Any Detector Can Detect Anything

Thomas E. Huang[1*] Siyuan Li[1*] Martin Danelljan[1] Henghui Ding[1]
Luc Van Gool[1,2] Fisher Yu[1]
[1] Computer Vision Lab, ETH Zürich [2] INSAIT, Sofia University "St. Kliment Ohridski"

Abstract

Visual prompt-based detection enables generalization to arbitrary novel instances in the target image by using one or a few visual templates. Previous methods rely on complex relation or explicit feature matching modules, and their designs are deeply coupled with specific detectors, greatly limiting their applicability. Instead, we propose our 'Any Detector can Detect Anything' framework that can enable any detector to detect any object given a single or a few visual templates. Specifically, we design an adapter called Template-Aware Adapter that can be added on top of any existing detector architecture to inject visual template information directly into the detection features. After integration, localization is done on the feature maps as in standard object detectors, effectively transforming any detector into a visual prompt-based detector. Furthermore, we revisit current visual prompt detection benchmarks and correct their unrealistic test assumptions and class splits, which limit the usability of the developed algorithms in the real world. We introduce a set of realistic benchmarks to remedy these issues. We comprehensively evaluate the proposed model on both existing and our new benchmarks, outperforming current state-of-the-art one-shot and few-shot detection methods by a large margin.

1. Introduction

Humans have the ability to generalize knowledge from a few or even a single example and apply them to any situation. Building machines with the same ability is critical for achieving more general-purpose artificial intelligence. Modern object detection systems are able to achieve impressive results [6, 46, 49], but they require copious amounts of annotated training data to be able to detect objects from a closed set of classes, and new classes that are not listed in the training set cannot be detected. We investigate a different detection paradigm that we call *visual prompt*-based detection, where the detector is given visual examples that can be used to detect new classes without re-training. The purpose is to enable the detection of any object in the real world, without requiring a predetermined set of class labels. Visual prompt-based detection encompasses existing tasks in low-shot or exemplar-based object detection, including One-Shot Object Detection (OSOD) and Few-Shot Object Detection (FSOD). Compared with language prompt-based detection [12], visual examples can provide more fine-grained descriptions of target objects, as a picture is worth a thousand words.

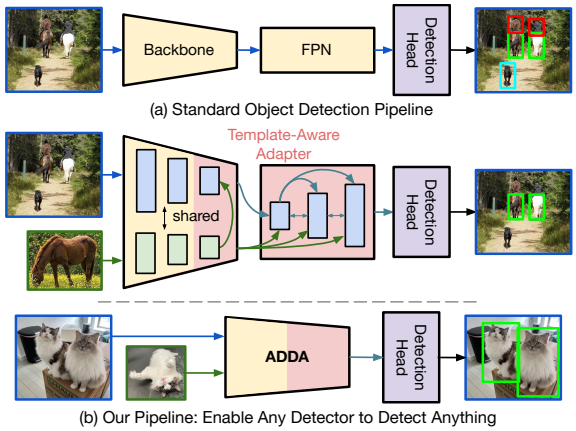

Figure 1. We follow (a) the standard detection pipeline by adding (b) our Template-Aware Adapter into the feature extractor. Our Any Detector can Detect Anything (ADDA) framework can enable any detector to detect any object when given visual templates.

The key challenge in visual prompt-based detection is how to effectively extract generalizable information from the given *visual templates* to detect novel objects in the *target image*. Most existing methods frame this task as a template-matching problem. Typically, region proposal features representing object regions are first extracted by a deep network (*e.g.*, Region Proposal Network [33]). Hand-crafted feature matching modules [5, 15, 45] are then employed to match each region feature with the template features, followed by a region-based detection head [33] to output corresponding object bounding boxes. Despite the recent advancements in OSOD and FSOD, most methods still rely on complex detector-specific designs (*e.g.*, proposal-based [5, 15, 45], DETR-based [8, 48]) and suboptimal matching modules that

*Equal contribution.

cannot be applied to a different detector.

We approach visual prompt-based detection from a unified perspective, adopting the standard object detection pipeline (Figure 1). Instead of complex correlation/matching modules, we employ our Template-Aware Adapter (TAA) within the feature extractor to model the interactions between the visual templates and the target image. TAA is composed of Feature Extractor and Integration Blocks (FEIB), which utilizes the power of self-attention and template-target cross-attention to automatically learn the important feature interactions between image and template. We embed FEIBs directly within the backbone and the multi-scale feature extractor. After integration, localization is done on the feature maps as in standard object detectors. Such a design enables our model to work with any detection architecture, training strategy, etc. Thus, our approach for visual prompt-based detection is perpendicular to advances in standard object detection, making our method much more applicable and scalable.

We further notice that current benchmarks make unrealistic assumptions that limits developed algorithms from operation in the real world. First, during the evaluation of OSOD methods, oracle knowledge is typically used by the detector, such as which classes are present in each image. This implicitly incentivizes detectors to generate many false positive predictions when the template class is not present in the image, as shown in Figure 2. However, such scenarios are fairly common in the real world. Second, current benchmarks contain limited class diversity, which makes them unsuitable for fully evaluating the performance of visual prompt-based detectors. To correct these assumptions, we introduce a set of Realistic One-Shot Object Detection (R-OSOD) benchmarks, which randomly inserts negative image-template pairs during evaluation and additionally utilize large-scale datasets [13, 20] for more class diversity.

We summarize the main contributions as follows:

- We propose ADDA, a unified visual prompt-based detection framework that utilizes the Template-Aware Adapter within the feature extractor to enable the detection of any object for any detector.
- We carefully examine existing benchmarks, identify their weaknesses, and propose more realistic evaluation benchmarks to better evaluate detectors' generalizability and performance in the real world.
- We comprehensively evaluate ADDA on both existing OSOD and FSOD benchmarks and our new benchmarks, outperforming current state-of-the-art by a significant margin across all settings.

2. Related Work

Object detection. Object detection involves classifying and localization every object in the image under a predefined class label set. There are a few common approaches to this task. Two-stage detectors [3, 17, 33] generate region proposals and then classify them, while one-stage detectors [24, 27, 36, 50] directly predict object classes and bounding boxes, optimizing for speed. DETR [4] and its variants [23, 28, 49], leveraging the Transformer architecture, further simplifies the detection process by eliminating components like non-maximum suppression.

