Reference Twice: A Simple and Unified Baseline for Few-Shot Instance Segmentation

Yue Han, Jiangning Zhang, Yabiao Wang, Chengjie Wang, Yong Liu, Lu Qi, Ming-Hsuan Yang, *Fellow, IEEE*, and Xiangtai Li

***Abstract*—Few-Shot Instance Segmentation (FSIS) requires detecting and segmenting novel classes with limited support examples. Existing methods based on Region Proposal Networks (RPNs) face two issues: 1) overfitting suppresses novel class objects and 2) dual-branch models require complex spatial correlation strategies to prevent spatial information loss when generating class prototypes. We introduce a unified framework, Reference Twice (RefT), to exploit the relationship between support and query features for FSIS and related tasks. Our three main contributions are: 1) a novel transformer-based baseline that avoids overfitting, offering a new direction for FSIS; 2) demonstrating that support object queries encode key factors after base training, allowing query features to be enhanced twice at both feature and query levels using simple cross-attention, thus avoiding complex spatial correlation interaction; and 3) introducing a class-enhanced base knowledge distillation loss to address the issue of DETR-like models struggling with incremental settings due to the input projection layer, enabling easy extension to incremental FSIS. Extensive experimental evaluations on the COCO dataset under three FSIS settings demonstrate that our method performs favorably against existing approaches across different shots, e.g., +8.2/+9.4 performance gain over state-of-the-art methods with 10/30-shots.**

***Index Terms*—Computer vision, few-shot learning, instance segmentation.**

I. Introduction

INSTANCE Segmentation aims to detect and segment each object in a scene, which is a core vision task widely used in scene understanding, autonomous driving, and image editing, to name a few. The recent years have witnessed significant success in designing models for a set of pre-defined classes for numerous vision tasks [1], [2], [3], [4], [5], [6]. However, deploying these methods for real-world scenarios is challenging since they are data-hungry, and most approaches need extra mask annotations. As such, numerous Few-Shot Learning (FSL) approaches have been developed. With several labeled data of base classes, FSL aims at learning and predicting novel classes in the given input data (query set of images) with only a few labeled exemplars (support set of images), i.e., FSL learns a conditional model that performs prediction by referring to support images.

To address the issue of lacking instance-wised mask annotation for novel classes, several Few-Shot Instance Segmentation (FSIS) have been developed [7], [8], mainly based on Region Proposal Networks [9]. Despite the promising results, these methods suffer from supervision collapse due to model overfitting. For example, existing RPN-based methods for FSIS tend to predict novel class foreground as the background. Some methods address these issues by mining additional samples [10] and unfreezing more parameters [11]. While [12] utilizes a transformer architecture, it still relies on RPN. Existing methods [13], [14], [15] employ the query-based detection transformer to circumvent RPN overfitting problems in FSOD. Nevertheless, they have not fully exploited the reference cues from the support branch at both feature and query levels in the dual-branch structure. In this work, we develop a query-based model based on mask-transformer model, i.e., Mask2Former [4] to handle FSIS problems. A naive solution is to freeze the model parameters after training on base classes and only fine-tune the class-specific ones on novel classes, similar to the fine-tuning based approaches using RPN-based models [16], [17]. Although this approach performs better than RPN-based methods, we observe that misclassification occurs among many semantically correlated classes on both low level and high level, as shown in Fig. 2.

To solve the issues mentioned above, we analyze the dual-branch architecture as shown in Fig. 1(a), where it typically extracts the class prototypes from support images to guide the query branch detector by feature aggregation [7], [8], [18]. The positions of the feature aggregation module can be summarized before and after the RPN, and the operation methods include channel-wise multiplication, addition, subtraction, and concatenation. A few methods [12], [19], [20] show that pooling operation would lead to complete loss of spatial information when extracting class prototypes. Thus, they directly use feature maps of the two branches to capture spatial correlation via the attention mechanism. These methods require the elaborate design of feature processing methods to avoid under-exploring visual cues from support data. One question ensues: *How to design an effective framework better to leverage the support guidance for query-based methods for FSIS?*

Manuscript received 10 October 2023; revised 2 May 2024; accepted 15 June 2024. Date of publication 1 July 2024; date of current version 5 November 2024. This work was supported by the Science and Technology Planning Project of Zhejiang Province, China under Grant 2024C01172. Recommended for acceptance by M.-M. Cheng. *(Yue Han and Jiangning Zhang contributed equally to this work.) (Corresponding authors: Yong Liu; Jiangning Zhang.)*

Yue Han and Yong Liu are with the Institute of Cyber-Systems and Control, Advanced Perception on Robotics and Intelligent Learning Lab (APRIL), Zhejiang University, Hangzhou 310027, China (e-mail: yongliu@iipc.zju.edu.cn).

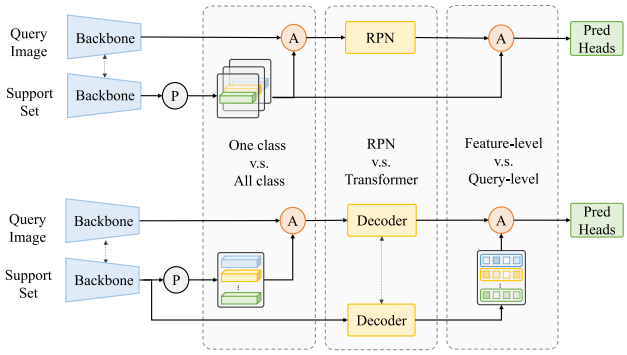
Jiangning Zhang, Yabiao Wang, and Chengjie Wang are with Youtu Lab, Tencent, Shenzhen 518057, China (e-mail: vtzhang@tencent.com).

Lu Qi and Ming-Hsuan Yang are with the Department of Computer Science and Engineering, University of California Merced, Merced, CA 95343 USA.

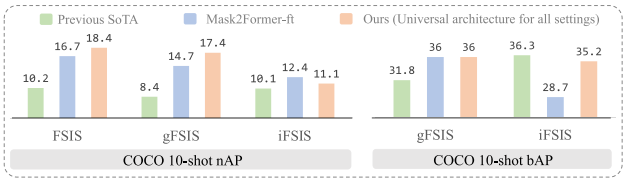
Xiangtai Li is with Tiktok, Singapore 048583.

Source code and models will be available at this github site.

Digital Object Identifier 10.1109/TPAMI.2024.3421340



(a) RPN-based dual-branch framework (Top) v.s. Our Reference Twice framework (Bottom)

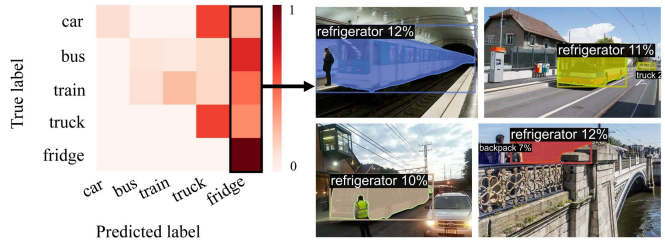


(b) Performance on COCO

Fig. 1. (a) Existing RPN-based dual-branch framework and the proposed mask-transformer-based framework. Our method better utilizes the support set on feature and query levels, with only one forward pass handling all classes. (b) Performance on COCO 10-shot. Our unified baseline performs favorably in all settings. Here, P in the circle denotes RoI align or mask pooling, and A in the circle denotes aggregation operation.

In this work, we examine the properties of Mask2Former after base training and identify two critical factors that help design a solution to leverage the visual cues of support samples better. First, we show that object queries from the support branch (with ground truth mask input) can locate objects for novel classes well, even *without* the fine-tuning process on novel classes. We term this as *support query localization*. Second, we calculate attention maps between object queries from support images and object queries from query images. We show that without fine-tuning, most object queries are highly correlated, even for most novel classes. We term this as *support query categorization*. Thus, we propose query-level feature aggregation using object queries that encode high-level instance-wise categories and positioning information to complement feature-level feature aggregation using class prototypes as used in prior works [21].

Motivated by the above findings, we present a *Reference Twice* (RefT) model better to exploit the support mask information and support object queries, as shown in Fig. 1(b). RefT adopts the Meta-Learning framework [7] with a two-stage pipeline, learning to quickly generalize to novel knowledge by referencing twice from the support branch. For the first reference, we adopt mask pooling to crop support features to generate class prototypes in a way similar to prior works. The difference is that we feed prototypes of all support classes simultaneously, unlike most existing methods that require multiple passes with one class at a time. Comparing multiple classes simultaneously helps alleviate the problem of misclassification (see Fig. 2). For the second reference, as shown by *support query categorization* and *support query localization*, object queries from support images already encode relevant classification and localization information, and thus we propose a multi-head attention module



(a) Confusion Matrix (b) Misclassification Examples

Fig. 2. Baseline results. (a) Confusion matrix and (b) visualization results of several semantically similar classes on COCO minival set (K = 10) from the fine-tuned Mask2Former.

to link both object queries, which enhances the classification and segmentation ability for novel classes. Both reference processes are well coupled and significantly improve segmenting instances from novel classes.

We note several works tackle new tasks with more stringent constraints based on FSIS, i.e., generalized FSIS (gFSIS) [22] and incremental FSIS (iFSIS) [17] using specific designs. Another question arises: *Is there a simple yet unified framework for FSIS, gFSIS, and iFSIS?* With our framework, RefT performs well in both FSIS and gFSIS, and we introduce a class-enhanced base knowledge distillation to address the issue of adaption to the incremental settings. This enables our framework to easily extend to incremental FSIS (Section III-C).

We summarize the contributions of this work as follows:

- We analyze the mask-transformer-based model for FSIS and identify two key factors, support query localization, and support query categorization, which are important for guiding the transformer framework design.
- We propose the RefT framework, which contains two aggregation modules at the feature and query level. We use cross-attention as a unified feature aggregation operation at both levels, thereby eliminating the need for complex spatial correlation interaction between features and better use of support features.
- We introduce a class-enhanced base knowledge distillation module to address model adaption issues and facilitate RefT for incremental FSIS.
- Extensive experimental evaluations on the COCO dataset under three FSIS settings demonstrate that our method performs favorably against existing approaches across different shots, e.g., $+8.2/+9.4$ performance gain over state-of-the-art methods with 10/30-shots.

II. Related Work

Few-Shot Classification: A variety of techniques have been devised to extend base models for classifying new classes using limited samples. Refs. [23], [24], [25], [26], [27], [28], [29], [30], [31], [32]. Existing approaches typically employ the N-way K-shot episodic training paradigm that helps adapt to multiple classification tasks. These approaches can be mainly based on optimization, e.g., meta learners [18], [33], [34] and metrics, e.g., transferable embeddings [35], [36], [37], [38]. Various embedding methods and distance functions are explored, including extracting categorical prototypes with a fixed distance metric

(cosine or Euclidean [21], [35], [36]), utilizing task-adaptive embedding functions with a learned distance metric [37], [38] and attention modules [39], [40]. Instead of studying the image-level classification task, we focus on instance-level few-shot learning.

Few-Shot Object Detection: A number of approaches enlarge the vocabulary of a detector with few samples [41], [42], [43], [44], [45], [46], [47], [48], [49]. TFA [16] proposes a two-phase fine-tuning approach, while DeFRCN [11] decouples the training of RPN features and RoI classification. SRR-FSD [50] combines multi-modal inputs and LVC [10] proposes a pipeline to enlarge novel detection examples for training a more robust model. Meta-DETR [13] uses a Correlational Aggregation Module (CAM) for simultaneous aggregation between query features and support class prototypes. While our method bears some similarties to this method, there are two main differences: (1) The support set is organized differently. Meta-DETR splits all classes into several class sets and requires multiple forward passes, while RefT deals with all classes at one time. (2) CAM entirely assumes the role of class matching and the sigmoid binary classification head outputs whether there is a match. This approach does not rely on embedding vectors before the classification head to capture class representative features. Therefore, it does not require embedding-based matching for classification and has better generalization performance, especially in low-shot scenarios. However, without class representative features, the model may forget the base classes more easily. This makes it unsuitable for generalized and incremental learning settings. To accommodate different few-shot learning settings for segmentation, RefT leverages the first reference module to filter the correct class features and employs the softmax classification head to perform class matching.

