ITC 1/55
Information Technology and Control
Vol. 55 / No. 1/ 2026
pp. 100-124
DOI 10.5755/j01.itc.55.1.42926

Knowledge Distillation and Distribution Calibration for Few-Shot Object Detection

Received 2025/09/30 | Accepted after revision 2025/12/21

HOW TO CITE: Yang, Q., Tian, Y., Xu, T., Wang, Z., Sun, J., He, F. (2026). Knowledge Distillation and Distribution Calibration for Few-Shot Object Detection. *Information Technology and Control*, 55(1), 100-124. https://doi.org/10.5755/j01.itc.55.1.42926

Knowledge Distillation and Distribution Calibration for Few-Shot Object Detection

Qinghua Yang, Yan Tian
School of Artificial Intelligence; China University of Mining and Technology-Beijing; Beijing, China; e-mails: snowicelean@163.com; 13681362249@163.com

Tingting Xu
School of Computer Science and Technology; China University of Mining and Technology; Xuzhou, China; e-mail: tingting_xu@cumt.edu.cn

Zehua Wang
Department of Electrical and Computer Engineering; The University of British Columbia; Vancouver, V6T 1Z4, Canada; e-mail: zwang@ece.ubc.ca

Jing Sun*
School of Basic Education; Beijing Polytechnic College; Beijing, China; e-mail: sjing@bgy.edu.cn

Fangyuan He
College of Applied Science and Technology of Beijing Union University; Beijing, China; e-mail: ktfangyuan@buu.edu.cn

Corresponding author: sjing@bgy.edu.cn

Few-shot object detection (FSOD) aims to recognize and localize novel categories using only a limited number of annotated samples. Existing transfer learning–based approaches have attracted considerable attention for their structural simplicity and computational efficiency. However, merely fine-tuning the pretrained model parameters is insufficient to capture inter-class and intra-class relationships, thereby limiting the exploitation of transferable knowledge and further performance improvement. Therefore, we propose a prototype-guided semantic learning framework. By incorporating knowledge distillation, the method explicitly models transferable knowledge among classes for both classification and localization tasks. Specifically, a queue-based memory mechanism constructs dynamic class prototypes and distribution statistics in the feature space, enabling the modeling of class relationships. Classification knowledge transfer is achieved via

Kullback-Leibler divergence, while localization knowledge transfer is guided through regression reweighting. Furthermore, to alleviate distribution bias from the scarcity of novel class samples, an adaptive distribution correction and augmentation strategy based on optimal transport is introduced to enhance novel class classification. Experimental results demonstrate that, compared with baseline methods, the proposed approach achieves 6% and 5% improvements in novel class mAP under the 1-shot setting on the VOC and COCO datasets, respectively.

KEYWORDS: Object detection, few-shot, knowledge distillation, distribution calibration

1. Introduction

The Object detection [1, 26] is a crucial area in artificial intelligence, typically relying on a substantial amount of training data to ensure optimal model performance. However, in real-world applications, data scarcity is a common issue, posing a significant challenge in accomplishing various downstream tasks with limited data. Consequently, Few-Shot Object Detection (FSOD) has been proposed, aiming to leverage the prior knowledge learned from abundant base-class data and achieve effective recognition and precise localization of novel objects using only a few annotated samples. By addressing data scarcity and reducing annotation costs, FSOD allows models to quickly adapt to novel classes, facilitating real-world applications in domains such as autonomous driving, industrial inspection, and medical imaging, while also driving research on knowledge transfer and model generalization.

The current mainstream approaches for few-shot object detection can be broadly categorized into two types: meta-learning methods and transfer learning methods. Meta-learning based FSOD methods [8, 17, 28, 44, 48] employ an episodic training strategy, where each task (or mini-batch) comprises a few query images and a small set of support images. By training on base-class data, the model acquires class-level meta-knowledge, enabling generalization to novel classes through feature reweighting or class-specific weight generation. However, these methods typically require specialized data organization and employ an episodic training style. In contrast, transfer-learning based approaches, such as TFA [43], fine-tune only the final layer and have achieved remarkable results with significantly simpler architectures. Due to its simple structure, several studies [2, 9, 31, 39, 46] adopt this two-stage fine-tuning mechanism. Although the fine-tuning paradigm assumes that a detector trained on base classes can implicitly transfer class-agnostic prior knowledge to novel classes, the pretrained detector often struggles to effectively disentangle class-specific and class-agnostic knowledge in the absence of explicit modeling, thereby limiting potential improvements in model performance.

To circumvent the above issue, MFDC [46] introduced a framework that distills commonality knowledge to capture the multifaceted relationships between base and novel classes. Building on a similar idea, Pei et al. [30] employed knowledge distillation on a bag-of-visual-words representation to model object similarities, thereby facilitating the training of detection models. To address the risk of propagating erroneous predictions from teacher to student, Li et al. [20] proposed a structural causal model that incorporates conditional causal interventions, ensuring more reliable detection performance. In parallel, episode-based meta-learning approaches extensively leverage class prototypes to establish inter-class relationships. Motivated by this paradigm, we explore the integration of prototype-based strategies into transfer learning-based approaches to modeling class relationships. DeFRCN [31] employs an offline prototype-based classification model to calibrate the original classification scores. However, the module is only utilized during the reference phase. Inspired by the queue mechanisms in [31, 46], we dynamically construct class prototypes during the fine-tuning stage by utilizing the diversity positive proposals from the region of interest (ROI) heads via a queue-based storage mechanism. To explicitly transfer the learnable knowledge between classes, we decouple the FSOD task into classification and localization components using a fully connected layer and a convolutional layer, respectively, and construct both classification and localization prototypes.

The core idea is illustrated in Figure 1. Specifically, our objective is to distill knowledge from similar base classes to novel classes in both the classification and localization feature spaces, while simultaneously transferring the associated intra-class statistical information to achieve distribution calibration for novel classes.

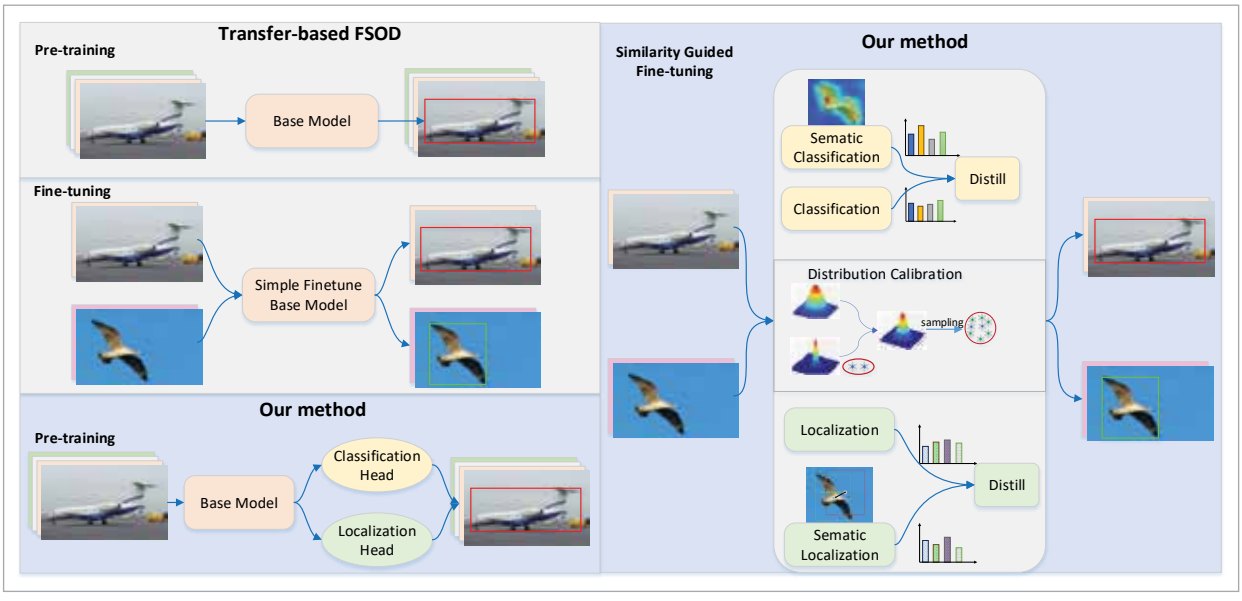
By utilizing class prototypes, we explicitly compute the semantic similarities between proposals and all classes, as well as the similarities among different classes. This establishes the semantic relationships between classes, as well as between classes and instances. Then, we apply Kullback-Leibler (KL) divergence to minimize the loss between these similarities and the linear classification results, which denotes the distillation of soft probabilities produced by classification models. Similarly, class localization similarities are used to reweight the object regression of novel classes, thereby explicitly enhancing class-specific localization features and distilling class-level localization knowledge into the proposals' regression. To address the data scarcity problem in novel classes, we dynamically transfer statistics from all related classes using optimal transport, which helps make the calibrated distribution more closely resemble the true distribution. Subsequently, new samples can be drawn from the calibrated distribution for data augmentation of novel classes, thereby further enhancing their detection performance.

Finally, we present the following contributions:

- This work introduces a queue-based storage mechanism to construct dynamic class prototypes, enabling the comprehensive characterization of both inter-class relationships and intra-class distribution features in transfer learning–based FSOD methods.
- This work proposes a decoupling of classification and localization tasks, performing knowledge distillation separately within their respective feature spaces to enable effective transfer of inter-class semantic information and localization knowledge.
- This work introduces a semantic relationship–driven adaptive class distribution calibration framework based on optimal transport, facilitating data augmentation from calibrated distributions and effectively addressing data scarcity in novel classes.
- Extensive experiments on the PASCAL VOC (VOC) and MS COCO (COCO) datasets demonstrate the competitiveness of our method, which is further validated on a real-world conveyor belt foreign object detection dataset.

Figure 1

A conceptual overview of knowledge transfer from base to novel classes.



The remainder of this paper is organized as follows. Section 2 reviews relevant work on FSL, FSOD, and distribution calibration methods in FSL. Section 3 elaborates the proposed architecture and training pipeline. Section 4 presents extensive experimental results and ablation studies to evaluate the effectiveness and robustness of our approach. Finally, Section 5 concludes the paper and outlines future research directions.

2. Related Work

This section reviews recent research progress from three key perspectives: few-shot Learning, few-shot object detection, distribution calibration in FSL.

2.1. Few-Shot Learning

Due to the risk of overfitting when only a limited number of samples are available, FSL typically employs paradigms that generalize prior knowledge—acquired from data-rich source tasks—to new tasks or novel classes with few labeled examples. Few-shot recognition/classification (FSC) aims to recognize new-class objects given only a few labeled examples for each class. In optimization-based meta-learning methods, the meta-learner gradually acquires generic meta-knowledge across tasks, subsequently helping the model quickly adapt to new tasks that were not encountered during training [10, 33], also known as 'learning-to-learn'. The training data typically uses episodes in the form of an N-way K-shot. Metric-based meta-learning approaches [37, 40] classify objects based on their nearest neighbors in the embedding space, utilizing distance metrics such as cosine similarity or Euclidean distance to labeled samples or class centers. Data augmentation approaches [13, 21], which employ generative adversarial networks (GANs) or features mixup, along with other feature transformations, are used to generate new samples and alleviate the data scarcity problem.

