Meta-DETR: Few-Shot Object Detection via Unified Image-Level Meta-Learning

Gongjie Zhang[†] Zhipeng Luo[†] Kaiwen Cui Shijian Lu[∗]
Nanyang Technological University, Singapore
gongjiezhang@ntu.edu.sg zhipeng001@e.ntu.edu.sg kaiwen001@e.ntu.edu.sg shijian.lu@ntu.edu.sg

Abstract

Few-shot object detection aims at detecting novel objects with only a few annotated examples. Prior works have proved meta-learning a promising solution, and most of them essentially address detection by meta-learning over regions for their classification and location fine-tuning. However, these methods substantially rely on initially well-located region proposals, which are usually hard to obtain under the few-shot settings. This paper presents a novel meta-detector framework, namely Meta-DETR, which eliminates region-wise prediction and instead meta-learns object localization and classification at image level in a unified and complementary manner. Specifically, it first encodes both support and query images into category-specific features and then feeds them into a category-agnostic decoder to directly generate predictions for specific categories. To facilitate meta-learning with deep networks, we design a simple but effective Semantic Alignment Mechanism (SAM), which aligns high-level and low-level feature semantics to improve the generalization of meta-learned representations. Experiments over multiple few-shot object detection benchmarks show that Meta-DETR outperforms state-of-the-art methods by large margins.

1. Introduction

Computer vision has witnessed significant progress in recent years. However, there still exists a huge gap between current computer vision techniques and human visual systems in learning new concepts from very few examples: most existing methods require large amounts of annotated samples, while humans can effortlessly recognize a new concept even with very little instruction [60, 55]. Such a human-like capability of generalizing from limited examples is highly desirable for machine vision systems, especially when sufficient training samples are not available or their annotations are hard to obtain [57, 14, 35, 20, 87, 19].

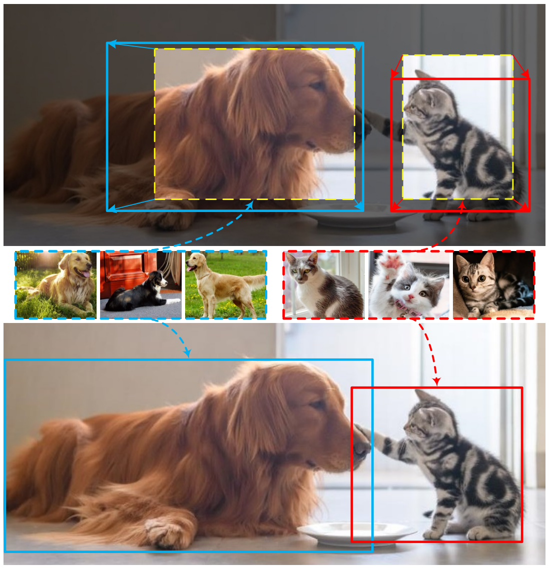

Figure 1. **Upper:** Most existing meta-detectors essentially perform *region-wise* predictions, which heavily rely on the quality of initial region proposals that cannot be guaranteed under the few-shot settings. **Lower:** The proposed Meta-DETR meta-learns object localization and classification at *image level* in a unified and complementary manner (without region-wise prediction), leading to superior few-shot object detection performance.

In this work, we explore the challenging *few-shot object detection* task, which requires both recognition and localization of novel objects within an image. Prior works [22, 46, 74, 81, 10, 80] have proved meta-learning a promising solution. As illustrated in the upper part of Fig. 1, they essentially address object detection by performing meta-learning over regions, including region proposals [81, 80], anchors [22], and window centers [46], for their classification and location fine-tuning. However, as identified in [10] and [91], these methods rely heavily on the quality of initial region proposals, which cannot be guaranteed in the few-shot setups with scarce training samples, thus producing inaccurate or missed detection. Though FSOD [10] proposes to meta-learn the generation of region proposals, this issue remains as the framework is still inherently region-based.

[†] denotes equal contribution.
[∗] denotes corresponding author.

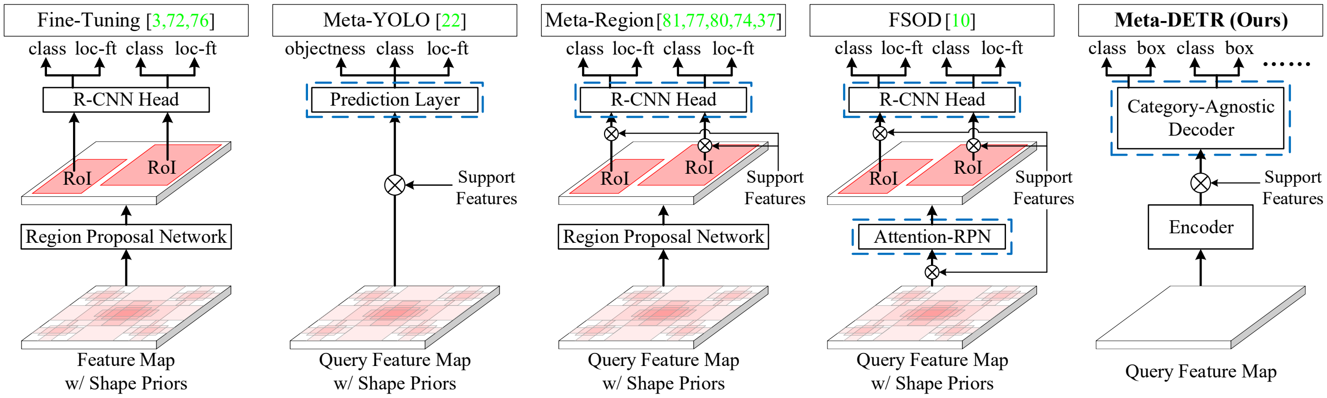

Figure 2. **Comparison of existing few-shot detectors with our Meta-DETR.** Dashed blue boxes indicate meta-learning components. ⊗ indicates feature aggregation. Unlike prior works that rely on region-wise predictions, Meta-DETR unifies the meta-learning of object localization and classification at image level with a single meta-learning module.

Based on the analysis above, a key limitation rooted in existing meta-detectors is the region-wise prediction approach. Besides, under the challenging settings of few-shot object detection where supervision from annotated examples is minimal, the complementary effect between classification and localization (as demonstrated in [93, 78, 53]) should be maximally exploited. Therefore, an ideal meta-detector should discard such region-based prediction and effectively leverage the synergistic relationship between classification and localization by meta-learning both sub-tasks in a fully end-to-end manner. However, such a framework is still absent to the best of our knowledge.

Recently, the emergence of fully end-to-end detection frameworks [2, 96] clears the way to such a framework. This paper presents *Meta-DETR*, a novel region-free framework for few-shot object detection that meta-learns image-level localization and classification in a unified and complementary manner. Concretely, it incorporates meta-learning into the DETR frameworks [2, 96] by first encoding support and query images into category-specific features and then feeding them into a category-agnostic decoder to directly generate detection results for the target categories. To facilitate meta-learning with deep networks, we design a simple but effective Semantic Alignment Mechanism (SAM) that aligns high-level and lower-level feature semantics and prevents reliance on category-specific representations with low generalization capability.

The contributions of this work are threefold. *First*, we propose Meta-DETR, a novel few-shot object detection framework that unifies image-level meta-learning of object localization and classification into a single module without requiring region-wise prediction. Such a design can effectively leverage the synergistic relationship between the two sub-tasks and avoid constraints caused by region-wise prediction. *Second*, we design a simple but effective Semantic Alignment Mechanism (SAM) that enhances the generalization capacity of meta-learning by aligning high-level and low-level semantics to avoid reliance on category-specific representations. *Third*, extensive experiments show that our method achieves state-of-the-art performance on multiple benchmarks for few-shot object detection.

2. Related Work

Object Detection. Generic object detection [38] is a joint task on object localization and classification. Modern object detectors can be broadly classified into two categories including two-stage detectors and single-stage detectors. The dominant two-stage detectors are Faster R-CNN [52] and its variants [21, 1, 31, 58, 59, 7, 89, 17, 82, 49], which first adopt a Region Proposal Network (RPN) to generate region proposals as coarse localization and then perform per-region classification and location fine-tuning. Differently, single-stage detectors [41, 51, 26, 33, 95, 70, 90, 40] employ densely placed anchors as region proposals and directly make predictions on them. These aforementioned methods still rely on many heuristics like anchor generation. Recently, DETR [2] and its variants [96, 6, 63, 32, 92] have received vast attention thanks to their merits of no heuristic design, fully end-to-end pipeline, and comparable or even better performance. However, these detectors still heavily rely on human supervision in the form of large amounts of annotated training samples, thus will suffer from huge performance drop in the context of few-shot learning.

Few-Shot Learning. Few-shot learning aims at bridging the gap between existing models and human intelligence in learning novel concepts from very few samples. One promising solution is meta-learning [18, 66], which aims to extract meta-level knowledge that can generalize across various tasks via ‘learning to learn’. Extensive researches [11, 68, 61, 64, 12, 44, 4, 43, 27, 62, 73, 56, 50, 79, 47, 8, 83, 5, 28, 36, 39] have proved the effectiveness

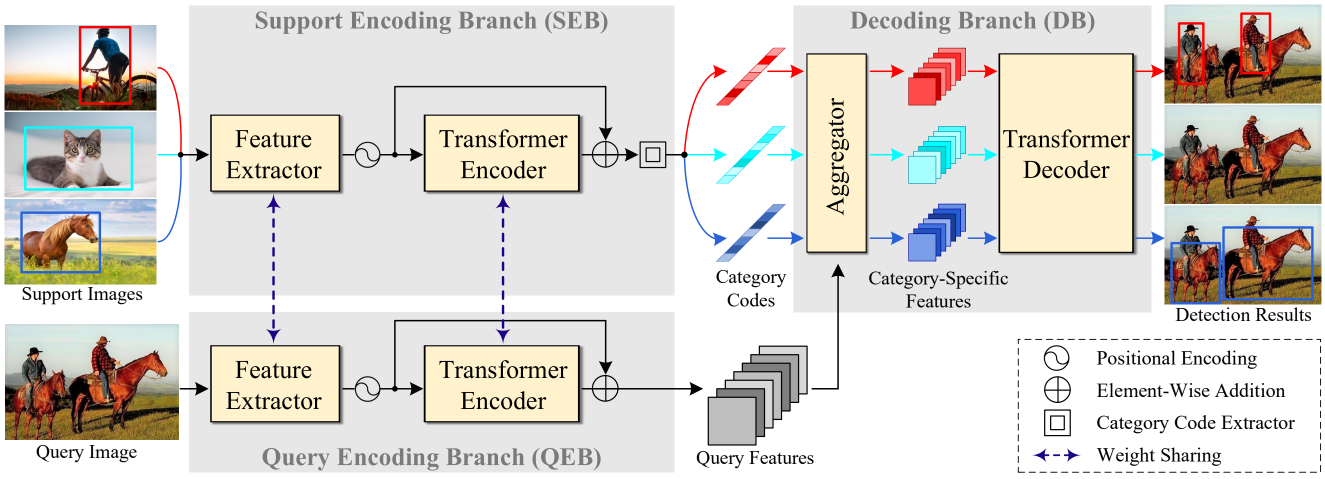

Figure 3. **The architecture of our proposed Meta-DETR.** It consists of a Query Encoding Branch (QEB), a Support Encoding Branch (SEB), and a Decoding Branch (DB). QEB receives a query image and generates its query features through a feature extractor and a transformer encoder. SEB, which shares all learnable parameters with QEB, extracts support category codes from the support images. Given the query features with a support category code, DB first aggregates them into category-specific features and then applies a category-agnostic transformer decoder to predict the detection results over the corresponding support category.

of the meta-learning paradigm for the few-shot classification task. However, other more complex few-shot learning tasks [65, 13, 45, 75, 69, 71] are still relatively under-explored.