Few-Shot Semantic Segmentation: These approaches require accurate pixel-level classification of query images based only on a limited number of labeled samples [51], [52], [53], [54], [55], [56], [57], [58]. To tackle this problem, prototypical feature learning and affinity learning approaches have been developed. Prototypical feature learning methods [59], [60], [61], [62], [63], [64], [65] condense masked support features into single or multiple prototypes, while affinity learning schemes consider fine-grained pairwise relationships between support and query features. However, relying solely on prototypes can lead to information loss and performance loss, while pixel-level correlation in affinity learning methods may suffer from false matches caused by intra-class variations and cluttered backgrounds. Recent methods [64], [65] have highlighted the limitation of using a single prototype to cover all regions of an object, particularly for pixel-wise dense segmentation tasks. To overcome this issue, affinity learning schemes [65], [66] mine dense correspondence between the query images and support annotations, thereby supplementing more detailed support context. AGNN [67] introduces attentive graph to thoroughly examines fine-grained semantic similarities between all the possible location pairs in two data instances.

Few-Shot Instance Segmentation: Existing approaches can be divided into single-branch and dual-branch architectures. The former [17], [68] focuses on the design of the classification head, while the latter [7], [8], [69] introduces an additional support branch to compute class prototypes or re-weigh vectors of support images to select target category features via feature aggregation. For example, Meta R-CNN [7] performs channel-wise multiplication on features belonging o region of interest, while FGN [8] aggregates channel-wise features at three stages, including RPN, detection head, and mask head. Several recent methods [10], [11], [13] show that RPN-based approaches tend to mistake the novel class objects as the background. In contrast, RefT builds on a query-based detector to avoid this issue.

Generalized Few-Shot Object Detection and Instance Segmentation: Several approaches take knowledge contained base classes into account for generalized few-shot instance segmentation (gFSIS). In [16] a replay strategy is used and an equal number of examples from base and new classes are used for fine-tuning to avoid class imbalance issues. However, the problem of forgetting base classes remains unaddressed. Several methods tackle the catastrophic forgetting issue [17], [22] by weighing base classes to retain knowledge, but suffer from performance loss in generalization. In contrast, RefT exploit reference information from the support set to address both generalization to novel classes and forgetting of base classes.

Incremental Few-Shot Object Detection and Instance Segmentation: Similar to the generalized setting, the incremental few-shot instance segmentation (iFSIS) also requires balancing the model performance on both base and novel classes. However, the incremental setting imposes more stringent constraints due to practical considerations such as data security and resource consumption. The learned base model does not have access to the training data, and fine-tuning can only be performed on novel class samples. RPN-based methods such as iMTFA [17] and iFS-RCNN [68] prevent catastrophic forgetting and adapt to the incremental setting by freezing all parameters except for the classification head. Specifically, iMTFA employs a cosine similarity classifier and models the novel class by averaging the embedding vectors of all previous classes; and iFS-RCNN proposes a logit classifier based on Bayesian probabilities to discriminate between base and novel classes.

In recent studies, the use of transformer, i.e. DETR, for incremental tasks has been explored [15]. However, it has been observed that designing the classification head alone is inadequate. In addition to the classification head, input projection layer is also a class-specific layer, which is referred to as class adaptation layer (CAL) in this paper to emphasize its significance in novel class learning. Without fine-tuning CAL, the model cannot learn novel classes, but fine-tuning CAL will cause serious forgetting issues. Incremental-DETR [15] addresses this by performing feature distillation on CAL, logits distillation on the classification head, and using selective search to provide pseudo labels to improve performance further. We show that CAL is the crucial factor in addressing the problem. Thus, we focus on CAL and propose a class-enhanced base class knowledge distillation module to tackle the problem.

Open-Vocabulary Methods: Open-Vocabulary learning [70], [71], [72], [73] increase their vocabulary size for object detection [74], [75], [76], [77], [78], [79], [80], [81], [82], [83], [84], [85], [86], [87], [88] or instance segmentation [89], [90] tasks by leveraging knowledge from pre-trained vision-language models.

TABLE I
DIFFERENT SETTINGS

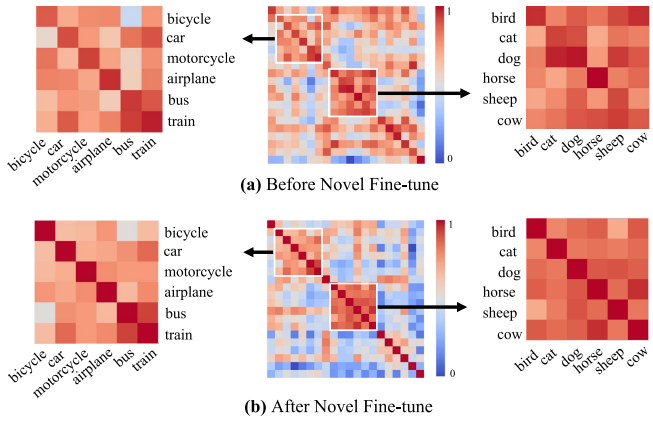
Settings	Fine-tune on		Test on	
	Base	Novel	Base	Novel
FSIS	✓	✓		✓
gFSIS	✓	✓	✓	✓
iFSIS		✓	✓	✓

FSIS, gFSIS, iFSIS: standard, generalized, incremental few-shot instance segmentation.

TABLE II
SUPPORT QUERY LOCALIZATION

# Top-k queries	Base Training		Novel Fine-tune	
	bIoU	nIoU	bIoU	nIoU
50	0.54	0.47	0.59	0.55
30	0.70	0.64	0.75	0.71
10	0.84	0.79	0.84	0.79

We analyze the mask quality of about 20 novel classes on COCO dataset and calculate the top-k IoU between the ground truth and predicted masks of the support branch. Object queries from support images can locate object masks for novel classes, even without novel fine-tuning. nIoU/ bIoU: average IoU of novel/ base class.



(a) Before Novel Fine-tune

(b) After Novel Fine-tune

Fig. 3. Support Query Categorization. We visualize the cosine similarity of object queries of the support branch belonging to COCO 20 novel classes. Most object queries are roughly distinguishable, even without fine-tuning. We zoom in on areas that contain highly correlated and easily misclassified classes.

Recent efforts are dedicated to exploring knowledge distillation techniques, including the use of pre-trained CLIP [91] and pseudo-labeling [92], [93], [94]. While both open-vocabulary and few-shot settings share the same goal of accommodating novel classes, the few-shot setting specifically focuses on utilizing limited examples to learn novel classes.

Foundation Models: Numerous foundation models have been proposed for vision, language, and multi-modal tasks [95], [96], [97], [98], [99], [100], [101], [102] with state-of-the-art generalization capability in zero-shot scenarios. Recently, Segment Anything (SAM) [103] develops a sophisticated data engine to collect 11 million image-mask data and trains a segmentation foundation model. This model introduces a novel segmentation paradigm based on prompts, including points, boxes, masks, and free-form texts, However, SAM does not inherently segment specific visual concepts, like precise parts in anomaly detection tasks (e.g., gears in industrial quality control) [104], personalized segmentation (with a specific individual) [105], [106], [107], [108] and specific organs in medical imaging (e.g., tumors in MRI scans) [109]. We note that SAM and RefT address *different aspects* of the segmentation problem. RefT focuses on solving the semantic matching challenge between limited novel instance annotations and the objects to be segmented. In contrast, SAM is a class-agnostic segmentation model that outputs binary masks *without* considering semantics. Thus, it cannot directly handle few-shot segmentation tasks like RefT, which benefits from its strong emphasis on semantic matching.

III. METHOD

We first introduce preliminaries, including settings, baselines, and support query localization and categorization. Next, we introduce our RefT framework and its application for FSIS and related tasks.

A. Preliminaries

Problem Settings: For the instance segmentation tasks considered in this work, the object classes are split into C_{Base} and C_{Novel} classes, where $C_{Base} \cap C_{Novel} = \varnothing$, $C_{Base} \cup C_{Novel} = C_{All}$. FSIS aims to segment objects belonging to C_{Test} in a query image after training over abundant samples of C_{Base} and a few samples of $C_{Finetune}$. RefT can be applied to all three settings for FSIS in the literature. As shown in Table I, for FSIS, $C_{Finetune} = C_{Base} \cup C_{Novel}$, $C_{Test} = C_{Novel}$. For gFSIS, $C_{Finetune} = C_{Base} \cup C_{Novel}$, $C_{Test} = C_{Base} \cup C_{Novel}$. For iFSIS, $C_{Finetune} = C_{Novel}$, $C_{Test} = C_{Base} \cup C_{Novel}$.

Mask2Former FSIS Baseline: In this work, we use the Mask2Former [16] as our baseline. In the first stage, we train the model on base classes with training samples. In the second stage, we fix most of the parameters and only fine-tune the class-specific layers, including the input projection layer, object queries, and the class head. For incremental setting, we use the same settings [16] and replace the class head with a Cosine-similarity classifier. As shown in Fig. 2, the simple fine-tuned baseline tends to misclassify semantically similar classes and confuse classes with similar shapes but completely different backgrounds.

Support Query Localization: After the base training stage, we directly infer on support images of novel classes to determine whether the model can recall novel classes with additional structural input. As shown in Table II, we find that the model is already capable of detecting and segmenting novel class objects after base training. This indicates that most queries have enough localization information for novel classes. Furthermore, we observe an improvement in the segmentation accuracy of more queries after the novel fine-tuning stage.

Support Query Categorization: We show the correlation maps among the support queries in Fig. 3, where most of these novel classes are highly correlated (more examples of correlation maps are presented in the supplementary materials). These correlation maps reveal *support query categorization* for novel classes as the support queries have the clustering effect even without fine-tuning. As such, these maps may be useful for query image

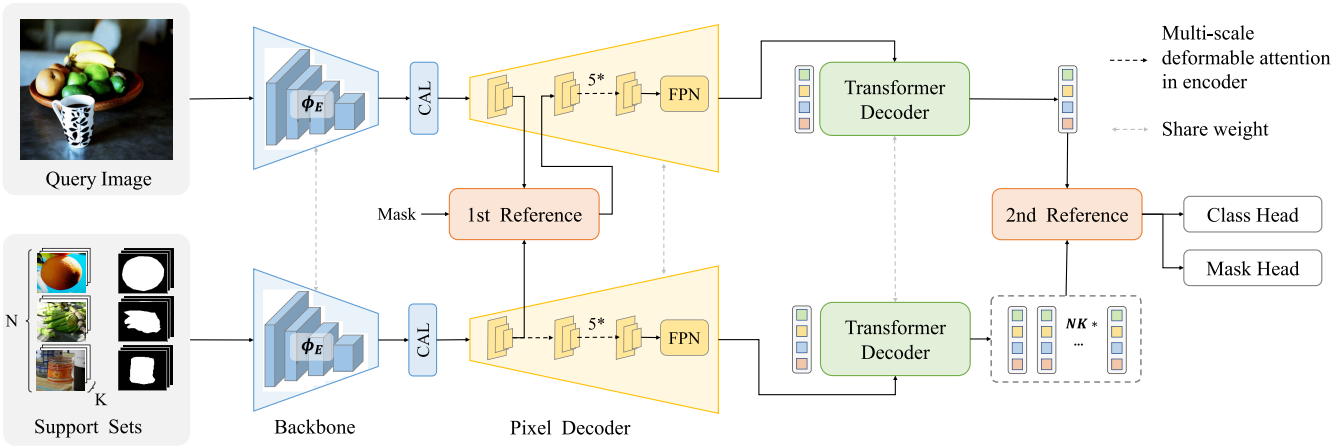


Fig. 4. Architecture of the proposed *Reference Twice* (RefT) for FSIS. The query branch refers to the support branch twice on the feature and query level. The first reference for feature-level enhancement performs simultaneous aggregation between the query features and all adaptive class prototypes obtained through mask pooling. The seoncd reference module for query-level feature aggregation links object queries from the query and support branch.

branch training. In our framework, as shown on the right side of Fig. 3, the categorization is more prominent.