2.2. Few-Shot Object Detection

We investigate FSOD through two prominent research directions: meta-learning methods and transfer learning methods.

FSRW [17] is the first meta-learning method for standard FSOD, using YOLOv2 [34] as the framework and designing a lightweight CNN as a reweighting module to enhance efficiency and facilitate learning. Meta R-CNN [48] extends the Faster/Mask R-CNN framework and applies channel-wise soft attention to the ROI features, remodeling the R-CNN predictor heads. FSOD [8] utilizes few-shot support information to filter out background boxes that do not match the desired categories, effectively measuring the similarity between query proposal boxes and support objects using a multi-relation detector. VFA [12] proposed a method that integrates class-agnostic feature aggregation with variational feature aggregation to mitigate category bias and sample variance sensitivity in few-shot object detection. ICPE [28] generate high-quality prototypes tailored to each query image with an information-coupled prototype elaboration network. FPD [44] extracts fine-grained support features into prototypes, establishing detailed feature relationships between prototypes and the query feature map. Additionally, it improves the method for aggregating high-level features.

In contrast, transfer learning-based methods operate on simpler input formats and do not require the episodic training structure typical of meta-learning approaches. LSTD [3] is a pioneering work that proposes training a framework based on the Faster R-CNN [35] model initially on a large base dataset, followed by fine-tuning it on small novel data. RepMet [18] integrates its DML embedding module as a classification head into the OD model to perform FSOD. MPSR [45] introduces a multi-scale positive sample refinement branch to mitigate scale bias in FSOD. TFA [43] is a seminal work in FSOD research. It significantly enhances the performance of transfer learning methods and revises FSOD evaluation protocols to enable stable comparisons. FSCE [39] presents a more robust FSOD approach via contrastive proposals encoding. DeFRCN [31] extends Faster R-CNN by adding two Gradient Decoupled Layers (GDL) to adjust the degree of decoupling between the backbone, RPN, and RCNN head, along with a prototypical calibration block for multitask decoupling during inference time, achieving impressive performance and establishing a strong baseline. To address the issue of scattered distribution of novel features, FADI [2] first aligns each novel class with an associative base class and then disentangles the

classification branches for base and novel classes during the discrimination step. CDKT [42] developed an inter-class correlation transfer branch to capture and align the intrinsic relationships among core representations, and an intra-class diversity transfer branch that augments training by generating hallucinated features, thereby enriching the intra-class distribution of novel categories. NIFF [11] proposes a data-free knowledge distillation method for G-FSOD, which synthesizes instance-level features by leveraging statistical information of RoI features from the base model, without requiring access to the original base images.

Our method achieves performance comparable to the current state-of-the-art approach, NIFF, under similar experimental settings and methodological frameworks. However, a performance gap remains when compared to other SOTA methods such as DeViT [51], which employ more powerful backbone architectures. Specifically, DeViT utilizes a vision transformer (ViT) [6] model pretrained with DINO, while our model achieves performance similar to ViT-S/14, but lags behind ViT-B and ViT-L. Moreover, when compared to approaches based on large-scale vision-language pretrained models, such as GroundingDINO [26], the performance gap becomes more pronounced. It is important to note, however, that these methods are grounded in fundamentally different research paradigms. The core idea of our work is to explicitly learn transferable semantic knowledge from base classes, rather than relying solely on increasingly powerful models to improve performance.

2.3. Calibrate Distribution in Few-Shot Learning

Due to limited labeled samples, the data distribution often deviates from the true class distribution. To address this, Salakhutdinov et al. [36] propose transferring prior knowledge of category means and variances from base to novel classes. Distribution calibration [49] extends this idea by introducing Gaussian modeling in FSL, calibrating novel-class distributions using statistics from similar base classes, which spurred interest in distribution-based methods. For example, [29, 27, 41] investigate this in FSL, while [46, 52] extend it to FSOD. In [52], the backbone is frozen during fine-tuning to compensate for shifts in base class distributions. However, since the pre-trained model is trained on base classes and is highly biased toward them, freezing the backbone hinders the learning of novel features.

Previous works [36, 41, 46] compute distances between novel features and base prototypes, select the most similar base classes, and transfer their averaged statistics to refine the novel feature distribution.

However, this approach has two main limitations: (1) Coarse top-k selection (e.g., $k = 2$ in [46, 49]) may overlook other relevant base classes, harming performance when the chosen ones have low similarity. (2) Since different base classes vary in similarity to the novel class, they should contribute unequally. To address this, we adopt an optimal transport framework that adaptively transfers statistics by weighting base classes according to the Sinkhorn distance in feature space, and calibrate the novel class mean and variance using the resulting weight matrix.

3. Methods

In this section, we first present the problem settings for few-shot object detection. Subsequently, we describe our model architecture for pre-training and fine-tuning. The construction of class prototypes is then discussed, followed by an explanation of distribution calibration for data augmentation. Finally, we describe knowledge distillation for both classification and localization tasks.

3.1. Problem Settings

We follow the standard settings for few-shot object detection established in previous works [31, 43, 46]. Specifically, there are two datasets, the base dataset D_b, which is large and contains abundant annotated instances of classes C_{base}, and the novel dataset D_n, which is smaller and contains only a few (K-shot, $K \in \{1, 2, 3, 5, 10, 30\}$ in the experiments) annotated instances for classes C_{novel}, $C_{base} \cap C_{novel} = \emptyset$. N_b and N_n represent the number of images in two datasets, respectively, with $N_b \gg N_n$. The objective is to train a model using the provided data (D_{base}, D_{novel}) to perform an object detection task on a test dataset D_{test}.

Faster RCNN [35] is a classic representative of a two-stage object detection model, consisting primarily of three modules: a backbone for feature extraction, a

region proposal network (RPN) to generate class-agnostic proposals, and a detection head for classification and localization tasks. Two-stage fine-tuning-based FSOD methods extend Faster R-CNN to FSOD, commonly employing ResNet-50 or ResNet-101 [15] as the backbone, occasionally in conjunction with FPN [23]. Our model adheres to this pipeline.

3.2. Two Stage Training Method

In the pre-training stage, we train the model on abundant data from base classes, using standard Faster R-CNN losses:

$$L_{det} = L_{rpn} + L_{cls} + \beta L_{reg} . \quad (1)$$

where L_{rpn} represents the RPN loss, which distinguishes foreground from background and applies the smoothed L1 loss [35] for anchor regression. L_{cls} represents the cross-entropy loss for classification in the detection head, L_{reg} denotes the smoothed L1 loss for bounding box regression, and β is a hyperparameter to balance the losses.

The architecture of the pre-training model is shown in Figure 2. We follow the approach of DeFRCN and add two GDLs. Additionally, we decouple the features in the detection head to address two primary challenges: 1) Multi-tasking in FSOD, where shared ROI features are used for both classification and localization tasks, but the classification task typically requires translation-invariant features, whereas localization tasks require translation-variant features [38]. Aggregated features may be detrimental to the regression task. 2) Decoupling features to facilitate knowledge transfer during the fine-tuning stage. We apply distinct methods to the classification and localization tasks. Inspired by the work of Double-Head [47], it was found that the Fully Connected (FC) head exhibits greater spatial sensitivity than the convolution head (conv-head) and is therefore more suitable for the classification task, while the conv-head is better suited for the localization task. We employ a fully connected layer in the classification head and a convolutional head in the regression head to decouple the features, allowing the classification and localization tasks to be treated independently during the fine-tuning stage.

As illustrated in Figure 3, the model during the fine-tuning stage primarily consists of class prototype construction, followed by three key modules: data augmentation, classification distillation, and localization distillation. The following sections provide a detailed explanation of each component.

3.3. Class Prototypes Construction

In the fine-tuning stage, to learn more semantic knowledge about novel classes, we unfreeze part of the backbone, the RPN, and the ROI head in the pre-trained model. However, this makes it challenging to preserve the pre-trained statistics of base classes, rendering the direct transfer of these statistics inappropriate. This occurs because the model parameters are dynamically updated during the fine-tuning process. Inspired by MoCo [14], which maintains a dictionary memory bank where the size is no longer constrained by the batch size, we randomly sample the dictionary for each mini-batch from the memory bank. The proposals in the detection head introduce

Figure 2
The architecture of the pre-training model.

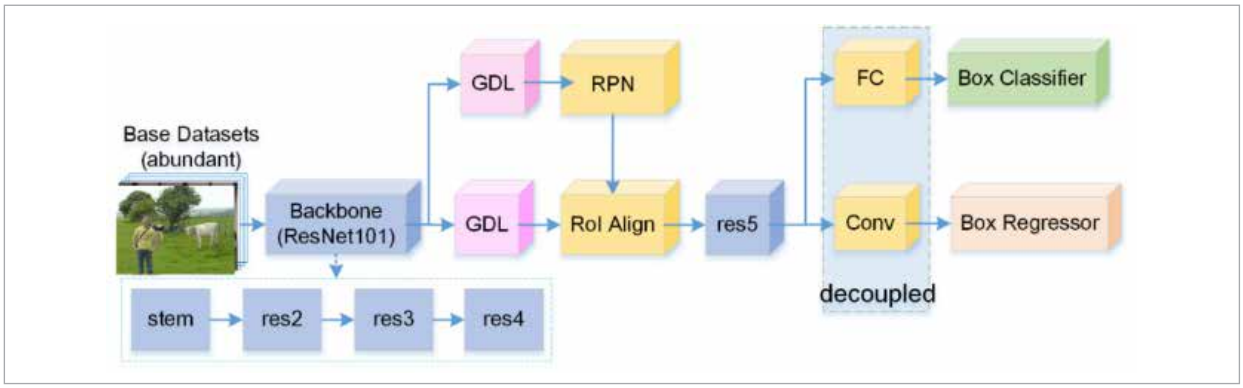
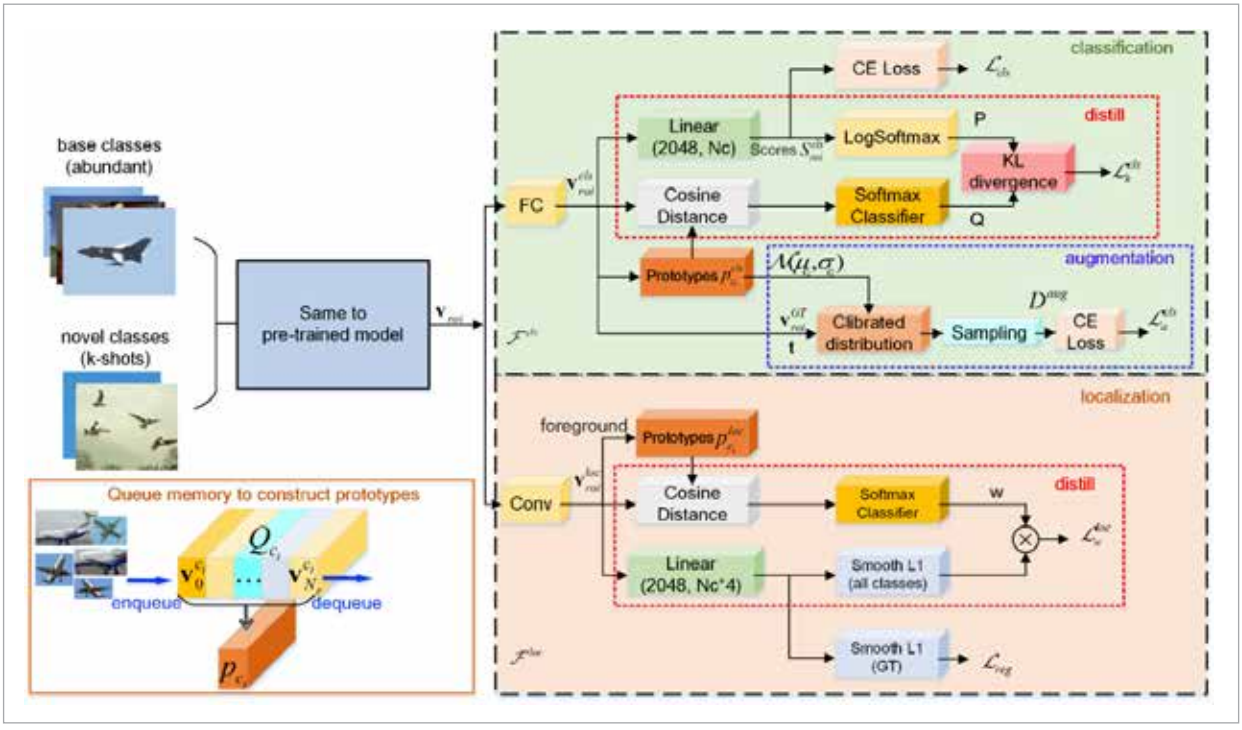


