DETA: Denoised Task Adaptation for Few-Shot Learning

Ji Zhang[1] Lianli Gao[2*] Xu Luo[1] Hengtao Shen[1] Jingkuan Song[2]

[1]University of Electronic Science and Technology of China (UESTC)
[2]Shenzhen Institute for Advanced Study, UESTC
{jizhang.jim,juana.alian}@gmail.com

Abstract

Test-time task adaptation in few-shot learning aims to adapt a pre-trained task-agnostic model for capturing task-specific knowledge of the test task, rely only on few-labeled support samples. Previous approaches generally focus on developing advanced algorithms to achieve the goal, while neglecting the inherent problems of the given support samples. In fact, with only a handful of samples available, the adverse effect of either the image noise (a.k.a. `X`*-noise) or the label noise (a.k.a.* `Y`*-noise) from support samples can be severely amplified. To address this challenge, in this work we propose* ***DE****noised* ***T****ask* ***A****daptation (****DETA****), a first, unified image- and label-denoising framework orthogonal to existing task adaptation approaches. Without extra supervision, DETA filters out task-irrelevant, noisy representations by taking advantage of both global visual information and local region details of support samples. On the challenging Meta-Dataset, DETA consistently improves the performance of a broad spectrum of baseline methods applied on various pre-trained models. Notably, by tackling the overlooked image noise in Meta-Dataset, DETA establishes new state-of-the-art results. Code is released at* https://github.com/JimZAI/DETA.

1. Introduction

Few-Shot Learning (FSL) refers to rapidly deriving new knowledge from a limited number of samples, a central capability that humans naturally possess, but "data-hungry" machines still lack. Over the past years, a community-wide enthusiasm has been ignited to narrow this gap, especially in fields such as computer vision [15,26,47], machine translation [4,30,54] and reinforcement learning [9,17,39].

The general formulation of FSL involves two stages: **1)** *training-time* task-agnostic knowledge accumulation, and **2)** *test-time* task-specific knowledge acquisition, a.k.a. task adaptation. In particular, the former stage seeks to pre-train a task-agnostic model on large amounts of training samples collected from a set of *base* classes. While the latter targets adapting the pre-trained model for capturing task-specific knowledge of the few-shot (or test) task with *novel* classes, given a tiny set of labeled support samples. Early progress in FSL has been predominantly achieved using the idea of meta-learning, which aligns the learning objectives of the two stages to better generalize the accumulated knowledge towards few-shot tasks [39,47,54]. Nevertheless, recent studies [13,25,35,42] revealed that a good test-time task adaptation approach with any pre-trained models – no matter what training paradigms they were learned by, can be more effective than sophisticated meta-learning algorithms. Furthermore, with the recent success in model pre-training techniques [14,16,33], designing efficient adapter-based [24,25,55] or finetuning-based [6,19,46] task adaptation algorithms that can flexibly borrow "free" knowledge from a wide range of pre-trained models is therefore of great practical value, and has made remarkable progress in FSL.

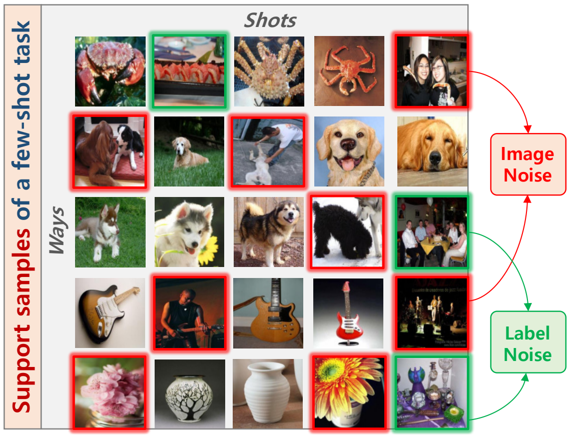


Figure 1. Dual noises in the support samples of a few-shot task. **Image noise** (a.k.a. `X`-noise): the target object regions are often obscured by interfering factors such as cluttered backgrounds, image corruption, etc. **Label noise** (a.k.a. `Y`-noise): mislabeled samples. The goal of this work is to develop a first, unified image- and label-denoising framework for reliable task adaptation.

[*]Corresponding author.

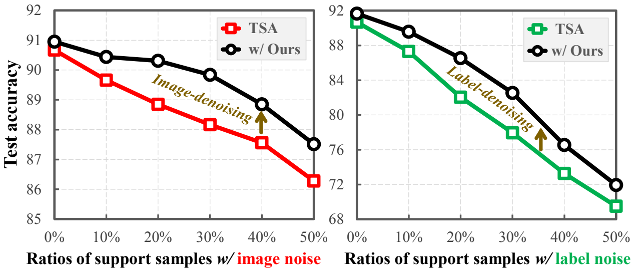


Figure 2. Quantitative evidence that image- or label-noisy support samples degrades test-time task adaptation performance. The results are averaged over 100 5-way 10-shot tasks sampled from the five classes in Figure 1. *Image-noisy* samples here are manually selected from all samples of the five classes. *Label-noisy* samples for each class are generated by uniformly changing the label to that of other four classes. The baseline scheme TSA [25] is applied to a RN-18 pre-trained on ImNet-MD [51] for task adaptation. As seen, the dual noises negatively impact task adaptation performance, and our method consistently improves the baseline under various ratios of image- or label-noisy support samples.