B. Reference Twice

Motivation: As discussed above, we aim to enable the model to exploit the cues from the support queries. Unlike prior works [15] that only exploit features for segmentation, we operate on the support queries as they can encode localization and categorization information.

Overview: For illustration, we use a query image I_Q and a support set S with $N \times K$ examples as input, as shown in Fig. 4. For the support branch, we mask out the support objects and drop the background together with other objects in the image. As such, the support features and object queries are more discriminative and contain more accurate relevant information. A shared feature extractor first encodes the query and support images into the same feature space. Subsequently, the *first reference aggregation* module performs simultaneous aggregation between the query and support features. In this step, the query features are coarsely filtered by support categories. Then the selected query features and support features are sent to the class-agnostic pixel decoder and transformer decoder to obtain object queries from both the query and support branches. Next, we consider object queries of both branches for better calibration via cross-attention. As we mask the support images with the ground truth mask, the obtained support object queries correspond precisely to the instances of the support categories.

First Reference for Feature-Level Enhancement: As shown in Fig. 5, we first use the Mask2Former encoder to extract multi-scale features. Given the multi-scale features from the query image $F_Q=\{x_Q^l\}_{l=1}^L$ and support features $F_S^{nk}=\{x_S^l\}_{l=1}^L$ $(n=1,\ldots,N, k=1,\ldots,K)$ (where L denotes the feature level), a weight-shared multi-head deformable attention [110] first encodes them into the same feature space, obtaining F'_Q and F'_S. The features of support instances are separated from the background and other instances through mask pooling with ground truth masks on the support features of each scale, respectively. Then, the adaptive class prototypes for all support

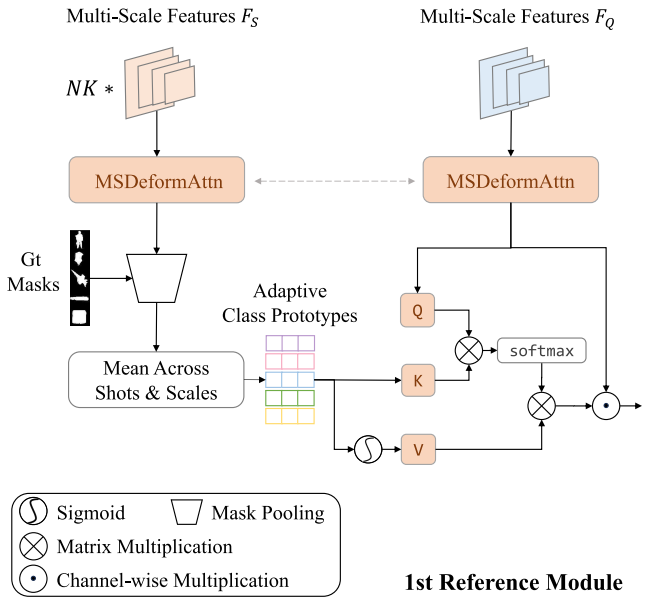


Fig. 5. First Reference Module for feature-level enhancement performs simultaneous aggregation between the query features and all adaptive class prototypes obtained through mask pooling.

classes are obtained by averaging all scales per image and K examples per class

$$c^n = \frac{1}{KL}\sum_{k=1}^{K}\sum_{l=1}^{L}\text{MaskPool}\left(F_S^{nk}\right), \quad n=1,\ldots,N. \tag{1}$$

After that, a multi-head attention module is used to generate the reweighting matrix for aggregating F'_Q with adaptive class prototypes $P=[c^1,\ldots,c^N]\in R^{N\times d}$

$$R = \text{softmax}\left(F'_Q P^\top\right)\sigma(P). \tag{2}$$

Here, the linear projection is omitted for simplicity. The query features are then multiplied with the obtained weights along the channel dimension as below

$$F_Q^{Enhanced} = F'_Q \odot R. \tag{3}$$

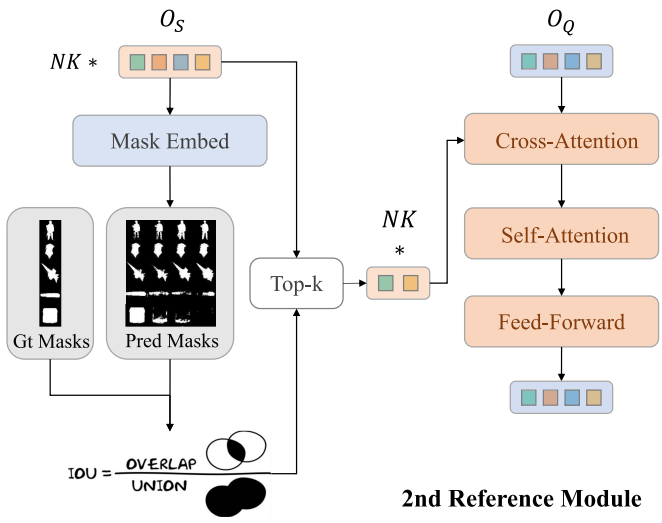


Fig. 6. Second Reference Module for query-level feature aggregation links object queries from the query and support branch.

Thus, the category-related query features are selected and enhanced. In this operation, the query branch is enhanced by support examples adaptively. We present the detailed design of choices of support features in the experiment part.

Second Reference for Query-Level Enhancement: Given the facts in (Section III-A), we add an extra query-level enhancement module to fully leverage the relevant classification and localization information of the support set. We first discard support object queries with irrelevant information and only keep ones that contain instance-level category and spatial information of high quality. The detailed process is shown in the bottom right of Fig. 4 and 6. Given query object queries $q_Q \in \mathbb{R}^{Q\times D}$ and support object queries $q_S \in \mathbb{R}^{NKQ\times D}$, Q is the number of object queries per image, and D is the feature dimension, we first obtain the predicted masks corresponding to the support object queries. Then, the top k out of the Q object queries per image are selected according to the IoU of the predicted and ground truth masks. As such, we avoid huge computation costs and also obtain better relevant cues to improve performance. We then use a multi-head cross-attention module to match the query and support object queries as follows:

$$q_Q^{Enhanced} = \text{softmax}\left(q_Q \,\text{TopK}(q_S)^\top\right)\text{TopK}(q_S). \quad (4)$$

A multi-head self-attention module followed by one feed-forward layer is used to adapt to the following prediction heads. Such simple multi-head attention is good enough to link support queries to the object queries from query images, as discussed in Section IV-E.

Discussions: Existing methods typically rely on carefully designed spatial correlation interaction and channel aggregation to extract accurate class prototypes from features. We do not perform query-level aggregation and keep this feature-level aggregation in the first reference module. The reason is that object queries encode intertwined classification and localization information, unlike channel-discriminative class prototypes. In this work, we only need to filter the correct class features at this stage. We use query-level aggregation in the second reference module to complement the spatial information loss in prototypes caused by pooling. Compared with other feature-level aggregation methods, the differences are for efficiency and adaptivity. For efficiency, we aggregate with one-forward-all-class via cross-attention, while previous dual-branch methods handle one-forward-one-class. For adaptivity, during each iteration, the classes present in the support set change adaptively based on the classes appearing in the query image. We sample negative classes (classes that do not appear in the query image) and include them in the support set to mimic the scenario in the inference stage where prototypes of all classes are available including both positive and negative ones.

C. Generalization and Training Details

FSIS and gFSIS: The common practice in prior RPN-based work is to freeze most parameters and fine-tune the class head on a balanced sampled dataset of base and novel classes to retain base class knowledge. However, as observed in [15], the DETR-like detector can barely generalize to novel classes with the input projection layer frozen. We show the layer is class-specific as it transforms the channel dimension with one convolution, and channel features are essential in distinguishing different classes. To emphasize the significance of the layer in novel class learning, we term it as Class Adaptation Layer (CAL), as shown in Fig. 4. We also alleviate the catastrophic forgetting problem by simply fine-tuning CAL, object queries, and the class head with all other parameters frozen and without additional modifications. The same losses in Mask2Former for classification and segmentation are used. In addition, we classify the adaptive class prototypes in the first reference using a cosine similarity cross-entropy loss [7], [13] to encourage prototypes to fall into the corresponding categories.

iFSIS: Existing RPN-based methods such as iMTFA [17] and iFS-RCNN [68] address the catastrophic forgetting issue and adapt to the incremental setting by freezing all parameters except for the classification head. In recent studies, the use of transformer, i.e. DETR, for incremental tasks has been explored [15]. It has been observed that fine-tuning the classification head alone is inadequate. The model cannot learn novel classes without fine-tuning CAL. However, fine-tuning CAL will cause serious forgetting issues. To solve this problem, Incremental-DETR [15] performs feature distillation on both CAL and the classification head. It further improves performance by using selective search to provide pseudo labels. We argue that CAL is crucial for adapting DETR-like models to the incremental setting. As a result, to adapt RefT to iFSIS as simply as possible, we focus on the distillation on CAL without using logits distillation and selective search. As shown in Fig. 7, during the fine-tuning stage in iFSIS, given an input image that may contain novel class foregrounds, base class foregrounds, and backgrounds, we enforce the model to learn novel class knowledge as much as possible while retaining base class knowledge, with only access to novel class annotations. To this end, we use the ground truth mas to mask out novel class objects and perform feature distillation only on base class foregrounds and backgrounds

$$\bar{F}^{freeze} = \left(1 - \sum_{p=1}^{P} M_p^{gt}\right) F^{freeze}, \quad (5)$$

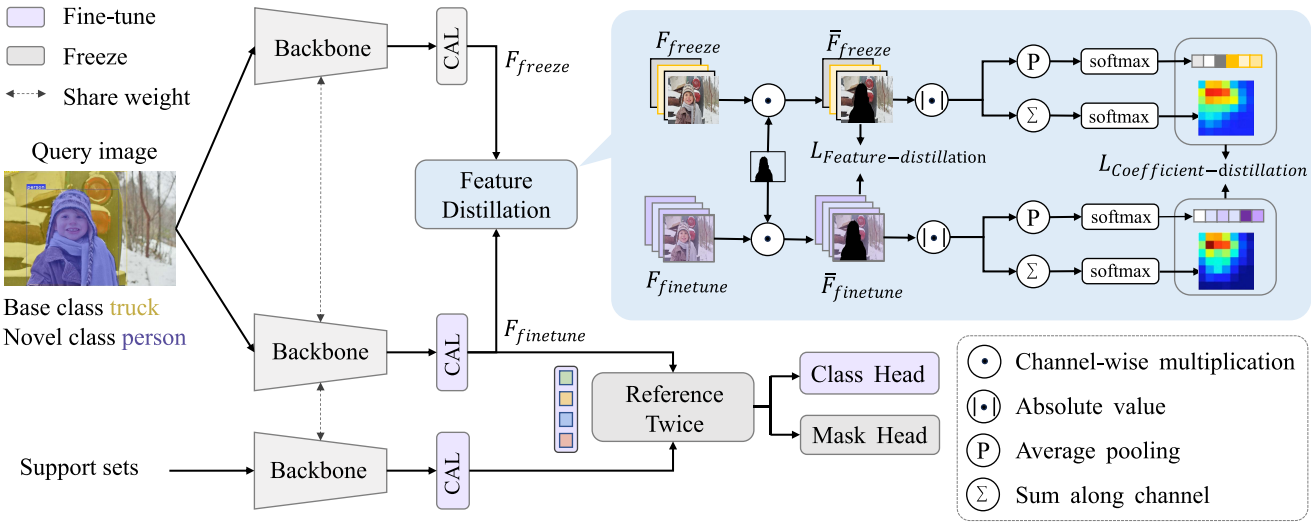


Fig. 7. Adaptation to iFSIS. In novel fine-tuning, all parameters except CAL, object queries, and the class head are frozen. The proposed Class-Enhanced BKD Loss is adopted to CAL to prevent overfitting, without hindering novel class generalization.

$$\bar{F}^{finetune} = \left(1 - \sum_{p=1}^{P} M_p^{gt}\right) F^{finetune}, \tag{6}$$

where M_p^{gt} is the ground truth mask of the pth foreground instance, belonging to the novel classes during fine-tuning, $p = 1, 2, \ldots, P$. F^{freeze}, $F^{finetune}$ is the CAL output feature of the model frozen after base training and during fine-tuning, respectively.