Figure 3

The model architecture during the fine-tuning stage, including the pre-training backbone, class prototype construction module, knowledge-distillation-based classification and localization modules, and the distribution-correction-based data augmentation module.



diversity to the classes. Therefore, we maintain class queues to store features of positive proposals, with each queue storing the features of a single class to obtain class prototypes.

3.3.1. Preprocess the Features of Positive Proposals

To construct class prototypes, we extract features from the positive proposals within the ROIs and exclude the negative proposals and background. Although we assume that the features follow a Gaussian distribution, the actual feature output from the network is likely to deviate from an ideal Gaussian distribution. To address this issue, following [49], we employ Tukey's Ladder of Powers transformation [16], which helps reduce the skewness of the distribution.

3.3.2. Queue Mechanism to Get Class Prototypes and Statistics

The queue memory mechanism dynamically stores feature vectors, as illustrated in Figure 4, after applying Tukey's Ladder of Powers transformation, the jth feature vector of class c_i is denoted as $\mathbf{v}_j^{c_i}$, where $j \in \{1, ..., N_{GT}\}$. In a mini-batch, ROI features are assigned to different class queues based on their corresponding ground truth labels, with each queue having a maximum length of L, to save memory and reduce subsequent computational complexity. Thus, following the 'first in, first out' rule of the queue, new values are enqueued one by one, and old values are dequeued when the queue reaches its maximum capacity. Therefore, the queue is dynamically updated, in contrast to the fixed queue mechanism used by [46].

Using the class queues, we can efficiently compute statistics for each class. Specifically, we compute both the mean and variance for each class. The mean of ROI feature vectors from the same class c_i, also referred to as the class prototype, is calculated by averaging each dimension of the vector:

$$\boldsymbol{\mu}_{c_i} = \frac{1}{N_r}\sum_{j=1}^{N_r}\mathbf{v}_j^{c_i}, \tag{2}$$

Figure 4
Overview of the class prototype module, which mainly includes positive proposal selection, Tukey transformation, and computation of Gaussian distribution statistics from the feature queue.

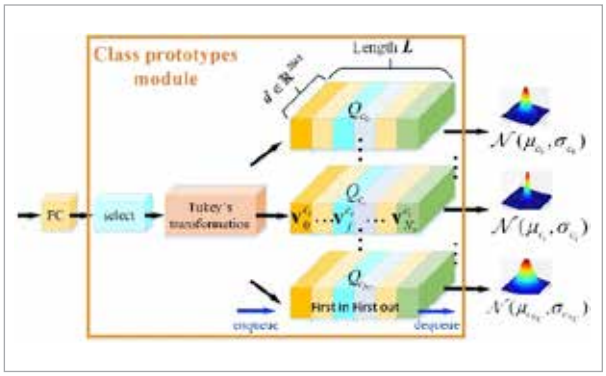
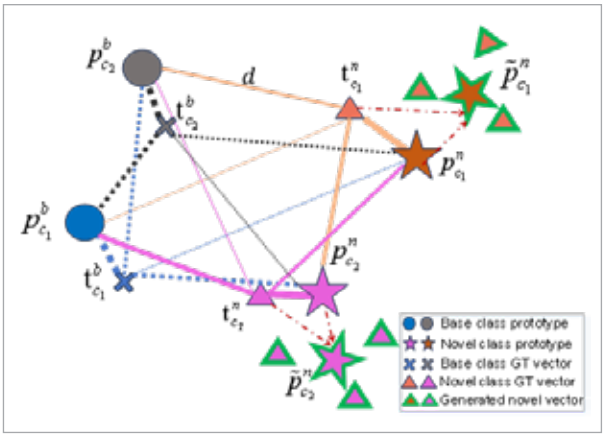


Figure 5
Data augmentation based on optimal transport between semantic features.



where N_r represents the total number of positive ROI feature vectors in class c_i, for $i \in \{1, ..., N_c\}$, where N_c denotes the total number of classes. We do not use a covariance matrix to represent class statistics as is commonly done in few-shot learning. This is due to the fact that the feature vectors have a dimensionality of 2048, making the computation computationally expensive. The class variance is obtained using the following equation:

$$\sigma_{c_i}^{\ 2} = \frac{1}{N_r - 1} \sum_{j=1}^{N_r} (\mathbf{v}_j^{c_i} - \boldsymbol{\mu}_{c_i})^2 . \quad (3)$$

Similarly, for the localization branch, we compute the prototypes of localization features using Equation (2), where the input data are the positive ROI features processed by the decoupled convolution layer.

3.4. Data Augmentation for Novel

Classes Data augmentation is a widely adopted strategy for mitigating data scarcity. Building on the idea that intra-class variance is shared across classes and can be modeled with common distributions [25], in this study, we propose a semantic relationship–based data augmentation strategy. As shown in Figure 5, feature vectors are treated as semantic entities, and a semantic graph encodes their similarity to class prototypes. Distances between GT box features and prototypes are computed, and optimal class weights are obtained via the Sinkhorn algorithm to calibrate novel class statistics. New samples are drawn from the calibrated distribution, and classification loss is computed using cross-entropy.

3.4.1. Estimating the Weight Matrix Using Optimal Transport

Once we have the statistics for all classes, the next step is to transfer this knowledge from the base classes to calibrate the distribution of the novel classes. In this paper, we focus on the relationships among all classes, rather than solely on those between the support vectors of novel and base classes.

As previously noted, more similar classes tend to exhibit more similar distribution statistics. We perform augmentation based on the GT boxes within the detection head. Specifically, within a mini-batch, the Euclidean distance between the j-th GT vector t_j and the prototypes of class c_i:

$$d_{\mathbf{t}_j, c_i} = \left\| \mathbf{t}_j - \boldsymbol{\mu}_{c_i} \right\| . \quad (4)$$

Then, calculate the distance to all classes for all GT vectors (in a batch) to obtain a distance matrix. The number of rows in this matrix corresponds to the number of GT vectors N_{GT} in the current mini-batch, with each row representing the distance between the feature vector of a specific GT and the prototypes of all classes.

About how much statistical knowledge from each base class should be transferred to each novel class. We decompose this problem into two parts: trans-

ferring the mean and variance separately. Theoretically, let $w_{\mathbf{t}_j,c_i}$ represent the weight coefficient to be transferred from class c_i to $\mathbf{t}_j$. This problem can be formulated as an optimal transport (OT) problem, where the objective is to identify similar classes and determine appropriate transfer weights that minimize the cost, also known as the Wasserstein distance (or Earth Mover's distance), denoted as:

$$\min_{w_{\mathbf{t}_j,c_i}} \sum_{j=1}^{N_{GT}} \sum_{i=1}^{N_C} d_{\mathbf{t}_j,c_i} w_{\mathbf{t}_j,c_i}, \tag{5}$$

where $w_{\mathbf{t}_j,c_i} > 0$. In the context of OT, the transfer weight provided by $\mathbf{t}_j$ is denoted as $u_{\mathbf{t}_j} = \sum_{i=1}^{N_C} w_{\mathbf{t}_j,c_i}$, and the transfer weight accepted by class c_i is denoted as $v_{c_i} = \sum_{j=1}^{N_{GT}} w_{\mathbf{t}_j,c_i}$.

Due to computational complexity, we employ the Sinkhorn distance [5], which allows for efficient computation of optimal transport distances:

$$\min_{w_{\mathbf{t}_j,c_i}} \sum_{j=1}^{N_{GT}} \sum_{i=1}^{N_C} d_{\mathbf{t}_j,c_i} w_{\mathbf{t}_j,c_i} - \frac{1}{\lambda} h(D), \tag{6}$$

where $h(D) = -\sum_{j=1}^{N_{GT}} \sum_{i=1}^{N_C} d_{\mathbf{t}_j,c_i} \log d_{\mathbf{t}_j,c_i}$ represents the information entropy of D. $\lambda \in [0, \infty]$ is a regularization parameter that controls the strength of the entropic regularization; decreasing its value encourages a more homogeneous distribution. As described in [27], we employ the smoothed version to solve the optimization process. The optimizer D^* is then formulated as:

$$D^* = diag(u^*) K diag(v^*), \tag{7}$$

where $K = exp(\lambda W)$, u^* and v^* are computed by iteration. Here, we obtain the weight matrix W, which represents the optimal weight matrix that transferred from all classes to the current GT box feature vectors. We refer to it as the OT mean, which is formally expressed as:

$$\boldsymbol{\mu}_{OT} = \sum_{i=1}^{N_C} W \mu_i, \tag{8}$$

Subsequently, we utilize it to calibrate the distribution of the novel classes.

3.4.2. Adaptive Calibration of Novel Class Statistics

The calibrated mean of the novel class is obtained using the previously derived $\boldsymbol{\mu}_{OT}$ and $\mathbf{t}$ that from novel classes, as expressed by the following equation:

$$\boldsymbol{\mu}' = \alpha \mathbf{t} + (1-\alpha) \boldsymbol{\mu}_{OT}, 0 \le \alpha \le 1. \tag{9}$$

The calculated variance is directly obtained from the weight matrix W:

$$\boldsymbol{\sigma}' = \sum_{i=1}^{N_C} W \boldsymbol{\sigma}_{c_i}. \tag{10}$$

The calibrated distribution of the novel class is $\tilde{p} \sim \mathrm{N}(\boldsymbol{\mu}', \boldsymbol{\sigma}')$, which alleviates the problem of biased distributions caused by the scarcity of novel class samples.