Few-Shot Object Detection. Prior works on few-shot object detection can be formulated in two paradigms: transfer-learning-based and meta-learning-based. Methods using transfer-learning include LSTD [3], PNPDet [88], TFA [72], and MPSR [76], where novel concepts are learned via fine-tuning. Differently, methods using meta-learning extract meta-level knowledge that can efficiently adapt to novel categories by constructing and learning on various auxiliary tasks, in which target categories are dynamically conditioned on support images. Of them, Meta-YOLO [22] and ONCE [46] are based on single-stage detectors, and Meta R-CNN [81] and its variants [77, 74, 23, 30, 80, 37] are built upon Faster R-CNN [52]. As shown in Fig. 2, existing meta-detectors essentially perform region-wise meta-learning, thus requiring initially well-located regions. However, such well-located regions for novel objects are usually hard to obtain with non-learnable shape priors and fine-tuned RPN when training samples are scarce. FSOD [10] attempts to mitigate this issue by meta-learning an Attention-RPN, but the issue remains as this framework and Attention-RPN are still innately region-based.

Our Meta-DETR follows the track of meta-learning. Unlike previous works, it discards region-wise prediction and instead unifies the meta-learning of localization and classification at image level with a category-agnostic decoder, thus leveraging global contexts and the synergistic relationship of the two sub-tasks to achieve superior performance.

3. Method

3.1. Problem Definition

Given two sets of categories $\mathcal{C}_{\text{base}}$ and $\mathcal{C}_{\text{novel}}$, where $\mathcal{C}_{\text{base}} \cap \mathcal{C}_{\text{novel}} = \varnothing$, a few-shot object detector aims at detecting objects of $\mathcal{C}_{\text{base}} \cup \mathcal{C}_{\text{novel}}$ by learning from a base dataset $\mathcal{D}_{\text{base}}$ with abundant annotated instances of $\mathcal{C}_{\text{base}}$ and a novel dataset $\mathcal{D}_{\text{novel}}$ with very few annotated instances of $\mathcal{C}_{\text{novel}}$. In the task of K-shot object detection, there are exactly K annotated object instances for each novel category in $\mathcal{D}_{\text{novel}}$.

3.2. Meta-DETR

3.2.1 Revisiting DETR Frameworks

Modern detectors like Faster R-CNN [52] address object detection by performing the surrogate task of classification and location fine-tuning on a number of regions. Such detectors require many heuristics and are not fully end-to-end. Recently, DETR [2] eliminates the need for such heuristic designs and achieves the first fully end-to-end detection framework. It is built upon the Transformer encoder-decoder architecture [67], combined with a set-based Hungarian loss that forces unique predictions for each object via bipartite matching. Besides, Deformable DETR [96] further extends DETR by mitigating its high complexity and slow convergence issue.

Meta-DETR extends the DETR frameworks [2, 96] by incorporating meta-learning into such fully end-to-end detection frameworks. Its innovative designs can help evade various issues such as the constraint of region-wise prediction under the context of few-shot object detection.

3.2.2 Network Description

Aiming at performing unified meta-learning for localization and classification at image level, our Meta-DETR is conceptually simple. As shown in Fig. 3, it consists of a *Query Encoding Branch (QEB)*, a *Support Encoding Branch (SEB)*, and a *Decoding Branch (DB)*. Given a *Query Image* and several *Support Images* with instance annotations, QEB and SEB first encode them into *Query Features* and *Category Codes*, respectively. DB then takes the query features and category codes as input and predicts *Detection Results* over the corresponding support categories. As target categories to detect are dynamically conditioned on the provided support images, Meta-DETR is able to extract category-agnostic meta-level knowledge that can easily adapt to novel categories.

Query Encoding Branch (QEB). The design of QEB follows Deformable DETR [96] except for a residual connection that will be introduced later. As illustrated in Fig. 3, it mainly consists of a feature extractor and a transformer encoder. Given a query image, the feature extractor (a CNN backbone such as ResNet [16]) generates its feature maps and then adopts 1×1 convolution to make the feature maps' channel dimension compatible with the downstream modules. Since the transformer encoder expects a sequence as input, we first inject positional encoding into the feature maps, collapse the feature maps' spatial dimensions into one dimension, and then feed them into the transformer encoder to produce the query features.

Support Encoding Branch (SEB). SEB shares all learnable parameters with QEB following the philosophy of Siamese Networks [25]. Unlike QEB that preserves image-level information within the query features, SEB aims at extracting category codes that mostly relate to certain object instances within the support images. We, therefore, introduce a Category Code Extractor (CCE) to filter out irrelevant information within the support images. CCE has no learnable parameters. It derives support category codes via three sequential operations: 1) restoring the features' spatial dimension from the transformer encoder, 2) locating support object instances with RoIAlign [15], and 3) global average pooling followed by a sigmoid function. When there are multiple support images for a category, it averages all category codes as the final category code.

Decoding Branch (DB). DB receives the outputs of QEB and SEB and produces object detection results, and its target categories are dynamically determined by the category codes. Concretely, it aggregates the query features and category codes into a set of category-specific features. The design of aggregator follows previous work [80]. A transformer decoder with a feed-forward network (FFN, omitted in Fig. 3 for simplicity) then takes the category-specific features and a small fixed number of object queries as input and produces detection results over the corresponding categories. Similar to the decoder in DETR frameworks, DB eliminates region-wise prediction and addresses object detection at image level. However, DB is category-agnostic with no intention to detect objects of specific categories. Such unique design enables joint meta-learning of object localization and classification at image level, which can avoid potential issues with region-wise prediction and achieve superior few-shot detection performance.

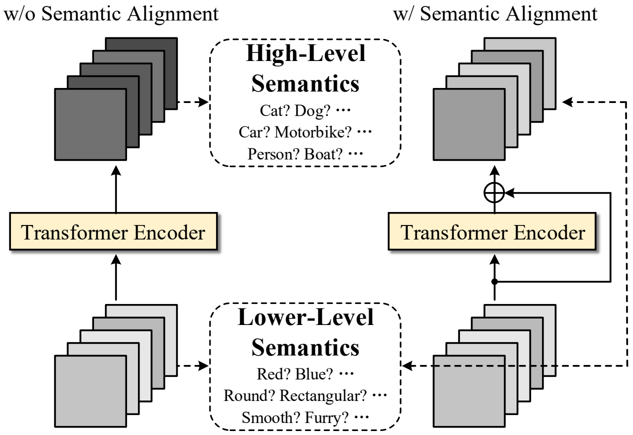

Figure 4. **Semantic Alignment Mechanism.** A simple residual connection acts as self-regularization to prevent the transformer encoder from relying on undesirable category-specific features by aligning the feature semantics of its input and output.

Semantic Alignment Mechanism (SAM). Meta-learning has been proved promising for few-shot learning. Its major motivation is to obtain meta-level knowledge that can generalize to various categories instead of focusing on specific categories. However, most works [27, 4, 29, 83, 86] perform meta-learning on relatively shallow networks, such as ResNet-12 and ResNet-18. There is also evidence [94, 12, 4, 48] that meta-learning a deeper network from scratch performs comparable or even worse than without meta-learning. One possible reason is that, even with meta-learning, deeper networks still tend to learn and rely on category-specific semantics with poor generalization undesirably. To mitigate this issue, we propose to incorporate a simple but effective Semantic Alignment Mechanism (SAM), which is essentially a residual connection as illustrated in Fig. 4, into the proposed Meta-DETR.

The motivation behind SAM is simple and straightforward. As observed in the feature visualization literature [85, 84], features from bottom layers relate to low-level cues such as colors and shapes that have better generalization; while features from top layers relate to more complex and specific concepts such as categories. To avoid reliance on such high-level category-specific features, SAM incorporates a shortcut connection to bypass the transformer encoder, which works as self-regularization to guide the feature semantics from the transformer encoder to align with its input feature semantics with better generalization.

It is worth mentioning that the motivation behind SAM is very different from the residual connections that have been widely used in various neural network architectures. The residual connections in ResNet [16] only bypass several convolutional layers and aim at improving the gradient flow and solving the gradient vanishing issue when training very deep neural networks. Meta-DETR does not suffer from gradient vanishing as its transformer [67] building blocks already incorporate such residual connections. In contrast, the residual connection used in SAM bypasses the entire transformer encoder, aiming to align its outputs' feature semantics with its inputs', thus acting as self-regularization to prevent reliance on category-specific semantics.

3.2.3 Training Objective

Detection Target Generation. Assume the fixed number of object queries is N, which means Meta-DETR infers N predictions over each category in a single pass through the decoder. Let us denote by x_{query} the query image, and $y=\{y_i\}_{i=1}^{N}=\{(c_i,b_i)\}_{i=1}^{N}$ the ground truth set of objects within the query image, which is a set of size N. When y_i indicates an object, $y_i=(c_i,b_i)$, where c_i denotes the target category label and b_i denotes the bounding box of the object. When y_i indicates no object, $y_i=(\varnothing,\varnothing)$.

Meta-DETR dynamically conditions its detection targets on support images. Given a support image x_{supp} along with its object annotation $(c_{\text{supp}}, b_{\text{supp}})$, the detection targets are defined as:

$$y' = \{y'_i\}_{i=1}^{N} = \{(c'_i, b'_i)\}_{i=1}^{N} = \{\psi(y_i, c_{\text{supp}})\}_{i=1}^{N} \tag{1}$$

where $\psi(y_i, c_{\text{supp}})$ acts to filter irrelevant object annotations, which can be formulated as:

$$\psi(y_i, c_{\text{supp}}) = \begin{cases} (\varnothing,\varnothing), & \text{if } \; y_i = (\varnothing,\varnothing) \\ (\varnothing,\varnothing), & \text{if } \; c_i \neq c_{\text{supp}} \\ (1, b_i), & \text{if } \; c_i = c_{\text{supp}} \end{cases} . \tag{2}$$

Note that y' can completely consist of $(\varnothing,\varnothing)$. In this case we call c_{supp} a negative target category.

Loss Function. Assume the N predictions for target category made by Meta-DETR are $\hat{y}=\{\hat{y}_i\}_{i=1}^{N}=\left\{(\hat{c}_i,\hat{b}_i)\right\}_{i=1}^{N}$. We adopt a pair-wise matching loss $\mathcal{L}_{\text{match}}(y'_i, \hat{y}_{\sigma(i)})$ to search for a bipartite matching between $\hat{y}$ and y' with the lowest cost:

$$\hat{\sigma} = \arg\min_{\sigma} \sum\nolimits_{i=1}^{N} \mathcal{L}_{\text{match}}(y'_i, \hat{y}_{\sigma(i)}) \tag{3}$$

where σ denotes a permutation of N elements, and $\hat{\sigma}$ denotes the optimal assignment between predictions and targets. Since the matching should consider both classification and localization, the matching loss is defined as:

$$\begin{aligned} \mathcal{L}_{\text{match}}(y'_i, \hat{y}_{\sigma(i)}) = & \mathbb{1}_{\{c'_i \neq \varnothing\}} \mathcal{L}_{\text{cls}}(c'_i, \hat{c}_{\sigma(i)}) + \\ & \mathbb{1}_{\{c'_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b'_i, \hat{b}_{\sigma(i)}) \ . \end{aligned} \tag{4}$$

With the optimal assignment $\hat{\sigma}$ obtained with Eq. 3 and Eq. 4, we optimize the network using the following loss function:

$$\mathcal{L}(y', \hat{y}) = \sum_{i=1}^{N} \left[\mathcal{L}_{\text{cls}}(c'_i, \hat{c}_{\hat{\sigma}(i)}) + \mathbb{1}_{\{c'_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b'_i, \hat{b}_{\hat{\sigma}(i)}) \right] \tag{5}$$

where we adopt sigmoid focal loss [33] for $\mathcal{L}_{\text{cls}}$ and a linear combination of $\ell 1$ loss and GIoU loss [54] for $\mathcal{L}_{\text{box}}$. Similar to [2] and [96], $\mathcal{L}(y', \hat{y})$ is applied to each layer of the transformer decoder.