Although without the accurate location of base class objects, we show that the learned features from the base class already indicate the approximate location. Thus, we directly use the relative prominence of the feature map in the spatial and channel dimensions to represent the positions of base class features and serve as reweighting weights for distillation. The degree of prominence is measured by the absolute mean value of the pixel or channel noarmlzed by a softmax function. The channel and spatial re-weighting coefficients R_k^C and R_k^S are obtained from

$$R_k^C = C \cdot \text{softmax}\left(\frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}\left|F_{k,i,j}^{freeze}\right|\right), \tag{7}$$

$$R_{ij}^S = HW \cdot \text{softmax}\left(\frac{1}{C}\sum_{i=1}^{C}\left|F_{k,i,j}^{freeze}\right|\right). \tag{8}$$

The Re-weighted Base Class Feature Distillation Loss (FD) is formulated as

$$\mathcal{L}_{FD} = \sum_{k=1}^{C}\sum_{i=1}^{H}\sum_{j=1}^{W} R_k^C R_{i,j}^S \left(\bar{F}_{k,i,j}^{freeze} - \bar{F}_{k,i,j}^{finetune}\right)^2, \tag{9}$$

where C, H, and W denote the channel, height, and width of the feature; $F_{k,i,j}^{freeze}$ is the masked CAL output feature of the model frozen after base training; and $F_{k,i,j}^{finetune}$ is the masked CAL output feature of the model during fine-tuning.

In addition, we add a Re-weighting Coefficient Distillation Loss (CD) to enhance the base class

$$\mathcal{L}_{CD} = \sum_{i=1}^{H}\sum_{j=1}^{W}\left(R_{i,j}^{finetune} - R_{i,j}^{freeze}\right)^2 + \sum_{k=1}^{C}\left(R_k^{finetune} - R_k^{freeze}\right)^2. \tag{11}$$

The final *Class-Enhanced Base Class Distillation Loss* (CE-BCD) is defined as

$$\mathcal{L}_{CE-BCD} = \mathcal{L}_{FD} + \lambda\mathcal{L}_{CD}, \tag{12}$$

where λ is the hyperparameter to balance the loss terms.

IV. Experiments

Dataset Setting and Metrics: We evaluate the proposed method on the MS-COCO 2014 dataset [111] with the same setups [16], where 80 classes are divided into 2 sets, including 20 novel classes that intersect with PASCAL VOC [112] and the remaining 60 base classes. The number of examples per class is set to $K = \{1, 3, 5, 10, \text{and } 30\}$. We adopt the typical metrics based on Average Precision (IoU $= 0.5 : 0.95$), for novel and base classes, abbreviated to nAP and bAP. As in [17], we carry out experiments 10 times with K examples of 10 seeds for each class, and report the averaged results. Since Pascal VOC has no annotations for instance segmentation, we evaluate RefT on the LVIS dataset [113] to demonstrate the generalization ability. We use frequent (≥ 100 samples per class) and common classes (11-100 samples per class) in LVIS as base classes and rare classes (≤ 10 samples per class) as novel classes. The corresponding COCO-style metrics are abbreviated as fAP, cAP, and rAP. Since the number of images of rare classes is too small to make different shots, we set $K \leq 10$ as done in prior work [16], [68].

Implementation Details and Strong Baselines: We use Mask2Former [4] as the main module, and ResNet-50 [115] is adopted as the backbone, similar to prior work for FSIS. For the base model, we train our model over COCO base classes for 50 epochs with a batch size of 8 on 4 RTX 3090 GPUs, using the AdamW optimizer [116] and the step learning rate schedule. We set the initial learning rate of 0.0001 and a weight decay at the last epoch by 0.05. The settings remain the same in the novel fine-tuning stage. For fair comparisons, we implement

TABLE III
FSOD AND FSIS RESULTS (NAP) ON COCO WITH K = {1, 3, 5, 10, 30}

Methods	Object Detection					Instance Segmentation				
	1	3	5	10	30	1	3	5	10	30
TFA [16]	1.9	7.0	9.1	12.1	-	-	-	-	-	-
FSDetView [114]	3.2	8.1	10.7	14.7	-	-	-	-	-	-
DeFRCN [11]	4.8	10.7	13.6	16.8	22.6	-	-	-	-	-
FCT [12]	5.1	9.8	12.0	15.3	20.2	-	-	-	-	-
Meta-DETR [13]	**7.5**	**13.5**	**15.4**	19.0	22.2	-	-	8.1	10.1	-
MRCN-ft-full [115]	0.7	-	1.3	2.5	11.1	0.6	-	1.2	1.9	-
Meta R-CNN [7]	-	-	3.5	5.6	12.4	-	-	2.8	4.4	-
MTFA [17]	-	-	6.6	8.5	-	-	-	6.6	8.4	-
iFS-RCNN [68]	6.3	8.9	10.5	11.3	14.7	**5.5**	7.8	9.4	10.2	13.1
Mask2Former-ft-full [16]	2.9	8.9	13.6	17.3	21.9	2.5	8.4	12.7	16.7	20.8
Mask2Former w/ CAM	4.1	10.5	14.8	18.6	22.9	3.9	11.1	13.4	17.5	21.6
RefT (Ours)	5.2	11.3	15.0	**19.3**	**24.0**	5.1	**12.2**	**14.2**	**18.4**	**22.5**

"-": unavailable corresponding result. Optimal and suboptimal results are highlighted in bold and underline, respectively. We use ResNet-50 as the backbone.

a fine-tune-based Mask2Former for FSIS similar to TFA [16]. We perform standard training on Mask2Former with the default settings over the COCO base classes and fine-tune the model over novel classes.

A. Training and Inference Details

We use the episodic-training [7] in both base training and the few-shot fine-tuning stage. The training stage comprises a series of episodes $\boldsymbol{E}_i = (\boldsymbol{I}_Q^i, \boldsymbol{S}^i)$, where i indicates the ith episode. Given a query image $\boldsymbol{I}_Q^i$, all objects present in the image belong to $\boldsymbol{N}_{pos}$ classes in $\boldsymbol{C}_{\text{train}}$. We also randomly add $\boldsymbol{N}_{neg}$ classes that are not present in the query image $\boldsymbol{I}_Q^i$. The support set $\boldsymbol{S}^i$ contains $\boldsymbol{N}$ classes, where $\boldsymbol{N} = \boldsymbol{N}_{pos} + \boldsymbol{N}_{neg}$, and varies between $\boldsymbol{N}_{pos}$ and $\boldsymbol{N}_{train}$. K samples per class along with the structural annotations are provided as additional input, which makes the N-way K-shot episode $\boldsymbol{E}_i$. In order to reduce computational costs, only the support features and queries for positive classes need to be calculated, while those for negative classes are sampled from the positive classes that are calculated in previous iterations.

During inference time, we compute adaptive class prototypes and object queries from support sets once and for all. Unlike prior works that require multiple forward passes for each query image, RefT only forwards once with all support classes, which is simpler and more efficient.

B. Main Results on COCO

FSIS Results: Table III shows that our method significantly outperforms previous works based on Mask R-CNN, which is expected given a more robust base model. For a fair comparison, we examine the Mask2Former baseline, and RefT still demonstrates noticeable performance gains. We also compare with the DETR-based approach, Meta-DETR [13], which achieves state-of-the-art results in FSOD. Our method yields substantially better results in FSIS and comparable or slightly improved results in FSOD.

We incorporate the Correlational Aggregation Module (CAM) from Meta-DETR into Mask2Former to further align the base model. Our first reference module is similar to CAM, as both use the attention mechanism for aggregation. However, CAM introduces a set of task encodings for each support class and maps them to instances, acting as a classifier. Following previous works, we maintain the classical design choice in the first reference and still outperform the stronger baseline. This highlights the superiority of our query-level aggregation module in supporting the feature-level one.

gFSIS Results: Few studies address the more challenging gFSIS, which necessitates retaining base class knowledge. Table IV demonstrates that RefT consistently surpasses recent state-of-the-art methods in both gFSOD and gFSIS. RefT significantly outperforms Mask2Former with CAM on base classes, highlighting the effectiveness of our feature- and query-level modules. We further explore alternative designs for our second reference module in Table VI. First, we replace cross-attention with adaptive convolution. Next, we introduce an auxiliary loss, as in [36], to enable the classification of concatenated queries and support branch object queries. Despite these modifications, no improvements are observed. This suggests that the simple and effective aggregation design is adequate for utilizing high-level information to support branch object queries.

iFSIS Results: In Table V, we first adapt Mask2Former to the iFSIS setting as our baseline by replacing the fully connected classifier with a cosine similarity classifier, as in iMTFA [17], and fine-tuning both CAL and the class head while keeping all other parameters frozen after base training. A challenge arises as the class-specific CAL needs fine-tuning to enable novel class learning. However, without access to base class samples, CAL may overfit and experience catastrophic forgetting. Introducing our Class Enhanced Base Knowledge Distillation Loss addresses this issue and achieves results comparable to recent state-of-the-art methods. Incremental-DETR [15] proposes a similar loss, but our approach is simpler and equally effective.

C. Results on LVIS

Table VII reports our results on LVIS with K $\leq$ 10 on FSIS, gFSIS, and iFSIS. Compared with the Mask2Former baseline, RefT improves by 1.5 and 0.9 on FSIS and gFSIS, respectively.

TABLE IV
GFSOD AND GFSIS RESULTS ON COCO WITH K = {1, 3, 5, 10, 30}

Backbone	Methods	Object Detection									
		nAP					bAP				
		1	3	5	10	30	1	3	5	10	30
R-50	MRCN-ft-full [115]	0.7	-	1.1	2.3	-	17.6	-	19.4	20.6	-
	MTFA [17]	2.1	-	6.2	8.3	-	31.7	-	33.1	34.0	-
	Retentive R-CNN [22]	-	-	8.3	10.5	13.8	-	-	**39.2**	**39.2**	**39.3**
	LVC [10]	-	-	-	17.6	**25.5**	-	-	-	29.7	33.3
	Mask2Former-ft-full	3.0	7.8	9.5	15.3	20.1	36.7	36.3	35.9	36.9	36.2
	Mask2Former w/ CAM	**5.2**	9.4	11.7	17.3	22.3	34.9	34.2	34.4	35.2	35.0
	RefT (Ours)	**5.2**	**10.2**	**13.1**	**18.6**	23.4	**38.4**	**36.5**	36.0	37.7	37.2
R-101	TFA [16]	1.9	5.1	7.0	9.1	12.1	31.9	32.0	32.3	32.4	34.2
	DeFRCN [11]	4.8	10.7	13.6	16.8	21.2	30.4	32.1	32.6	34.0	34.8
	LVC [10]	-	-	-	17.8	24.5	-	-	-	31.9	33.0
	Mask2Former-ft-full	3.1	8.0	10.2	16.5	20.2	37.2	36.0	35.3	36.9	36.5
	Mask2Former w/ CAM	**5.2**	9.8	13.2	17.7	22.4	35.4	34.0	34.8	35.7	34.8
	RefT (Ours)	**5.2**	**10.8**	**14.1**	**18.9**	**23.6**	**38.5**	**36.4**	**36.2**	**37.7**	**37.4**
Backbone	Methods	Instance Segmentation									
		nAP					bAP				
		1	3	5	10	30	1	3	5	10	30
R-50	MRCN-ft-full [115]	0.6	-	1.2	1.9	-	15.6	-	17.9	18.1	-
	MTFA [17]	2.3	-	6.4	8.4	-	29.9	-	31.3	31.8	-
	Mask2Former-ft-full	2.9	7.6	8.5	14.7	17.9	36.1	35.6	33.4	36.0	35.2
	Mask2Former w/ CAM	**5.2**	9.3	11.2	16.1	20.3	33.4	33.2	32.9	33.3	33.5
	RefT (Ours)	**5.2**	**10.2**	**12.4**	**17.4**	**21.7**	**36.3**	**36.5**	**34.4**	**36.0**	**35.5**

"-": unavailable corresponding result. Optimal and suboptimal results are highlighted in bold and underline, respectively.

