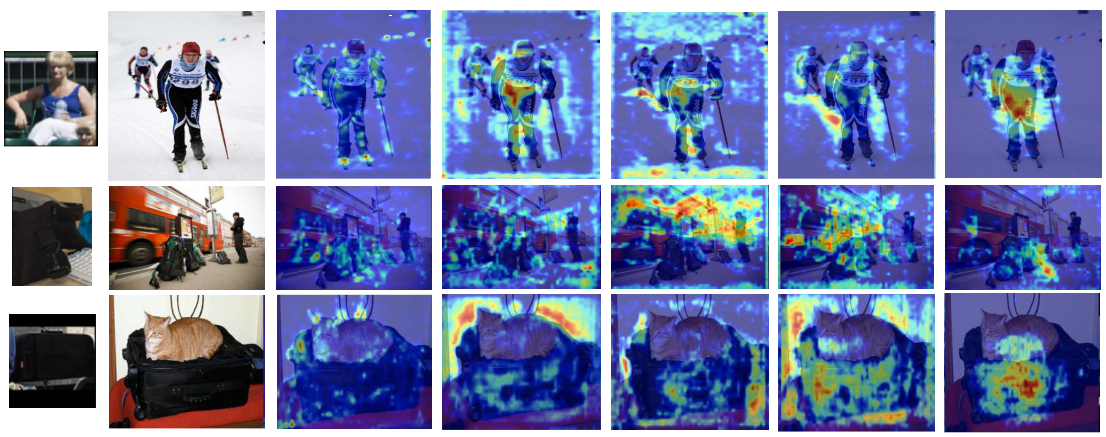


Fig.3. Visualization results of the intermediate feature maps. (a) Query images. (b) Target images. (c) Backbones outputs. (d) Layer-1. (e) Layer-2. (f) Layer-3. (g) Layer-4. We visualize the outputs of each layer in our CAT on several target-query pairs.

visualization demonstrates the importance of our proposed CAT module on exploiting the correspondence between the target and the query.

4.5 Comparison with Few-Shot Detection Methods

We also compare our method with several state-of-the-art few-shot object detection methods on PASCAL VOC and MS-COCO, under the one-shot setting.

Experimental Setup. PASCAL VOC 2007 and VOC 2012 consist of 16.5k training and validation images and 5k test images covering 20 categories. Consistent with the previous few-shot learning setup in TFA[29], we use the VOC 2007 and VOC 2012 train/val sets for training and the VOC 2007 test set for test. Fifteen classes are considered as base classes, and the remaining five classes as novel classes. We report the mean average precision (mAP) with intersection over union (IoU) threshold at 0.5 (AP50). For MS-COCO, we set the 20 PASCAL VOC categories as novel classes and the remaining 60 categories as base classes. Since our method is a one-way one-shot object detection method, we keep the evaluation protocol of few-shot methods consistent with CoAE for a fair comparison, which means we only test classes present in each test image.

Quantitative Results. The results are summarized in Tables 9 and 10. Our method outperforms state-of-the-art methods in most cases for the three different dataset splits of PASCAL VOC, and it achieves the best results on the 20 novel classes of COCO, which demonstrates the effectiveness of our approach. In Table 10, we list the FPS of various methods on COCO, among which our CAT is the fastest. Moreover, our approach does not need to fine-tune on novel classes compared with the previous few-shot methods, which leads to a much faster adaptation speed.

5 Conclusions

In this work, we proposed a Cross-Attention Transformer module (CAT) to deeply exploit bidirectional correspondence between the query and target pairs for one-shot object detection. By combining the proposed CAT module with a two-stage framework, we constructed a simple yet effective one-shot detector. The proposed model achieves state-of-the-art performance on three one-shot detection benchmarks and meanwhile runs 2.5 times faster than CoAE[3], a major strong baseline, demonstrating a superiority over both effectiveness and efficiency.