Following [81], we also adopt a conventional cross-entropy loss, denoted as $\mathcal{L}_{\text{SEB}}$, to classify the category codes produced by SEB. This encourages category codes that belong to different categories to be distinguished from each other.

3.2.4 Training and Inference Scheme

The training procedure consists of two stages. The first stage is *base training stage*. During this stage, the model is trained on the base dataset $\mathcal{D}_{\text{base}}$ with abundant training samples for each base category. The second stage is *few-shot fine-tuning stage*. In this stage, we train the model on both base and novel categories with limited training samples. Only K object instances are available for each novel category in K-shot object detection. Following [72, 81, 80], we also include several object instances for each base category to prevent performance drop for base categories. In both stages, we optimize the network in an end-to-end manner using the loss functions described in Section 3.2.3.

In both training stages, multiple auxiliary tasks, also known as episodes, are formed to train the proposed Meta-DETR. Specifically, each episode contains one query image and 10 support images representing different target categories to detect. Target categories include both positive categories and negative categories. Support images are randomly sampled from the training dataset.

Before inference, we first use SEB to obtain the category codes for all categories once and for all. For each category, if there are multiple support images, we average all corresponding category codes as the final category code. After acquiring the category codes, SEB can be detached. During inference, Meta-DETR does not need to repeatedly compute category codes as in the training stage, which promises the efficient inference of Meta-DETR.

		Category Split 1					Category Split 2					Category Split 3				
Method	multi-scale	1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
FRCN-ft-full [52, 72]	✓	15.2	20.3	29.0	40.1	45.5	13.4	20.6	28.6	32.4	38.8	19.6	20.8	28.7	42.2	42.1
D-DETR-ft-full [96] †	✓	5.6	13.3	21.7	34.2	45.0	10.9	13.0	18.4	27.3	39.4	7.3	16.6	20.8	32.2	41.8
LSTD [3]	✓	8.2	1.0	12.4	29.1	38.5	11.4	3.8	5.0	15.7	31.0	12.6	8.5	15.0	27.3	36.3
RepMet [56]	✓	26.1	32.9	34.4	38.6	41.3	17.2	22.1	23.4	28.3	35.8	**27.5**	31.1	31.5	34.4	37.2
TFA w/ fc [72] †	✓	22.9	34.5	40.4	46.7	52.0	16.9	26.4	30.5	34.6	39.7	15.7	27.2	34.7	40.8	44.6
TFA w/ cos [72] †	✓	25.3	36.4	42.1	47.9	52.8	18.3	27.5	30.9	34.1	39.5	17.9	27.2	34.3	40.8	45.6
MPSR [76] †	✓	**34.7**	**42.6**	46.1	49.4	56.7	**22.6**	30.5	31.0	36.7	43.3	**27.5**	32.5	38.2	44.6	50.0
Meta-YOLO [22]		14.8	15.5	26.7	33.9	47.2	15.7	15.3	22.7	30.1	40.5	21.3	25.6	28.4	42.8	45.9
Meta Det [74]		18.9	20.6	30.2	36.8	49.6	21.8	23.1	27.8	31.7	43.0	20.6	23.9	29.4	43.9	44.1
Meta R-CNN [81]		19.9	25.5	35.0	45.7	51.5	10.4	19.4	29.6	34.8	45.4	14.3	18.2	27.5	41.2	48.1
FsDetView [80]		24.2	35.3	42.2	49.1	57.4	21.6	24.6	31.9	37.0	45.7	21.2	30.0	37.2	43.8	49.6
Meta-DETR (Ours)		17.5	36.0	45.1	51.2	57.1	18.5	27.5	34.7	41.1	49.8	15.4	32.6	39.4	49.0	54.3
Meta-DETR (Ours)	✓	20.4	35.0	**46.3**	**52.2**	**57.8**	20.2	**30.9**	**38.2**	**44.0**	**52.6**	22.8	**34.9**	**43.0**	**50.2**	**54.9**

Table 1. Few-shot detection performance (mAP@0.5) on Pascal VOC *test 07* set for novel categories. Results are averaged over multiple repeated runs with different randomly sampled support datasets. † indicates results are re-evaluated using official codes for multiple runs since original results are evaluated with a single run.

4. Experiments

4.1. Datasets

We follow the data setups of prior works for few-shot object detection [22, 74, 81, 72, 80, 76]. Concretely, two widely used few-shot object detection benchmarks are evaluated in our experiments.

Pascal VOC [9] consists of images with object annotations of 20 categories. We use *trainval 07+12* for training and perform evaluations on *test 07*. Following [22, 81, 72, 80], we use 3 novel / base category splits, *i.e.*, ("bird", "bus", "cow", "motorbike", "sofa" / others); ("aeroplane", "bottle","cow","horse","sofa" / others) and ("boat", "cat", "motorbike","sheep", "sofa" / others). The number of shots is set to 1, 2, 3, 5 and 10. Mean average precision (mAP) at IoU threshold 0.5 is used as the evaluation metric. Results are averaged over 10 randomly sampled support datasets.

MS COCO [34] is a more challenging object detection dataset, which contains 80 categories including those 20 categories in Pascal VOC. We adopt the 20 shared categories as novel categories, and adopt the remaining 60 categories in MS COCO dataset as base categories. The number of shots is 10 and 30. We use *train 2017* for training, and perform evaluations on *val 2017*. Standard evaluation metrics for MS COCO are adopted. Results are averaged over 5 randomly sampled support datasets.

Shot	Method	multi-scale	Base	Novel
3	LSTD [3]	✓	66.3	12.4
	TFA w/ cos [72] †	✓	**77.3**	42.1
	MPSR [76] †	✓	65.9	46.1
	Meta-YOLO [22]		64.8	26.7
	Meta R-CNN [81]		64.8	35.0
	Meta-DETR (Ours)		65.2	45.1
	Meta-DETR (Ours)	✓	66.5	**46.3**
10	LSTD [3]	✓	66.3	38.5
	TFA w/ cos [72] †	✓	**77.5**	52.8
	MPSR [76] †	✓	69.8	56.7
	Meta-YOLO [22]		63.6	47.2
	Meta R-CNN [81]		67.9	51.5
	Meta-DETR (Ours)		67.1	57.1
	Meta-DETR (Ours)	✓	67.4	**57.8**

Table 2. Few-shot detection performance (mAP@0.5) for base and novel categories on category split 1 of Pascal VOC. Results are averaged over multiple runs. † indicates re-evaluated results.

4.2. Implementation Details

We adopt commonly used ResNet-101 [16] as the feature extractor in both QEB and SEB. The network architectures and hyper-parameters of transformer encoder and decoder remain the same as Deformable DETR [96]. The feed-forward network (FFN) after the transformer decoder is a 3-layer MLP for box prediction and a 1-layer MLP for object confidence prediction. Thanks to the multi-scale attention module introduced in Deformable DETR [96], Meta-DETR supports multi-scale features as input by nature without any modification. For a comprehensive comparison, we present results of Meta-DETR with both single-scale and multi-scale features in benchmarking. For ablation study, we only adopt the single-scale setting for Meta-DETR.

We train our model using the AdamW [24, 42] optimizer with an initial learning rate of 2×10^{-4} and a weight decay of 1×10^{-4}. We adopt a batch size of 32 and each query image is associated with 10 support images to form an episode. Conventional data augmentation as used in [2, 96] is adopted during training. In the base training stage, we train the model for 100 epochs for Pascal VOC and 50 epochs for MS COCO. Learning rate is decayed at the 85^{th} and 40^{th} epoch by a factor of 0.1, respectively. In the few-

			Average Precision						Average Recall					
Shot	Method	multi-scale	$\mathrm{AP}_{0.5:0.95}$	$\mathrm{AP}_{0.5}$	$\mathrm{AP}_{0.75}$	AP_{S}	AP_{M}	AP_{L}	AR_{1}	AR_{10}	AR_{100}	AR_{S}	AR_{M}	AR_{L}
10	LSTD [3]	✓	3.2	8.1	2.1	0.9	2.0	6.5	7.8	10.4	10.4	1.1	5.6	19.6
	TFA w/ fc [72] †	✓	9.1	17.3	8.5	-	-	-	-	-	-	-	-	-
	TFA w/ cos [72] †	✓	9.1	17.1	8.8	-	-	-	-	-	-	-	-	-
	MPSR [76]	✓	9.8	17.9	9.7	**3.3**	9.2	16.1	15.7	21.2	21.2	4.6	19.6	34.3
	Meta-YOLO [22]		5.6	12.3	4.6	0.9	3.5	10.5	10.1	14.3	14.4	1.5	8.4	28.2
	Meta Det [74]		7.1	14.6	6.1	1.0	4.1	12.2	11.9	15.1	15.5	1.7	9.7	30.1
	Meta R-CNN [81]		8.7	19.1	6.6	2.3	7.7	14.0	12.6	17.8	17.9	7.8	15.6	27.2
	FSOD [10] †		12.0	22.4	11.8	2.9	12.2	20.7	18.8	26.4	26.4	3.6	23.6	45.6
	FsDetView [80]		12.5	27.3	9.8	2.5	13.8	19.9	20.0	25.5	25.7	7.5	27.6	38.9
	Meta-DETR (Ours)		16.7	**29.0**	17.1	2.7	13.7	27.0	19.6	30.4	32.7	7.7	29.3	52.8
	Meta-DETR (Ours)	✓	**17.8**	28.8	**18.5**	**3.3**	**14.0**	**29.3**	**21.0**	**32.2**	**34.1**	**7.9**	**29.9**	**56.0**
30	LSTD [3]	✓	6.7	15.8	5.1	0.4	2.9	12.3	10.9	14.3	14.3	0.9	7.1	27.0
	TFA w/ fc [72] †	✓	12.0	22.2	11.8	-	-	-	-	-	-	-	-	-
	TFA w/ cos [72] †	✓	12.1	22.0	12.0	-	-	-	-	-	-	-	-	-
	MPSR [76]	✓	14.1	25.4	14.2	4.0	12.9	23.0	17.7	24.2	24.3	5.5	21.0	39.3
	Meta-YOLO [22]		9.1	19.0	7.6	0.8	4.9	16.8	13.2	17.7	17.8	1.5	10.4	33.5
	Meta Det [74]		11.3	21.7	8.1	1.1	6.2	17.3	14.5	18.9	19.2	1.8	11.1	34.4
	Meta R-CNN [81]		12.4	25.3	10.8	2.8	11.6	19.0	15.0	21.4	21.7	8.6	20.0	32.1
	FsDetView [80]		14.7	30.6	12.2	3.2	15.2	23.8	22.0	28.2	28.4	8.3	30.3	42.1
	Meta-DETR (Ours)		21.3	**36.0**	22.0	3.8	17.8	35.5	22.2	33.8	36.3	9.1	34.0	59.0
	Meta-DETR (Ours)	✓	**22.9**	35.8	**23.8**	**4.7**	**20.9**	**36.5**	**23.3**	**36.0**	**38.4**	**12.5**	**36.0**	**59.9**

Table 3. Few-shot detection performance on MS COCO *val 2017* set for novel categories. Results are averaged over multiple repeated runs with different randomly sampled support datasets. † indicates results are re-evaluated using official codes for multiple runs since original results are evaluated with a single run.

shot fine-tuning stage, the same settings (excluding the total number of epochs and the learning rate decay epochs) are applied to train the model until full convergence.