Despite the encouraging progress, existing approaches mostly focus on developing advanced algorithms to mine task-specific knowledge for few-shot tasks, while neglecting the inherent problems of the given support samples. Unfortunately, the set of support samples collected from the open world, no matter how small, can be unavoidably polluted by noises. As illustrated in Figure 1, either the image noise (a.k.a. X-noise) or the label noise (a.k.a. Y-noise) could arise at possibly every phase of the task lifecycle[1]. It has been well recognized that a tiny portion of image-noisy [21, 34] or label-noisy [31, 48] samples can compromise the model performance to a large extent. When it comes to test-time task adaptation, the adverse effects of the dual noises can be remarkably magnified owing to the *scarcity* of support samples, as quantitatively proven in Figure 2. Despite being harmful and inevitable, as far as we know, both image noise and label noise have received considerably less attention in test-time task adaptation. **This begs the following questions: 1)** Is it possible to design a method to tackle the two issues in a unified framework? **2)** Whether the designed method can be orthogonal to existing task adaptation approaches, so as to achieve robust FSL?

In this work, we answer the above questions by proposing **DE**noised **T**ask **A**daptation (**DETA**), a first, unified image- and label-denoising framework for FSL. The key idea of DETA is to simultaneously filter out task-irrelevant (i.e. noisy) local region representations of *image-noisy* samples, as well as global image representations of *label-noisy* samples, relying only on the interrelation among the given support samples of few-shot tasks. To this end, a parameter-free *contrastive relevance aggregation (CoRA)* module is first designed to determine the weights of regions and images in support samples, based on which two losses are proposed for noise-robust (or reliable) task adaptation: a *local compactness* loss $\mathcal{L}_l$ that promotes the intra-class compactness of *clean* regions, along with a *global dispersion* loss $\mathcal{L}_g$ that encourages the inter-class dispersion of *clean*, image-level class prototypes. The two losses complement each other to take advantage of both global visual information and local region details of support samples to softly ignore the dual noises during the optimization. An overview of our DETA framework is shown in Figure 3.

Flexibility and Strong Performance. The proposed DETA is orthogonal to existing *adapter*-based task adaptation (A-TA) and *finetuning*-based task adaptation (F-TA) paradigms, therefore can be plugged into any types of these approaches to improve model robustness under the joint (image, label)-noise. On average, by performing image-denoising on the vanilla Meta-Dataset (MD) [51], DETA improves the classification accuracy of A-TA, F-TA baselines by **1.8%**∼**1.9%**, **2.2%**∼**4.1%**, respectively (Table 1). In particular, by tackling the overlooked image noise in the vanilla MD, DETA further boosts the state-of-the-art TSA [25] by **1.8%**∼**2.1%** (Table 5). Also, by conducting label-denoising on the label-corrupted MD, DETA outperforms A-TA, F-TA baselines by **1.8%**∼**4.2%**, **2.8%**∼**6.1%**, respectively (Table 2).

Contributions. To summarize, our contributions are three-fold. **1)** We propose DETA, a first, unified image- and label-denoising framework for FSL. **2)** Our DETA can be flexibly plugged into both adapter-based and finetuning-based task adaptation paradigms. **3)** Extensive experiments on Meta-Dataset show the effectiveness and flexibility of DETA.

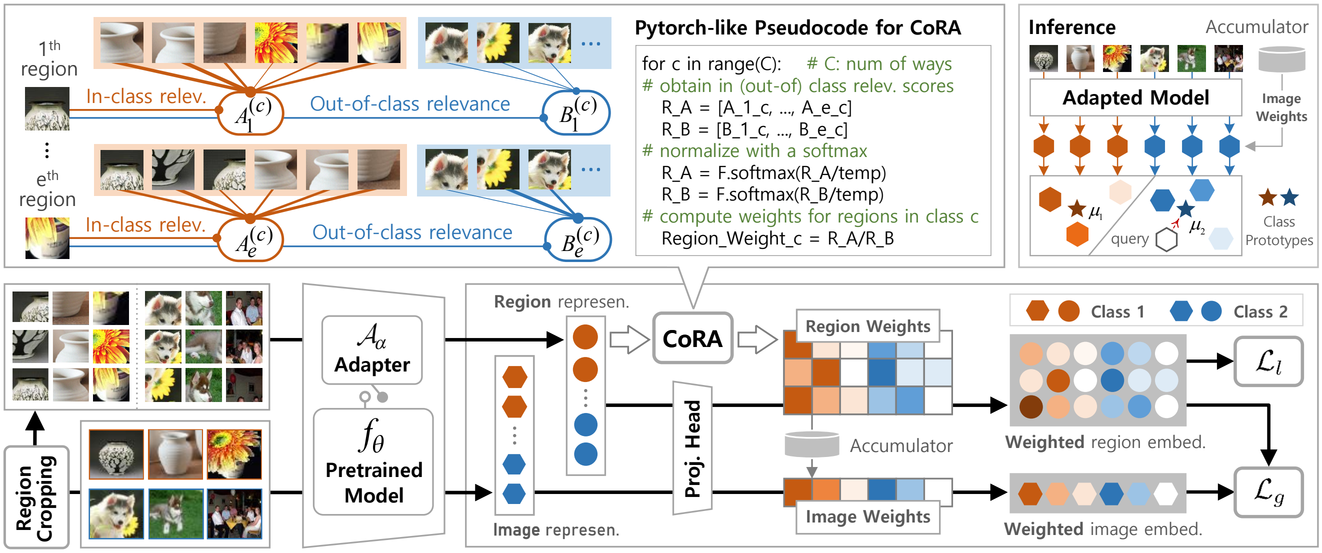
2. Related Work

Few-shot Learning. Generalizing from a limited amount of samples has been proven challenging for most existing deep learning models. Prevalent FSL approaches learn new concepts under scarce supervision by a meta-learning setting [11, 12, 18, 27, 40, 45, 57, 59–61]. In **Sup. Mat. (E.1)**, we present a review of the literature on FSL approaches.