Table 9. Comparison with Few-Shot Methods on Three PASCAL VOC Novel Split Sets

Model	Backbone	Novel Split 1	Novel Split 2	Novel Split 3	Adaptation Time (s)
Meta R-CNN ICCV2019[30]	R-101	39.9	31.4	35.3	327
FSOD CVPR2020[4]	R-101	53.5	45.2	57.5	407
TFA w/fc ICML2020[29]	R-101-FPN	61.2	44.8	57.2	2 672
TFA w/cos ICML2020[29]	R-101-FPN	58.1	49.2	58.8	2 672
FsDetView ECCV2020[31]	R-101	59.8	47.6	56.4	319
FSCE CVPR2021[32]	R-101-FPN	**69.0**	50.1	59.2	586
MPSR ECCV2020[33]	R-101-FPN	60.5	47.9	58.7	407
Ours	R-101	61.9	**53.2**	**60.8**	0

Note: All results come from official released codes of these methods. For a fair comparison, we modify their evaluation protocols to be the same as CoAE[3]. "R-101" refers to ResNet-101 backbone and "R-101-FPN" refers to ResNet-101 with FPN[34] and adaptation time represents the time the model requires for fine-tuning on novel classes on eight NVIDIA RTX-2080Ti GPUs.

Table 10. Comparison with Few-Shot Methods on COCO

Model	Backbone	Novel AP	Novel AP50	FPS	Adaptation Time (s)
Meta R-CNN ICCV2019[30]	R-101	10.9	20.4	11.4	339
FSOD CVPR2020[4]	R-101	14.7	24.8	10.6	402
TFA w/fc ICML2020[29]	R-101-FPN	15.3	25.2	16.1	2 597
TFA w/cos ICML2020[29]	R-101-FPN	15.5	25.9	16.1	2 597
FsDetView ECCV2020[31]	R-101	16.4	26.8	9.5	311
FSCE CVPR2021[32]	R-101-FPN	15.8	26.5	16.2	902
MPSR ECCV2020[33]	R-101-FPN	14.5	24.2	15.9	643
Ours	R-101	**17.8**	**30.1**	**16.3**	0

Conflict of Interest The authors declare that they have no conflict of interest.

References

[1] Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp.580–587. DOI: 10.1109/CVPR.2014.81.

[2] Ren S Q, He K M, Girshick R, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2017, 39(6): 1137–1149. DOI: 10.1109/TPAMI.2016.2577031.

[3] Hsieh T I, Lo Y C, Chen H T, Liu T L. One-shot object detection with co-attention and co-excitation. In *Proc. the 33rd International Conference on Neural Information Processing Systems*, Dec. 2019, Article No. 245.

[4] Fan Q, Zhuo W, Tang C K, Tai Y W. Few-shot object detection with attention-RPN and multi-relation detector. In *Proc. the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2020, pp.4012–4021. DOI: 10.1109/CVPR42600.2020.00407.

[5] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser L, Polosukhin I. Attention is all you need. In *Proc. the 31st International Conference on Neural Information Processing Systems*, Dec. 2017, pp.6000–6010. DOI: 10.5555/3295222.3295349.

[6] Chen H, Wang Y L, Wang G Y, Qiao Y. LSTD: A low-shot transfer detector for object detection. In *Proc. the 32nd AAAI Conference on Artificial Intelligence*, Feb. 2018, pp.2836–2843. DOI: 10.1609/aaai.v32i1.11716.

[7] Kang B Y, Liu Z, Wang X, Yu F, Feng J S, Darrell T. Few-shot object detection via feature reweighting. In *Proc. the 2019 IEEE/CVF International Conference on Computer Vision*, Oct. 27–Nov. 2, 2019, pp.8419–8428. DOI: 10.1109/ICCV.2019.00851.

[8] Karlinsky L, Shtok J, Harary S, Schwartz E, Aides A, Feris R, Giryes R, Bronstein A M. RepMet: Representative-based metric learning for classification and few-shot object detection. In *Proc. the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.5192–5201. DOI: 10.1109/CVPR.2019.00534.

[9] Osokin A, Sumin D, Lomakin V. OS2D: One-stage one-shot object detection by matching anchor features. In *Proc. the 16th European Conference on Computer Vision*, Aug. 2020, pp.635–652. DOI: 10.1007/978-3-030-58555-6_38.

[10] Tay Y, Dehghani M, Bahri D, Metzler D. Efficient transformers: A survey. *ACM Computing Surveys*, 2023, 55(6): Article No. 109. DOI: 10.1145/3530811.

[11] Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X H, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. the 9th International Conference on Learning Representations*, May 2021.