4.3. Comparison with State-of-the-Art Methods

Pascal VOC. Table 1 shows the few-shot detection performance for novel categories of Pascal VOC. It can be seen that Meta-DETR outperforms existing methods for most cases except when training samples are extremely scarce. We conjecture that the unsatisfactory performance for extremely low-shot settings is largely attributed to the large search space that comes with Meta-DETR's image-level prediction, which may lead to overfitting when training samples are extremely insufficient. However, when there are slightly more training samples for novel categories, *e.g.*, 3-shot, 5-shot, and 10-shot, Meta-DETR performs significantly better across all category splits. Such experimental results demonstrate the superior robustness and generalization capability of our method.

Table 2 shows experimental results while taking base categories into consideration. While achieving good performance for novel categories with limited training samples, Meta-DETR can still detect objects of base categories with competitive performance. TFA [72] produces outstanding performance for base categories since it works more like conventional detectors with fine-tuning, thus having constrained capacity in generalizing on novel categories.

MS COCO. Table 3 shows experimental results on MS COCO. It can be seen that, although MS COCO is more challenging with higher complexity like occlusions and large scale variations, Meta-DETR still outperforms all existing methods for all setups by even larger margins. Specifically, on the primary metric $\mathrm{AP}_{0.5:0.95}$, Meta-DETR outperforms state-of-the-art methods by 5.3% for 10-shot and 8.2% for 30-shot. On the strict metric $\mathrm{AP}_{0.75}$, Meta-DETR almost doubles the state-of-the-art method's performance from 9.8% to 18.5% for 10-shot and from 12.2% to 23.8% for 30-shot. This demonstrates Meta-DETR's precise localization, which is largely attributed to the unified image-level meta-learning that exploits the synergistic effects of localization and classification. Besides, Meta-DETR achieves the best performance for objects of all scales, especially for large objects, largely because Meta-DETR exploits global contexts via image-level predictions effectively.

Except for Average Precision (AP) that directly measures the performance of a detector, Average Recall (AR) is also an important metric. Higher AR indicates less missed detection. As shown in Table 3, Meta-DETR also outperforms the state-of-the-art by large margins regarding AR_{100} ($+8.4\%$ for 10-shot and $+10.0\%$ for 30-shot). It is noteworthy that FSOD [10] achieves the highest AR_{100} among the region-based counterparts, thanks to its meta-learning-based AttentionRPN that generates more accurate region proposals. However, FSOD still suffers from inaccurate or missed detection as it is fundamentally region-based, rely-

Design Choice			Shot		
CCE	SAM	$\mathcal{L}_{\text{SEB}}$	1	3	10
	✓	✓	11.6	37.6	54.2
✓			15.1	39.6	53.2
✓		✓	15.8	40.4	53.4
✓	✓		17.2	43.0	56.7
✓	✓	✓	**17.5**	**45.1**	**57.1**

Table 4. Ablation studies over several design choices of Meta-DETR. Results for novel categories are averaged over multiple runs on the category split 1 of Pascal VOC.

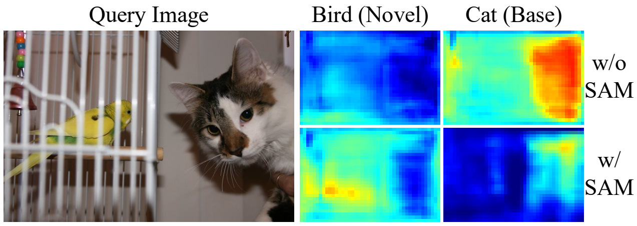

Figure 5. Visualization of correlations between query features and category codes. With semantic alignment mechanism (SAM) introduced, clear responses for both base category (cat) and novel category (bird) are observed, demonstrating SAM's effectiveness in enhancing generalization of meta-learned representations.

ing on high-quality region proposals that are hard to obtain under the few-shot scenarios. In contrast, Meta-DETR fully eliminates region-wise prediction and makes predictions at image level, thus avoiding this constraint and achieving superior performance.

4.4. Ablation Study

We design extensive ablation experiments to study how our designed technical components contribute to the overall few-shot object detection performance.

Effect of Category Code Extractor (CCE). We introduce CCE into SEB to extract object-level instead of image-level information for generating category codes, thus solving the task mismatch issue between the two encoding branches. Another strategy adopted by prior works [22, 81, 80] is to directly use support images with an extra channel representing objects' locations as input. As shown in Table 4, CCE achieves better performance, which shows CCE can effectively filter out redundant information and generate more accurate category codes compared with previous strategy.

Effect of Semantic Alignment Mechanism (SAM). Meta-learning does not aim to learn specific categories. However, with a limited number of base categories, it still inevitably learns category-specific features that only perform well on certain categories and fail to generalize to novel categories. As shown in Table 4, SAM consistently boosts few-shot detection performance for novel categories, which demonstrates its effectiveness in preventing reliance on category-specific features. In Fig. 5, we further visualize the attention maps of correlations between query features and category codes. Without SAM, our method produces strong responses for the base category (cat) with the learned category-specific features, while failing to produce clear responses for the novel category (bird). With SAM included, clear responses are produced for both base and novel categories, which implies that more generalizable representations are learned effectively.

Transfer-Learning		Meta-Learning		Shot		
cls	loc	cls	loc	1	3	10
✓	✓			5.4	21.0	44.8
	✓	✓		11.0	33.9	53.9
		✓	✓	9.8	32.5	52.7
Unified Meta-Learning for cls & loc				**17.5**	**45.1**	**57.1**

Table 5. Ablation studies over the effect of unified meta-learning. Results for novel categories are averaged over multiple runs on the category split 1 of Pascal VOC.

Effect of $\mathcal{L}_{\text{SEB}}$. We introduce $\mathcal{L}_{\text{SEB}}$, which is essentially a conventional cross-entropy loss, to classify the category codes of different categories for better discrimination. As shown in Table 4, $\mathcal{L}_{\text{SEB}}$ slightly but consistently boosts the performance. When there are relatively more training samples for novel categories (10-shot), the performance gain brought by $\mathcal{L}_{\text{SEB}}$ is marginal, which means Meta-DETR already can discriminate novel categories even without $\mathcal{L}_{\text{SEB}}$.

Effect of Unified Meta-Learning. We also study the effect of unified meta-learning in Table 5. Specifically, we make modifications to Meta-DETR to perform separated learning for localization and classification, the two sub-tasks of object detection. Detailed architectures for this study are presented in appendices. As shown in Table 5, unified meta-learning significantly outperforms other design choices, which proves the synergistic effect of the two subtasks. Interestingly, separate meta-learning for both subtasks performs slightly worse than meta-learning classification alone. This can be attributed to the intrinsic difficulty in meta-learning image-level localization with no support from the classification task.

5. Conclusion

This paper presents Meta-DETR, a novel few-shot object detection framework that unifies the meta-learning of object localization and classification at image level. By eliminating the region-wise prediction that is problematic in the few-shot scenarios and effectively leveraging the synergistic relationship between localization and classification, it overcomes the common weaknesses rooted in existing methods. Extensive experiments validate that Meta-DETR establishes new state-of-the-art and outperforms prior works by large margins without bells and whistles.

References

[1] Zhaowei Cai and Nuno Vasconcelos. Cascade R-CNN: Delving into high quality object detection. In *CVPR*, 2018. 2

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 2, 3, 5, 6

[3] Hao Chen, Yali Wang, Guoyou Wang, and Yu Qiao. LSTD: A low-shot transfer detector for object detection. In *AAAI*, 2018. 3, 6, 7

[4] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019. 2, 4

[5] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *ArXiv*, 2003.04390, 2020. 2

[6] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. UP-DETR: Unsupervised pre-training for object detection with transformers. In *CVPR*, 2021. 2

[7] Jian Ding, Nan Xue, Yang Long, Gui-Song Xia, and Qikai Lu. Learning RoI transformer for oriented object detection in aerial images. In *CVPR*, 2019. 2

[8] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Diversity with cooperation: Ensemble methods for few-shot classification. In *ICCV*, 2019. 2

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 6

[10] Qi Fan, Wei Zhuo, Chi-Keung Tang, and Yu-Wing Tai. Few-shot object detection with attention-RPN and multi-relation detector. In *CVPR*, 2020. 1, 3, 7

[11] Chelsea Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017. 2

[12] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 2, 4

[13] Liang-Yan Gui, Yu-Xiong Wang, D. Ramanan, and José M. F. Moura. Few-shot human motion prediction via meta-learning. In *ECCV*, 2018. 3

[14] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *CVPR*, 2019. 1

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 4

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4, 5, 6

[17] Yihui He, Chenchen Zhu, Jianren Wang, Marios Savvides, and Xiangyu Zhang. Bounding box regression with uncertainty for accurate object detection. In *CVPR*, 2019. 2

[18] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *ArXiv*, 2004.05439, 2020. 2

[19] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Cross-view regularization for domain adaptive panoptic segmentation. In *CVPR*, 2021. 1

[20] Jiaxing Huang, Shijian Lu, Dayan Guan, and Xiaobing Zhang. Contextual-relation consistent domain adaptation for semantic segmentation. In *ECCV*, 2020. 1

[21] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *ECCV*, 2018. 2

[22] Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. Few-shot object detection via feature reweighting. In *ICCV*, 2019. 1, 3, 6, 7, 8

[23] Geonuk Kim, Honggyu Jung, and Seong-Whan Lee. Few-shot object detection via knowledge transfer. In *SMC*, 2020. 3

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[25] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, 2015. 4

[26] Tao Kong, Fuchun Sun, Anbang Yao, Huaping Liu, Ming Lu, and Yurong Chen. RON: Reverse connection with objectness prior networks for object detection. In *CVPR*, 2017. 2

[27] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *CVPR*, 2019. 2, 4

[28] Aoxue Li, Weiran Huang, Xu Lan, Jiashi Feng, Zhenguo Li, and Liwei Wang. Boosting few-shot learning with adaptive margin loss. In *CVPR*, 2020. 2

[29] Hongyang Li, D. Eigen, Samuel F. Dodge, Matthew D. Zeiler, and Xiaogang Wang. Finding task-relevant features for few-shot learning by category traversal. In *CVPR*, 2019. 4

[30] Yuewen Li, Wenquan Feng, Shuchang Lyu, Qi Zhao, and Xuliang Li. MM-FSOD: Meta and metric integrated few-shot object detection. *ArXiv*, 2012.15159, 2020. 3

[31] Minghui Liao, Pengyuan Lyu, Minghang He, Cong Yao, Wenhao Wu, and Xiang Bai. Mask TextSpotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(2):532–548, 2021. 2

[32] Matthieu Lin, Chuming Li, Xingyuan Bu, Ming Sun, Chen Lin, J. Yan, Wanli Ouyang, and Z. Deng. DETR for pedestrian detection. *ArXiv*, 2012.06785, 2020. 2

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

[34] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 6

[35] Suiyi Ling, Andreas Pastor, Jing Li, Zhaohui Che, Junle Wang, Jieun Kim, and Patrick Le Callet. Few-shot pill recognition. In *CVPR*, 2020. 1

[36] Bo Liu, Hao Kang, Haoxiang Li, Gang Hua, and Nuno Vasconcelos. Few-shot open-set recognition using meta-learning. In *CVPR*, 2020. 2

[37] Longyao Liu, Bo Ma, Yulin Zhang, Xin Yi, and Haozhi Li. AFD-Net: Adaptive fully-dual network for few-shot object detection. *ArXiv*, 2011.14667, 2020. 3

[38] Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. Deep learning for generic object detection: A survey. *International Journal of Computer Vision*, 128:261–318, 2020. 2

[39] Qing Liu, Orchid Majumder, Alessandro Achille, Avinash Ravichandran, Rahul Bhotika, and Stefano Soatto. Incremental few-shot meta-learning via indirect discriminant alignment. In *ECCV*, 2020. 2