3.4.3. Sampling New Data for Novel Classes

In a mini-batch, we obtain a set of calibrated statistics for the novel class c_i denoted as $\mathrm{S}^{c_i} = \left(\boldsymbol{\mu}_1', \boldsymbol{\sigma}_1'\right), \cdots, \left(\boldsymbol{\mu}_{N_{GT}}', \boldsymbol{\sigma}_{N_{GT}}'\right)$. For each set, we can sample additional samples with labels consistent with those of the corresponding GT feature vectors:

$$\begin{aligned} &D_{aug}^{c_i} = \{(x, y) \mid x \sim \mathrm{N}(\boldsymbol{\mu}', \boldsymbol{\sigma}')\} \\ &\forall (\boldsymbol{\mu}', \boldsymbol{\sigma}') \in \mathrm{S}^{c_i} \ \& \ c_i \in C_{novel} \end{aligned}. \tag{11}$$

The total number of samples generated per class is set as a hyperparameter within a batch. For a novel class, the generated data is used to train the classifier for detection with a cross-entropy loss. We then introduce a separate loss function for data augmentation, denoted as:

$$\mathrm{L}_a^{cls} = \sum_{(x,y) \sim D_{aug}} -\log P(y \mid x; \theta). \tag{12}$$

Formally, this is identical to the standard cross-entropy loss function, except that the input data differs. The above method is summarized in the following Algorithm 1.

Algorithm 1 Data Augmentation Pipeline

Requires: GT feature vectors $\mathbf{t}$ of novel classes, mean μ and variance σ of class prototypes.

Outputs: The augmented data vectors of the novel classes.

1: Calculate the Euclidean distance d between $\mathbf{t}$ and μ. Equation (4).
2: Calculate the weight matrix $\mathbf{W}$ using the Sinkhorn distance based on d. Equations (5), (6) and (7).
3: **for** each $\mathbf{t}$ **do**
4: Calculate the OT mean $\boldsymbol{\mu}_{OT}$. Equation (8).
5: Calibrate the distribution of the novel class N ($\boldsymbol{\mu}'$, $\boldsymbol{\sigma}'$). Equations (9) and (10).
6: Sample new data. Equation (11).
7: **end for**
8: Calculate the classification loss of the new samples. Equation (12).

To reduce potential bias from base class statistics, we introduce a balancing hyperparameter α in Equation (9). We evaluate its effect by varying α from 0.4 to 0.9 in increments of 0.1 and report the averaged optimal values across different data splits. The best performance is achieved with α = 0.6 for the 1-and 3-shot settings and α = 0.7 for the 5-shot setting. In high-shot settings (e.g., 10-shot), calibrating novel class distributions with base class statistics tends to introduce boundary samples, leading to misclassification. To prevent this, we discard base-to-novel transfer and rely exclusively on novel class statistics with augmentation, as formally defined below:

$$\begin{aligned}\boldsymbol{\mu}' &= \alpha\mathbf{t} + (1-\alpha)\boldsymbol{\mu}^{\text{novel}} \\ \boldsymbol{\sigma}' &= \boldsymbol{\sigma}^{\text{novel}}\end{aligned}. \tag{13}$$

In the 10-shot VOC setting, we set α = 0.5 and sample 10 instances per novel-class GT box feature with a sampling ratio of 0.6.

3.5. Distilling Knowledge for Classification

Based on the class prototypes, we calculate the cosine similarities between novel objects and all classes. We treat the cosine similarity distribution as a teacher to guide the novel classes to learn knowledge from base classes. Specifically, the KL divergence is computed between the predicted probability distributions from a linear classifier and those from a cosine similarity-based classifier. Formally, the loss is minimized as follows:

$$\mathrm{L}_k^{cls} = D_{KL}(\mathbf{P} \,||\, \mathbf{Q}), \tag{14}$$

where $\mathbf{P}$ represents the normalized distribution of the classification probabilities from a linear classifier, and $\mathbf{Q}$ denotes the cosine similarity distribution, which computes the similarity between proposals and class prototypes. This distribution retains the similarities between proposals of novel classes and all novel classes, as well as between proposals of base classes and their ground truth and novel classes. The similarity between the proposals and the background class is set to zero.

3.6. Distilling Knowledge for Localization

In object localization, beyond the bounding box regressor for the associated ground truth, similar classes also share similar regressors. Similar to prototypes used for classification, we aggregate the localization offsets for each class and predict the localization distribution based on the cosine similarity between ROI feature vectors and aggregated class feature vectors, which serve as coefficients to reweight the standard regressors. Formally, the localized reweighting loss L_w^{loc} is defined as follows:

$$\mathrm{L}_w^{loc} = \sum_{i=1}^{N_C} \mathbf{W}_{\mathbf{v}}^{c_i} \times Smooth_{L1}\left(\delta_p^{c_i}, \delta_{GT}^{c_i}\right), \tag{15}$$

where $\mathbf{W}_{\mathbf{v}}$ represents the weights derived from the cosine similarity between ROI feature vectors for location $\mathbf{v}^{loc}$ and the localization features of all classes. $\delta_p^{c_i}$ represents the standard predicted regression offset for foreground proposals of class c_i, while $\delta_{GT}^{c_i}$ represents the offsets between the foreground proposal vectors and the corresponding regression ground truths.

In addition to the standard Faster R-CNN losses L_{det} used during the pre-training stage, our model incorporates three additional losses, as shown in Figure 3, and the total loss during the fine-tuning stage is:

$$\mathrm{L}_{ft} = \mathrm{L}_{\text{det}} + \lambda_k \mathrm{L}_k^{cls} + \lambda_a \mathrm{L}_a^{cls} + \lambda_r \mathrm{L}_w^{loc}, \tag{16}$$

where λ_k, λ_a, and λ_r are hyperparameters that control the relative importance of each loss term.

4. Experiments

In this section, we first describe the experimental settings, followed by a comparison of our method with previous approaches to demonstrate its effectiveness. Finally, ablation studies are presented.

4.1. Experimental Setting

The following section provides a detailed description of the datasets, experimental settings, and evaluation metrics.

4.1.1. Datasets

We evaluate our method on popular benchmark datasets in the FSOD domain: PASCAL VOC [7] and MS COCO [24], using the same data splitting settings as in [19, 31, 43] for a fair comparison. For PASCAL VOC, there are a total of 20 categories, which are divided into three randomly split groups, each containing 15 base classes and 5 novel classes. The training data, which includes all base class data and the given support novel class data, is sampled from the combined VOC07 and VOC12 train/val sets. The VOC07 test set is used for evaluation purposes. For each novel class, K-shot support samples are provided, with $K \in \{1, 2, 3, 5, 10\}$ in the experiments. This implies that only K object samples are available for each novel class. For COCO, the 20 classes that overlap with PASCAL VOC are selected as novel classes, while the remaining 60 classes are used as base classes, with $K \in \{1, 2, 3, 5, 10, 30\}$. We use 5,000 images from the validation set for evaluation and the remaining images for training.

4.1.2. Evaluation Protocols

The earlier research in FSOD [3, 17, 48] has primarily focused on the performance of novel classes. However, the performance of base classes also plays a crucial role. The improved performance of novel classes is often associated with a potential decrease in the performance of base classes. Therefore, it is more equitable to compare the performance across both categories, a concept referred to as generalized few-shot object detection (G-FSOD). In this context, following the evaluation protocol revised in TFA [43] and adopted by DeFRCN, FSCE, MFDC, and others, we report the mean AP50 (matching threshold = 0.5) for all classes (AP), base classes (bAP), and novel classes (nAP) for the PASCAL VOC dataset. Additionally, we report the COCO-style mean average precision (mAP) for all classes, base classes, and novel classes for the COCO dataset.

4.1.3. Implementation Details

We utilize Faster R-CNN as the primary detection framework, with a ResNet-101 backbone pretrained on the ImageNet dataset. The detector is trained using a mini-batch size of 8 on 2 GPUs and 16 on 4 GPUs for VOC and COCO, respectively. We employ SGD for optimization, with a momentum of 0.9 and weight decay of $5e^{-5}$. The learning rate is set to 0.01 during base training and 0.005 during few-shot fine-tuning. During pre-training, we decouple classification and localization, utilizing a fully connected layer and a convolutional layer. During fine-tuning, we maintain dynamic queues to store the proposal representations of classification and localization, which are then used to compute the prototypes for each class. The maximum length of these queues is 2048. The weight hyperparameters in the loss function of Equation (16) are set to λ_k=0.1, λ_a=0.1 and λ_r=0.1. Initially, we run 200 iterations to construct class prototypes that can fill the queues with novel class prototypes, and subsequently use these prototypes to more explicitly guide the model in transferring knowledge from base classes.

4.2. Comparison with FSOD Methods

We compare our method with other approaches on standard benchmarks, including VOC and COCO.

4.2.1. Results on PASCAL VOC

Following the previous work, we conduct experiments on three different base/novel class splits of the dataset. The evaluation results of VOC on three different data splits are presented in Table 1. The nAP50 performance across three novel sets is compared using different FSOD methods. The results demonstrate a significant improvement over other approaches, showcasing the effectiveness of our method. Specifically, in low-shot settings, compared with the competitive DeFRCN baseline, we observe consistent improvements of up to 9.1%, 5.9%, and 3.1% for the 1-shot scenario across the three split groups, which proved the effectiveness of data augmentation.

Table 1
Experimental results on the Pascal VOC dataset (mAP50 for novel classes).

method	Novel Set 1 (shot)					Novel Set 2 (shot)					Novel Set 3 (shot)				
	1	2	3	5	10	1	2	3	5	10	1	2	3	5	10
LSTD	8.2	1.0	12.4	29.1	38.5	11.4	3.8	5.0	15.7	31.0	12.6	8.5	15.0	27.3	36.3
FSRW	14.8	15.5	26.7	33.9	47.2	15.7	15.3	22.7	30.1	40.5	21.3	25.6	28.4	42.8	45.9
MetaDet	18.9	20.6	30.2	36.8	49.6	21.8	23.1	27.8	31.7	43.0	20.6	23.9	29.4	43.9	44.1
Meta R-CNN	19.9	25.5	35.0	45.7	51.5	10.4	19.4	29.6	34.8	45.4	14.3	18.2	27.5	41.2	48.1
TFA w/cos	39.8	36.1	44.7	55.7	56.0	23.5	26.9	34.1	35.1	39.1	30.8	34.8	42.8	49.5	49.8
Retentive	42.4	45.8	45.9	53.7	56.1	21.7	27.8	35.2	37.0	40.3	30.2	37.6	43.0	49.7	50.1
FSCE	44.2	43.8	51.4	61.9	63.4	27.3	29.5	43.5	44.2	50.2	37.2	41.9	47.5	54.6	58.5
FADI	50.3	54.8	54.2	59.3	63.2	30.6	35.0	40.3	42.8	48.0	45.7	49.7	49.1	55.0	59.6
DeFRCN	57.0	58.6	64.3	67.8	67.0	35.8	42.7	51.0	54.5	52.9	52.5	56.6	55.8	60.7	62.5
FCT	49.9	57.1	57.9	63.2	67.1	27.6	34.5	43.7	49.2	51.2	39.5	54.7	52.3	57.0	58.7
CDKT-DeFRCN	48.6	60.6	64.3	69.0	70.8	33.0	42.1	46.6	52.4	53.3	40.2	52.9	55.2	61.6	63.7
ICPE	54.3	59.5	62.4	65.7	66.2	33.5	40.1	48.7	51.7	52.5	50.9	53.1	55.3	60.6	60.1
VFA	57.7	64.6	64.7	67.2	67.4	41.4	46.2	51.1	51.8	51.6	48.9	54.8	56.6	59.9	58.9
ours	**66.1**	**67.7**	**67.8**	**69.9**	**71.1**	**41.7**	**47.5**	**53.0**	**54.7**	**53.9**	**55.6**	**60.6**	**60.8**	**62.7**	**65.0**

Table 2
The AP, bAP and nAP results under the different shot settings for VOC novel split 1.