[12] Touvron H, Cord M, Douze M, Massa F, Sablayrolles A, Jégou H. Training data-efficient image transformers & distillation through attention. In *Proc. the 38th International Conference on Machine Learning*, Jul. 2021, pp.10347–10357.

[13] Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. End-to-end object detection with transformers. In *Proc. the 16th European Conference on Computer Vision*, Aug. 2020, pp.213–229. DOI: 10.1007/978-3-030-58452-8_13.

[14] Zhu X Z, Su W J, Lu L W, Li B, Wang X G, Dai J F. Deformable DETR: Deformable transformers for end-to-end object detection. In *Proc. the 9th International Conference on Learning Representations*, May 2021.

[15] Ye L W, Rochan M, Liu Z, Wang Y. Cross-modal self-attention network for referring image segmentation. In *Proc. the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2019, pp.10494–10503. DOI: 10.1109/CVPR.2019.01075.

[16] Tan H, Bansal M. LXMERT: Learning cross-modality encoder representations from transformers. In *Proc. the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Nov. 2019, pp.5100–5111. DOI: 10.18653/v1/D19-1514.

[17] Su W J, Zhu X Z, Cao Y, Li B, Lu L W, Wei F R, Dai J F. VL-BERT: Pre-training of generic visual-linguistic representations. In *Proc. the 8th International Conference on Learning Representations*, Apr. 2020.

[18] Guo M H, Cai J X, Liu Z N, Mu T J, Martin R R, Hu S M. PCT: Point cloud transformer. *Computational Visual Media*, 2021, 7(2): 187–199. DOI: 10.1007/s41095-021-0229-5.

[19] Yuan L, Chen Y P, Wang T, Yu W H, Shi Y J, Jiang Z H, Tay F E H, Feng J S, Yan S C. Tokens-to-token ViT: Training vision transformers from scratch on ImageNet. In *Proc. the 2021 IEEE/CVF International Conference on Computer Vision*, Oct. 2021, pp.538–547. DOI: 10.1109/ICCV48922.2021.00060.

[20] He K M, Zhang X Y, Ren S Q, Sun J. Deep residual learning for image recognition. In *Proc. the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2016, pp.770–778. DOI: 10.1109/CVPR.2016.90.

[21] Zhang Z M, Warrell J, Torr P H S. Proposal generation for object detection using cascaded ranking SVMs. In *Proc. the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2011, pp.1497–1504. DOI: 10.1109/CVPR.2011.5995411.

[22] 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 *Proc. the 13th European Conference on Computer Vision*, Sept. 2014, pp.740–755. DOI: 10.1007/978-3-319-10602-1_48.

[23] Everingham M, Van Gool L, Williams C K I, Winn J,

Zisserman A. The Pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 2010, 88(2): 303–338. DOI: 10.1007/s11263-009-0275-4.

[24] Chen K, Wang J Q, Pang J M, Cao Y H, Xiong Y, Li X X, Sun S Y, Feng W S, Liu Z W, Xu J R, Zhang Z, Cheng D Z, Zhu C C, Cheng T H, Zhao Q J, Li B Y, Lu X, Zhu R, Wu Y, Dai J F, Wang J D, Shi J P, Ouyang W L, Loy C C, Lin D H. MMDetection: Open MMLab detection toolbox and benchmark. arXiv: 1906.07155, 2019. https://arxiv.org/abs/1906.07155, March 2024.

[25] Michaelis C, Ustyuzhaninov I, Bethge M, Ecker A S. One-shot instance segmentation. arXiv: 1811.11507, 2018. https://arxiv.org/abs/1811.11507, March 2024.

[26] Fu K, Zhang T F, Zhang Y, Sun X. OSCD: A one-shot conditional object detection framework. *Neurocomputing*, 2021, 425: 243–255. DOI: 10.1016/j.neucom.2020.04.092.

[27] Cen M B, Jung C. Fully convolutional Siamese fusion networks for object tracking. In *Proc. the 25th IEEE International Conference on Image Processing*, Oct. 2018, pp.3718–3722. DOI: 10.1109/ICIP.2018.8451102.

[28] Li B, Yan J J, Wu W, Zhu Z, Hu X L. High performance visual tracking with Siamese region proposal network. In *Proc. the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Jun. 2018, pp.8971–8980. DOI: 10.1109/CVPR.2018.00935.