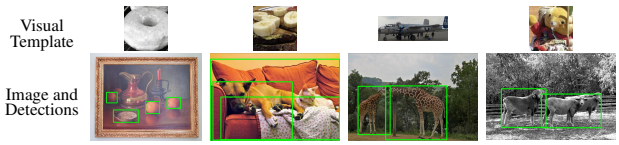

Figure 2. Current state-of-the-art methods [45, 52] fail to discriminate negative templates and output many false positive predictions.

Few-shot object detection. Few-shot object detection (FSOD) aims to train a detector with limited training examples. Classes are split into base classes, which have abundant data, and novel classes, which only have a few examples. FSOD approaches mainly utilize either transfer learning or meta-learning. Transfer learning methods [32, 39, 42, 43, 47, 51, 55] follow a two-stage training protocol, where the detector is first trained on the base classes and then fine-tuned on novel classes. These methods often suffer from catastrophic forgetting of the base classes and require retraining to detect new classes. Meta-learning methods [11, 16, 19, 44, 48] follow standard few-shot learning [35, 38] and utilize meta-level knowledge for adaptation to new classes. These methods can generalize to novel classes without additional training, though most still fine-tune on the novel set to pursue higher performance.

One-shot object detection. One-shot object detection (OSOD) is closely related to FSOD, with the restriction of only a single example. However, OSOD only uses base data for training and evaluates on novel classes without further fine-tuning. In this way, it evaluates pure generalization performance on novel data. Most methods [5, 8, 18, 30, 45, 54] frame the problem as a template-matching problem and compare region proposals with template features to find matches. CoAE [18] utilizes a co-attention and co-excitation framework to reweight object proposals before matching. BHRL [45] uses an instance-level hierarchical relation module to better extract template-relevant features through modeling different relations. Instead, we design a unified pipeline that forgoes complex relation or explicit feature matching modules. BSPG [52] proposes a base-class suppression module to remove bias towards base classes and a prior guidance module to guide detection with feature correlation.

3. Method

In this section, we introduce our 'Any Detector can Detect Anything' (ADDA) framework, which enables any detector to detect any object.

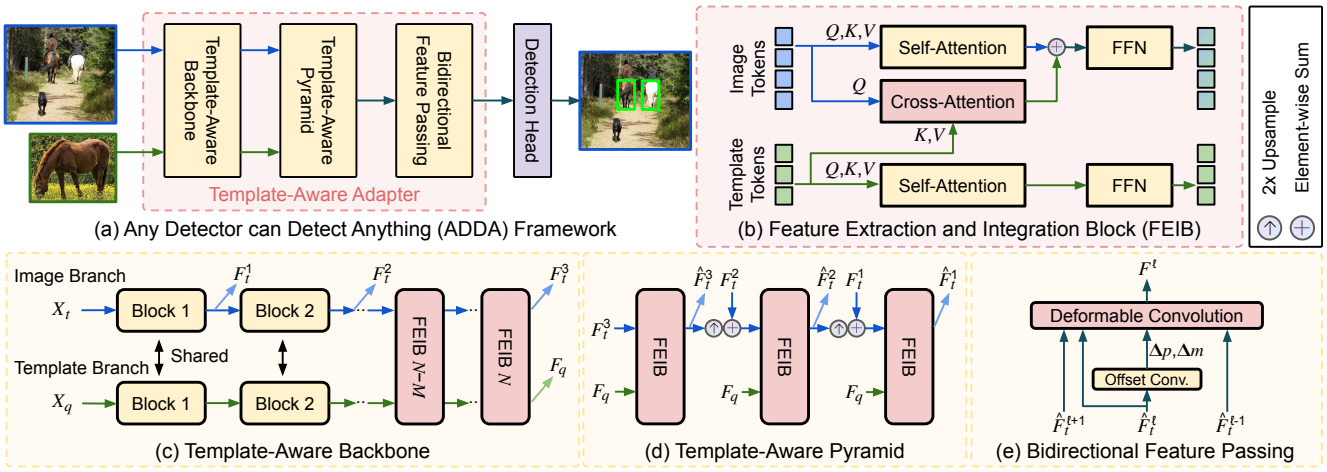

Figure 3. The overall pipeline of our Any Detector can Detect Anything (ADDA) framework. (a) ADDA consists of a Template-Aware Adapter and a standard detection head. We use (b) cross-attention modules to perform feature integration between image and template. FEIBs are inserted into (c) the last stage of the backbone and (d) the feature pyramid to directly inject template information into the image features. We also consider (e) a feature passing scheme to propagate template-aware features across scales.

3.1. Problem Setting

We investigate the problem of visual prompt-based detection, *i.e.*, detecting any object given a single or a few visual examples. At training, we are given a detection dataset sufficiently annotated with a set of *base classes* C_{B}. At test-time, the trained detector is primarily evaluated on a set of *novel classes* C_{N}, which do not overlap with those seen during training ($C_{\mathrm{B}} \cap C_{\mathrm{N}} = \varnothing$). During inference, templates of novel classes are provided to enable generalization to arbitrary new classes without the need for re-training.

Visual prompt-based detection is related to two main tasks: One-Shot Object Detection (OSOD) and Few-Shot Object Detection (FSOD). OSOD focuses on generalization to novel classes and does not use data from novel classes at all during training. It treats each class independently and does not assume a predefined class list is given. Furthermore, OSOD is characterized by its restriction to a single visual example for detection, and OSOD methods generally lack support for multiple examples by default. In contrast, FSOD primarily focuses on swiftly adapting to a predetermined set of novel classes and is not limited by the number of visual examples. Consequently, FSOD methods usually include an additional fine-tuning stage on novel classes to further enhance performance on the predefined classes. As we are focusing on detecting any object, our architecture and evaluation methodology predominantly aligns with the principles of OSOD, though our method supports multiple templates.