Test-time Task Adaptation in FSL. Recent progress revealed that when there exists severe *category/domain shift* between base classes and few-shot tasks, without performing test-time task adaptation, the generalization of any pre-trained models would decrease remarkably [3, 35, 42]. Various attempts have been made to adapt the pre-trained models to few-shot tasks by devising model-specific adapters, *e.g.*, the residual adapter TSA [25] for ResNets [15], the self-attention adapter eTT [55] for ViTs [7]. A survey of test-time task adaptation is presented in **Sup. Mat. (E.2)**.

Data-denoising for FSL. The training data collected from the open world are unavoidably polluted by image noise or

[1] In more challenging FSL scenarios, some or even all of the few examples are collected by an agent from a dynamic environment rather than relying on humans, the dual noises become more common in this context.



Pytorch-like Pseudocode for CoRA

```
for c in range(C):     # C: num of ways
# obtain in (out-of) class relev. scores
   R_A = [A_1_c, ..., A_e_c]
   R_B = [B_1_c, ..., B_e_c]
# normalize with a softmax
   R_A = F.softmax(R_A/temp)
   R_B = F.softmax(R_B/temp)
# compute weights for regions in class c
   Region_Weight_c = R_A/R_B
```

Figure 3. An overview of the proposed **DETA** (in a 2-way 3-shot exemple). During each iteration of task adaptation, the images together with a set of randomly cropped local regions of support samples are first fed into a pre-trained model f_θ (w/ or w/o a model-specific adapter $\mathcal{A}_\alpha$) to extract image and region representations. Next, a **Contrastive Relevance Aggregation(CoRA)** module takes the region representations as input to determine the weight of each region, based on which we can refine the image weights by a momentum accumulator. Finally, a **Local Compactness** loss $\mathcal{L}_l$, along with a **Global Dispersion** loss $\mathcal{L}_g$ are devised in a weighted embedding space for promoting the mining of task-specific (or clean) representations. At inference, we only retain the adapted model f_{θ^*} (or $f_{[\theta;\alpha^*]}$) to produce image representations of support samples, on which we can build a noise-robust classifier guided by the refined image weights in the accumulator.

label noise, which may compromise the performance of the learned models [1, 20, 48]. Limited works in FSL considered the influence of image noise [34, 56] or label noise [28, 36] on model generalization. Additionally, they mainly focus on dealing with noises in base classes rather than in the few-shot task. Particularly, Liang et al. [28] for the first time explored the label noise problem in FSL. Differences between the work [28] and ours are threefold. **1)** We aim to address both the image and label noises from support samples, where every sample is of great value in characterizing the few-shot task. **2)** We take advantage of both global visual information and local region details to achieve the goal. **3)** Our method is orthogonal to both adapter-based and finetuning-based task adaptation methods. Even so, Liang et al. [28] do bring a lot of inspiration to our method.

Cross-image Alignment for Representation Learning. A plethora of cross-image alignment based FSL methods have recently been developed to extract more discriminative representations [18, 23, 38, 52, 53, 58]. Those methods highlight important local regions by aligning the local features between the support and query samples of few-shot tasks. Despite the impressive performance, those *none-adaptation* methods are unable to capture task-specific representations when there exists severe *category shift* or *domain shift* between base classes and few-shot tasks [19, 35]. Moreover, we often overlook the fact that owing to the small sample size in few-shot tasks, negligible computational cost is required to model the relationships of the support samples.

3. Methodology

In this section, we elaborate our proposed DETA. Before that, we introduce some preliminary concepts about test-time task adaptation in FSL, and the mechanism of adapter-based or finetuning-based task adaptation.

3.1. Preliminary

Assume we have a pre-trained task-agnostic model f_θ parameterized by θ, which serves as a feature backbone to output a d-dimensional representation for each input image. Test-time task adaptation seeks to adapt f_θ to the test task $T = \{S, Q\}$, by deriving task-specific knowledge on the few-labeled support samples $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N_s}$, consisting of N_s *image-label* pairs from C novel classes, *i.e.*, $y_i \in \{1, ..., C\}$. It is expected that the adapted model can correctly partition the N_q query samples $Q = \{(\boldsymbol{x}_i)\}_{i=1}^{N_q}$ to the C classes in the *representation* space. If there are exactly K support samples in each of these C classes, the task is also called a C-way K-shot task.