[40] Songtao Liu, Di Huang, and Yunhong Wang. Receptive field block net for accurate and fast object detection. In *ECCV*, 2018. 2

[41] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *ECCV*, 2016. 2

[42] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[43] Tiange Luo, Aoxue Li, Tao Xiang, Weiran Huang, and L. Wang. Few-shot learning with global class representations. In *ICCV*, 2019. 2

[44] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. In *ICLR*, 2018. 2

[45] Khoi Nguyen and Sinisa Todorovic. Feature weighting and boosting for few-shot segmentation. In *ICCV*, 2019. 3

[46] Juan-Manuel Perez-Rua, Xiatian Zhu, Timothy M Hospedales, and Tao Xiang. Incremental few-shot object detection. In *CVPR*, 2020. 1, 3

[47] Limeng Qiao, Yemin Shi, Jia Li, Yonghong Tian, Tiejun Huang, and Yaowei Wang. Transductive episodic-wise adaptive metric for few-shot learning. In *ICCV*, 2019. 2

[48] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L. Yuille. Few-shot image recognition by predicting parameters from activations. In *CVPR*, 2018. 4

[49] Shafin Rahman, Salman Khan, and Fatih Porikli. Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. In *ACCV*, 2018. 2

[50] A. Ravichandran, Rahul Bhotika, and Stefano Soatto. Few-shot learning with embedded class models and shot-free meta training. In *ICCV*, 2019. 2

[51] Joseph Redmon and Ali Farhadi. YOLO 9000: Better, faster, stronger. In *CVPR*, 2017. 2

[52] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 2, 3, 6

[53] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G. Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *CVPR*, 2020. 2

[54] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, 2019. 5

[55] L. Samuelson and L. Smith. They call it like they see it: Spontaneous naming and attention to shape. *Developmental Science*, 8(2):182–198, 2005. 1

[56] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Sharathchandra Pankanti, Rogerio Feris, Abhishek Kumar, Raja Giries, and Alex M Bronstein. RepMet: Representative-based metric learning for classification and one-shot object detection. In *CVPR*, 2019. 2, 6

[57] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, 19(1):221–248, 2017. 1

[58] Bharat Singh and Larry S Davis. An analysis of scale invariance in object detection - SNIP. In *CVPR*, 2018. 2

[59] Bharat Singh, Mahyar Najibi, and Larry S Davis. SNIPER: Efficient multi-scale training. In *NeurIPS*, 2018. 2

[60] L. Smith, S. Jones, B. Landau, Lisa Gershkoff-Stowe, and L. Samuelson. Object name learning provides on-the-job training for attention. *Psychological Science*, 13:13–19, 2002. 1

[61] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017. 2

[62] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *CVPR*, 2019. 2

[63] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris Kitani. Rethinking transformer-based set prediction for object detection. *ArXiv*, 2011.10881, 2020. 2

[64] 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 *CVPR*, 2018. 2

[65] Hung-Yu Tseng, Shalini De Mello, Jonathan Tremblay, Sifei Liu, Stan Birchfield, Ming-Hsuan Yang, and Jan Kautz. Few-shot viewpoint estimation. In *BMVC*, 2019. 3

[66] Joaquin Vanschoren. Meta-learning: A survey. *ArXiv*, 1810.03548, 2018. 2

[67] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3, 5

[68] Oriol Vinyals, Charles Blundell, T. Lillicrap, K. Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, 2016. 2

[69] Guangting Wang, Chong Luo, Xiaoyan Sun, Zhiwei Xiong, and Wenjun Zeng. Tracking by instance detection: A meta-learning approach. In *CVPR*, 2020. 3

[70] Tiancai Wang, Rao Muhammad Anwer, Hisham Cholakkal, F. Khan, Y. Pang, and Ling Shao. Learning rich features at high-speed for single-shot object detection. In *ICCV*, 2019. 2

[71] Tao Wang, Xiaopeng Zhang, Li Yuan, and Jiashi Feng. Few-shot adaptive Faster R-CNN. In *CVPR*, 2019. 3

[72] Xin Wang, Thomas E. Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. In *ICML*, 2020. 3, 5, 6, 7, 13

[73] Xin Wang, Fisher Yu, Ruth Wang, Trevor Darrell, and Joseph E Gonzalez. TAFE-Net: Task-aware feature embeddings for low shot learning. In *CVPR*, 2019. 2

[74] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Meta-learning to detect rare objects. In *ICCV*, 2019. 1, 3, 6, 7

[75] Davis Wertheimer and Bharath Hariharan. Few-shot learning with localization in realistic settings. In *CVPR*, 2019. 3

[76] Jiaxi Wu, Songtao Liu, Di Huang, and Yunhong Wang. Multi-scale positive sample refinement for few-shot object detection. In *ECCV*, 2020. 3, 6, 7

[77] Xiongwei Wu, Doyen Sahoo, and Steven Hoi. Meta-RCNN: Meta learning for few-shot object detection. In *MM*, 2020. 3

[78] Yue Wu, Yinpeng Chen, Lu Yuan, Zicheng Liu, Lijuan Wang, Hongzhi Li, and Yun Fu. Rethinking classification and localization for object detection. In *CVPR*, 2020. 2

[79] Ziyang Wu, Yuwei Li, Lihua Guo, and Kui Jia. PARN: Position-aware relation networks for few-shot learning. In *ICCV*, 2019. 2

[80] Yang Xiao and Renaud Marlet. Few-shot object detection and viewpoint estimation for objects in the wild. In *ECCV*, 2020. 1, 3, 4, 5, 6, 7, 8, 12, 13

[81] 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 *ICCV*, 2019. 1, 3, 5, 6, 7, 8

[82] Xue Yang and Junchi Yan. Arbitrary-oriented object detection with circular smooth label. In *ECCV*, 2020. 2

[83] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *CVPR*, 2020. 2, 4

[84] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. In *ICML*, 2015. 4

[85] Matthew D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014. 4

[86] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. DeepEMD: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *CVPR*, 2020. 4

[87] Gongjie Zhang, Kaiwen Cui, Tzu-Yi Hung, and Shijian Lu. Defect-GAN: High-fidelity defect synthesis for automated defect inspection. In *WACV*, 2021. 1

[88] 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 *WACV*, 2021. 3

[89] Gongjie Zhang, Shijian Lu, and Wei Zhang. CAD-Net: A context-aware detection network for objects in remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 57(12):10015–10024, 2019. 2

[90] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z Li. Single-shot refinement neural network for object detection. In *CVPR*, 2018. 2

[91] Weilin Zhang, Yu-Xiong Wang, and D. Forsyth. Cooperating RPN's improve few-shot object detection. *ArXiv*, 2011.10142, 2020. 1

[92] Minghang Zheng, Peng Gao, Xiaogang Wang, Hongsheng Li, and Hao Dong. End-to-end object detection with adaptive clustering transformer. *ArXiv*, 2011.09315, 2020. 2

[93] Bolei Zhou, A. Khosla, Àgata Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 2

[94] Fengwei Zhou, Bin Wu, and Zhenguo Li. Deep meta-learning: Learning to learn in the concept space. *ArXiv*, 1802.03596, 2018. 4

[95] Peng Zhou, Bingbing Ni, Cong Geng, Jianguo Hu, and Yi Xu. Scale-transferrable object detection. In *CVPR*, 2018. 2

[96] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 2, 3, 4, 5, 6, 12, 13

6. Appendix

This section provides more details of our proposed method and experimental setups, which are omitted in the main paper due to space limitation.

6.1. Detailed Architecture of Meta-DETR

The transformer encoder and decoder in the proposed Meta-DETR have similar setups as Deformable DETR [96]. Concretely, both transformer encoder and decoder have 6 layers and adopt the multi-scale deformable attention module [96] as their attention mechanism. The channel dimension is 256, and the intermediate dimension of fully-connected layers (FC) inside the transformer is 1024. The dropout probability, number of attention heads, and number of object queries are set at 0.1, 8, and 300, respectively.

Fig. 6 shows the architecture of the Aggregator inside the Decoding Branch (DB). The architecture has similar design as FsDetView [80], except that the query features represent whole-image rather than region-level information. Aggregation is conducted between category codes and each position of query features. Fig. 7 illustrates the feed-forward network (FFN) in Decoding Branch (DB) that produces final predictions (omitted for simplicity in Fig. 3 in the manuscript). It consists of a 1-layer MLP for confidence prediction and a 3-layer MLP for box prediction. FFN is shared for all the embeddings that are generated from the transformer decoder.

6.2. Modified Meta-DETR for Ablation Study

In Section 4.4, we modified the proposed Meta-DETR to study the effect of unified meta-learning. In Table 5, *transfer-learning* means that the specific sub-tasks (classification or localization, or both) are learned via naive fine-tuning strategy. Separated *meta-learning* means that the specific sub-tasks are learned via a *standalone* meta-learning-based component. To achieve this, we move the Aggregator after the transformer decoder and perform feature aggregation between category codes and the embeddings generated from the transformer decoder. Therefore, FFN becomes meta-learning-based components for specific sub-tasks, which manages to disentangle the meta-learning for the two sub-tasks. This design enables us to explore the effect of unified meta-learning.

6.3. Detailed Training and Inference Setups

Base Training Stage. All essential setups are provided in Section 4.2. For further details, please refer to our codes.

Few-Shot Fine-Tuning Stage. The few-shot fine-tuning stage shares the same setups as the base training stage, except for the total number of epochs and the learning rate decay epochs. Such differences are due to the significantly smaller number of training samples under the few-shot scenarios, so that more training epochs are required to reach full convergence. Detailed setups are presented in Table 6. These numbers are empirically set solely based on the training loss trajectory, so we expect further performance gain if comprehensive hyper-parameter search is conducted.

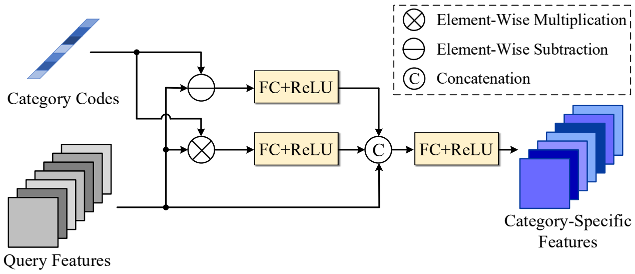

Figure 6. Illustration of the detailed architecture of Aggregator in Decoding Branch (DB). Aggregation is performed between category codes and each position of query features.

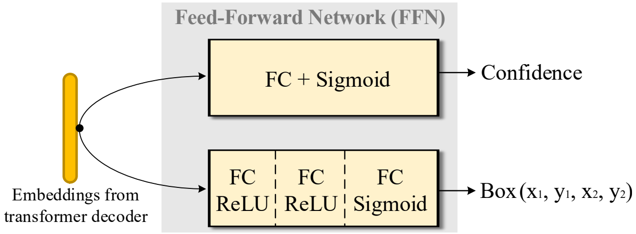

Figure 7. Illustration of the feed-forward network (FFN) in Decoding Branch (DB) to produce final predictions. FFN is shared for all the embeddings generated from the transformer decoder.

Setups	Pascal VOC					MS COCO	
	1	2	3	5	10	10	30
Total Epochs	700	600	600	500	500	500	500
Decay Epochs	600	500	500	425	425	425	425

Table 6. Setups of total number of epochs and learning rate decay epochs for the few-shot fine-tuning stage.

Inference. Given a query image, Meta-DETR produces 300 predictions for each category when performing inference. However, both Pascal VOC and MS COCO accept only 100 predictions per image. We choose the top-scored 100 predictions across all categories as the final predictions.

6.4. Evaluation Metrics

Pascal VOC. For Pascal VOC, mean average precision (mAP) at IoU threshold 0.5 is used as the evaluation metric. In the context of few-shot object detection, mAP is averaged over all novel categories.