3.2. Unified Architecture

We design our unified architecture following the standard object detection pipeline (Figure 1), which typically consists of a backbone network, multi-scale feature extractor, and a detection head. To inject template information into the existing image features for visual prompt-based detection, we design an adapter, termed Template-Aware Adapter (TAA), that conducts feature integration between the image and the template directly within the feature extractor.

The building blocks of TAA are feature integration modules that harness the power of self-attention and template-target cross-attention to enable automatic learning of feature interactions, which significantly simplifies the framework and improves performance (section 3.3). We embed such modules in the final blocks of the backbone and in the multi-scale feature extractor (section 3.4). After feature extraction and integration, a detection head can be directly added on top to retrieve object bounding boxes corresponding to the class of the input template. Thus, ADDA can operate with any detector, whether one-stage, two-stage, or query-based. An overview of our framework is depicted in Figure 3 (a).

3.3. Feature Extraction and Integration

We first define the building block of our adapter, Feature Extraction and Integration Block (FEIB), that enables the modeling of feature interactions between template and target along with feature extraction, depicted in Figure 3 (b).

Given a target image X_t and visual template X_q, we first compute image and template tokens by using a shared convolutional tokenizer and flattening the resulting feature maps into two sets of discretized tokens, denoted by $z_t^l \in \mathbb{R}^{HW \times C}$ for image tokens and $z_q^l \in \mathbb{R}^{nhw \times C}$ for template tokens. We use n to indicate the number of visual templates and l to indicate the l-th block. To seamlessly handle multiple visual templates at the same time, we concatenate and tokenize the templates together, increasing the number of template tokens by n times.

In each block, we apply three projection layers to both the image and template tokens to generate query, key, and

value tokens q_t, k_t, v_t and q_q, k_q, v_q (l omitted for clarity). Then, we compute the Multi-Head Attention (MHA) [37],

$$\begin{aligned} k_c = \text{Concat}(k_t, k_q), v_c = \text{Concat}(v_t, v_q), \\ \text{Attention}_t(z_t) = \text{MHA}\left(k_c, q_t, v_c\right), \\ \text{Attention}_q(z_q) = \text{MHA}\left(k_q, q_q, v_q\right), \end{aligned} \tag{1}$$

where Attention_t and Attention_q are the attention operations for the image and template, respectively. Only self-attention is used for the template, whereas both self- and cross-attention are considered for the image. This ensures the template features are intact during the integration process, providing better guidance for finding target objects and enhancing generalization for novel objects. MHA can be any variant of Multi-Head Attention, such as convolutional based attention used in CvT [41] or shifted window based attention used in Swin Transformer [29].

Afterwards, feed-forward networks (FFN) are applied to obtain the output image and template tokens for block l,

$$\begin{aligned} \hat{z}_t^l = \text{Attention}_t(\text{LN}(z_t^l)) + z_t^l, \\ z_t^{l+1} = \text{FFN}(\text{LN}(\hat{z}_t^l)) + \hat{z}_t^l, \end{aligned} \tag{2}$$

where LN denotes layer normalization [1]. Template tokens also follow Equation 2 except for changing the attention operation to Attention_q defined in Equation 1. To obtain image feature maps, we reshape the resulting image and template tokens back to 2D $F_t^{l+1} \in \mathbb{R}^{\frac{H}{s} \times \frac{W}{s} \times C'}$ and $F_q^{l+1} \in \mathbb{R}^{n \times \frac{h}{s} \times \frac{w}{s} \times C'}$, where s denotes the stride and C' denotes output channels.

3.4. Template-Aware Adapter

To adapt target image features to support visual prompting, we build TAA by inserting FEIBs directly within the backbone and the multi-scale feature extractor to generate template-aware features for detection.

Template-Aware Backbone. We first incorporate FEIBs within the backbone, depicted in Figure 3 (c). Standard Transformer encoders [29, 41] utilize a sequence of attention blocks to extract features separately for image and template. We instead replace these blocks with FEIBs to perform joint feature extraction and integration of image and template, inspired by [7]. By doing so, we can re-use the existing pre-trained weights [9], which serves as a good initialization for measuring the feature interactions between image and template. FEIB does not introduce any additional learnable parameters over the original Transformer block.

To save computation, we only replace the final M blocks with FEIBs ($M = 4$ in our experiments). By dropping the low-level high resolution features during integration, we save memory usage significantly, which enable our method to support multiple visual templates and high resolution inputs; these are crucial for object detection accuracy.

Template-Aware Pyramid. Solely using the backbone features for object detection yields inferior performance, as objects exist in a wide range of different scales. The Feature Pyramid Network (FPN) [26] is typically used to incorporate high-level semantically-rich features into lower-level features for detection. FPN uses top-down aggregation operations to fuse successive feature maps, starting from the top-most level. However, FPN merely propagates high-level features, but low-level feature integration that is crucial for discovering small-scale objects relevant to the template is still missing.

To remedy this, we build a Template-Aware Pyramid (TAP) by incorporating template features into the FPN and replacing the existing convolutional layers with FEIBs to add further feature integration between image and template at earlier levels,

$$\begin{aligned} \tilde{F}_t^l = \text{Conv}_{1\times1}\left(F_t^l\right) + \text{Up}\left(\hat{F}_t^{l+1}\right), \\ \hat{F}_t^l = \text{FEIB}\left(\tilde{F}_t^l, \tilde{F}_q\right), \end{aligned} \tag{3}$$

where $\text{Conv}_{k\times k}$ denotes convolution with kernel size of k and Up denotes upsampling by a factor of 2. This process is depicted in Figure 3 (d). This structure reintroduces missing low-level feature integrations into the FPN, enabling better modeling of feature interactions across different scales.

Bidirectional Feature Passing. A simple FPN is not sufficient for obtaining expressive multi-scale features, as there are limited vertical connections between different feature levels. This restricts the template-aware information that can be passed across scales, limiting performance. To enhance cross-scale feature aggregation, we introduce the Bidirectional Feature Passing (BFP) module. For each feature level l, BFP aggregates its features F_t^l with its neighbors F_t^{l-1} and F_t^{l+1} at each spatial location by summing the features in the spatial neighborhood. Instead of using static operations to determine which features locations to aggregate for each location, we use a convolutional layer to generate dynamic offsets and modulation factors that enables more expressive aggregation based on the perceived importance of each location.