TABLE V
IFSOD AND IFSIS RESULTS ON COCO WITH K = {1, 3, 5, 10, 30}

Methods	Object Detection									
	nAP					bAP				
	1	3	5	10	30	1	3	5	10	30
Incremental-DETR [15]	-	-	-	14.4	-	-	-	-	27.3	-
Mask2Former w/ Cos	**4.9**	**8.9**	**13.7**	**16.1**	**21.2**	24.3	24.6	23.8	24.0	24.2
RefT (Ours) w/ Cos	4.0	8.8	12.0	14.9	18.9	**32.2**	**32.0**	**31.8**	**33.4**	**33.2**
Methods	Instance Segmentation									
iMTFA [17]	2.8	-	5.2	5.9	-	25.9	-	22.6	21.9	-
iFS-RCNN [68]	**4.0**	-	8.8	10.1	-	36.4	-	**36.3**	**36.3**	-
Mask2Former w/ Cos	3.1	**6.3**	**9.0**	**12.4**	**18.5**	29.0	28.4	28.4	28.7	28.6
RefT (Ours) w/ Cos	3.1	6.0	8.8	11.1	17.7	**37.0**	36.3	35.3	35.2	32.1

"-": unavailable corresponding result. Optimal and suboptimal results are highlighted in bold and underline, respectively. We use ResNet-50 as backbone.

TABLE VI
COMPARISON OF DIFFERENT FEATURE AGGREGATION OPERATIONS FOR THE SECOND REFERENCE IN GFSIS ON COCO WITH K = 10

Methods	Instance Segmentation	
	nAP	bAP
RefT	17.4	36.0
w/ Adaptive Conv	16.8	35.4
w/ Auxiliary Loss	17.5	35.2

We use ResNet-50 as the backbone.

TABLE VII
FSIS, GFSIS AND IFSIS RESULTS ON LVIS WITH K ≤ 10

Settings	FSIS	gFSIS			iFSIS		
Test on	rAP	rAP	cAP	fAP	rAP	cAP	fAP
MRCN-ft-full [115]	18.3	12.8	25.4	27.8	-	-	-
iFS-RCNN [68]	21.1	-	-	-	18.3	26.3	**28.5**
M2Former-ft-full	22.0	21.2	26.9	28.3	**19.3**	19.9	22.00
Reft (Ours)	**23.5**	**22.1**	**27.1**	**28.6**	18.6	**26.9**	28.1

"-": unavailable corresponding result. Optimal and suboptimal results are highlighted in bold and underline.

TABLE VIII
CROSS-DOMAIN EVALUATION ON COCO2FSS-1000 WITH K = {1, 3, 5}

Methods	Instance Segmentation		
	1	3	5
MRCN-ft-full [115]	79.7	80.3	81.1
iFS-RCNN [68]	81.5	82.2	83.6
Mask2Former-ft-full [16]	81.9	82.5	83.4
RefT (Ours)	**82.7**	**83.3**	**84.1**

Optimal and suboptimal results are highlighted in bold and underline, respectively. We use ResNet-50 as backbone.

TABLE IX
ABLATION STUDY ON EACH COMPONENT

Baseline	+first reference	+second reference	nAP	bAP
Mask2Former-ft			14.7	36.0
	✓		15.6	36.3
		✓	16.3	35.5
	✓	✓	17.4	36.0

In the more challenging iFSIS, RefT effectively retains base class knowledge, achieving a substantial increase of 7.0 cAP and 5.9 fAP, while maintaining a comparable performance on novel classes.

D. Cross-Domain Evaluation on COCO2FSS-1000

To demonstrate the scalability and generalization ability of RefT to new datasets, we conduct a cross-domain evaluation from COCO to FSS-1000. The FSS-1000 dataset is originally designed for few-shot semantic segmentation, but it also supports instance-level segmentation with instance labels in 758 out of the 1,000 classes in the dataset. We select 100 classes with only instance labels from the FSS-1000 test categories (designed to be disjoint with COCO 60 base classes) as the test set. We train on COCO 60 base classes and few-shot fine-tune on FSS-1000 test classes. Since FSS-1000 has only 10 images per class, we only provide results with shot $K = \{1, 3, 5\}$. Table VIII demonstrates the scalability and generalization ability of RefT to novel classes in FSS-1000.

E. Ablation Study and Analysis

In this section, we present ablation studies for FSIS, gFSIS, and iFSIS. All experiments are conducted on the MS-COCO minival dataset with K = 10, using ResNet-50 as the backbone. We employ standard MS-COCO metrics, specifically Average Precision (IoU = 0.5 : 0.95), for both novel and base classes, denoted as nAP and bAP, respectively. Each test is executed 10 times with K examples from 10 seeds for each class, and the averaged results are reported.

1) FSIS and gFSIS: Ablation study on each component: In Table IX, we focus on the effectiveness of each component. For a fair comparison, we use the single-branch Mask2Former as our baseline. Incorporating our query-based second reference module results in a 1.6 improvement in novel classes. Additionally, including our image-based first reference module further enhances nAP by 1.1. It is worth noting that while longer training contributes to greater performance gains in nAP at the expense of a rapid decline in bAP, we are not sacrificing bAP for nAP. This supports our observations in Section III-A, indicating that both branches exhibit a coupled effect for novel classes.

Another interesting point is that the first reference yields a 0.9 improvement in nAP and a 0.3 improvement in bAP, while the second reference yields a 1.6 improvement in nAP but a 0.5 loss in bAP. We assume that this may be due to the fact that the detection of base classes mainly relies on the knowledge encoded in the model parameters during base training with large amount of data, while the detection of novel classes relies mainly on the reference information provided by the support branch. The first reference is the backbone feature aggregation. Since the two branches share the same backbone, the features of the base classes in the support branch should still be more prominent. Therefore, no loss in bAP is observed, and the additional guidance has an improvement on both base and novel classes. The second reference is the query feature aggregation. Here, the model is encouraged to rely more on the support object query features for classification, which reduces the model bias towards the base classes. This is particularly beneficial for the classification of novel classes but may result in some loss for the base classes.

Effect of fine-tuning CAL: To illustrate the significance of CAL in enabling DETR-like models to learn novel classes, we freeze CAL during the novel fine-tuning stage and only fine-tune the object queries and the class head. Table XII reveals that the model struggles to generalize with a frozen CAL, resulting in a substantial gap compared to the model that fine-tunes CAL (2.6 versus 17.4 nAP). Moreover, the findings suggest that additional learnable parameters are unnecessary to ensure adequate transferability to the novel domain, as fine-tuning an existing CAL can achieve both novel class generalization and base knowledge retention.

TABLE X
SUPPORT FEATURES IN FIRST REFERENCE

Method	nAP	bAP
Res3	17.2	36.6
Res4	17.0	36.0
Res5	16.4	35.2
Enc1	17.4	36.0

Support features in first reference In Table X, we conduct an ablation study on various features used to compute adaptive class prototypes, including three stages of features from ResNet-50 (denoted as Res3, Res4, and Res5) and the flattened multi-scale features from the first transformer encoder layer (denoted as Enc1). Utilizing support features of a larger scale results in a notable performance improvement (17.2 versus 16.4 nAP), potentially due to the increased detection of smaller objects. Employing multi-scale features further contributes to a 0.2 nAP gain.

Support features in second reference: To showcase the efficacy of our query-based aggregation in the second reference, we implement an image-based version that employs the

TABLE XI
SUPPORT FEATURES IN SECOND REFERENCE

Features	nAP	bAP
Feature-level	16.3	35.7
Query-level	17.4	36.0

TABLE XII
EFFECT OF FINE-TUNING CAL

Fine-tune	nAP	bAP
	2.6	39.4
✓	17.4	36.0

TABLE XIII
POOLING IN FIRST REFERENCE

Method	nAP	bAP
GAP	17.0	35.4
RoiAlign	17.2	35.3
MaskPool	17.4	36.0

TABLE XIV
QUERY SELECTION IN SECOND REFERENCE

Sort by	nAP	bAP
Score	17.2	35.8
Mask	17.4	36.0

same adaptive class prototypes from the first reference as support guidance and uses cross-attention as the aggregation operation. Table XI demonstrates that the query-level enhancement module outperforms the feature-level one, resulting in a 1.1 nAP increase. This validates the superiority of our RefT framework over a purely feature-level enhancement framework.

Pooling in first reference: In Table XIII, we compare various pooling methods for extracting pertinent information from support images. As the accuracy of the feature region improves, the performance increases correspondingly. This is attributed to the more accurate instance features without noise, which provide more distinctive class prototypes for the query branch classification.

Query selection in second reference: In Table XIV, we compare the results of selecting top 10 object queries in second reference sorted by scores or mask IoU. The results are comparable when k is small, as object queries exhibit higher accuracy for both classification and segmentation.

Number of Object Queries in Second Reference: In the second reference, we choose the top-k object queries for each support image. Table XV compares the performance impact of different k values. The results indicate that when $k \leq 50$, a relatively stable outcome close to the best can be achieved. This finding aligns with Table II, as the support branch mask quality is ensured when $k \leq 50$.

TABLE XV
NUMBER OF OBJECT QUERIES IN THE SECOND REFERENCE

# Queries	3	10	50	100
nAP	17.4	17.4	17.5	16.8
bAP	35.8	36.0	35.8	35.5

TABLE XVI
ABLATION STUDY ON FIRST REFERENCE *VERSUS* CAM

Methods	Support Set & Loss Aligned with	nAP	bAP
Meta-DETR [13]	Meta-DETR	10.1	-
RefT w/ CAM	Meta-DETR	16.3	26.8
RefT w/ CAM	RefT	15.2	32.3
RefT w/ first reference	RefT	17.4	36.0

TABLE XVII
EFFECT OF FINE-TUNING DIFFERENT LAYERS

Res5	CAL	Pixel Decoder	Transformer Decoder	Object Queries & Class Head	nAP	bAP
		✓		✓	1.8	31.0
		✓	✓	✓	2.5	17.2
✓				✓	11.7	32.9
	✓			✓	17.4	36.0

Comparison with previous feature-level aggregation module: Meta-DETR [13] introduces a Correlational Aggregation Module (CAM) for concurrent aggregation of query features and support class prototypes, akin to our first reference module. The differences are twofold: (1) the support set organization varies. Meta-DETR divides all classes into class sets, requiring multiple forward passes, while the first reference handles all classes simultaneously. (2) The classification loss differs. Meta-DETR uses a sigmoid binary cross-entropy loss, transforming the classification task into matching between query and support classes, whereas we utilize the standard softmax cross-entropy loss. As demonstrated in Table XVI, replacing our first reference module with CAM and maintaining the support set and loss in line with Meta-DETR leads to a substantial 9.2 bAP decrease (36.0 versus 26.8). Even with our support set and loss function unchanged, bAP drops by 3.7 (36.0 versus 32.3). We speculate that CAM primarily handles class matching in Meta-DETR, while the class head operates in a class-agnostic manner. Consequently, the base class feature representations are not stored in the class head as in prior works, resulting in suboptimal bAP outcomes.

Effect of fine-tuning different layers: In the main results, we only unfreeze CAL, object queries, and the class head during novel fine-tuning to enable the model to learn novel classes without forgetting previous knowledge. Table XVII presents a detailed investigation of how fine-tuning different layers impacts the results. We observe that when CAL is frozen, even if both the pixel decoder and transformer decoder are fine-tuned, the model struggles to learn novel knowledge (2.5 nAP). Moreover, fine-tuning more parameters on extremely limited samples can cause overfitting, leading to a significant decline in bAP ($-18.8\downarrow$).