Adapter-based Task Adaptation (`A-TA`). The goal of `A-TA` is to capture the knowledge of the test task by attaching a model-specific adapter $\mathcal{A}_\alpha$ parameterized by α to the pre-trained model f_θ. During task adaptation, the parameters of f_θ, θ, are frozen and only the parameters α are optimized from scratch using the support samples:

$$\alpha := \alpha - \gamma \nabla_\alpha \mathcal{L}^S\big([\theta; \alpha]\big), \qquad (1)$$

where γ is the learning rate, and

$$\mathcal{L}^S([\theta;\alpha]) = \frac{1}{N_s}\sum_{(\boldsymbol{x},y)\in S}\ell\big(h(f_{[\theta;\alpha]}(\boldsymbol{x});S),y\big), \quad (2)$$

where ℓ is cross-entropy loss, $f_{[\theta;\alpha]}$ indicates the feature backbone appended with the adapter, h is a non-parametric classifier head capable of producing a softmax probability vector whose dimensionality equals C. Notably, the recent `A-TA` scheme TSA [25] achieved state-of-the-art results on Meta-Dataset [51], by integrating a residual-adapter into the pre-trained URL [24] model (w/ RN-18), and setting h to the nonparametric Nearest Centroid Classifier (NCC) [37].

Finetuning-based Task Adaptation (`F-TA`). `A-TA` requires model-specific adapters to adapt different pre-trained models, e.g., the residual adapter TSA [25] for ResNets [15], the self-attention adapter eTT [55] for ViTs [7]. In contrast, `F-TA`, originated from transfer learning literature [22] and introduced into FSL by MAML [9], directly finetunes the parameters θ of any pre-trained model f_θ at test time, i.e., $\theta := \theta - \gamma\nabla\mathcal{L}^S(\theta)$, and is thus model-agnostic.

3.2. Overview

Our framework DETA is illustrated in Figure 3, which mainly consists of the following steps. **For each iteration:**

Step-1. A feature backbone f takes the images and a set of randomly cropped image regions of the support samples as input to obtain image and region representations.

Step-2. A *contrastive relevance aggregation* (CoRA) module takes the region representations as input to calculate the weights of different regions, based on which we can determine the image weights by a momentum accumulator.

Step-3. A projection head maps the high-dimensional image and region representations to a lower dimensional embedding space, where a *local compactness* loss $\mathcal{L}_l$ and a *global dispersion* loss $\mathcal{L}_g$ are developed on the weighted region and image embeddings to promote the mining of task-specific knowledge from support samples.

Step-4. The calculated $\mathcal{L}_l$ and $\mathcal{L}_g$ are jointly used to update the parameters of the projection head and the feature backbone f, i.e., α in $f_{[\theta;\alpha]}$ for `A-TA`, θ in f_θ for `F-TA`.

3.3. Contrastive Relevance Aggregation

The motivation of CoRA is that a region, which shows higher relevance (or similarity) to *in-class* regions while lower relevance to *out-of-class* regions, is more likely to be the object region and should be assigned a larger weight.

Given the support samples $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N_s}$ of a test-time task, we first randomly crop k local regions of size $M \times M$ for every image $\boldsymbol{x}_i$. Next, the original image together with all cropped regions of each sample are fed into f to generate image representation $\boldsymbol{z}_i$ and region representations $Z_i = \{\boldsymbol{z}_{ij}\}_{j=1}^{k}$. Let $Z^{(c)} = \bigcup_{i=1}^{N_c} Z_i$ denote the collection of representations of cropped regions in class c, $\mathbf{Z} = \bigcup_{c=1}^{C} Z^{(c)}$ the set of all representations of cropped regions, where N_c is the number of images in class c. For each region representation $\boldsymbol{z}_{ij}$ in Z_i, we construct its *in-class* and *out-of-class* region representation sets as $I(\boldsymbol{z}_{ij}) = Z^{(c)} \setminus Z_i$ and $O(\boldsymbol{z}_{ij}) = \mathbf{Z} \setminus Z^{(c)}$, respectively. Note that in $I(\boldsymbol{z}_{ij})$, the other $k-1$ intra-image representations are dropped to alleviate their dominating impacts. CoRA calculates the weight of each region based on the global statistics of in-class and out-of-class relevance scores, respectively formulated as

$$\phi(\boldsymbol{z}_{ij}) = \frac{1}{|I(\boldsymbol{z}_{ij})|}\sum_{\boldsymbol{z}'\in I(\boldsymbol{z}_{ij})}\zeta(\boldsymbol{z}_{ij},\boldsymbol{z}'), \quad (3)$$

$$\psi(\boldsymbol{z}_{ij}) = \frac{1}{|O(\boldsymbol{z}_{ij})|}\sum_{\boldsymbol{z}'\in O(\boldsymbol{z}_{ij})}\zeta(\boldsymbol{z}_{ij},\boldsymbol{z}'), \quad (4)$$

where $\zeta(\cdot)$ indicates cosine similarity. These scores are then normalized inside each class:

$$\widetilde{\phi}(\boldsymbol{z}_{ij}) = \frac{e^{\phi(\boldsymbol{z}_{ij})}}{\sum_{\boldsymbol{z}'\in Z^{(c)}} e^{\phi(\boldsymbol{z}')}},\ \widetilde{\psi}(\boldsymbol{z}_{ij}) = \frac{e^{\psi(\boldsymbol{z}_{ij})}}{\sum_{\boldsymbol{z}'\in Z^{(c)}} e^{\psi(\boldsymbol{z}')}}. \quad (5)$$

Therefore, the final calculated region weight for $\boldsymbol{z}_{ij}$ can be defined as $\lambda_{ij} = \widetilde{\phi}(\boldsymbol{z}_{ij})/\widetilde{\psi}(\boldsymbol{z}_{ij}) \in \mathbb{R}$. A pytorch-like pseudocode for CoRA is illustrated in Figure 3.