In implementation, this is akin to the modulated deformable convolution [56] operation, adapted to operate across feature levels:

$$\begin{aligned} \{\Delta p^j, \Delta m^j\} = \text{Conv}_{3\times3}(\hat{F}_t^j + \text{Up}(\hat{F}_q^j)), \\ F^l(p) = \sum_{j=l-1}^{l+1} \sum_{k=1}^{K} w_k \cdot \hat{F}_t^j(p + p_k + \Delta p_k^j) \cdot \Delta m_k^j, \end{aligned} \tag{4}$$

where K is the number of sampling locations for a convolutional kernel, w_k and p_k are the weight and predefined offset for the k-th location, Δp_k^j and Δm_k^j are the learnable offset and modulation factor, and bilinear interpolation is

Table 1. One-shot detection performance comparison with state-of-the-art methods on the PASCAL VOC 2007 test set using AP_{50}. **Black** / *blue* indicate best / second best. ADDA outperforms other methods significantly across both base and novel categories.

Method	Base																	Novel				
	Plant	Sofa	TV	Car	Bottle	Boat	Chair	Person	Bus	Train	Horse	Bike	Dog	Bird	Mbike	Table	Avg.	Cow	Sheep	Cat	Aero	Avg.
CoAE [18]	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	83.9	67.1	75.6	46.2	68.2
UP-DETR [8]	46.7	61.2	75.7	81.5	54.8	57.0	44.5	80.7	74.5	86.8	79.1	80.3	80.6	72.0	70.9	57.8	69.0	80.9	71.0	80.4	59.9	73.1
AIT [5]	47.7	62.7	71.9	76.1	51.8	63.5	31.5	70.3	84.0	87.2	81.2	80.8	84.5	72.2	78.7	*62.8*	69.2	86.6	74.3	83.7	47.7	73.1
BHRL [45]	57.5	49.4	76.8	80.4	61.2	58.4	48.1	83.3	74.3	87.3	80.1	81.0	87.2	73.0	78.8	38.8	69.7	81.0	67.9	*86.9*	59.3	73.8
BSPG [52]	55.0	55.6	78.3	81.4	62.5	59.5	50.9	81.7	74.7	87.5	82.1	81.0	85.2	73.9	79.1	39.5	70.5	80.6	67.4	84.6	*61.4*	73.5
SaFT [54]	*59.7*	**81.3**	*82.4*	*86.9*	**73.0**	*72.0*	**62.3**	*83.7*	*85.9*	*88.1*	*86.7*	*87.7*	*87.7*	*83.5*	**86.1**	**75.1**	*80.1*	*88.1*	*77.0*	84.3	48.5	*74.5*
ADDA	**60.7**	*79.3*	**86.1**	**91.9**	*72.5*	**81.1**	*62.1*	**88.0**	**88.0**	**90.6**	**92.2**	**92.3**	**94.7**	**89.1**	*83.9*	52.0	**81.6**	**93.4**	**86.6**	**92.0**	**61.5**	**83.4**

Table 2. Comparison with state-of-the-art methods on the COCO 2017 validation set using AP_{50}. **Black** / *blue* indicate best / second best. † indicates using ImageNet-1K pre-trained weights, and ‡ indicates using DINOv2 [31] pre-trained weights.

Method	Base					Novel				
	S1	S2	S3	S4	Avg.	S1	S2	S3	S4	Avg.
Siam Mask [30]	38.9	37.1	37.8	36.6	37.6	15.3	17.6	17.4	17.0	16.8
CoAE [18]	42.2	40.2	39.9	41.3	40.9	23.4	23.6	20.5	20.4	22.0
AIT [5]	50.1	47.2	45.8	46.9	47.5	26.0	26.4	22.3	22.6	24.3
SaFT [54] †	49.2	47.2	47.9	49.0	48.3	27.8	27.6	21.0	23.0	24.9
BHRL [45]	56.0	52.1	52.6	53.4	53.5	26.1	29.0	22.7	24.5	25.6
BHRL [45] †	57.1	52.8	53.4	54.1	54.4	26.6	28.7	21.9	25.0	25.6
BSPG [52]	57.1	54.1	54.0	54.6	55.0	*27.7*	30.7	24.6	*26.3*	27.3
DE-ViT [53] ‡	*59.4*	*57.0*	**61.3**	**60.7**	*59.6*	27.4	**33.2**	*27.1*	26.1	*28.4*
ADDA †	**61.2**	**58.3**	*59.4*	*60.2*	**59.8**	**31.9**	*31.0*	**28.4**	**28.8**	**30.0**

applied to obtain $P^j(p + p_k + \Delta p_k^j)$. In our experiments, we use $K = 9$ and $p_k \in \{(-1,-1),(-1,0),...,(1,1)\}$, which represents a 3×3 convolutional kernel with dilation factor of 1. Our BFP module can be applied iteratively to continuously refine each feature map, and we use six iterations in our experiments.

3.5. Detection Head

On top of the template-aware features from TAA, ADDA can utilize any object detection head. Unlike standard detection, ADDA treats classification as a binary classification problem instead (*i.e.*, foreground vs. background, equivalent to setting the number of classes to 1), as it only needs to retrieve object bounding boxes with the same class as the input templates. Besides this, we do not make any other modifications, and we use the exact same training losses, optimizer, and hyperparameters as the original detector.

4. Realistic Benchmarks

Our main goal is to develop detectors that are capable of detecting any object in the real world. While our evaluation protocol aligns well with that of OSOD, we revisit existing benchmarks and find key issues that prevent a complete understanding of the behaviors of developed algorithms. We detail these issues below and introduce new Realistic One-Shot Object Detection (R-OSOD) benchmarks to better assess detectors' ability to handle novel classes in the wild.