[29] Wang X, Huang T E, Darrell T, Gonzalez J E, Yu F. Frustratingly simple few-shot object detection. In *Proc. the 37th International Conference on Machine Learning*, Jul. 2020, Article No. 920.

[30] Wu X W, Sahoo D, Hoi S. Meta-RCNN: Meta learning for few-shot object detection. In *Proc. the 28th ACM International Conference on Multimedia*, Oct. 2020, pp.1679–1687. DOI: 10.1145/3394171.3413832.

[31] Xiao Y, Marlet R. Few-shot object detection and viewpoint estimation for objects in the wild. In *Proc. the 16th European Conference on Computer Vision*, Aug. 2020, pp.192–210. DOI: 10.1007/978-3-030-58520-4_12.

[32] Sun B, Li B H, Cai S C, Yuan Y, Zhang C. FSCE: Few-shot object detection via contrastive proposal encoding. In *Proc. the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2021, pp.7348–7358. DOI: 10.1109/CVPR46437.2021.00727.

[33] Wu J X, Liu S T, Huang D, Wang Y H. Multi-scale positive sample refinement for few-shot object detection. In *Proc. the 16th European Conference on Computer Vision*, August 2020, pp.456–472. DOI: 10.1007/978-3-030-58517-4_27.

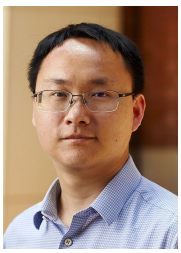
[34] Lin T Y, Dollár P, Girshick R, He K M, Hariharan B, Belongie S. Feature pyramid networks for object detection. In *Proc. the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2017, pp.936–944. DOI: 10.1109/CVPR.2017.106.

Wei-Dong Lin received his B.S. degree in hydroacoustic engineering from Northwestern Polytechnical University, Xi'an, in 2019. He is now a Master student in computer science and technology, Northwestern Polytechnical University, Xi'an. His current research interests mainly focus on computer vision and object detection.

Yu-Yan Deng received his B.S. degree in computer science and technology from Xidian University, Xi'an, in 2019. He is now a Master student in computer science and technology, Northwestern Polytechnical University, Xi'an. His current research interests mainly focus on computer vision and obejct detection.

Yang Gao received his B.S. degree in computer science and technology from Northwestern Polytechnical University, Xi'an, in 2019. He is now a Master student in computer science and technology, Northwestern Polytechnical University, Xi'an. His current research interests mainly focus on computer vision and auto machine learning.

Ning Wang received his B.S. degree in computer science and technology from Northwestern Polytechnical University, Xi'an, in 2019. He is now a Master student in computer science and technology, Northwestern Polytechnical University, Xi'an. His current research interests mainly focus on computer vision and neural architecture search.



Ling-Qiao Liu received his B.S. and M.S. degrees in communication engineering from the University of Electronic Science and Technology of China, Chengdu, in 2006 and 2009, respectively, and his Ph.D. degree in computer science from the Australian National University, Canberra, in 2014. In 2016, he was awarded the Discovery Early Career Researcher Award from the Australian Research Council and the University Research Fellow from the University of Adelaide, Adelaide. He is now a senior lecturer at the University of Adelaide and the Australian Institute for Machine Learning, Adelaide. His current research interests include low-supervision learning and various topics in computer vision and natural language processing.

Lei Zhang received his Ph.D. degree in computer science and technology from Northwestern Polytechnical University, Xi'an, in 2018. He was a research staff in the School of Computer Science, the University of Adelaide, Adelaide, between 2017 and 2019. He was a research scientist in the Inception Institute of Artificial Intelligence (IIAI), Abu Dhabi, United Arab Emirates between 2019 and 2020. He is currently a professor with the School of Computer Science, Northwestern Polytechnical University, Xi'an. His research interests include image processing, machine learning and video analysis.

Peng Wang received his B.S. degree in electrical engineering and automation from Beihang University, Beijing, in 2004, and his Ph.D. degree in control science and engineering from Beihang University, Beijing, in 2011. He is now a professor at School of Computer Science, Nothwestern Polytechnical University, Xi'an. He was with School of Computer Science, the University of Adelaide, Adelaide, for about four years. His research interests include computer vision, machine learning and artificial intelligence.