A Momentum Accumulator for Image-weighting. Aside from weighting the local regions, we also need to assess the *quality* of the images themselves for filtering out label-noisy samples. Intuitively, the most direct way to determine the weight of an image $\boldsymbol{x}_i$, ω_i, is to average the weights of all k cropped regions belonging to it, i.e., $\omega_i = \frac{1}{k}\sum_{j=1}^{k}\lambda_{ij}$.

However, the randomly cropped regions in different task adaptation iterations may have large variations, resulting the frailty of the calculated image weights. A momentum accumulator is thus developed to cope with this issue by

$$\omega_i^t = \begin{cases} \frac{1}{k}\sum_{j=1}^{k}\lambda_{ij}, & \text{if } t = 1 \\ \gamma\omega_i^{t-1} + \frac{1-\gamma}{k}\sum_{j=1}^{k}\lambda_{ij}, & \text{if } t > 1 \end{cases} \quad (6)$$

where ω_i^t denotes the accumulated image weight of $\boldsymbol{x}_i$ in the t-th iteration of task adaptation, γ is the momentum hyperparameter, and we set it to 0.7 in our method. For brevity, we omit the superscript t in the following sections.

3.4. Noise-robust Task Adaptation

DETA performs image- and label-denoising in a unified framework to achieve noise-robust task adaptation. To this end, DETA simultaneously ① **promotes the intra-class compactness of *clean* regions** – to filter out noisy local representations (e.g. cluttered backgrounds of image-noisy samples), and ② **encourages the inter-class dispersion of *clean*, image-level class prototypes** – to filter out noisy global representations (i.e. images of label-noisy samples). To formalize our idea, we first map each image representation $\boldsymbol{z}_i$ and its region representations $Z_i = \{\boldsymbol{z}_{ij}\}_{j=1}^{k}$ to a low-dimensional embedding space by a projection head. The l_2 normalized image embedding and k region embeddings are denoted as $\boldsymbol{e}_i$ and $E_i = \{\boldsymbol{e}_{ij}\}_{j=1}^{k} = \{\boldsymbol{r}_\iota\}_{\iota=1}^{k}$, respectively. Define $E^{(c)}$ and $\boldsymbol{E}$ similar to $Z^{(c)}$ and $\boldsymbol{Z}$.

To achieve ①, we softly pull together (resp. push away) *clean* regions from the same class (resp. different classes), guided by the calculated region weights of CoRA. For every pair of region embeddings $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ from the same class and their region weights λ_i and λ_j, the loss function is

$$l(\boldsymbol{r}_i, \boldsymbol{r}_j) = -\log \frac{\exp(\lambda_i \boldsymbol{r}_i \cdot \lambda_j \boldsymbol{r}_j / \tau)}{\sum_{\boldsymbol{r}_v \in \boldsymbol{E} \backslash \boldsymbol{r}_i} \exp(\lambda_i \boldsymbol{r}_i \cdot \lambda_v \boldsymbol{r}_v / \tau)}, \quad (7)$$

where τ is a temperature parameter. The objective function is equivalent to minimizing the following loss:

$$\mathcal{L}_l = \frac{1}{\sum_{c=1}^{C} \frac{kN_c \times (kN_c - 1)}{2}} \sum_{c=1}^{C} \sum_{\boldsymbol{r}_i, \boldsymbol{r}_j \in E^{(c)}} \mathbb{1}_{\boldsymbol{r}_i \neq \boldsymbol{r}_j} l(\boldsymbol{r}_i, \boldsymbol{r}_j). \quad (8)$$

We term $\mathcal{L}_l$ *local compactness* loss, since it encourages the intra-class compactness of clean local regions. By regularizing the task adaptation process with $\mathcal{L}_l$, task-irrelevant local representations from support samples (e.g. cluttered backgrounds of image-noisy samples) can be effectively filtered out during the optimization.

To achieve ②, we propose a *global dispersion* loss that encourages large distances among different class prototypes aggregated by clean images. Inspired by ProtoNet [47], we assign region-level queries to image-level class prototypes in a soft manner, guided by the calculated image and region weights. Concretely, we first use all image embeddings $\{\boldsymbol{e}_i\}_{=1}^{N_s}$ to construct C aggregated class prototypes as

$$\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{y_i = c} \omega_i \boldsymbol{e}_i, \quad c = 1, 2, ..., C, \quad (9)$$

where the impact of label-noisy samples from each class c are weakened by a lower image weight ω_i. Next, we estimate the likelihood of every region embedding $\boldsymbol{r}_j$, based on a softmax over distances to the prototypes:

$$p(y = m | \boldsymbol{r}_j) = \frac{\exp\left(\zeta(\boldsymbol{r}_j, \boldsymbol{\mu}_m)\right)}{\sum_{c=1}^{C} \exp\left(\zeta(\boldsymbol{r}_j, \boldsymbol{\mu}_c)\right)}. \quad (10)$$

The *global dispersion* loss, $\mathcal{L}_g$, thus can be expressed as

$$\mathcal{L}_g = -\frac{1}{N_s \times k} \sum_{i=1}^{N_s \times k} \lambda_i \log\left(p(y = y_i | \boldsymbol{r}_i)\right), \quad (11)$$

where λ_i is used to constrain the contribution of region i. We experimentally found that using different collections of region embeddings, rather than a fixed set of image embeddings (i.e. $\{\boldsymbol{e}_i\}_{=1}^{N_s}$) as queries to enlarge distances among image-level class prototypies in different iterations is more effective (in Eq. 10). One possible reason is that in addition to promoting the inter-class dispersion of clean, image-level class prototypes, $\mathcal{L}_g$ also complements $\mathcal{L}_l$ to improve the intra-class compactness of clean regions by Eq. 10.