Oracle knowledge during evaluation. Existing benchmarks are only evaluated over classes that exist within each test image. Visual templates of classes that do not exist in the test image (*i.e.*, negative templates) are not used, thus avoiding the false positive predictions associated with such classes. However, as this protocol do not punish these false positive predictions, current OSOD methods produce confidently wrong detections when given negative templates. Visualizing detections of state-of-the-art methods [45] show that, when given negative templates, many incorrect detections that localize foreground objects are produced. This is problematic, as in the wild it is typically unknown whether objects of the template class exist in each test image.

We aim to remove access to such oracle knowledge during testing. In addition to existing class templates, we randomly sample η negative templates during evaluation. Detectors are asked to retrieve objects with each negative template, and the detections will be treated as false positive predictions for the template class, which will be punished by the standard average precision (AP) metric. The goal is to avoid predictions for each negative template. We determine η based on the average number of object classes in each dataset to balance the number of positive and negative templates. We term the AP metric with negative templates as $AP^{\eta N}$.

Furthermore, we introduce negative templates during training by simply sampling them with a probability p_{neg} at each training iteration. The training targets in this case will be empty, and thus the detector will be optimized to not output any detections when given negative templates. With negative sampling, the amount of false positives produced by the detector is greatly reduced, improving the detector's ability to handle negative templates (Figure 4).

Limited class diversity. Existing benchmarks use datasets (PASCAL VOC [10] and COCO [25]) that contain a low number of classes (20 and 80). This results in two issues: First, some frequently appearing classes (*e.g.*, person) are used as novel classes for evaluation, which negatively affects performance as they are still present in the training images. We found accuracy on frequently appearing novel classes to be close to zero, as the detector learns to treat them as background. Second, it is difficult to estimate performance in the wild without sufficient class diversity.

We remedy these issues by constructing new benchmarks based on the LVISv1 dataset [13] and the OpenImages dataset [20]. The amount of annotations for each class in both datasets reflects actual rarity in the real world, which can better evaluate a detector's performance in the wild. For LVISv1, we treat frequent and common classes as the base classes and rare classes as the novel classes to evaluate the performance of existing algorithms under higher class diver-

Table 3. Few-shot detection performance on the COCO dataset. Methods that perform fine-tuning on the novel set are colored in gray and are provided for reference. All evaluation scores are averaged over 10 random seeds.

Method	1-Shot			2-Shot			3-Shot			5-Shot			10-Shot		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Novel Fine-Tuning															
TFA w/fc [39]	2.9	5.7	2.8	4.3	8.5	4.1	6.7	12.6	6.6	8.4	16.0	8.4	10.0	19.2	9.2
FCT [16]	5.1	-	-	7.2	-	-	9.8	-	-	12.0	-	-	15.3	-	-
Meta-DETR [48]	7.5	12.5	7.7	-	-	-	13.5	21.7	14.0	15.4	25.0	15.8	19.0	30.5	19.7
DeFRCN [32]	9.3	-	-	12.9	-	-	14.8	-	-	16.1	-	-	18.5	-	-
DeFRCN+TSF [21]	9.9	-	-	13.5	-	-	14.8	-	-	16.3	-	-	18.3	-	-
SNIDA [40]	12.0	-	-	15.4	-	-	16.4	-	-	17.8	-	-	20.7		
No Fine-Tuning															
Fan *et al.* [11]	4.0	8.5	3.5	5.4	11.6	4.6	5.9	12.5	5.0	6.9	14.3	6.0	7.6	15.4	6.8
QA-FewDet [14]	5.1	10.5	4.5	7.8	16.4	6.6	8.6	17.7	7.5	9.5	19.3	8.5	10.2	20.4	9.0
Meta FRCNN [15]	5.0	10.2	4.6	7.0	13.5	6.4	-	-	-	-	-	-	9.7	18.5	9.0
AirDet [22]	6.0	10.5	6.0	6.6	12.0	6.3	7.0	13.0	6.7	7.8	14.3	7.3	8.7	15.3	8.8
FS-DETR [2]	7.0	13.6	7.5	8.9	17.5	9.0	10.0	18.8	10.0	10.9	20.7	10.8	11.3	21.7	11.1
ADDA	**11.1**	**16.2**	**11.8**	**14.8**	**22.1**	**15.6**	**16.6**	**24.6**	**17.5**	**17.7**	**26.6**	**18.7**	**18.7**	**28.1**	**19.6**

sity. For OpenImages, we construct a zero-shot cross-domain benchmark by evaluating the COCO-trained models directly on OpenImages to further evaluate the generalizability of existing algorithms. We filter the OpenImages classes to ensure there are no overlaps with COCO. Additional details are provided in the supplementary material.

5. Experiments

We conduct extensive results on both existing OSOD and FSOD benchmarks in section 5.1 and our R-OSOD benchmarks in section 5.2. We also provide ablation studies and visualizations in section 5.3.

Implementation details. We use Swin-T [29] and CvT-13 [41] as the backbones pre-trained on ImageNet-1K [9]. We use a resolution of 640×640 for CvT-13 and 1024×1024 for Swin-T. Our models are trained for 24 epochs on VOC and LVISv1 and 12 epochs on COCO with a batch size of 16. We use a negative sampling probability of $p_{\text{neg}} = 0.4$ on COCO and $p_{\text{neg}} = 0.2$ on LVISv1. We provide additional details in the supplementary material.

5.1. Existing Benchmarks

To compare with previous methods, we first follow the standard experimental setting from prior work in OSOD [18, 30, 45, 54] and FSOD [19, 39] using PASCAL VOC [10] and COCO [25] datasets. We use the CvT-13 [41] backbone and the FCOS [36] detector for ADDA for all experiments.

OSOD evaluation setting. VOC uses 16 base classes and 4 novel classes. On COCO, there are four different splits (S1, S2, S3, and S4), each with 60 base classes and 20 novel classes. We use standard AP_{50} as the metric for evaluation for both base and novel classes. For these benchmarks, we do not use negative sampling as negative templates are not used during evaluation. We follow previous work [5, 18, 45] to generate the image-template pairs during evaluation to ensure consistent comparison. The final score is the average over five sampled templates.

OSOD results. In Table 1, we compare ADDA with previous state-of-the-art OSOD methods on the PASCAL VOC test set. ADDA improves over the previous state-of-the-art SaFT [54] by almost 10 points on average for novel class AP_{50} and 1.8 points on average for base class AP_{50}.