Method	single-scale	multi-scale
Deformable DETR [96]	17.8 FPS	11.3 FPS
Meta-DETR (Ours)	11.0 FPS	5.3 FPS

Table 7. Inference speed comparison. Results are obtained using NVIDIA GeForce RTX 2080Ti GPU with single batch size on Pascal VOC.

MS COCO. MS COCO's standard metrics are used for evaluation. Specifically, $\mathrm{AP}_{0.5:0.95}$ is the primary metric that directly measures detectors' performance, which adopts 10 different IoU thresholds to reward detectors with better localization. Standard metrics also include $\mathrm{AP}_{0.5}$ and $\mathrm{AP}_{0.75}$, which correspond to the Pascal VOC metric and a more strict metric, respectively. In addition to average precision (AP), average recall (AR) also serves as an important evaluation metric, which measures the percentage of detected objects among all ground truth objects. Higher AR indicates less missed detection. Concretely, AR_1, AR_{10}, and AR_{100} correspond to AR given 1 detection per image, 10 detections per image, and 100 detections per image, respectively. The MS COCO metrics also evaluate the performance for objects of different sizes (small, medium, and large), including AP_S, AP_M, AP_L, AR_S, AR_M, and AR_L. Similar to Pascal VOC, all these metrics are averaged over all novel categories in our experiments.

Evaluation with Multiple Repeated Runs. More and more researchers have realized that few-shot object detection performance often comes with a large variance. The lower the number of shots, the more unstable the results are. This is because few-shot detection performance relies heavily on the quality of the training samples for novel categories. Therefore, with results from a single run, it is not easy to draw convincing conclusions. To address this issue, following [72] and [80], our results, as reported in Table 1-5, are averaged over multiple repeated runs with different randomly sampled support datasets. Specifically, as we observe large performance variances in Pascal VOC, especially for 1-shot, 2-shot, and 3-shot, all our results on Pascal VOC are averaged over 10 randomly sampled support datasets. For MS COCO, we observe smaller variances with repeated runs, which can be attributed to the larger number of categories and shots. Therefore, we average our results on MS COCO over 5 randomly sampled support datasets.

6.5. Inference Speed of Meta-DETR

During inference, the category codes for all base and novel categories can be computed once and for all. This enables efficient inference of Meta-DETR. Table 7 presents the inference speed of Meta-DETR and Deformable DETR [96]. We can see that Meta-DETR only introduces moderate extra computational costs as compared with the naive fine-tuning approach.

6.6. Qualitative Results

We provide multiple qualitative visualizations of Meta-DETR's few-shot detection results in Figs. 8-15, which give a straightforward illustration of the performance of our method. Note that only detection results of novel categories are presented, as the major focus is to detect objects of novel categories. In addition, we only show results with confidence scores higher than 0.3. White boxes indicate correct detections, red solid boxes indicate false positives, and red dashed boxes indicate false negatives. It can be observed that the proposed Meta-DETR is able to detect novel objects even with scarce training samples. In addition, Meta-DETR performs exceptionally well on large objects and we will investigate how to handle small objects and cluttered objects in our future research.

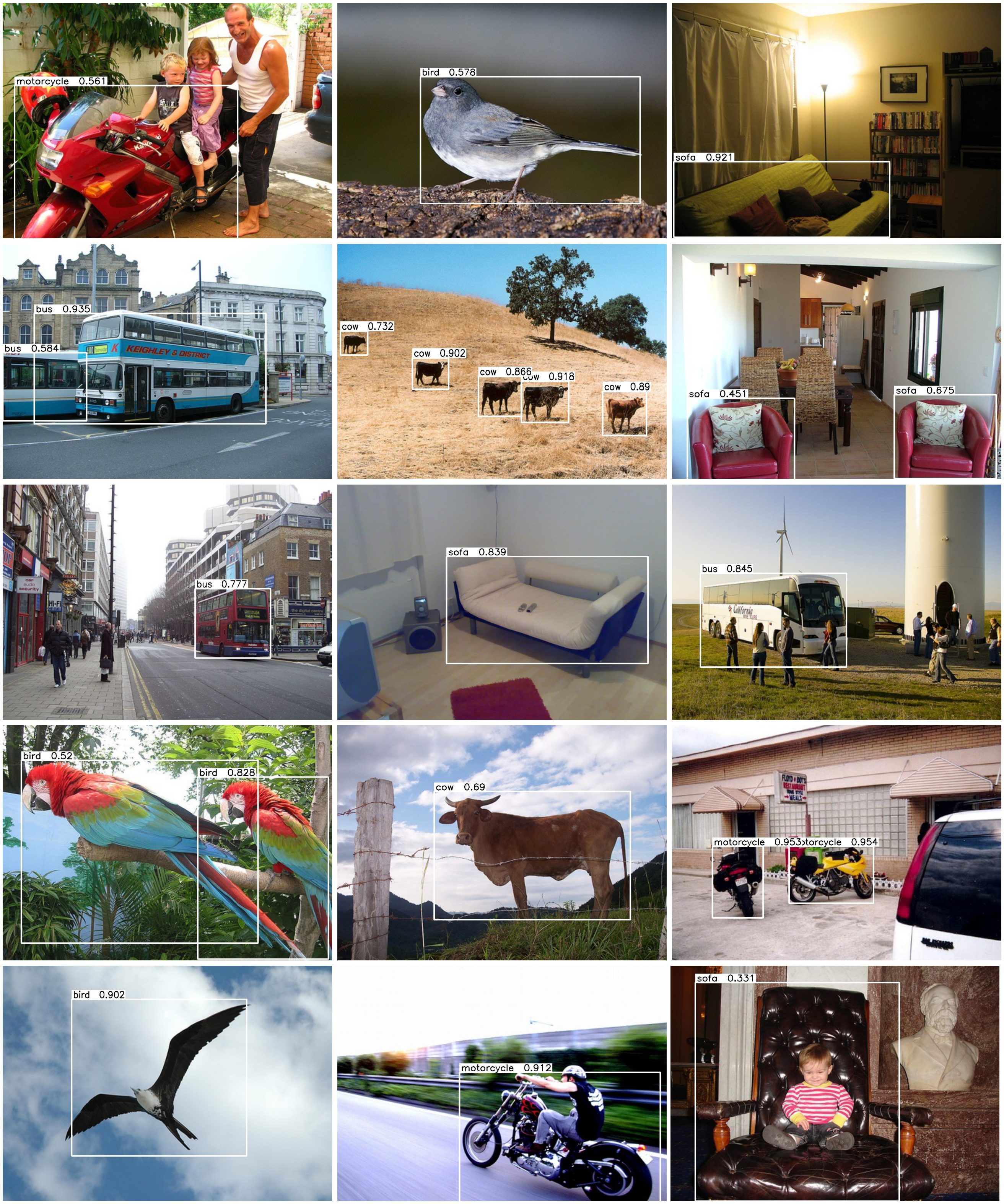

Figure 8. Visualization of multi-scale Meta-DETR's 10-shot object detection results on Pascal VOC category split 1. Novel categories include bird, bus, cow, motorcycle, and sofa. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.

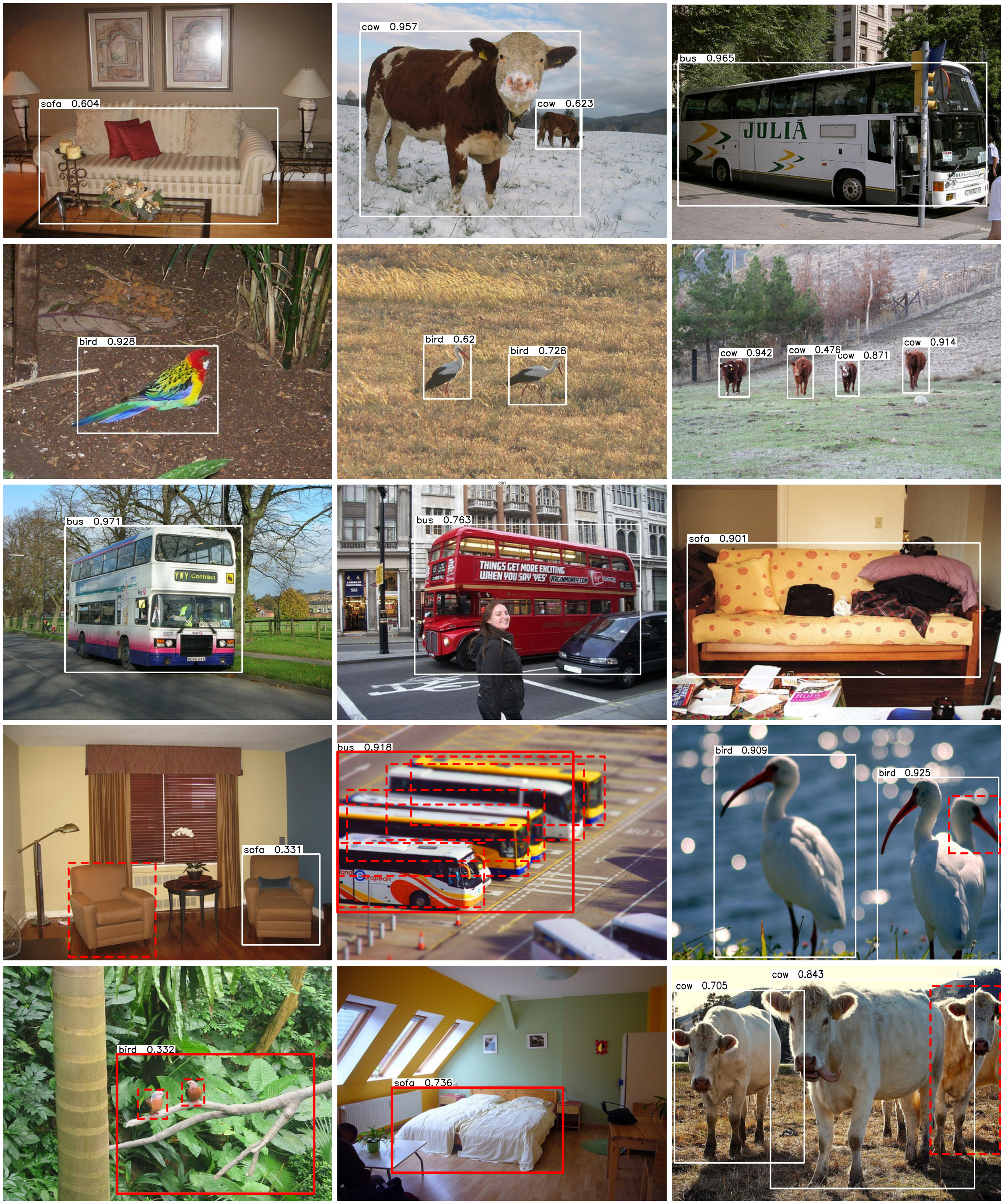

Figure 9. Visualization of multi-scale Meta-DETR's 10-shot object detection results on Pascal VOC category split 1. Novel categories include bird, bus, cow, motorcycle, and sofa. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.