Finally, the two losses are complementary to strengthen the mining of more discriminative representations from support samples, by optimizing the following objective:

$$\mathcal{L} = \beta \mathcal{L}_l + \mathcal{L}_g, \quad (12)$$

where β is used to balance the importance of the two losses.

3.5. Task Adaptation and Inference

During task adaptation, we iteratively construct a set of local regions from the inner-task support samples, and perform SGD update using $\mathcal{L}$. At inference, we only retain the adapted model to produce *image representations* of support samples, on which we build a noise-robust prototypical classifier guided by the refined image weights in the momentum accumulator. More details are in **Sup. Mat. (A)**.

Discussion. In terms of computational efficiency, DETA is better equipped to handle the dual noises in few-shot tasks than in training-time base classes or other generic scenarios with large training datasets. Computational issues in DETA caused by **1)** the weighting of inner-task images and regions, and **2)** the multiplicative expansion of support samples (brought by cropped regions) for task adaptation, can be substantially weakened due to the much smaller number of samples in few-shot tasks. Please refer to **Sup. Mat. (D)** for an analysis of DETA w.r.t. computational efficiency.

4. Experiments

In this section, we perform extensive experiments to demonstrate the flexibility and effectiveness of DETA.

Datasets. We conduct experiments on Meta-Dataset (MD) [51], the most comprehensive and challenging large-scale FSL benchmark, which subsumes ten image datasets from various vision domains in one collection, including ImN-MD, Omglot, *etc*. Please refer to [51] for details of MD. We consider two versions of MD in our experiments. *Vanilla MD for image-denoising*: We assume the labels of the ten vanilla MD datasets are clean – a commonly-used assumption in generic label-denoising tasks [10, 21, 28], and directly use the vanilla MD to verify the image-denoising performance of our method. *Label-corrupted MD for label-denoising*: following [21, 28] we scrutinize the robustness of DETA to label noise, by manually corrupting the labels of various ratios (10%∼70%) of support samples. Yet, it is worth mentioning that in the standard task-sampling protocol for MD, the generated test tasks are way/shot imbalanced, *a.k.a. varied-way varied-shot*. To avoid cases where the number of support samples in a class is less than 10, we adopt a unified task sampling protocol for the two MD versions by fixing the shot of every inner-task class to 10, *i.e.*, *varied-way 10-shot*. However, when conducting comparisons with state-of-the-arts, we still employ the standard varied-way varied-shot protocol for fair comparison.

Baseline Methods. We verify the effectiveness and flexibility of DETA by applying it to a broad spectrum of baseline methods applied on various diverse pre-trained models. For A-TA, we consider the two strong baselines TSA [25] and eTT [55]. Both of them integrate a model-specific adapter to the pre-trained model: TSA integrates a residual adapter to the single-domain URL (w/ RN-18 pre-trained on 84×84 ImN-MD) [24] and eTT attaches a self-attention adapter to DINO (ViT-S) [5]. As for F-TA, motivated by [25][2], we use the NCC head instead of a linear classifier which is common in transfer learning literature. We denote this F-TA scheme F-NCC, and use it for adapting different pre-trained models including MOCO (w/ RN-50) [16], CLIP (w/ RN-50) [41], DeiT (w/ ViT-S) [49] and Swin Transformer (Tiny) [33]. All models are trained on Imagenet-1k, except for CLIP, which is trained on large-scale image captions. For all baseline methods, we match the image size in model pre-training and task adaptation, *i.e.*, the image size is set to 84×84 for *TSA* [25], and 224×224 for other methods.

Implementation Details. Following [25, 55], we perform task adaptation by updating the pre-trained model (or the appended task adapter) for 40 iterations on each few-shot task. During each iteration of our DETA, 4 and 2 image regions are cropped from every support sample for TSA and other methods, respectively. The projection head in our network is a two-layer MLP, and the embedding dimension is 128. The two temperatures τ and π, are set to 0.5 and 0.07, respectively. The hyperparameter β is set to 0.1. More detailed settings are provided in **Sup. Mat. (B)**.

Evaluation Metric. We evaluate our method on 600 randomly sampled test tasks for each MD dataset, and report average accuracy (in %) and 95% confidence intervals.

4.1. Experimental Results

In this part, we seek to answer the following questions.

Q1. Can DETA consistently enhance task adaptation results for any types of baselines by performing image-denoising on support samples?

Q2. Can DETA perform robustly in the presence of various ratios of label-noisy support samples?

Q3. Can DETA boost the current state-of-the-art, after tackling the overlooked image noise in the MD benchmark?

Image-denoising. To validate the effectiveness of DETA on image-denoising, we conduct experiments on the vanilla MD with six baseline approaches shown before. The quantitative results of the baseline methods w/ or w/o DETA are reported in Table 1. We can observe from the results: **1)** DETA consistently improves adapter-based and finetuning-based task adaptation methods, which confirms that DETA is orthogonal to those methods and able to improve model robustness to image noise for them. **2)** DETA achieves significant performance gains on both TSA (for 84×84-size input images) and other methods (for 224×224-size images), suggesting DETA's flexibility. **3)** DETA can tackle both the two types of image noise: *background clutter* (in ImgN-MD, etc) and *image corruption* (in Omglot and QkDraw), qualitative results are shown in Section 4.3.