We perform the same comparison on the COCO validation set in Table 2. Despite the increase in difficulty, ADDA achieves consistent improvements over previous OSOD methods across all splits and over both base and novel classes, setting a new state-of-the-art on this dataset. In particular, ADDA outperforms the previous state-of-the-art BHRL [45] by 5.4 points on average on base class AP_{50} and 4.4 points on average on novel class AP_{50}. DE-ViT [53] uses DINOv2 [31] pre-trained weights with the much larger ViT-L backbone, so the comparison is not fair; despite this, ADDA still outperforms their method by over 1.6 points in average novel AP_{50}.

FSOD evaluation setting. We follow previous work [19, 39] for evaluation on the COCO dataset. Evaluation is conducted over a different number of shots, including 1, 2, 3, 5, and 10. We use standard AP as the metric for evaluation and average the scores over 10 random seeds (consistent with [39]). We mainly compare with methods that do not require re-training on the novel set, but we provide methods that do novel fine-tuning as reference.

FSOD results. The results on the COCO dataset are shown in Table 3. Across all shots, ADDA obtains the best performance when compared with previous state-of-the-art methods. ADDA shows a significant improvement in AP across all shots, marking a substantial advancement over the previous best methods. Moreover, in lower shots (*i.e.*, 1, 2, 3, 5), ADDA not only outperforms methods without novel fine-tuning but also those with novel fine-tuning. For instance,

Table 4. Comparison on the R-OSOD COCO benchmark using the COCO 2017 validation set. FRCNN denotes Faster R-CNN. ADDA achieves significantly better performance across all splits, both base and novel classes, and different types of detectors.

Method	Detector	Type	Backbone	Base AP_{50}^{5N}					Novel AP_{50}^{5N}				
				S1	S2	S3	S4	*Avg.*	S1	S2	S3	S4	*Avg.*
BHRL [45]	FRCNN [33]	Two-Stage	ResNet-50	45.5	39.9	40.0	41.1	41.6	13.9	17.3	11.1	14.0	14.1
			Swin-T	53.9	50.2	50.6	50.2	51.2	11.6	13.8	8.1	10.5	11.0
			CvT-13	41.9	35.8	36.1	37.4	37.8	14.8	15.9	10.9	13.2	13.7
ADDA	FCOS [36]	One-Stage	Swin-T	**55.5**	51.9	52.0	53.1	53.1	13.5	14.6	11.6	11.9	12.9
	FCOS [36]	One-Stage	CvT-13	52.5	47.9	48.5	49.6	49.6	17.7	**18.8**	**16.3**	17.1	**17.5**
	GFL [24]	One-Stage	CvT-13	51.1	47.2	48.0	49.2	48.9	**18.3**	18.4	15.7	**17.3**	17.4
	FRCNN [33]	Two-Stage		46.0	44.1	44.0	45.8	45.0	16.8	18.2	15.6	16.1	16.7
	SRCNN [34]	Query		54.4	49.2	49.5	50.8	51.0	13.2	14.0	13.6	13.0	13.5
	DINO [49]	Query		54.6	**52.4**	**53.5**	**54.3**	**53.7**	13.7	14.1	10.7	12.6	12.8

Table 5. Comparison on the R-OSOD LVIS benchmark.

Method	Detector	Type	Base AP_f^{5N}	Base AP_c^{5N}	Novel AP_r^{5N}
BHRL [45]	FRCNN	Two-Stage	13.2	15.5	19.3
ADDA	FCOS	One-Stage	**21.3**	**22.9**	**27.2**
	GFL	One-Stage	20.5	22.6	26.5
	FRCNN	Two-Stage	14.6	16.9	23.5

Table 6. Comparison on the R-OSOD OpenImages benchmark. We evaluate models trained on each split of R-OSOD COCO.

Method	Detector	AP^{3N}				
		S1	S2	S3	S4	*Avg.*
BHRL [45]	FRCNN	17.2	10.9	15.8	12.0	14.0
ADDA	FCOS	25.6	24.0	24.4	23.7	24.4
	GFL	**26.7**	**24.9**	**24.7**	**24.5**	**25.2**
	FRCNN	23.3	22.0	22.5	22.1	22.5

it achieves a 3.1 points and 2.3 points improvement in AP over Meta-DETR [48] for the 3- and 5-shot settings, respectively. This underlines the superior generalization ability of ADDA. Overall, the performance of ADDA on the COCO dataset demonstrates its effectiveness even in scenarios with multiple visual templates.

5.2. Realistic Benchmarks

We provide additional experimental results on our new R-OSOD benchmarks.

Evaluation setting. On COCO, we use the same class splits as in the existing benchmarks. We use AP_{50}^{5N} as the main metric for COCO, AP^{5N} for LVISv1, and AP^{3N} for OpenImages. We also use the same protocol as before for generating image-template pairs for each test image. For comparison, we evaluate the previous state-of-the-art method BHRL [45] retrained with ImageNet-1K [9] pre-trained weights. All methods use negative sampling by default.

COCO. We first show results on COCO in Table 4. With negative templates, both methods suffer large degradation in performance compared to in Table 2, showing that negative templates is a big problem in current algorithms. Across both base and novel classes and backbones, ADDA outperforms BHRL in all of the splits. In particular, when comparing the best models, ADDA achieves around 8 points improvement in average base AP_{50}^{5N} and 3.5 points improvement in average novel AP_{50}^{5N} over BHRL [45]. We observe that ADDA with Swin-T achieves the best base accuracy, but CvT-13 achieves considerable improvement in novel accuracy.

Unlike previous methods which can only operate on one specific type of detector, we further demonstrate that ADDA can work with any detector, whether one-stage, two-stage, or query-based. For each detector, we only modify the detection head, using standard hyperparameters. Both one-stage and two-stage detectors achieve a similar level of performance on novel classes, while Faster R-CNN [33] observes a drop on base classes. We also observe that query-based detectors, Sparse R-CNN [34] and DINO [49], are extremely powerful on base classes but generalizes poorly to novel classes. This suggests that learnable queries are easier to overfit to the base classes, thus leading to weaker generalization.