Method	1-shot			2-shot			3-shot			5-shot			10-shot		
	AP	bAP	nAP	AP	bAP	nAP	AP	bAP	nAP	AP	bAP	nAP	AP	bAP	nAP
FRCN+ft	55.4	68.9	15.2	57.1	69.4	20.3	56.8	66.1	29.0	60.1	66.7	40.1	60.9	66.0	45.5
TFA w/cos	69.7	79.6	39.8	68.2	78.9	36.1	70.5	79.1	44.7	73.4	79.3	55.7	72.8	78.4	56.0
DeFRCN	73.1	78.4	57.0	73.2	78.1	58.6	73.7	76.8	64.3	75.1	77.6	67.8	74.4	76.8	67.0
NIFF	75.9	-	-	76.9	-	-	77.3	-	-	70.6	-	-	77.5	-	-
Ours	75.8	79.1	66.1	76.3	79.1	67.7	75.6	78.1	68.0	76.4	78.6	69.9	76.4	78.3	70.8

We observe that improvements in novel class performance are often accompanied by a slight drop in base class performance. To ensure a fair comparison, we also assess the model's performance on base classes. In Table 2, we report the AP, bAP, and nAP results under the different shot settings for VOC novel split 1. Compared with NIFF, our method achieves a 5.8% improvement under the 5-shot setting, while exhibiting slight decreases in other few-shot scenarios. This difference can be attributed to the distinct objectives of the two approaches: NIFF is specifically designed to mitigate catastrophic forgetting in general few-shot object detection, whereas our work is primarily dedicated to enhancing detection performance on novel classes. While compared to the baseline DeFRCN, our method continues to perform well on base classes while improving performance on novel classes.

4.2.2. Results on MS COCO

We observe consistent performance improvements on the MS COCO benchmark. As shown in Table 3, compared to MFDC, our method consistently outperforms it in the low-shot range. Compared with the baseline DeFRCN, our method achieves performance improvements of 5.0%, 2.5%, 2.6%, 2.2%, and 1.1% under the 1-, 2-, 3-, 5-, and 10-shot settings, respectively. It can be observed that the improvement is more pronounced with fewer samples, which further demonstrates the effectiveness of the proposed method in extremely low-data regimes. Notably, no new samples are generated for data augmentation in the 30-shot setting, as experimental results indicate no performance gain. As the number of samples for novel classes increases, the intra-class distribution becomes more stable, thereby reducing the effectiveness of distribution calibration.

4.2.3. Average Recall (AR) Results and Precision-recall (PR) Curves

For evaluation on the VOC test set, we compute mAP using the 11-point interpolation method. We report average recall when the Intersection over Union (IoU) is 0.5 for the VOC dataset under the split 1, seed 0 settings. Results for all classes, base classes, and novel classes across different shot settings are shown in Table 4. We observe that the AR values for all classes are high, although the values for novel classes are slightly lower than those for base classes. As the number of shots increases, the AR values for both novel and all classes tend to improve.

Table 3

Experimental results on the COCO dataset.

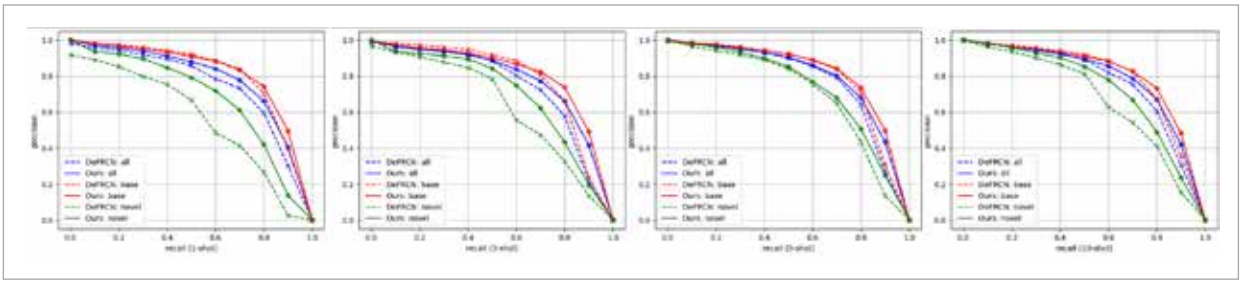
Method	1-shot		2-shot		3-shot		5-shot		10-shot		30-shot	
	AP	AP75	AP	AP75	AP	AP75	AP	AP75	AP	AP75	AP	AP75
FSRW	–	–	–	–	–	–	–	–	5.6	4.6	9.1	7.6
FSCE	–	–	–	–	–	–	–	–	11.9	10.5	16.4	16.2
CME	–	–	–	–	–	–	–	–	15.1	16.4	16.9	17.8
TFA w/cos	3.4	3.8	4.6	4.8	6.6	6.5	8.3	8.0	10.0	9.3	13.7	13.4
MPSR	2.3	2.3	3.5	3.4	5.2	5.1	6.7	6.4	9.8	9.7	14.1	14.2
FADI	5.7	6.0	7.0	7.0	8.6	8.3	10.1	9.7	12.2	11.9	16.1	15.8
DeFRCN	6.5	6.9	11.8	12.4	13.4	13.6	15.3	14.6	18.6	17.6	22.5	22.3
MFDC	10.8	11.6	13.9	14.8	15.0	15.5	16.4	17.3	19.4	20.2	22.7	23.2
CDKT-DeFRCN	–	–	–	–	–	–	–	–	18.5	17.9	22.3	22.0
NIFF	-	-	-	-	-	-	15.9	-	18.8	-	20.9	-
ours	11.5	12.7	14.3	15.5	16.0	16.2	17.5	17.4	19.7	20.3	22.5	23.3

Table 4

Average recall on all classes, base classes, novel classes when IoU=0.5 for VOC.

shot	$AR_{all}^{IoU=.50}$	$AR_{base}^{IoU=.50}$	bird	bus	cow	motorbike	sofa	$AR_{novel}^{IoU=.50}$
1	0.943	0.953	0.800	0.934	0.992	0.902	0.925	0.910
2	0.943	0.949	0.828	0.962	0.984	0.926	0.929	0.926
3	0.944	0.947	0.828	0.967	0.975	0.938	0.954	0.933
5	0.950	0.952	0.839	0.972	0.984	0.951	0.967	0.942
10	0.946	0.950	0.843	0.962	0.984	0.945	0.946	0.936

Figure 6
PR curves under different shots on the dataset VOC.



The comparison of PR curves between DeFRCN and our method for different shot settings is shown in Figure 6. Our method outperforms others on VOC across different shot settings. Each subfigure includes the PR curves (IoU=0.5) of the average of all classes, base classes and novel classes for DeFRCN and our method.

We report the average recall for IoU values in the range [0.50:0.05:0.95] with a maximum of 100 detections (maxDets) for the COCO dataset under seed 0 settings. Results for all classes, base classes, and novel classes across different shot settings are presented in Table 5. We further observe that the overall recall rate increases with the number of shots, while the gap between base classes and novel classes diminishes as the number of shots increases. The comparison of PR curves between DeFRCN and our method for different shot settings is shown in Figure 7.

In summary, compared to the baseline, the improvement is evident, particularly in low-shot scenarios. Compared to the current state-of-the-art work, MFDC, our method shows improvements in most results. The inference time is 0.08 seconds per image on an RTX 3090 GPU. Methods based on large models, such as [22], which relies on CLIP [32] and uses synthetic data, are not included in this paper.

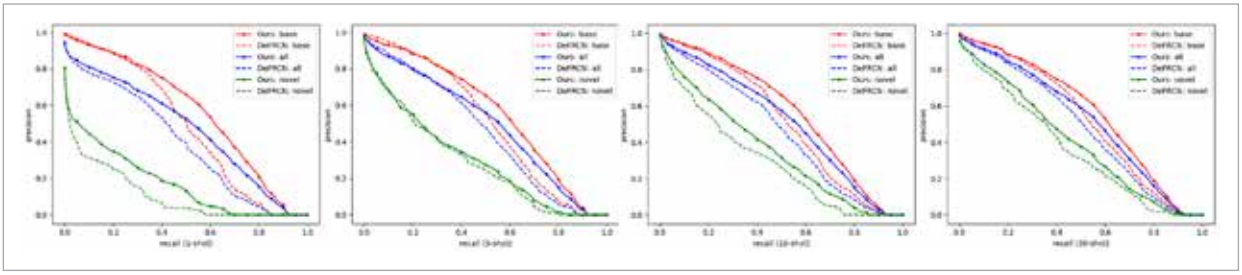
Table 5
AR results on different shots for COCO.

shot	classes	small	medium	large	all area
2	novel	0.080	0.241	0.416	0.265
	base	0.307	0.588	0.714	0.523
	all	0.250	0.501	0.639	0.459
10	novel	0.131	0.358	0.543	0.364
	Base	0.303	0.581	0.698	0.515
	All	0.260	0.525	0.660	0.477
30	Novel	0.146	0.401	0.589	0.406
	Base	0.296	0.580	0.693	0.512
	all	0.258	0.535	0.667	0.485

4.3. Ablation Study

In this section, we first analyze the details of the data augmentation strategies. Then, we conduct ablation experiments on the loss function, model stability, model size, and inference speed. Finally, we visualize the detection results to further demonstrate the effectiveness of the model.

Figure 7
PR curves for different shot settings on the COCO dataset are provided.



4.3.1. Semantic Similarity Comparison Between Novel Instances and Classes – Using Euclidean Distance

Euclidean distance is employed to quantify the similarity between feature vectors and class prototypes. To illustrate this relationship, we conduct a detailed analysis of the similarities between the ROI feature vectors and the class prototypes within a mini-batch. Specifically, in the split 1 group of the VOC dataset, we randomly select a ground-truth (GT) feature vector from the novel class and compute its Euclidean distance to all class prototypes, followed by normalization. As illustrated in Figure 8, the object in the left image belongs to the novel class 'sofa', while the right image depicts its distances to all class prototypes. These distances are sorted in ascending order. It is evident that the closest class is the novel class 'sofa', followed by several base classes. Thus, in contrast to other distribution calibration methods that consider only the top-k most similar base classes or solely the mean of the novel classes when computing the mean of the calibrated distribution, the figure clearly demonstrates that it is more rational to consider both novel and base classes simultaneously.

4.3.2. Weight Matrix by Sinkhorn Distance

Optimization is performed based on the semantic relevance between instance features and class prototypes, using Sinkhorn distance to compute the weighted matrix. As depicted in Figure 9(a), 'v0, v1, v2, v3' represent the feature vectors of the novel ground-truth objects in the current mini-batch. For the row corresponding to v3, the heatmap displays the Euclidean distances between v3 and all class prototypes, which align with the values presented in the bar chart in Figure 7. The resulting analysis of the weights to be transferred from various classes is illustrated in Figure 9(b). By comparing (a) and (b) in terms of color mapping in Figure 9, it is evident that higher values in the distance heatmap correspond to lower values in the OT weight heatmap, as indicated by the blue arrow. This observation aligns with our objective of transferring more knowledge from closer classes. In contrast to the top-k (k=2) approach in Distribution Calibration [49], our optimal transport-based method adaptively determines both which base classes to transfer and the extent of knowledge to leverage, thereby facilitating a more precise modeling of the novel class distribution.