TABLE XVIII
ABLATION OF THE CLASS-ENHANCED BKD LOSS

Methods	nAP	bAP
RefT	12.4	28.7
RefT w/ Weight Regularization	11.5	29.4
RefT w/ BKD Loss	10.1	33.2
RefT w/ Class-Enhanced BKD Loss	11.1	35.2

TABLE XIX
EMPLOYING DIFFERENT CLASS HEADS FOR IFSIS

Methods	nAP	bAP
RefT w/ Cosine Similarity Classifier	11.1	35.2
RefT w/ Logit Classifier	12.7	36.0

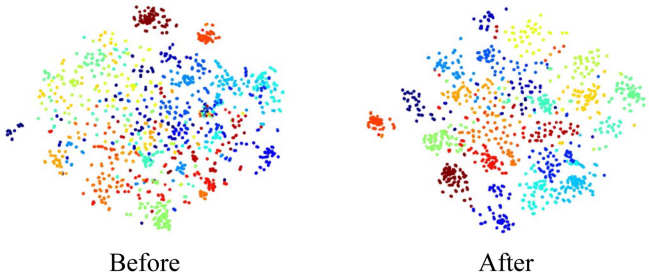


Fig. 8. t-SNE visualization of object queries belonging to COCO 20 novel classes. The results are obtained from the support branch before and after novel fine-tuning.

Although fine-tuning layers in the feature extractor allows the model to generalize, fine-tuning CAL with fewer parameters achieves better results for both novel (+5.7↑) and base classes (+3.1↑).

2) iFSIS: Ablation of the Class-Enhanced BKD Loss: In Table XVIII, we first use L2 regularization to constrain the parameters of CAL as our baseline. Using BKD Loss [15] recovers bAP by 3.8 with a drop in nAP (−1.4↓) while applying our Class-Enhanced BKD Loss significantly recovers bAP (+5.8↑) with a marginal drop in nAP (−0.4↓).

Employing different class heads: In Table XIX, we also experiment with the logit classifier based on Bayesian probabilities proposed by the recent state-of-the-art iFS-RCNN [68]. Although performance improvements are observed in both nAP(+1.6) and bAP(+0.8), they are not as significant as in region-based methods. This suggests that there is potential for further improvement in DETR-like [118] models for iFSIS.

F. Fair Comparison With Previous Methods

To ensure a fair comparison, we transfer the modules proposed in the RPN-based methods to the query-based architecture, i.e., Mask2Former. The transferable methods mainly include those that improve the backbone feature aggregation and the predictor head. The methods that improve RPN and ROI-feature aggregation cannot be transferred to the query-based architecture. Specifically, we transfer the module from DCNet (i.e., Dense Relation Distillation Module) and FCT (i.e., Fully Cross Transformer) for improving backbone feature aggregation, and iMTFA (i.e., Cosine-Similarity Head) and iFS-RCNN (i.e., Bayesian Head) for improving the classification head. Table XX shows that the improvement brought by these methods is slightly lower than or comparable to our approach. However, it is worth noting that our method is not in conflict with these methods. With the addition of these heads, our method can achieve even higher results. Our goal is to provide a simple query-based baseline for (general/ incremental) FSIS to avoid the overfitting issue of RPN. Therefore, we propose query-level feature aggregation as

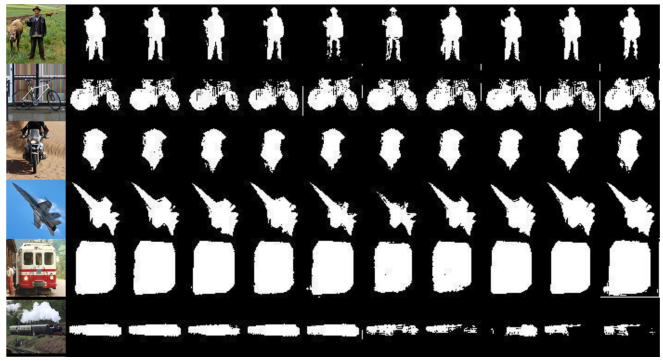


Fig. 9. Visualization of the mask head predictions of support branch object queries. Object queries that achieve top-10 IoU with the ground truth masks are shown.

Fig. 10. Visualization of 10-shot FSIS results on COCO minival set. Results of novel classes are mainly displayed.

TABLE XX
COMPARISON WITH MORE BASELINES INTEGRATED WITH MODULES OF PREVIOUS METHODS ON COCO WITH K = {1, 3, 5, 10, 30}

Improve	Methods	Object Detection									
		nAP					bAP				
		1	3	5	10	30	1	3	5	10	30
/	Mask2Former-ft-full	3.0	7.8	9.5	15.3	20.1	36.7	36.3	35.9	36.9	36.2
Backbone FA	Mask2Former w/ DCNet	3.5	8.2	10.4	16.2	21.0	36.9	36.0	36.0	37.0	36.6
Backbone FA	Mask2Former w/ FCT	5.1	10.0	12.2	17.0	21.8	37.5	36.2	35.9	36.9	36.8
Head	Mask2Former w/ Cos	4.5	9.4	12.0	17.6	22.9	37.8	36.5	36.0	37.0	36.8
Head	Mask2Former w/ Bay	**5.5**	**10.3**	12.5	18.1	23.3	38.1	**36.9**	**36.2**	37.2	37.0
Query FA	RefT (Ours)	5.2	10.2	**13.1**	**18.6**	**23.4**	**38.4**	36.5	36.0	**37.7**	**37.2**
Improve	Methods	Instance Segmentation									
		nAP					bAP				
		1	3	5	10	30	1	3	5	10	30
/	Mask2Former-ft-full	2.9	7.6	8.5	14.7	17.9	36.1	35.6	33.4	36.0	35.2
Backbone FA	Mask2Former w/ DCNet	3.5	8.0	10.1	15.7	18.4	36.1	36.0	33.6	35.9	35.2
Backbone FA	Mask2Former w/ FCT	5.0	9.8	12.0	16.6	19.0	36.1	36.1	33.7	35.9	35.2
Head	Mask2Former w/ Cos	4.5	9.3	11.8	16.7	20.1	36.2	36.2	34.0	36.0	35.4
Head	Mask2Former w/ Bay	**5.4**	**10.2**	12.3	17.0	20.9	**36.3**	36.4	**34.6**	**36.0**	35.4
Query FA	RefT (Ours)	5.2	**10.2**	**12.4**	**17.4**	**21.7**	**36.3**	**36.5**	34.4	**36.0**	**35.5**

Optimal and suboptimal results are highlighted in bold and underline, respectively.

TABLE XXI
EVALUATION RESULTS ON COCO WITHOUT RE-TRAINING WITH K = {1, 3, 5, 10, 30}

	nAP					bAP				
	1	3	5	10	30	1	3	5	10	30
	Object Detetion									
iMTFA [17]	3.3	-	6.2	7.1	-	-	-	-	-	-
AirDet [117]	6.0	7.0	7.8	8.7	-	-	-	-	-	-
FS-DETR [14]	**7.0**	**10.0**	10.9	11.3	-	-	-	-	-	-
RefT (Ours)	6.6	9.7	**11.3**	**12.7**	**13.5**	**33.9**	**33.7**	**34.2**	**33.9**	**34.5**
	Instance Segmentation									
iMTFA	2.8	-	5.2	7.1	-	-	-	-	-	-
RefT (Ours)	**6.5**	**9.5**	**10.8**	**12.9**	**13.2**	**32.7**	**32.5**	**33.6**	**32.8**	**33.1**

Optimal and suboptimal results are highlighted in bold and underline, respectively.

a replacement for ROI feature aggregation and try to solve the challenge of query-based architecture in incremental setting. We hope that this will encourage further exploration of query-based structures in future research.

G. Generalizability Without Re-Training

RefT employs a vanilla class-specific classification head, which means it does not support the "without re-training" setting. However, by simply adding a learnable support class embedding for each support class on the support branch query features during training, RefT can easily adapt to the "without re-training" setting. By doing this, the classification head no longer predicts the score of a fixed category, but instead predicts the probability of whether the object belongs to the given support class. The model dynamically classifies based on the input support classes. Is is worth noting that our second reference module is functionally almost equivalent to FS-DETR [14] when adapted to "without re-training" setting. The only difference is that we implement cross-attention to input support queries and class embeddings, while FS-DETR concatenates them with object queries.

Since FS-DETR has not been open-sourced, in order to provide a comparative result in the "without re-training" setting, we did not evaluate on new dataset but still on the COCO dataset. As demonstrated in Table XXI, the class-agnostic training strategy greatly enhances the model generalization ability for novel classes especially in low-shot situations, but at the cost of a decrease in base class performance and a gradual reduction in performance improvement as the number of shots increases. FS-DETR also exhibits similar behavior according to its reported results.

H. Scale Up

In Table XXII, we further showcase the effectiveness and generalization capability of RefT by scaling up the backbone. Using more powerful Transformer backbone models, such as Swin-T, Swin-S, and Swin-B, our approach continues to generalize well and avoids severe overfitting.

Fig. 11. Visualization of results for base classes on COCO with K = 10.

TABLE XXII
GFSOD AND GFSIS RESULTS USING THE SCALED-UP BACKBONE ON COCO WITH K = {1, 5, 10}

Backbone	Methods	Object Detection					
		nAP			bAP		
		1	5	10	1	5	10
Swin-T	LVC [10]	-	-	18.6	-	-	29.2
	RefT (Ours)	5.3	16.8	20.0	39.6	37.0	37.9
Swin-S	LVC [10]	-	-	19.0	-	-	28.7
	RefT (Ours)	5.2	21.0	24.2	40.4	39.8	40.5
Swin-B	RefT (Ours)	7.4	20.2	26.4	42.6	40.9	41.0
Backbone	Methods	Instance Segmentation					
		nAP			bAP		
		1	5	10	1	5	10
Swin-T	RefT (Ours)	5.2	15.4	18.5	37.1	35.0	36.2
Swin-S	RefT (Ours)	5.0	19.4	22.7	38.3	37.9	38.3
Swin-B	RefT (Ours)	7.1	20.7	24.8	40.2	38.7	39.2

"-": unavailable corresponding result.

I. Visualization and More Analysis

Understanding the effect of second reference: In Fig. 8, we visualize the t-SNE results of object queries for COCO's 20 novel classes. We observe that the object queries of novel classes can be roughly distinguished even without the novel fine-tuning process, and the clustering becomes more apparent after fine-tuning. In our second reference module, we utilize this *support query categorization* to guide the query branch classification. In Fig. 9, we visualize the mask head predictions of support branch object queries from the model before the novel fine-tuning stage. Only object queries with top-10 IoU values compared to ground truth masks are shown. We find that the object queries of novel classes encode accurate localization information even without the novel fine-tuning process. We leverage this *support query localization* in our second reference module to provide guidance for the query branch localization.

Visualizations for all three settings are presented. Predictions for base class instances are shown in Fig. 11, while those for novel class instances are displayed in Fig. 12. Although class misclassification remains prevalent, most predicted masks are accurate across all settings. Even in the most challenging iFSIS setting, numerous novel instances are recalled.

Additional Qualitative Results: In Fig. 10, we provide visual results on MS-COCO datasets corresponding to Table IX. We observe that incorporating our query-based second reference effectively reduces both missed and misclassified instances, primarily due to the guidance from *support query localization* and *support query categorization*. By aligning query features with adaptive class prototypes, the inclusion of our first reference further decreases misclassification between highly correlated classes.

J. Computational Cost

Comparison of computational cost is provided in Table XXIII

Parameter and GFLOPs: Compared to the robust Mask2Former baseline, RefT achieves a notable +1.7↑ increase in nAP with only a 5.8% increment in GFLOPs (226.2 versus 239.4) and a 6.2% rise in parameters (43.7 versus 46.4), while processing a 1,024 × 1,024 image input.

Speed: During training, RefT is slightly slower than the fine-tuning-based iFS-RCNN (0.10 versus 0.31) due to the additional forward pass required to process a randomly sampled support set, which guides the query branch. However, RefT outperforms the meta-based Meta-FRCN (0.97 versus 0.31) by using object queries, thereby eliminating the need for pixel space processing.