Label-denoising. We further demonstrate the effectiveness of DETA on label-denoising on the label-corrupted MD. Concretely, we manually corrupt the labels of different ratios (10%∼70%) of support samples for each task, by uniformly changing the correct image labels to the other $C-1$ classes. Table 2 reports the average accuracy of different baselines methods w/ or w/o our DETA on the ten MD datasets, under different ratios of corrupted support samples. We have the following observations. **1)** The few-shot classification performance gradually decreases as the ratio of label-noisy support samples increases. **2)** DETA consistently improves the baseline methods by a large margin in all settings, demonstrating its effectiveness to improve model robustness to label noise. **3)** Compared with the obtained image-denoising results in Table 1, the performance gains of DETA w.r.t. label-denoising are more significant. Possible reasons are twofold. **i)** The negative impact of label noise on performance is more significant than that of image noise, as the label-noisy samples contain almost no valuable object features associated with the correct classes. **ii)** When one class contains samples from other classes, our designed CoRA can identify the harmful regions more precisely by taking advantage of out-of-class relevance information.

State-of-the-art Comparison. So far, we can see that our DETA can be flexibly plugged into both adapter-based and finetuning-based task adaptation methods to improve model robustness to the dual noises. It is interesting to investigate whether DETA can further boost the current state-of-the-art after tackling the image-noisy samples in the vanilla MD. Hence, we apply our DETA to the state-of-the-art scheme TSA [25] and conduct experiments on MD with a group of competitors, *e.g.*, FLUTE [50], URL [24], eTT [55]. In Table 5, we can observe DETA considerably improves the strong baseline TSA and establishes new state-of-the-art results on nearly all ten MD datasets, which further confirm the effectiveness and flexibility of our DETA. More importantly, the achieved results also uncover the ever-overlooked image noise problem of the MD benchmark. More qualita-

[2] The nonparametric NCC has been proven in [25] to be more effective for adapter-based or finetuning-based task adaptation than other competitors such as logistic regression, support vector machine and Mahal. Dist.

Model	Method	ImN-MD	Omglot	Acraft	CUB	DTD	QkDraw	Fungi	Flower	COCO	Sign	*Avg*
URL	*TSA* [25]	58.3 ± 0.9	80.7 ± 0.2	61.1 ± 0.7	83.2 ± 0.5	72.5 ± 0.6	78.9 ± 0.6	64.7 ± 0.8	92.3 ± 0.3	75.1 ± 0.7	87.7 ± 0.4	75.5
(RN-18)	***+ DETA***	**58.7 ± 0.9**	**82.7 ± 0.2**	**63.1 ± 0.7**	**85.0 ± 0.5**	**72.7 ± 0.6**	**80.4 ± 0.6**	**66.7 ± 0.8**	**93.8 ± 0.3**	**76.3 ± 0.7**	**92.1 ± 0.4**	**77.3** (+1.8)
DINO	*eTT* [55]	73.2 ± 0.8	93.0 ± 0.4	**68.1 ± 0.7**	89.6 ± 0.3	74.9 ± 0.5	79.3 ± 0.7	76.2 ± 0.5	96.0 ± 0.2	72.7 ± 0.6	86.3 ± 0.7	80.9
(ViT-S)	***+ DETA***	**75.6 ± 0.8**	**93.6 ± 0.4**	67.7 ± 0.8	**91.8 ± 0.3**	**76.0 ± 0.5**	**81.9 ± 0.7**	**77.2 ± 0.5**	**96.9 ± 0.3**	**78.5 ± 0.6**	**88.5 ± 0.7**	**82.8** (+1.9)
MoCo	*F-NCC*	70.7 ± 1.0	82.5 ± 0.4	55.1 ± 0.8	67.0 ± 0.8	**81.3 ± 0.5**	73.8 ± 0.7	54.8 ± 0.9	89.2 ± 0.5	76.8 ± 0.7	79.6 ± 0.6	73.0
(RN-50)	***+ DETA***	**73.6 ± 1.0**	**83.9 ± 0.4**	**59.1 ± 0.8**	**73.9 ± 0.8**	80.9 ± 0.5	**76.1 ± 0.7**	**60.7 ± 0.9**	**92.3 ± 0.5**	**79.0 ± 0.7**	**84.2 ± 0.6**	**76.4** (+3.4)
CLIP	*F-NCC*	67.0 ± 1.0	89.2 ± 0.5	61.2 ± 0.8	84.0 ± 0.7	74.5 ± 0.6	75.5 ± 0.7	57.6 ± 0.9	92.1 ± 0.4	72.1 ± 0.8	79.8 ± 0.7	75.3
(RN-50)	***+ DETA***	**69.6 ± 0.9**	**92.2 ± 0.5**	**59.7 ± 0.8**	**88.5 ± 0.7**	**76.2 ± 0.6**	**77.2 ± 0.7**	**64.5 ± 0.9**	**94.5 ± 0.3**	**72.6 ± 0.8**	**80.7 ± 0.7**	**77.6** (+2.3)
DeiT	*F-NCC*	90.0 ± 0.6	92.5 ± 0.2	65.3 ± 0.7	89.8 ± 0.4	73.9 ± 0.6	83.3 ± 0.5	70.3 ± 0.8	92.2 ± 0.4	83.0 ± 0.6	85.0 ± 0.6	82.5
(ViT-S)	***+ DETA***	**90.8 ± 0.6**	**93.3 ± 0.2**	**71.6 ± 0.7**	**92.4 ± 0.4**	**78.0 ± 0.6**	**84.1 ± 0.6**	**75.2 ± 0.8**	**84.4 ± 0.4**	**95.5 ± 0.6**	**90.0 ± 0.6**	**85.2** (+2.7)
Vanilla	*F-NCC*	90.8 ± 0.8	91.2 ± 0.3	57.6 ± 1.0	88.3 ± 0.5	76.4 ± 0.6	81.9 ± 0.8	67.8 ± 0.9	92.3 ± 0.4	82.5 ± 0.6	83.9 ± 0.8	81.3
SwinT	***+ DETA***	**91.8 ± 0.9**	**92.5 ± 0.3**	**68.9 ± 0.9**	**92.7 ± 0.5**	**79.5 ± 0.7**	**82.8 ± 0.6**	**76.6 ± 0.8**	**96.4 ± 0.4**	**82.9 ± 0.4**	**89.9 ± 0.7**	**85.4** (+4.1)