4.3.4. Augmentation Loss Comparison with MFDC

We compare the classification losses of augmented data generated by MFDC and T-CLD to evaluate the robustness of our method. As shown in Figure 10, the losses in (a) correspond to the VOC split1,

Figure 8

Euclidean Distances between a feature vector of a novel class object and all classes.

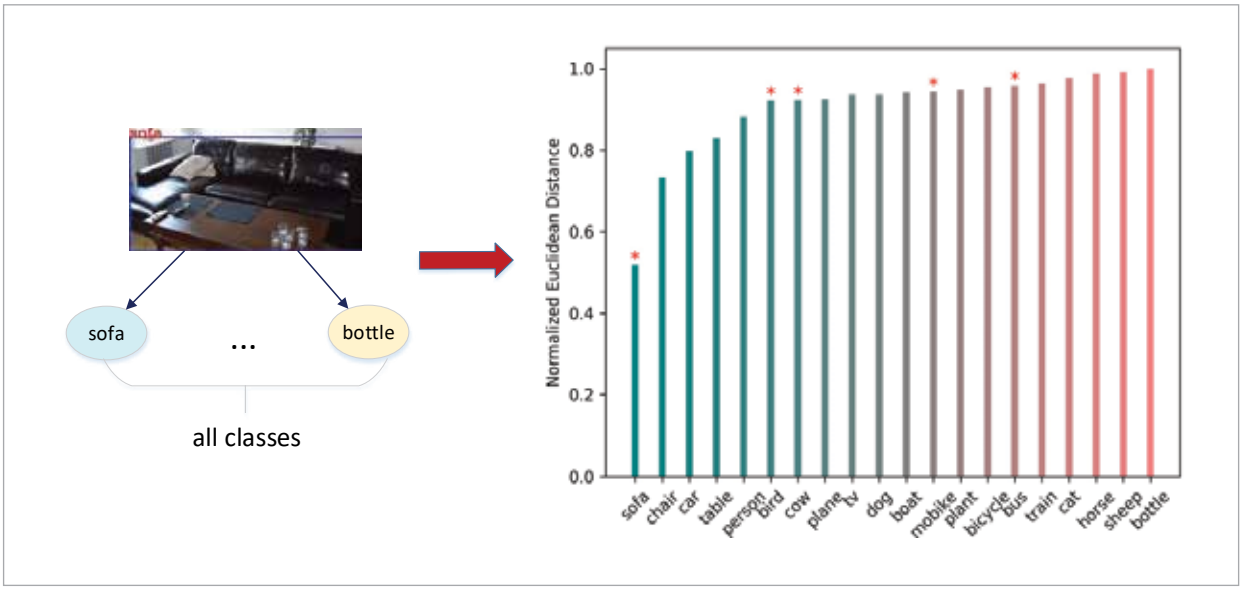


Figure 9

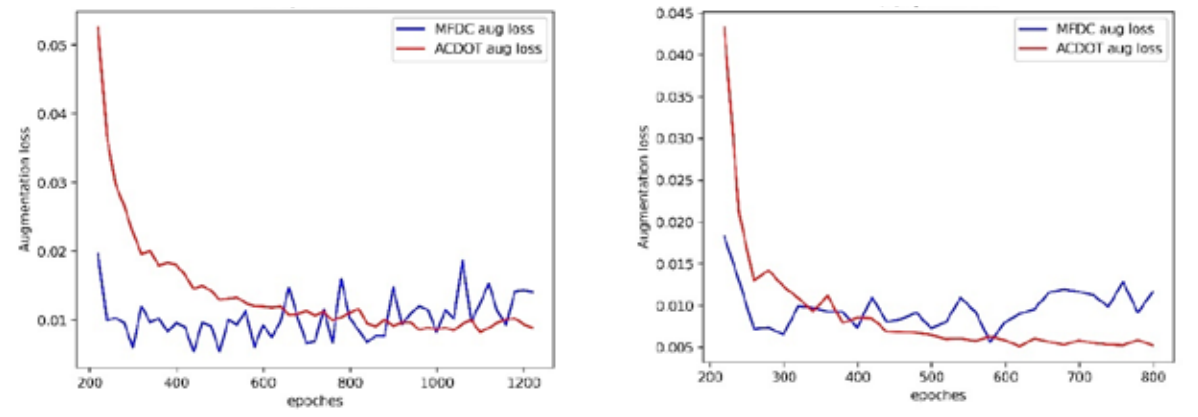
Use Sinkhorn algorithm to compute the weight matrix from the Euclidean distance matrix.

(a) Distance heatmap: GT vectors to classes

(b) OT weight heatmap: GT vectors to classes

Figure 10

Comparison of classification loss of augmented data from MFDC and ours.



(a) split1 3shot

(b) split2 2shot

3-shot settings, while those in (b) correspond to the VOC split2, 2-shot settings; it is evident that the augmentation loss of our model decreases and remains much more stable compared to MFDC.

4.3.5. Ablation Study of Each Loss

To evaluate the effectiveness of each loss in the model, we conduct experiments on each loss individually, as well as on pairwise combinations, respectively, as shown in Table 6. The experimental settings involve VOC split1 with 1-shot and 2-shot configurations. The first row shows the results from DeFRCN, which serves as the baseline. The optimal performance is achieved when all three losses are used together.

Table 6

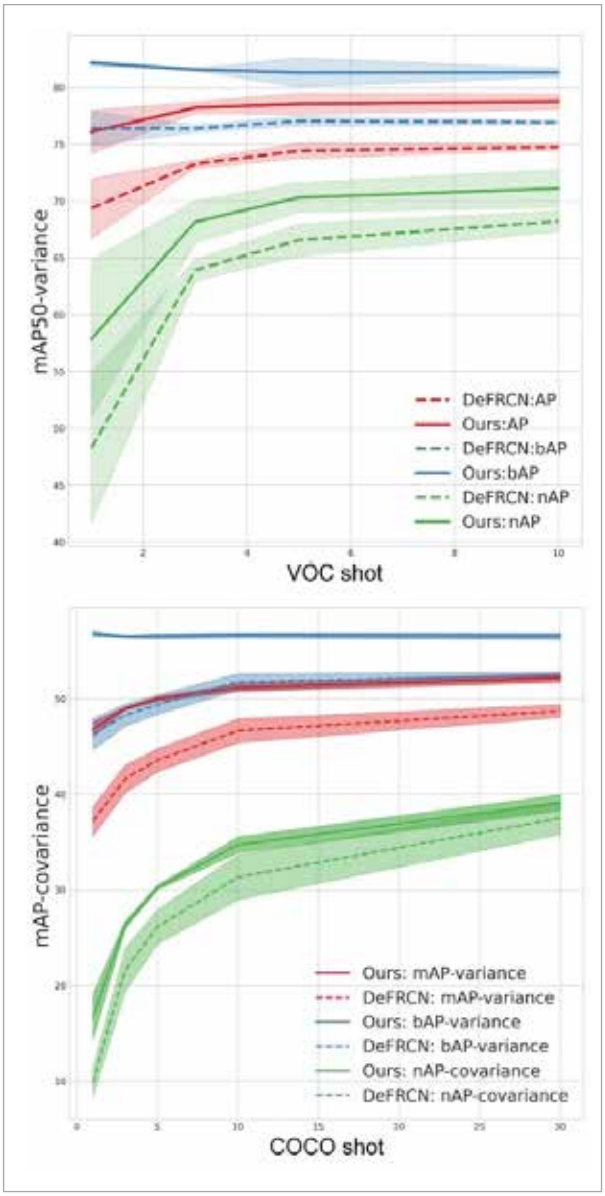
Quantifies the effectiveness of each loss.

L_k^{cls}	L_w^{loc}	L_a^{cls}	1-shot	2-shot
			57.03	58.57
✓			60.85	65.36
	✓		62.73	65.68
		✓	62.86	63.41
✓		✓	62.82	66.45
✓	✓		62.37	66.84
	✓	✓	65.00	65.46
✓	✓	✓	66.10	68.45

4.3.6. Steadiness Analysis: Compare AP and Variance among Different Seeds

Using the same model, varying training samples can lead to different precision results on the test dataset. To demonstrate the stability of the model, we report the AP and the variance across different seeds and compare the results with those of the DeFRCN method. In this context, different seeds refer to different training samples for novel classes, consistent with previous works [31, 39, 43, 46]. The results for VOC and COCO are presented in Figure 11. It can be observed that our model consistently improves performance across different shot settings. Additionally, as the number of shots increases, the variance tends to decrease, leading to more stable results. Compared to VOC, the variance fluctuations on COCO are smaller across different shot settings, indicating improved stability.

Figure 11
Average mAP and variance among different seeds.



4.3.7. Model Size and Inference Speed Comparisons

We compare the model sizes and the number of learnable parameters at different stages on the VOC and COCO datasets, as shown in Table 7. Compared to DeFRCN, our method achieves improved performance, but the introduction of knowledge distillation and distribution calibration modules increases the number of parameters in both the pre-training and fine-tuning stages.

Table 7
Model size and learnable params comparison. The numerical unit is M.

Dataset	Stage	Method	Model Size	Learnable
VOC	pre-train	DeFRCN	198.6	48.3
		Ours	230.7	56.3
	fine-tune	DeFRCN	198.8	34.1
		Ours	230.8	42.1
COCO	pre-train	DeFRCN	200.4	48.7
		Ours	232.4	56.7
	fine-tune	DeFRCN	198.8	34.1
		Ours	233.2	42.7

Although the number of parameters increases during both the pre-training and fine-tuning stages due to the model design, no knowledge distillation or data augmentation is required during inference. As a result, the inference speed remains comparable to that of DeFRCN, as shown in the comparison results in Table 8. In Contrast to the 'inference speed', 'pure computation' excludes Image Preprocessing Time.

Table 8
Inference speed comparison. The unit is image per second.

Dataset	Shot	Method	Inference Speed	Pure Compute
VOC	1	DeFRCN	0.082	0.080
		MFDC	0.081	0.077
	3	DeFRCN	0.090	0.087
		MFDC	0.076	0.074
	5	DeFRCN	0.088	0.085
		MFDC	0.075	0.073
	10	DeFRCN	0.088	0.085
		MFDC	0.083	0.079
COCO	1	DeFRCN	0.084	0.081
		MFDC	0.102	0.099
	5	DeFRCN	0.082	0.080
		MFDC	0.087	0.084
	10	DeFRCN	0.081	0.080
		MFDC	0.087	0.085
	30	DeFRCN	0.082	0.081
		MFDC	0.087	0.085

4.3.8. Queue Storage and Optimal Transport Complexity Analysis

Queue Storage.

The queue storage mechanism constructs and stores class prototypes. For N classes, each with a 2048-dim float32 vector (8 KB per class), total memory is M = N×8KB (e.g., 160 KB for 20 classes in VOC). If higher-dimensional structures like 2048×2048 matrices are stored, memory rises to M = N×16MB (320 MB for 20 classes). Thus, storage scales linearly with the number of classes.