LVISv1. We provide the full results on LVISv1 in Table 5 with various detectors. Even with the increase in number of classes in the benchmark, ADDA can still obtain significantly better results across frequent, common, and rare classes. ADDA outperforms BHRL by around 8 points in AP_r^{5N}, demonstrating that it is much better at detecting novel classes. This illustrates that ADDA scales much better to larger number of classes than previous methods and is thus a more effective option for real-world applications.

OpenImages. We also conduct a cross-dataset experiment by evaluating the COCO-trained models on OpenImages [20] in a zero-shot manner (no re-training) in Table 6. We evaluate models trained on each split of R-OSOD COCO. ADDA performs much better compared to BHRL across all detectors, achieving over 8 points improvement in AP^{3N} over all splits. Furthermore, the accuracy of ADDA is stable across different COCO splits, demonstrating the strong generalizability of ADDA regardless of the training categories. In contrast, the accuracy of BHRL varies by over 6 points.

Table 7. Ablation study of model components of ADDA on PASCAL VOC and R-OSOD LVISv1 benchmarks.

Backbone # FEIB 1	2	4	6	Multi-Scale TAP	BFP	VOC Base AP_{50}	Novel AP_{50}	LVISv1 Base AP_f^{5N}	AP_c^{5N}	Novel AP_r^{5N}
✓	✗	✗	✗	✗	✗	64.7	77.1	16.0	18.0	22.5
✗	✓	✗	✗	✗	✗	67.9	80.0	16.7	18.7	24.3
✗	✗	✓	✗	✗	✗	74.8	82.1	17.8	19.7	24.6
✗	✗	✗	✓	✗	✗	75.7	82.0	17.4	19.1	24.3
✗	✗	✓	✗	✓	✗	75.3	82.7	18.1	20.1	25.0
✗	✗	✓	✗	✗	✓	80.4	81.7	19.8	22.3	27.1
✗	✗	✓	✗	✓	✓	**81.6**	**83.4**	**21.3**	**22.9**	**27.2**

Table 8. Ablation study of negative sampling on *split 1* of the R-OSOD COCO benchmark.

p_{neg}	ADDA Base AP_{50}^{5N}	ADDA Novel AP_{50}^{5N}	p_{neg}	BHRL Base AP_{50}^{5N}	BHRL Novel AP_{50}^{5N}
0.0	45.1	13.9	0.0	45.5	12.4
0.1	50.5	16.6	0.1	**46.8**	13.4
0.2	51.9	17.2	0.2	44.6	**14.2**
0.4	**52.5**	**17.7**	0.4	42.4	14.0

5.3. Ablation Studies

We conduct several ablation studies to validate the effect of our model components. We use the CvT-13 [41] backbone and the FCOS [36] detector for all experiments.

Impact of model components. We evaluate the impact of each component in our model on PASCAL VOC and R-OSOD LVISv1 in Table 7. First, we evaluate the effect of the number of FEIBs in the backbone, without using our proposed multi-scale template-aware components (replaced with a standard FPN). With more integration blocks, the base performance consistently improves, while the novel performance saturates. We choose 4 blocks for the best balance between performance, generalization, and efficiency.

Next, we assess the effectiveness of our multi-scale template-aware components. With only TAP, the performance improves on both base and novel AP^{5N}. When we further propagate the fused features using BFP, we can gain additional improvements on all classes. On both benchmarks, both components together provide a significant 2.6 AP^{5N} gain on novel classes.

Impact of negative sampling. The effect of negative sampling on ADDA and BHRL [45] on split 1 of the R-OSOD COCO benchmark is shown in Table 8. Without negative sampling ($p_{neg} = 0.0$), performance of both methods on all classes suffer as the model cannot handle negative templates during inference. Increasing the proportion of negative samples significantly improves novel AP_{50}^{5N}, while base AP_{50} drops for BHRL. We choose 0.4 for ADDA and 0.2 for BHRL for all other experiments as they achieve the best novel performance for each model.

Impact of adapter design. We evaluate different possible designs of the feature adapter and compare them against TAA in Table 9. First, only using FEIBs in the first and first two stages of the backbone (rows 1 and 2) result in significantly worse detection accuracy. Adding FEIBs in the third stage (row 3) can greatly improve performance, demonstrating that high-level feature integrations are necessary for visual prompt-based detection accuracy. Using a Siamese structure instead where feature integration at all feature levels only occurs after the backbone gives a better base score but lower novel score. Finally, using our TAA achieves the best scores across all classes while being the most memory efficient, showing that our adapter design is effective for our task.

Table 9. Ablation study of adapter design on *split 1* of the R-OSOD COCO benchmark.

Adapter Design	Base AP_{50}^{5N}	Novel AP_{50}^{5N}	Memory Usage (GB)
Early stage (1)	13.6	1.7	7.2
Early stages (1 + 2)	23.5	5.1	7.5
All blocks (1 + 2 + 3)	48.5	16.6	7.5
Siamese	51.0	14.2	8.3
TAA (Ours)	**52.5**	**17.7**	**7.1**

Figure 4. Qualitative comparison between BHRL [45] and ADDA on novel classes in COCO [25]. ADDA produces more accurate detections and fewer false positives.

Qualitative results. We qualitatively compare the detections of BHRL [45] and ADDA on several novel classes in COCO [25] in Figure 4. ADDA produces more accurate detections overall and fewer false positive detections.

6. Conclusion

In this work, we present a unified solution to visual prompt-based detection that follows the standard detection pipeline to enable any detector to detect any object. This is achieved through our Template-Aware Adapter, which utilizes feature integration blocks directly within the feature extractor to model feature interactions between image and template. In addition, we address weaknesses of current benchmarks to construct more realistic one-shot detection benchmarks that are more relevant for real-world applications. We hope our new framework and benchmark motivate the design of more unified and realistic detectors in the future.