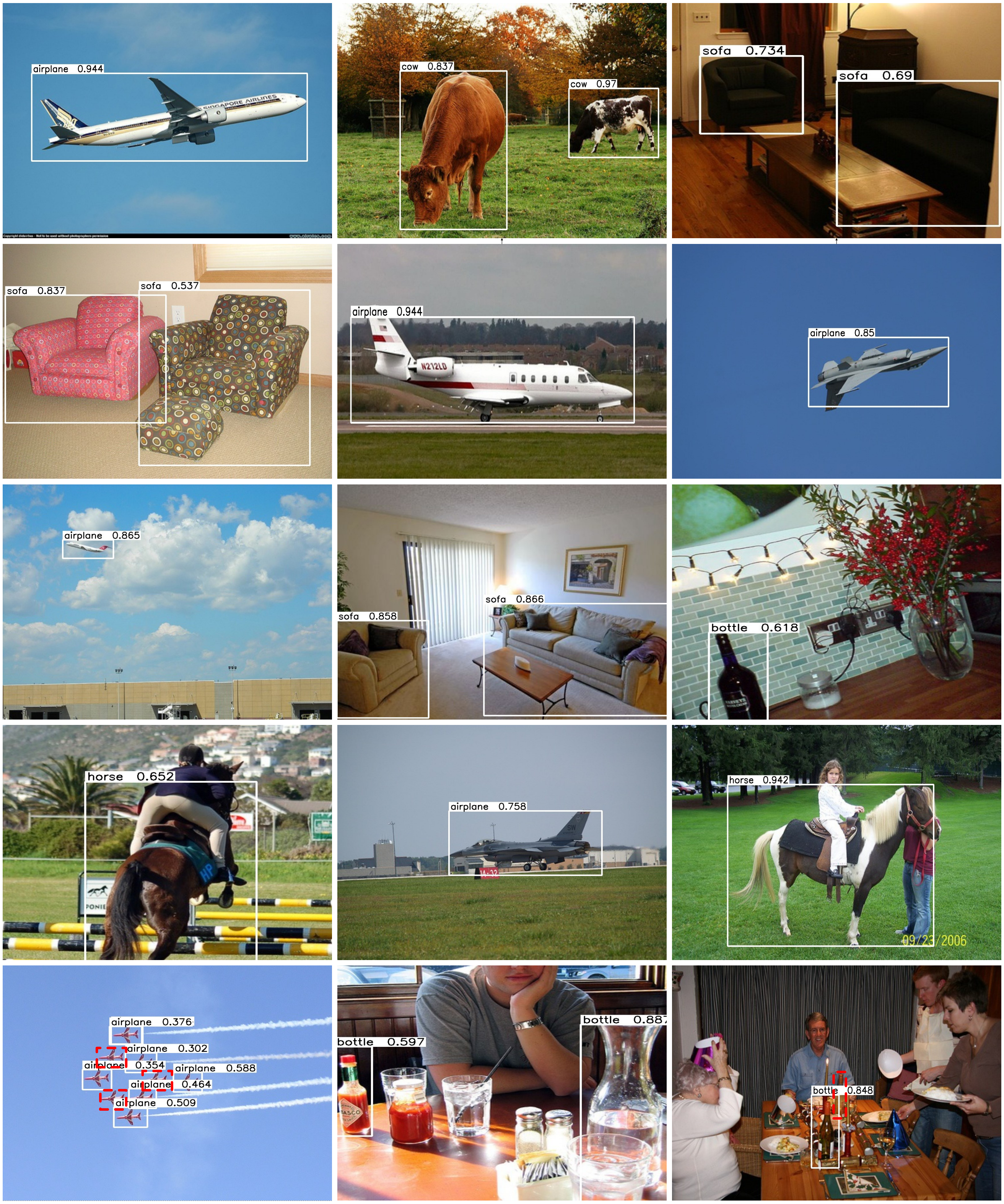

Figure 10. Visualization of multi-scale Meta-DETR's 10-shot object detection results on Pascal VOC category split 2. Novel categories include airplane, bottle, cow, horse, and sofa. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.

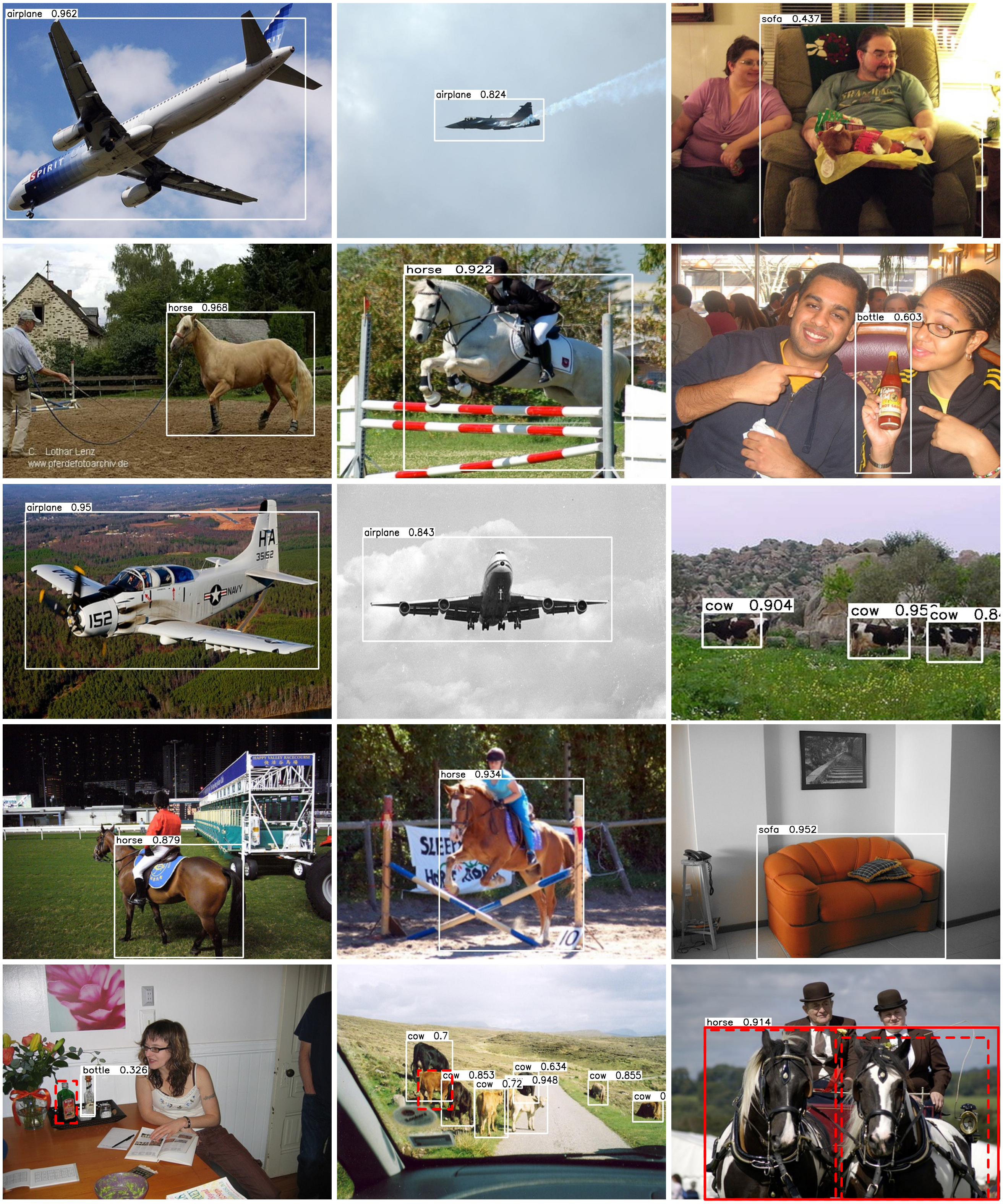

Figure 11. Visualization of multi-scale Meta-DETR's 10-shot object detection results on Pascal VOC category split 2. Novel categories include airplane, bottle, cow, horse, and sofa. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.

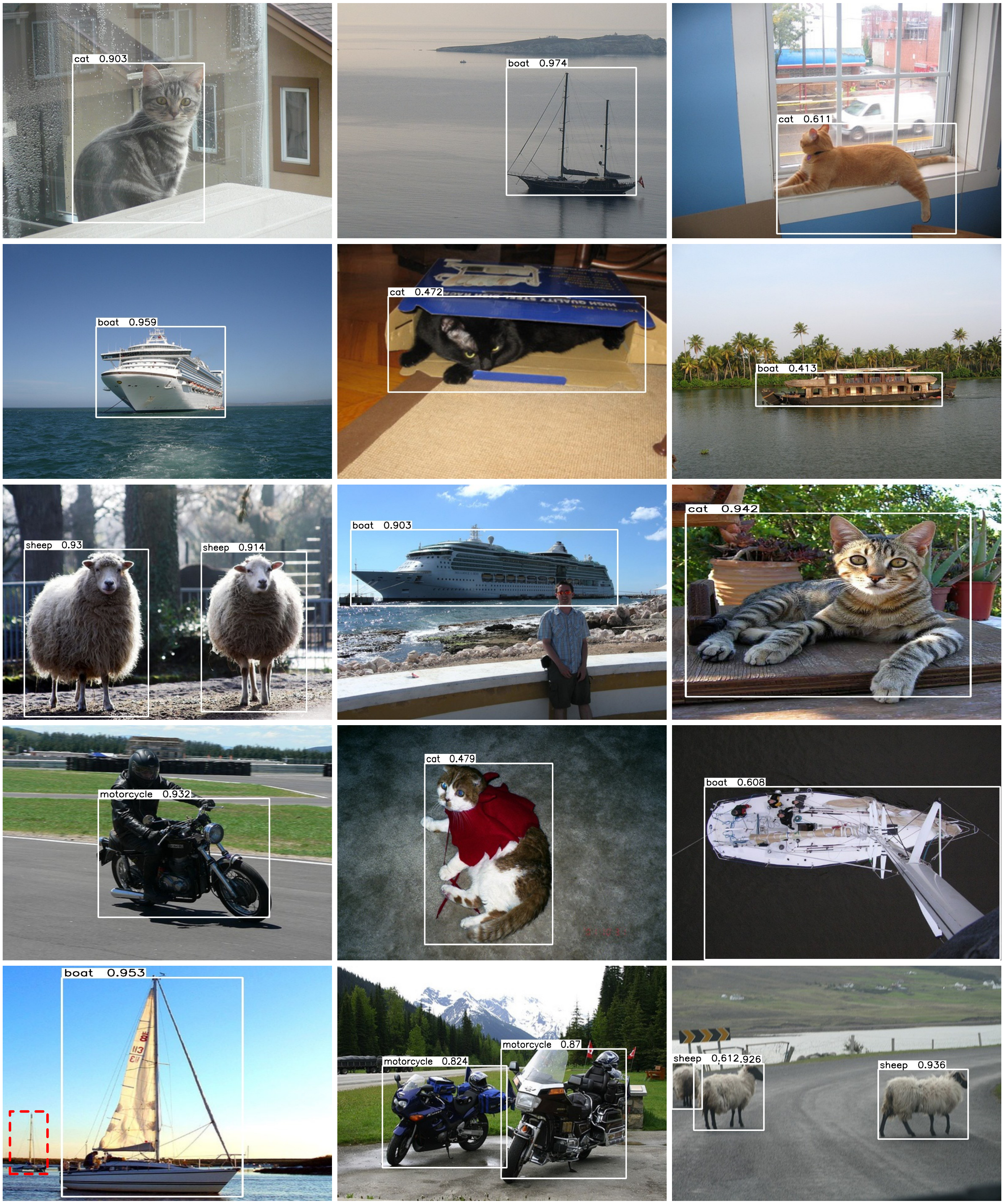

Figure 12. Visualization of multi-scale Meta-DETR's 10-shot object detection results on Pascal VOC category split 3. Novel categories include boat, cat, motorcycle, sheep, and sofa. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.