Optimal Transport (OT) Complexity Analysis.

Assume n positive proposal features [n,2048] and 5 class prototypes [5, 2048] (VOC). Constructing the cost matrix $C \in R^{n\times 5}$ requires $O(2048\times n\times 5) = O(10{,}240n)$ operations. Each of 50 Sinkhorn iterations costs $O(10n)$, totaling $O(500n)$. As Sinkhorn iterations are lightweight, the overall optimal transport complexity is approximated as $O(10{,}000n)$.

4.3.9. Visualization of Detection Results

To provide a clearer visualization, we compare the detection results of the same images from GT, DeFRCN, and our method under the split1 1-shot settings of VOC. The quantitative results presented in Table 2 demonstrate that, compared to the

Figure 12
Visualization comparison of results on base classes.

Figure 13
Visualization comparison of results on novel classes.

baseline, our method improves performance by 2.7%, 0.7%, and 9.1% on all classes, base classes, and novel classes, respectively. Figure 12 further illustrates the performance on base classes. Our method successfully detects additional objects missed by DeFRCN, mitigates misclassification, and yields more precise localization. Notably, it can even identify objects absent from the original annotations, such as the 'potted plant' in the column row of the figure.

As shown in Table 2, under the Split1 1-shot setting on the VOC dataset, our method outperforms the baseline on novel classes by 9.1%. The corresponding qualitative results are illustrated in Figure 13. Compared to DeFRCN, our approach detects previously missed objects—such as the bird and person in column 1, the person in column 3, and the person and bottle in column 4—while also achieving more precise localization, as demonstrated by the bus in column 2 and the sofa in column 4.

4.4. Reality Verification of Foreign Object Detection on Conveyor Belts

Based on DsLMF+ [50], we constructed the CoalMine dataset with four categories—miner, safety helmet, towline, and hydraulic support guard plate ("guard plates")—containing 4,824 training and 1,342 test images. The Foreign dataset, collected from underground conveyor belt videos (including part of Cheng et al. [4]), contains elongated objects (e.g., anchor rods, iron bars) and large blocks (e.g., coal gangue), with 500 training and 202 test images.

4.4.1. Experiments Results

We conducted two transfer experiments: from the CoalMine base dataset to the Foreign dataset, and from the COCO base dataset to the Foreign dataset. In each case, few-shot fine-tuning was performed under 5-shot and 10-shot settings. The results are summarized in Table 9. Transferring from the CoalMine dataset to the Foreign dataset performed significantly worse than from COCO, with AP on new classes under 5-shot and 10-shot settings lower by 21.9% and 22.5%, respectively, mainly due to CoalMine's limited four classes, small sample size, and lack of diversity, compared with COCO's 60 base classes and much larger, more diverse data. This highlights the importance of data and the crucial role of a strong base-class pre-trained model in few-shot task generalization.

Table 9
Two transfer methods result on few-shot Foreign object detection.

Method	Shot	AP	AP50	bAP	bAP50	nAP	nAP50
Coal-Foerign	5	45.7	68.0	65.1	91.4	7.0	21.2
	10	46.8	71.0	64.6	90.6	11.2	31.8
COCO-Foreign	5	38.1	58.6	38.4	58.8	28.9	51.7
	10	38.2	58.8	38.4	58.7	33.7	60.9

Figure 14
Visualization of detection results on the CoalMine-Foreign dataset.

4.4.2. Visualization of Detection Results

The detection results under the 10-shot setting were visualized, focusing primarily on coal mine–related categories, as shown in Figure 14. The first two rows depict the visualization of predictions from transferring CoalMine to the Foreign classes, while the last row shows the predictions from transferring COCO to the Foreign classes.

5. Conclusion

In this work, we propose a simple yet effective few-shot detection model based on transfer learning. By constructing dynamic prototypes, the model is guided to fully learn transferable knowledge in both the classification and localization feature spaces, thereby significantly enhancing few-shot detection performance. We first decouple the classification and localization tasks during the pre-training stage, enabling the model to separately learn more discriminative semantic representations and more precise bounding box regressions. In the fine-tuning stage, we introduce a queue-based storage mechanism to dynamically construct class prototypes within both the classification and localization feature spaces, while employing a self-distillation strategy to facilitate knowledge transfer

from base classes to novel classes. Furthermore, optimal transport-based distance metrics are used to adaptively calibrate the feature distributions of novel classes, combined with resampling-based augmentation to further enhance their classification performance. Relative to the strong DeFRCN baseline, our approach demonstrates consistent and significant improvements in the 1-shot setting, yielding gains of up to 9.1%, 5.9%, and 3.1% across the three split groups. While the model demonstrates improved performance, its current inference speed of approximately 80 millisecond per image remains insufficient for real-time applications. Future work will focus on methodological and architectural optimizations to achieve the latency requirements necessary for real-time deployment.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, author-ship, and/or publication of this article.

Data Availability

Data will be made available on request.

Funding

This study was supported by the Academic Research Projects of Beijing Union University (No. ZK90202106).

Appendix

A. Visualization of Newly Generated Samples

To analyze the feasibility of our method, we visualize the GT, OT mean, and Calibrated mean by t-sne. In the split1, 3-shot, and seed0 settings of VOC, we generate 100 samples for the GT bounding box feature vectors of novel classes in a mini-batch. As shown in Figure A1, we can see that for a GT from class cow, shown as the black square in the rightmost position, the OT mean which is shown as the blue circle, makes it have a greater distance from class motorbike. The final Calibrated mean, showed as the upper red square in Figure A1, is used to generate new data. The ground-truth (GT) vectors from the same class, as depicted in the bot-

Figure A1
Visualization of GT, OT Mean, Calibrated mean, and generated samples.

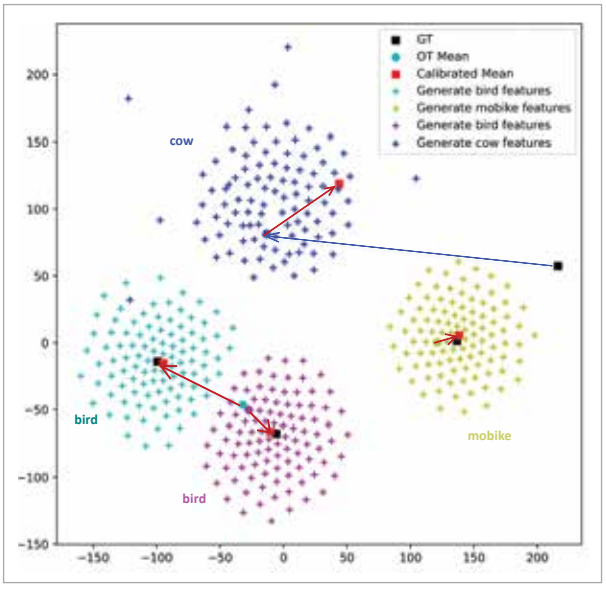
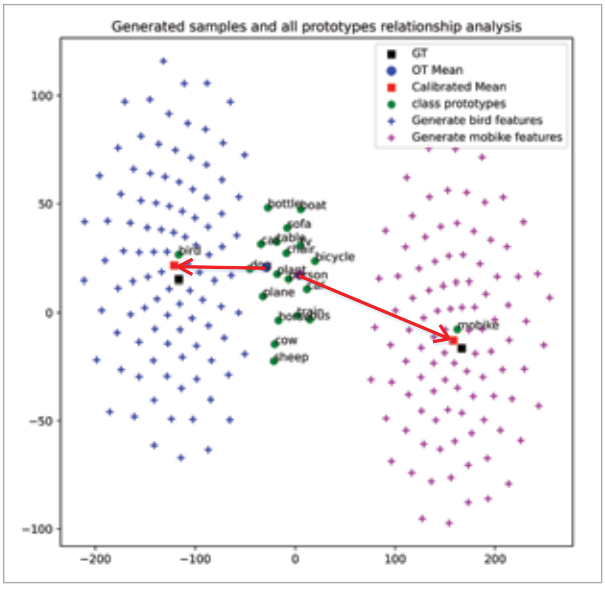


Figure A2
Relations between generated samples and all class prototypes.



tom left of Figure A1, both belong to the 'bird' class. Although the two GT vectors are separated, their OT means are closely positioned, as shown by the two adjacent points in cyan round dot and magenta round dot. However, the calibrated distribution, as described in Equation (9) and (10), subsequently moves the centers further apart, as indicated by the two red squares at the bottom left. This procedure prevents the overlap of generated samples and promotes diversity in the generated samples.

To further clarify the relationships between the ground-truth (GT) vectors, their corresponding means, and class prototypes across all classes, we select two GTs for detailed analysis, as shown in Figure A2. It is noteworthy that, although the two OT means, represented by the blue round dots and magenta round dots, lie among the class prototypes, the Calibrated means (denoted by the two red square) are not as close to each other as their OT means, and effectively separate the generated samples from each other. The calibrated means are respectively close to their corresponding class prototypes, while remaining distinct from those of other classes.

References

1. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D. Language Models Are Few-Shot Learners. Advances in Neural Information Processing Systems, 2020, 33. [Online]. Available: <Go to ISI>://WOS:001207690601035.
2. Cao, Y. H., Wang, J. Q., Jin, Y., Wu, T., Chen, K., Liu, Z. W., Lin, D. H. Few-Shot Object Detection via Association and Discrimination. Advances in Neural Information Processing Systems, 2021, 34. [Online]. Available: <Go to ISI>://WOS:000922928201045.
3. Chen, H., Wang, Y. L., Wang, G. Y., Qiao, Y. LSTD: A Low-Shot Transfer Detector for Object Detection. In Thirty-Second AAAI Conference on Artificial Intelligence / Thirtieth Innovative Applications of Artificial Intelligence Conference / Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, 2018, 2836-2843. [Online]. Available: <Go to ISI>://WOS:000485488902112. https://doi.org/10.1609/aaai.v32i1.11716
4. Cheng, D., Xu, J., Kou, Q., Ma, L., Liu, H. Classification of Foreign Objects on Coal Conveyor Belts Using a Lightweight Network with Fused Residual Information. Journal of Coal Science & Engineering, 2022, 47(3), 1361-1369.
5. Cuturi, M. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. Advances in Neural Information Processing Systems, 2013, 26.
6. Dosovitskiy, A. An Image Is Worth 16×16 Words: Transformers for Image Recognition at Scale. arXiv Preprint arXiv:2010.11929, 2020.
7. Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision, 2010, 88(2), 303-338. https://doi.org/10.1007/s11263-009-0275-4
8. Fan, Q., Zhuo, W., Tang, C. K., Tai, Y. W. Few-Shot Object Detection with Attention-RPN and Multi-Relation Detector. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, 4012-4021. https://doi.org/10.1109/CVPR42600.2020.00407
9. Fan, Z. B., Ma, Y. C., Li, Z. M., Sun, J. Generalized Few-Shot Object Detection Without Forgetting. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021, 4525-4534. https://doi.org/10.1109/CVPR46437.2021.00450
10. Finn, C., Abbeel, P., Levine, S. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. International Conference on Machine Learning, 2017, 70. [Online]. Available: <Go to ISI>://WOS:000683309501022.
11. Guirguis, K., Meier, J., Eskandar, G., Kayser, M., Yang, B., Beyerer, J. NIFF: Alleviating Forgetting in Generalized Few-Shot Object De-

tection via Neural Instance Feature Forging. In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, 24193-24202. https://doi.org/10.1109/CVPR52729.2023.02317