TABLE XXIII
SPEED AND MEMORY ANALYSIS OF DIFFERENT FEW-SHOT ARCHITECTURES

Venue	Methods	Architecture	Training Strategy	Train		Inference	
				Speed	VRAM	Speed	VRAM
CVPR'22	iFS-RCNN [68]	RPN	Finetune	0.10s	3543M	0.14s	5363M
Baseline	Mask2Former-ft	DETR	Finetune	0.19s	4973M	0.10s	4523M
AAAI'22 Oral	Meta Faster R-CNN [68]	RPN	Meta	0.97s	9583M	0.92s	8483M
Ours	RefT	DETR	Meta	0.31s	5655M	0.11s	4705M

All tested on 1 RTX 3090 with batchsize 1 and image size 1,024*1,024.

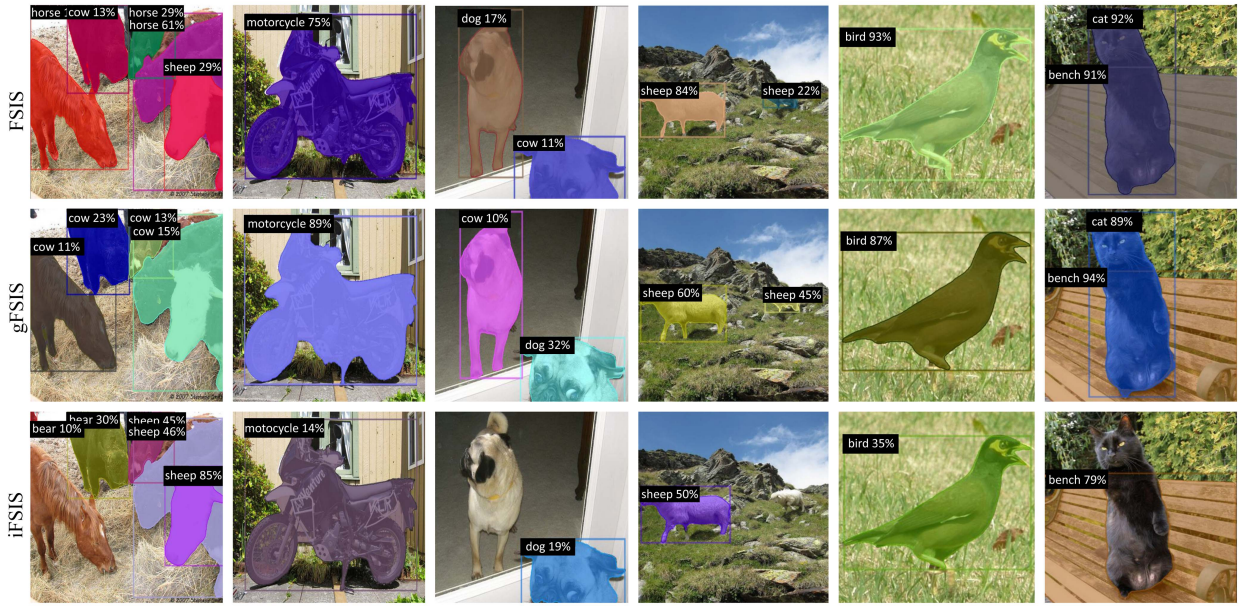


Fig. 12. Visualization of results for novel classes on COCO with K = 10.

Inference Efficiency: During inference, RefT becomes more efficient as the support set is preprocessed and fixed.

VRAM: The memory cost of RefT is limited due to the following factors: *1)* ReFT mainly finetunes class-specific parameters while keeping most parameters fixed; *2)* RefT adopts a smaller 320 × 320 image size in the support branch, significantly reducing memory consumption compared to the query branch.

V. CONCLUSION

In this paper, we present a simple and unified baseline for few-shot instance segmentation, namely *Reference Twice* (RefT). We carefully examine the mask-based DETR framework in FSIS and identify two key factors named *support query localization* and *support query categorization*. Specifically, we first design a mask-based dynamic weighting module to aggregate support features and then propose to link object queries for better calibration via cross-attention. Additionally, RefT can be easily extended to all three settings including FSIS, gFSIS, and iFSIS with a simple Class-Enhanced BKD loss to solve the difficulty of adapting DETR-like models to incremental settings without selective search or logits distillation. To the best of our knowledge, we are the first unified architecture to support three different few-shot settings Despite its simplicity, RefT achieves state-of-the-art or second-best performance on MS-COCO and LVIS benchmarks, across all settings and all shots. We hope our framework can be a solid baseline for instance-level few-shot segmentation problems.

Limitation and Future Work: Currently, RefT fails to perform well in the one-shot setting. One shot makes the guidance of the reference-twice modules unreliable because the support features deviate from the true class distribution. One potential solution is to improve the single object matching ability to fix this issue.

REFERENCES

[1] M.-H. Guo, C.-Z. Lu, Q. Hou, Z. Liu, M.-M. Cheng, and S.-M. Hu, "SegNeXt: Rethinking convolutional attention design for semantic segmentation," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022, Art. no. 84.

[2] F. Li et al., "Mask DINO: Towards a unified transformer-based framework for object detection and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 3041–3050.

[3] J. Jain, J. Li, M. T. Chiu, A. Hassani, N. Orlov, and H. Shi, "OneFormer: One transformer to rule universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 2989–2998.

[4] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1280–1289.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.

[6] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-time instance segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9156–9165.

[7] X. Yan, Z. Chen, A. Xu, X. Wang, X. Liang, and L. Lin, "Meta R-CNN: Towards general solver for instance-level low-shot learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9576–9585.

[8] Z. Fan et al., "FGN: Fully guided network for few-shot instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9169–9178.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[10] P. Kaul, W. Xie, and A. Zisserman, "Label, verify, correct: A simple few shot object detection method," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 14217–14227.

[11] L. Qiao, Y. Zhao, Z. Li, X. Qiu, J. Wu, and C. Zhang, "DeFRCN: Decoupled faster R-CNN for few-shot object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 8661–8670.

[12] G. Han, J. Ma, S. Huang, L. Chen, and S.-F. Chang, "Few-shot object detection with fully cross-transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 5311–5320.

[13] G. Zhang, Z. Luo, K. Cui, S. Lu, and E. P. Xing, "Meta-DETR: Image-level few-shot detection with inter-class correlation exploitation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 11, pp. 12832–12843, Nov. 2023.

[14] A. Bulat, R. Guerrero, B. Martinez, and G. Tzimiropoulos, "FS-DETR: Few-shot detection transformer with prompting and without re-training," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 11759–11768.

[15] N. Dong, Y. Zhang, M. Ding, and G. H. Lee, "Incremental-DETR: Incremental few-shot object detection via self-supervised learning," in *Proc. 37th AAAI Conf. Artif. Intell.*, 2023, Art. no. 60.

[16] X. Wang, T. Huang, J. Gonzalez, T. Darrell, and F. Yu, "Frustratingly simple few-shot object detection," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, Art. no. 920.

[17] D. A. Ganea, B. Boom, and R. Poppe, "Incremental few-shot instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1185–1194.

[18] S. Baik, M. Choi, J. Choi, H. Kim, and K. M. Lee, "Meta-learning with adaptive hyperparameters," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, Art. no. 1743.

[19] H. Hu, S. Bai, A. Li, J. Cui, and L. Wang, "Dense relation distillation with context-aware aggregation for few-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10185–10194.

[20] G. Han, S. Huang, J. Ma, Y. He, and S.-F. Chang, "Meta faster r-CNN: Towards accurate few-shot object detection with attentive feature alignment," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 780–789.

[21] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4080–4090.

[22] Z. Fan, Y. Ma, Z. Li, and J. Sun, "Generalized few-shot object detection without forgetting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 4527–4536.

[23] R. Zhang et al., "Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 15211–15222.

[24] A. Parnami and M. Lee, "Learning from few examples: A summary of approaches to few-shot learning," 2022, *arXiv:2203.04291*.

[25] J.-B. Alayrac et al., "Flamingo: A visual language model for few-shot learning," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022, Art. no. 1723.

[26] S. X. Hu, D. Li, J. Stühmer, M. Kim, and T. M. Hospedales, "Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9058–9067.

[27] R. Zhang et al., "Tip-adapter: Training-free adaption of CLIP for few-shot classification," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 493–510.

[28] J. Xie, F. Long, J. Lv, Q. Wang, and P. Li, "Joint distribution matters: Deep Brownian distance covariance for few-shot classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 7962–7971.

[29] Z. Yang, J. Wang, and Y. Zhu, "Few-shot classification with contrastive learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 293–309.

[30] J. Xu and H. Le, "Generating representative samples for few-shot classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8993–9003.

[31] A. Roy, A. Shah, K. Shah, P. Dhar, A. Cherian, and R. Chellappa, "FeLMi: Few shot learning with hard mixup," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022Art. no. 1777.

[32] Y. Lu, L. Wen, J. Liu, Y. Liu, and X. Tian, "Self-supervision can be a good few-shot learner," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 740–758.

[33] C. Simon, P. Koniusz, R. Nock, and M. Harandi, "On modulating the gradient for meta-learning," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 556–572.

[34] S. Baik, S. Hong, and K. M. Lee, "Learning to forget for meta-learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2376–2384.

[35] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 3637–3645.

[36] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1199–1208.

[37] S. W. Yoon, J. Seo, and J. Moon, "Tapnet: Neural network augmented with task-adaptive projection for few-shot learning," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 7115–7123.

[38] H. Li, D. Eigen, S. Dodge, M. Zeiler, and X. Wang, "Finding task-relevant features for few-shot learning by category traversal," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1–10.

[39] F. Hao, F. He, J. Cheng, L. Wang, J. Cao, and D. Tao, "Collect and select: Semantic alignment metric learning for few-shot learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 8459–8468.

[40] R. Hou, H. Chang, B. MA, S. Shan, and X. Chen, "Cross attention network for few-shot classification," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, Art. no. 360.

[41] Y. Li et al., "Few-shot object detection via classification refinement and distractor retreatment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15395–15403.

[42] H. Lee, M. Lee, and N. Kwak, "Few-shot object detection by attending to per-sample-prototype," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2022, pp. 1101–1110.

[43] S. Zhang, L. Wang, N. Murray, and P. Koniusz, "Kernelized few-shot object detection with efficient integral aggregation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 19185–19194.

[44] Y. Cao et al., "Few-shot object detection via association and discrimination," in *Proc. 35th Int. Conf. Neural Inf. Process. Syst.*, 2021, Art. no. 1267.

[45] A. Li and Z. Li, "Transformation invariant few-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 3094–3102.

[46] A. Wu, S. Zhao, C. Deng, and W. Liu, "Generalized and discriminative few-shot object detection via svd-dictionary enhancement," in *Proc. 35th Int. Conf. Neural Inf. Process. Syst.*, 2021, Art. no. 486.

[47] G. Han, Y. He, S. Huang, J. Ma, and S.-F. Chang, "Query adaptive few-shot object detection with heterogeneous graph convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 3243–3252.

[48] T.-I. Chen et al., "Dual-awareness attention for few-shot object detection," *IEEE Trans. Multimedia*, vol. 25, pp. 291–301, 2021.

[49] L. Zhang, S. Zhou, J. Guan, and J. Zhang, "Accurate few-shot object detection with support-query mutual guidance and hybrid loss," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14424–14432.

[50] C. Zhu, F. Chen, U. Ahmed, Z. Shen, and M. Savvides, "Semantic relation reasoning for shot-stable few-shot object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8782–8791.

[51] C. Lang, G. Cheng, B. Tu, C. Li, and J. Han, "Base and meta: A new perspective on few-shot segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 10669–10686, Sep. 2023.

[52] C. Lang, G. Cheng, B. Tu, and J. Han, "Learning what not to segment: A new perspective on few-shot segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 8047–8057.

[53] D. Kang and M. Cho, "Integrative few-shot learning for classification and segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9969–9980.

[54] Q. Fan, W. Pei, Y.-W. Tai, and C.-K. Tang, "Self-support few-shot semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 701–719.

[55] Y. Liu, N. Liu, X. Yao, and J. Han, "Intermediate prototype mining transformer for few-shot semantic segmentation," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022, Art. no. 2755.

[56] A. Afrasiyabi, H. Larochelle, J.-F. Lalonde, and C. Gagné, "Matching feature sets for few-shot image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9004–9014.