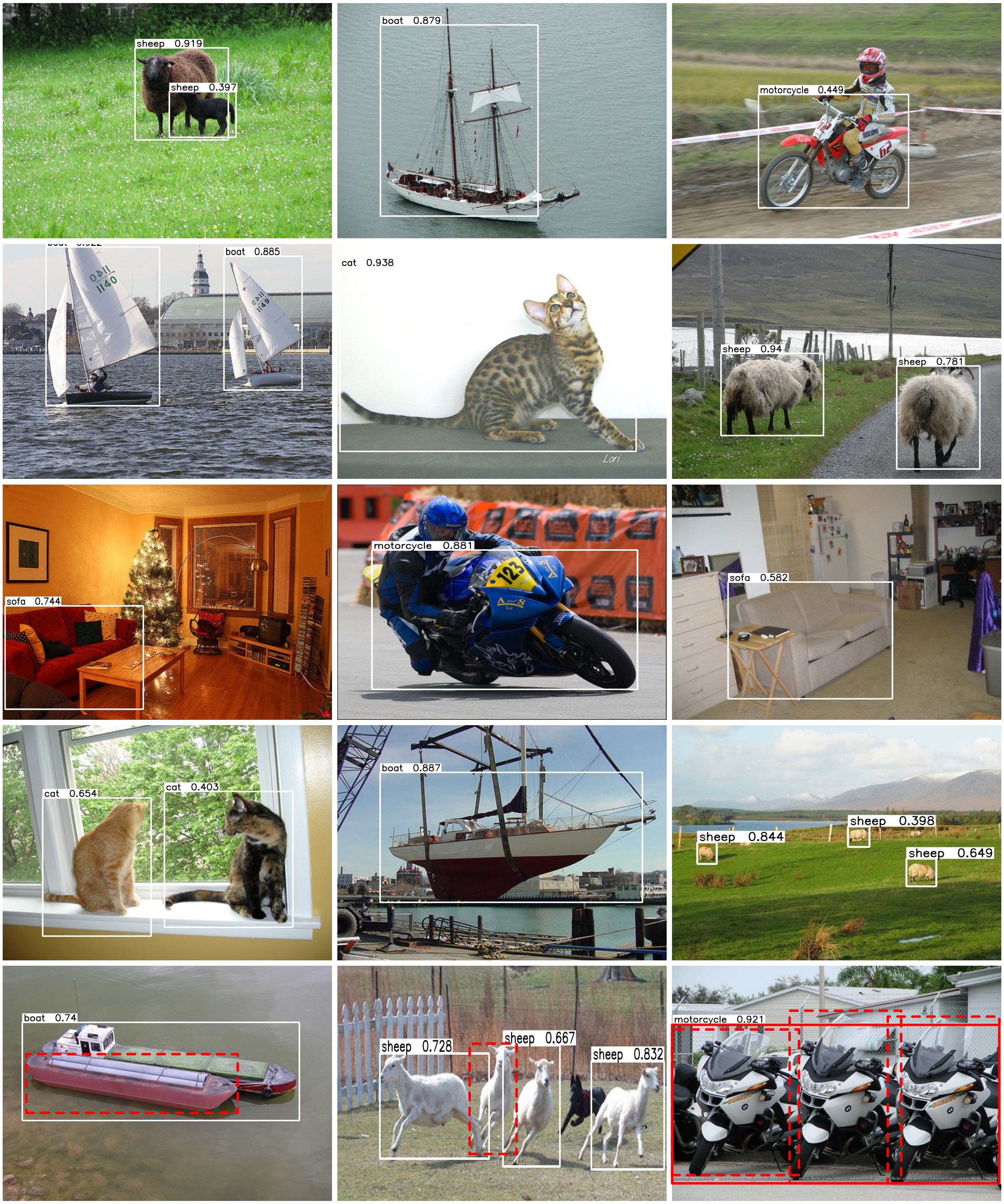

Figure 13. Visualization of multi-scale Meta-DETR's 10-shot object detection results on Pascal VOC category split 3. Novel categories include boat, cat, motorcycle, sheep, and sofa. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.

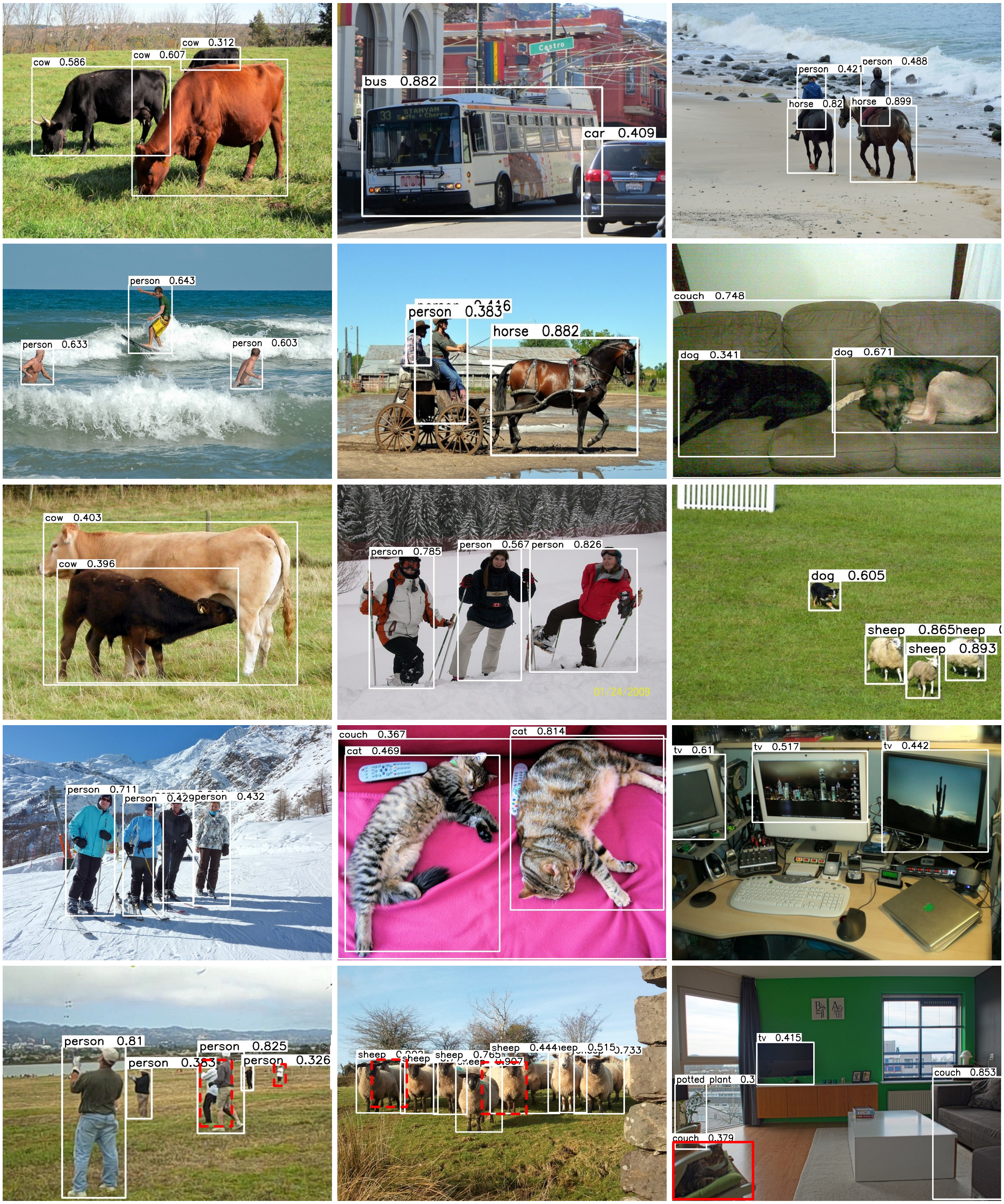

Figure 14. Visualization of multi-scale Meta-DETR's 30-shot object detection results on MS COCO. Novel categories include person, bicycle, car, motorcycle, airplane, bus, train, boat, bird, cat, dog, horse, sheep, cow, bottle, chair, couch, potted plant, dining table, and tv. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.

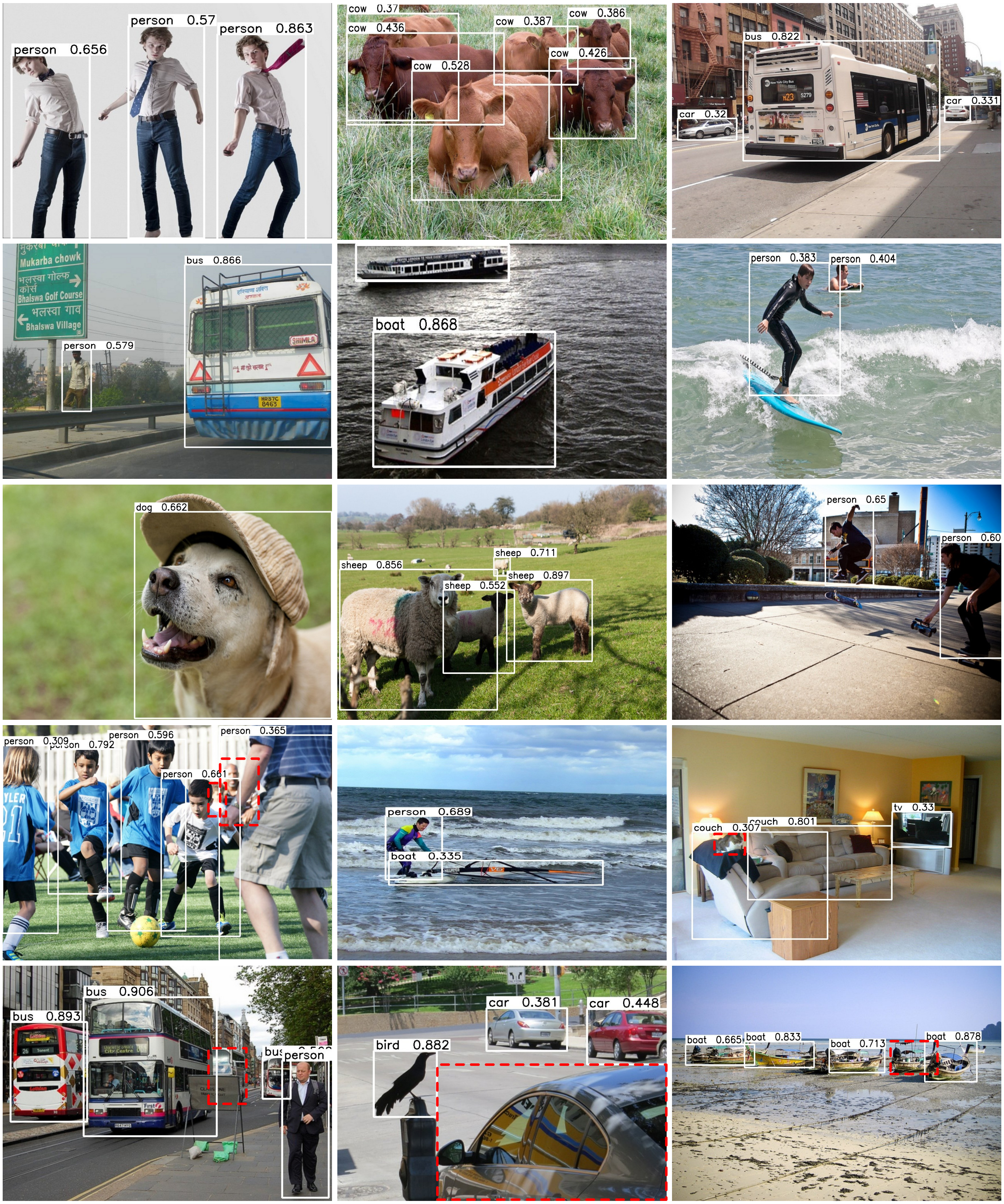

Figure 15. Visualization of multi-scale Meta-DETR's 30-shot object detection results on MS COCO. Novel categories include person, bicycle, car, motorcycle, airplane, bus, train, boat, bird, cat, dog, horse, sheep, cow, bottle, chair, couch, potted plant, dining table, and tv. For simplicity, only results of novel categories are illustrated. White boxes indicate correct detections. Red solid boxes indicate false positives. Red dashed boxes indicate false negatives.