References

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 4

[2] Adrian Bulat, Ricardo Guerrero, Brais Martinez, and Georgios Tzimiropoulos. Fs-detr: Few-shot detection transformer with prompting and without re-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11793–11802, 2023. 6

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

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 2

[5] Ding-Jie Chen, He-Yen Hsieh, and Tyng-Luh Liu. Adaptive image transformer for one-shot object detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12242–12251, 2021. 1, 2, 5, 6

[6] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019. 1

[7] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. Mixformer: End-to-end tracking with iterative mixed attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 4

[8] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1601–1610, 2021. 1, 2, 5

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 4, 6, 7

[10] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. In *IJCV*, 2010. 5, 6

[11] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. Few-shot object detection with attention-rpn and multi-relation detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2020. 2, 6

[12] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 1

[13] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019. 2, 5

[14] Guangxing Han, Yicheng He, Shiyuan Huang, Jiawei Ma, and Shih-Fu Chang. Query adaptive few-shot object detection with heterogeneous graph convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3263–3272, 2021. 6

[15] Guangxing Han, Shiyuan Huang, Jiawei Ma, Yicheng He, and Shih-Fu Chang. Meta faster r-cnn: Towards accurate few-shot object detection with attentive feature alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 780–789, 2022. 1, 6

[16] Guangxing Han, Jiawei Ma, Shiyuan Huang, Long Chen, and Shih-Fu Chang. Few-shot object detection with fully cross-transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5321–5330, 2022. 2, 6

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 2

[18] Ting-I Hsieh, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. One-shot object detection with co-attention and co-excitation. *Advances in neural information processing systems*, 32, 2019. 2, 5, 6

[19] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8420–8429, 2019. 2, 6

[20] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 2, 5, 7

[21] Jinxiang Lai, Siqian Yang, Wenlong Liu, Yi Zeng, Zhongyi Huang, Wenlong Wu, Jun Liu, Bin-Bin Gao, and Chengjie Wang. tSF: Transformer-Based Semantic Filter for Few-Shot Learning. In *ECCV*, pages 1–19. Springer, 2022. 6

[22] Bowen Li, Chen Wang, Pranay Reddy, Seungchan Kim, and Sebastian Scherer. Airdet: Few-shot detection without fine-tuning for autonomous exploration. In *European Conference on Computer Vision*, pages 427–444. Springer, 2022. 6

[23] Feng Li, Hao Zhang, Shilong Liu, Jian Guo, Lionel M Ni, and Lei Zhang. Dn-detr: Accelerate detr training by introducing query denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13619–13627, 2022. 2

[24] 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. *Advances in Neural Information Processing Systems*, 33:21002–21012, 2020. 2, 7

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 5, 6, 8

[26] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 4

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

[28] Shilong Liu, Feng Li, Hao Zhang, Xiao Yang, Xianbiao Qi, Hang Su, Jun Zhu, and Lei Zhang. Dab-detr: Dynamic anchor boxes are better queries for detr. *arXiv preprint arXiv:2201.12329*, 2022. 2

[29] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 4, 6

[30] Claudio Michaelis, Ivan Ustyuzhaninov, Matthias Bethge, and Alexander S Ecker. One-shot instance segmentation. *arXiv preprint arXiv:1811.11507*, 2018. 2, 5, 6

[31] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 5, 6

[32] Limeng Qiao, Yuxuan Zhao, Zhiyuan Li, Xi Qiu, Jianan Wu, and Chi Zhang. Defrcn: Decoupled faster r-cnn for few-shot object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8681–8690, 2021. 2, 6

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

[34] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14454–14463, 2021. 7

[35] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1199–1208, 2018. 2

[36] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019. 2, 6, 7, 8

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[38] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016. 2

[39] Xin Wang, Thomas E. Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. 2020. 2, 6

[40] Yanjie Wang, Xu Zou, Luxin Yan, Sheng Zhong, and Jiahuan Zhou. Snida: Unlocking few-shot object detection with non-linear semantic decoupling augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12544–12553, 2024. 6

[41] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22–31, 2021. 4, 6, 8

[42] Jiaxi Wu, Songtao Liu, Di Huang, and Yunhong Wang. Multi-scale positive sample refinement for few-shot object detection. In *European conference on computer vision*, pages 456–472. Springer, 2020. 2

[43] Yang Xiao and Renaud Marlet. Few-shot object detection and viewpoint estimation for objects in the wild. In *European conference on computer vision*, pages 192–210. Springer, 2020. 2

[44] Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaodan Liang, and Liang Lin. Meta r-cnn: Towards general solver for instance-level low-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9577–9586, 2019. 2

[45] Hanqing Yang, Sijia Cai, Hualian Sheng, Bing Deng, Jianqiang Huang, Xiansheng Hua, Yong Tang, and Yu Zhang. Balanced and hierarchical relation learning for one-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 5, 6, 7, 8

[46] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 1

[47] Gongjie Zhang, Kaiwen Cui, Rongliang Wu, Shijian Lu, and Yonghong Tian. Pnpdet: Efficient few-shot detection without forgetting via plug-and-play sub-networks. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3822–3831, 2021. 2

[48] Gongjie Zhang, Zhipeng Luo, Kaiwen Cui, and Shijian Lu. Meta-detr: Image-level few-shot object detection with inter-class correlation exploitation. *arXiv preprint arXiv:2103.11731*, 2021. 1, 2, 6, 7

[49] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022. 1, 2, 7

[50] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020. 2

[51] Weilin Zhang and Yu-Xiong Wang. Hallucination improves few-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13008–13017, 2021. 2

[52] Wenwen Zhang, Yun Hu, Hangguan Shan, and Eryun Liu. Exploring base-class suppression with prior guidance for bias-

free one-shot object detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(7):7314–7322, 2024. 2, 5

[53] Xinyu Zhang, Yuhan Liu, Yuting Wang, and Abdeslam Boularias. Detect everything with few examples, 2024. 5, 6

[54] Yizhou Zhao, Xun Guo, and Yan Lu. Semantic-aligned fusion transformer for one-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7601–7611, 2022. 2, 5, 6

[55] Chenchen Zhu, Fangyi Chen, Uzair Ahmed, Zhiqiang Shen, and Marios Savvides. Semantic relation reasoning for shot-stable few-shot object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8782–8791, 2021. 2

[56] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9308–9316, 2019. 4