[57] J.-W. Zhang, Y. Sun, Y. Yang, and W. Chen, "Feature-proxy transformer for few-shot segmentation," in *Proc. 36th Int. Conf. Neural Inf. Process. Syst.*, 2022, Art. no. 476.

[58] H. Gao, J. Xiao, Y. Yin, T. Liu, and J. Shi, "A mutually supervised graph attention network for few-shot segmentation: The perspective of fully utilizing limited samples," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 4826–4838, Apr. 2024.

[59] Z. Tian, H. Zhao, M. Shu, Z. Yang, R. Li, and J. Jia, "Prior guided feature enrichment network for few-shot segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 1050–1065, Feb. 2022.

[60] K. Rakelly, E. Shelhamer, T. Darrell, A. Efros, and S. Levine, "Conditional networks for few-shot semantic segmentation," 2018. [Online]. Available: https://openreview.net/forum?id=SkMjFKJwG

[61] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, "PANet: Few-shot image semantic segmentation with prototype alignment," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9196–9205.

[62] X. Zhang, Y. Wei, Y. Yang, and T. S. Huang, "SG-one: Similarity guidance network for one-shot semantic segmentation," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3855–3865, Sep. 2020.

[63] Y. Liu, X. Zhang, S. Zhang, and X. He, "Part-aware prototype network for few-shot semantic segmentation," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K., 2020, pp. 142–158.

[64] G. Li, V. Jampani, L. Sevilla-Lara, D. Sun, J. Kim, and J. Kim, "Adaptive prototype learning and allocation for few-shot segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8334–8343.

[65] G. Zhang, G. Kang, Y. Yang, and Y. Wei, "Few-shot segmentation via cycle-consistent transformer," in *Proc. 35th Int. Conf. Neural Inf. Process. Syst.*, 2021, Art. no. 1683.

[66] S. Hong, S. Cho, J. Nam, S. Lin, and S. Kim, "Cost aggregation with 4D convolutional swin transformer for few-shot segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 108–126.

[67] X. Lu, W. Wang, J. Shen, D. J. Crandall, and L. Van Gool, "Segmenting objects from relational visual data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7885–7897, Nov. 2022.

[68] K. Nguyen and S. Todorovic, "iFS-RCNN: An incremental few-shot instance segmenter," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 7000–7009.

[69] C. Michaelis, I. Ustyuzhaninov, M. Bethge, and A. S. Ecker, "One-shot instance segmentation," 2018, *arXiv: 1811.11507*.

[70] J. Wu et al., "Towards open vocabulary learning: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 7, pp. 5092–5113, Jul. 2024.

[71] H. Zhang et al., "A simple framework for open-vocabulary segmentation and detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 1020–1031.

[72] J. Qin et al., "FreeSeg: Unified, universal and open-vocabulary image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 19446–19455.

[73] X. Wang, S. Li, K. Kallidromitis, Y. Kato, K. Kozuka, and T. Darrell, "Hierarchical open-vocabulary universal image segmentation," in *Proc. 37th Int. Conf. Neural Inf. Process. Syst.*, 2024, Art. no. 936.

[74] A. Zareian, K. D. Rosa, D. H. Hu, and S.-F. Chang, "Open-vocabulary object detection using captions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14393–14402.

[75] M. Minderer et al., "Simple open-vocabulary object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 728–755.

[76] X. Gu, T.-Y. Lin, W. Kuo, and Y. Cui, "Open-vocabulary object detection via vision and language knowledge distillation," in *Proc. Int. Conf. Learn. Representations*, 2021, pp. 1–20.

[77] Y. Du, F. Wei, Z. Zhang, M. Shi, Y. Gao, and G. Li, "Learning to prompt for open-vocabulary object detection with vision-language model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 14064–14073.

[78] M. Minderer, A. Gritsenko, and N. Houlsby, "Scaling open-vocabulary object detection," in *Proc. 37th Int. Conf. Neural Inf. Process. Syst.*, 2024, Art. no. 3191.

[79] T. Wang, "Learning to detect and segment for open vocabulary object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7051–7060.

[80] S. Wu, W. Zhang, S. Jin, W. Liu, and C. C. Loy, "Aligning bag of regions for open-vocabulary object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 15254–15264.

[81] C. Shi and S. Yang, "EdaDet: Open-vocabulary object detection using early dense alignment," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 15678–15688.

[82] D. Kim, A. Angelova, and W. Kuo, "Region-aware pretraining for open-vocabulary object detection with vision transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 11144–11154.

[83] J. Wang et al., "Open-vocabulary object detection with an open corpus," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 6736–6746.

[84] L. Wang et al., "Object-aware distillation pyramid for open-vocabulary object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 11186–11196.

[85] C. Feng et al., "PromptDet: Towards open-vocabulary detection using uncurated images," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 701–717.

[86] L. Yao et al., "DetCLIPv2: Scalable open-vocabulary object detection pre-training via word-region alignment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 23497–23506.

[87] C. Ma, Y. Jiang, X. Wen, Z. Yuan, and X. Qi, "CoDet: Co-occurrence guided region-word alignment for open-vocabulary object detection," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2024, pp. 71078–71094.

[88] S. Wu et al., "CLIPSelf: Vision transformer distills itself for open-vocabulary dense prediction," in *Proc. Int. Conf. Learn. Representations*, 2024, pp. 1–20.

[89] V. VS et al., "Mask-free OVIS: Open-vocabulary instance segmentation without manual mask annotations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 23539–23549.

[90] J. Wu et al., "Betrayed by captions: Joint caption grounding and generation for open vocabulary instance segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 21881–21891.

[91] F. Liang et al., "Open-vocabulary semantic segmentation with mask-adapted clip," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7061–7070.

[92] M. Gao et al., "Open vocabulary object detection with pseudo bounding-box labels," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 266–282.

[93] D. Huynh, J. Kuen, Z. Lin, J. Gu, and E. Elhamifar, "Open-vocabulary instance segmentation via robust cross-modal pseudo-labeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 7010–7021.

[94] S. Xu et al., "DST-Det: Simple dynamic self-training for open-vocabulary object detection," 2023, *arXiv:2310.01393*.

[95] W. Wang et al., "InternImage: Exploring large-scale vision foundation models with deformable convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 14408–14419.

[96] H. Touvron et al., "LLaMA: Open and efficient foundation language models," 2023, *arXiv:2302.13971*.

[97] Y. Fang et al., "EVA: Exploring the limits of masked visual representation learning at scale," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 19358–19369.

[98] C. Zhou et al., "A comprehensive survey on pretrained foundation models: A history from BERT to ChatGPT," 2023, *arXiv:2302.09419*.

[99] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, "COCA: Contrastive captioners are image-text foundation models," 2022, *arXiv:2205.01917*.

[100] A. Singh et al., "FLAVA: A foundational language and vision alignment model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 15617–15629.

[101] X. Li et al., "OMG-seg: Is one model good enough for all segmentation?," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 27948–27959.

[102] X. Li et al., "Transformer-based visual segmentation: A survey," 2023, *arXiv:2304.09854*.

[103] A. Kirillov et al., "Segment anything," 2023, *arXiv:2304.02643*.

[104] W. Ji, J. Li, Q. Bi, W. Li, and L. Cheng, "Segment anything is not always perfect: An investigation of SAM on different real-world applications," 2023, *arXiv:2304.05750*.

[105] J. Ma and B. Wang, "Segment anything in medical images," 2023, *arXiv:2304.12306*.

[106] S. He, R. Bao, J. Li, P. E. Grant, and Y. Ou, "Accuracy of segment-anything model (SAM) in medical image segmentation tasks," 2023, *arXiv:2304.09324*.

[107] J. Wu et al., "Medical SAM adapter: Adapting segment anything model for medical image segmentation," 2023, *arXiv:2304.12620*.

[108] K. Zhang and D. Liu, "Customized segment anything model for medical image segmentation," 2023, *arXiv:2304.13785*.
[109] R. Zhang et al., "Personalize segment anything model with one shot," 2023, *arXiv:2305.03048*.
[110] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–16.
[111] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[112] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, pp. 303–338, 2010.
[113] A. Gupta, P. Dollar, and R. Girshick, "LVIS: A dataset for large vocabulary instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5356–5364.
[114] Y. Xiao and R. Marlet, "Few-shot object detection and viewpoint estimation for objects in the wild," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 192–210.
[115] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[116] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv: 1711.05101*.
[117] B. Li, C. Wang, P. Reddy, S. Kim, and S. Scherer, "AirDet: Few-shot detection without fine-tuning for autonomous exploration," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 427–444.
[118] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.

Yue Han received the BEng degree in automation from the School of Southeast University, Nanjing, China, in 2021. She is currently working toward the PhD degree with the Institute of Cyber-Systems and Control, Advanced Perception on Robotics and Intelligent Learning Lab (APRIL), Zhejiang University, China, under the supervision of Dr. Yong Liu. Her primary research interests include computer vision, few-shot instance segmentation, and talking face generation.

Jiangning Zhang received the BS degree from Electronic Information School, Wuhan University, Wuhan, China, in 2017, and the PhD degree from the College of Control Science and Engineering, Zhejiang University, Hangzhou, China, in 2022. He is currently a research scientist with Youtu Lab, Tencent, Shanghai, China. His research interests include artificial intelligence generated content and deep learning.

Yabiao Wang received the master's degree from Zhejiang University, in 2016. He is currently a research scientist with Tencent Youtu Lab, China. He published more than 50 conference papers including CVPR, ICCV, ECCV, and AAAI etc. He won more than 20 challenge titles. His research interests are object detection, segmentation, few-shot learning, and AI generated content.

Chengjie Wang received the BS degree in computer science from Shanghai Jiao Tong University, China, in 2011, and double MS degrees in computer science from Shanghai Jiao Tong University, China and Weseda University, Japan, in 2014, and currently working toward the PhD degree with Shanghai Jiao Tong University. He is currently the research director of Tencent YouTu Lab. His research interests include computer vision and machine learning. He has published more than 100 papers on major Computer Vision and Artificial Intelligence Conferences such as CVPR, ICCV, ECCV, AAAI, IJCAI, and NeurIPS, and holds more than 100 patents in these areas.

Yong Liu received the BS degree in computer science and engineering and the PhD degree in computer science from Zhejiang University, Zhejiang, China, in 2001 and 2007, respectively. He is currently a professor with the Institute of Cyber-Systems and Control, Zhejiang University. His main research interests include: robot perception and vision, deep learning, Big Data analysis, and multi-sensor fusion. His research interests on machine learning, computer vision, information fusion, and robotics.

Lu Qi received the PhD degree from The Chinese University of Hong Kong, in 2021. He works as a postdoc with UC Merced and has about 8000 citations in Google Scholar. He obtained the Hong Kong PhD Fellowship, in 2017. His research interests include instance-level detection, image generation, and cross-modal pretraining. He was the senior program chair of AAAI 2023/2024 and area chair of ICLR 2024.

Ming-Hsuan Yang (Fellow, IEEE) is a professor of electrical engineering and computer science with the University of California, Merced. He serves as a program co-chair of the IEEE International Conference on Computer Vision (ICCV), in 2019, program co chair of the Asian Conference on Computer Vision (ACCV), in 2014, and general co-chair of ACCV 2016. He served as an associate editor of the *IEEE Transactions on Pattern Analysis and Machine Intelligence* from 2007 to 2011, and is an associate editor of *International Journal of Computer Vision*, *Image and Vision Computing*, and *Journal of Artificial Intelligence Research*. He received the NSF CAREER award, in 2012 and Google Faculty Award, in 2009.

Xiangtai Li received the PhD degree from Peking University, in 2022. He is working as a research scientist with Tiktok, Singapore. Previously, he worked as a research fellow with MMLab@NTU and a member of the Multimedia Laboratory with Nanyang Technological University. His research interests include computer vision and machine learning with a focus on scene understanding, segmentation, video understanding, and multi-modal learning. He regularly reviews top-tier conferences and journals, including CVPR, ICCV, ICLR, ECCV, ICML, NeurIPS, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, and *International Journal of Computer Vision*.