12. Han, J. M., Ren, Y. Q., Ding, J., Yan, K., Xia, G. S. Few-Shot Object Detection via Variational Feature Aggregation. In Thirty-Seventh AAAI Conference on Artificial Intelligence, 2023, 37(1), 755-763. [Online]. Available: <Go to ISI>://WOS:001243759700084. https://doi.org/10.1609/aaai.v37i1.25153
13. Hariharan, B., Girshick, R. Low-Shot Visual Recognition by Shrinking and Hallucinating Features. In 2017 IEEE International Conference on Computer Vision (ICCV), 2017, 3037-3046. https://doi.org/10.1109/ICCV.2017.328. [Online]. Available: <Go to ISI>://WOS:000425498403011. https://doi.org/10.1109/ICCV.2017.328
14. He, K. M., Fan, H. Q., Wu, Y. X., Xie, S. N., Girshick, R. Momentum Contrast for Unsupervised Visual Representation Learning. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, 9726-9735. https://doi.org/10.1109/CVPR42600.2020.00975
15. He, K. M., Zhang, X. Y., Ren, S. Q., Sun, J. Deep Residual Learning for Image Recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90
16. Hoaglin, D. C. John W. Tukey and Data Analysis. Statist. Sci., 2003, 18(3), 311-318. https://doi.org/10.1214/ss/1076102418
17. Kang, B., Liu, Z., Wang, X., Yu, F., Feng, J. S., Darrell, T. Few-Shot Object Detection via Feature Reweighting. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, 8419-8428. https://doi.org/10.1109/ICCV.2019.00851
18. Karlinsky, L., Shtok, J., Harary, S., Schwartz, E., Aides, A., Feris, R., Giryes, R., Bronstein, A. M. RepMet: Representative-Based Metric Learning for Classification and Few-Shot Object Detection. In 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019, 5192-5201. https://doi.org/10.1109/CVPR.2019.00534
19. Koch, G., Zemel, R., Salakhutdinov, R. Siamese Neural Networks for One-Shot Image Recognition. In ICML Deep Learning Workshop, 2015, 2(1), 1-30.
20. Li, J. M., Zhang, Y. N., Qiang, W. W., Si, L. Y., Jiao, C. B., Hu, X. H., Zheng, C. W., Sun, F. C. Disentangle and Remerge: Interventional Knowledge Distillation for Few-Shot Object Detection from a Conditional Causal Perspective. Thirty-Seventh AAAI Conference on Artificial Intelligence, 2023, 37(1), 1323-1333. [Online]. Available: <Go to ISI>://WOS:001243759700147. https://doi.org/10.1609/aaai.v37i1.25216
21. Li, K., Zhang, Y. L., Li, K. P., Fu, Y. Adversarial Feature Hallucination Networks for Few-Shot Learning. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, 13467-13476. https://doi.org/10.1109/CVPR42600.2020.01348
22. Lin, S. B., Wang, K., Zeng, X. Y., Zhao, R. Explore the Power of Synthetic Data on Few-Shot Object Detection. In 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2023, 638-647. https://doi.org/10.1109/CVPRW59228.2023.00071
23. Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, 2117-2125. https://doi.org/10.1109/CVPR.2017.106
24. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C. L. Microsoft COCO: Common Objects in Context. Computer Vision - ECCV 2014, Pt V, 2014, 740-755. https://doi.org/10.1007/978-3-319-10602-1_48
25. Lin, X. D., Duan, Y. Q., Dong, Q. Y., Lu, J. W., Zhou, J. Deep Variational Metric Learning. Computer Vision - ECCV 2018, Pt 15, 2018, 714-729. https://doi.org/10.1007/978-3-030-01267-0_42
26. Liu, S. L., Zeng, Z. Y., Ren, T. H., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C. Y., Yang, J. W., Su, H., Zhu, J., Zhang, L. Grounding DINO: Marrying DINO With Grounded Pre-Training for Open-Set Object Detection. In Computer Vision - ECCV 2024, Pt XLVII, 2025, 15105, 38-55. https://doi.org/10.1007/978-3-031-72970-6_3

27. Liu, X., Zhou, K. R., Yang, P. B., Jing, L. P., Yu, J. Adaptive Distribution Calibration for Few-Shot Learning via Optimal Transport. Information Sciences, 2022, 611, 1-17. https://doi.org/10.1016/j.ins.2022.07.189

28. Lu, X., Diao, W., Mao, Y., Li, J., Wang, P., Sun, X., Fu, K. Breaking Immutable: Information-Coupled Prototype Elaboration for Few-Shot Object Detection. In Proceedings of the AAAI Conference on Artificial Intelligence, 2023, 37(2), 1844-1852. https://doi.org/10.1609/aaai.v37i2.25274

29. Park, S.-J., Han, S., Baek, J.-W., Kim, I., Song, J., Lee, H. B., Han, J.-J., Hwang, S. J. Meta Variance Transfer: Learning to Augment from the Others. In International Conference on Machine Learning, 2020, PMLR, 7510-7520.

30. Pei, W. J., Wu, S., Mei, D. W., Chen, F. L., Tian, J. D., Lu, G. M. Few-Shot Object Detection by Knowledge Distillation Using Bag-of-Visual-Words Representations. In Computer Vision - ECCV 2022, Pt X, 2022, 13670, 283-299. https://doi.org/10.1007/978-3-031-20080-9_17

31. Qiao, L. M., Zhao, Y. X., Li, Z. Y., Qiu, X., Wu, J. N., Zhang, C. DeFRCN: Decoupled Faster R-CNN for Few-Shot Object Detection. In 2021 IEEE/CVF International Conference on Computer Vision (ICCV 2021), 2021, 8661-8670. https://doi.org/10.1109/ICCV48922.2021.00856

32. Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I. Learning Transferable Visual Models from Natural Language Supervision. Proceedings of Machine Learning Research, 2021, 139. [Online]. Available: <Go to ISI>://WOS:000768182704084.

33. Ravi, S., Larochelle, H. Optimization as a Model for Few-Shot Learning. In International Conference on Learning Representations, 2017.

34. Redmon, J., Farhadi, A. YOLO9000: Better, Faster, Stronger. In 30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), 2017, 6517-6525. https://doi.org/10.1109/CVPR.2017.690

35. Ren, S. Q., He, K. M., Girshick, R., Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Advances in Neural Information Processing Systems, 2015, 28. [Online]. Available: <Go to ISI>://WOS:000450913100006.

36. Salakhutdinov, R., Tenenbaum, J., Torralba, A. One-Shot Learning with a Hierarchical Nonparametric Bayesian Model. In Proceedings of ICML Workshop on Unsupervised and Transfer Learning, 2012, 195-206.

37. Snell, J., Swersky, K., Zemel, R. Prototypical Networks for Few-Shot Learning. Advances in Neural Information Processing Systems, 2017, 30. [Online]. Available: <Go to ISI>://WOS:000452649404015.

38. Song, G. L., Liu, Y., Wang, X. G. Revisiting the Sibling Head in Object Detector. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020), 2020, 11560-11569. https://doi.org/10.1109/CVPR42600.2020.01158

39. Sun, B., Li, B. H., Cai, S. C., Yuan, Y., Zhang, C. FSCE: Few-Shot Object Detection via Contrastive Proposal Encoding. In 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021), 2021, 7348-7358. https://doi.org/10.1109/CVPR46437.2021.00727

40. Sung, F., Yang, Y. X., Zhang, L., Xiang, T., Torr, P. H. S., Hospedales, T. M. Learning to Compare: Relation Network for Few-Shot Learning. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018), 2018, 1199-1208. https://doi.org/10.1109/CVPR.2018.00131

41. Vigneswaran, R., Law, M. T., Balasubramanian, V. N., Tapaswi, M. Feature Generation for Long-Tail Classification. In Proceedings of the Twelfth Indian Conference on Computer Vision, Graphics and Image Processing, 2021, 1-9. https://doi.org/10.1145/3490035.3490300

42. Wang, M., Wang, Y., Liu, H. P. Explicit Knowledge Transfer of Graph-Based Correlation Distillation and Diversity Data Hallucination for Few-Shot Object Detection. Image and Vision Computing, 2024, 143, 104958. https://doi.org/10.1016/j.imavis.2024.104958

43. Wang, X., Huang, E. T., Darrell, T., Gonzalez, J. Y., Yu, F. Frustratingly Simple Few-Shot Object Detection. arXiv, 2020. arXiv:2003.06957.

44. Wang, Z., Yang, B., Yue, H., Ma, Z. Fine-Grained Prototypes Distillation for Few-Shot Object Detection. In Proceedings of the AAAI Conference on Artificial Intelligence, 2024, 38(6), 5859-5866. https://doi.org/10.1609/aaai.v38i6.28399

45. Wu, J., Liu, S., Huang, D., Wang, Y. Multi-Scale Positive Sample Refinement for Few-Shot Object Detection. In Computer Vision - ECCV 2020: 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XVI, 2020, 456-472. https://doi.org/10.1007/978-3-030-58517-4_27

46. Wu, S., Pei, W. J., Mei, D. W., Chen, F. L., Tian, J. D., Lu, G. M. Multi-Faceted Distillation of Base-Novel Commonality for Few-Shot Object Detection. In Computer Vision - ECCV 2022, Pt IX, 2022, 13669, 578-594. https://doi.org/10.1007/978-3-031-20077-9_34 https://doi.org/10.1007/978-3-031-20077-9_34

47. Wu, Y., Chen, Y. P., Yuan, L., Liu, Z. C., Wang, L. J., Li, H. Z., Fu, Y. Rethinking Classification and Localization for Object Detection. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020), 2020, 10183-10192. https://doi.org/10.1109/CVPR42600.2020.01020

48. Yan, X. P., Chen, Z. L., Xu, A. N., Wang, X. X., Liang, X. D., Lin, L. Meta R-CNN: Towards General Solver for Instance-Level Low-Shot Learning. In 2019 IEEE/CVF International Conference on Computer Vision (ICCV 2019), 2019, 9576-9585. https://doi.org/10.1109/ICCV.2019.00967

49. Yang, S., Liu, L., Xu, M. Free Lunch for Few-Shot Learning: Distribution Calibration. arXiv Preprint arXiv:2101.06395, 2021.

50. Yang, W. J., Zhang, X. H., Ma, B., Wang, Y. Q., Wu, Y. J., Yan, J. X., Liu, Y. W., Zhang, C., Wan, J. C., Wang, Y., Huang, M. Y., Li, Y. Y., Zhao, D. An Open Dataset for Intelligent Recognition and Classification of Abnormal Condition in Longwall Mining. Scientific Data, 2024, 11(1), 848. https://doi.org/10.1038/s41597-024-03713-2

51. Zhang, X., Liu, Y., Wang, Y., Boularias, A. Detect Everything with Few Examples. arXiv Preprint arXiv:2309.12969, 2023.

52. Zhao, Z. Y., Liu, Q. J., Wang, Y. H. Exploring Effective Knowledge Transfer for Few-Shot Object Detection. In Proceedings of the 30th ACM International Conference on Multimedia (MM 2022), 2022, 6831-6839. https://doi.org/10.1145/3503161.3548062