Table 1. Few-shot classification results of different methods on MD. The A-TA methods *TSA* [25] and *eTT* [55] integrate a model-specific adapter to the pre-trained model, while the F-TA method *F-NCC* use a model-agnostic NCC head for adapting different pre-trained models.

Model	Method	*Ratio of noisy labels*			
		10%	30%	50%	70%
URL	*TSA* [25]	72.8	65.0	54.1	38.3
(RN-18)	***+ DETA***	**74.8** (+2.0)	**67.2** (+2.2)	**56.0** (+1.9)	**40.1** (+1.8)
DINO	*eTT* [55]	78.0	67.7	53.8	37.8
(ViT-S)	***+ DETA***	**80.3** (+2.3)	**70.7** (+3.0)	**58.0** (+4.2)	**41.9** (+4.1)
MoCo	*F-NCC*	70.4	63.3	52.4	36.6
(RN-50)	***+ DETA***	**74.1** (+3.7)	**68.0** (+4.7)	**57.8** (+5.4)	**40.1** (+3.5)
CLIP	*F-NCC*	73.0	65.5	53.3	36.9
(RN-50)	***+ DETA***	**75.7** (+2.7)	**69.7** (+4.2)	**58.5** (+5.2)	**40.8** (+3.9)
DeiT	*F-NCC*	80.0	74.3	64.1	44.9
(ViT-S)	***+ DETA***	**83.3** (+3.3)	**77.2** (+2.9)	**67.1** (+3.0)	**47.7** (+2.8)
Vanilla	*F-NCC*	78.8	71.6	59.8	42.2
SwinT	***+ DETA***	**83.9** (+5.1)	**77.3** (+5.7)	**65.9** (+6.1)	**46.8** (+4.6)

Table 2. Average few-shot classification results of different models on MD, with various ratios of label-noisy support samples.

Image	Region	Image-denoising	Label-denoising (30%)
✓	✗	73.0	63.3
✓	✓	73.8 (+0.8)	63.9 (+0.6)

Table 3. The impact of data augmentation caused by cropped regions on model performance. The baseline is MoCo (w/ RN-50).

ID	Setting	Img-denois.	Label-denois.
$\mathbb{A}$	Baseline	73.8	63.9
$\mathbb{B}$	+ CoRA*	74.1	64.3
$\mathbb{C}$	+ CoRA	74.4	64.8
$\mathbb{D}$	+ CoRA + $\mathcal{L}_l$	75.5	65.4
$\mathbb{E}$	+ CoRA + $\mathcal{L}_g$	75.2	66.6
$\mathbb{F}$	+ CoRA + $\mathcal{L}_l$ + $\mathcal{L}_g$	75.8	67.1
$\mathbb{G}$	+ CoRA + $\mathcal{L}_l$ + $\mathcal{L}_g$ + MA (DETA)	**76.4** (+2.6)	**68.0** (+4.1)

Table 4. Ablation studies for the designed components of DETA on MD. The baseline model is MoCo (w/ RN-50), the ratio of label-noisy support samples is set to 30%.

tive evidence for this problem are discussed in Section 4.3 and demonstrated in Figure 4.

4.2. Ablation Studies

Here, we conduct ablative analysis to investigate the designed components of DETA in Table 4. We also study the impact of data augmentation caused by cropped regions on model performance in Table 3. Unless stated otherwise, the baseline is MoCo (w/ RN-50), the ratio of label-noisy support samples is 30%, and the average results on the ten MD datasets are reported.

Influence of Data Augmentation. DETA leverages both the images and cropped regions of support samples to perform test-time task adaptation. It is important to answer the question: are the performance improvements are mostly attributed to data augmentation? To this end, we remove all the designed components of DETA, and jointly use the images and cropped regions for task adaptation. The results are reported in Table 3. Not surprisingly, without filtering out task-irrelevant, noisy representations, the joint utilization of images and regions for task adaptation does not result in significant performance gains.

Effectiveness of the Designed Components. DETA contains three key components, including a CoRA module, a *local compactness* loss $\mathcal{L}_l$ and a *global dispersion* loss $\mathcal{L}_g$. We conduct component-wise analysis by alternatively adding one of them to understand the influence of each component in Table 4. We take the random crop data augmentation as baseline ($\mathbb{A}$). $\mathbb{B}$ or $\mathbb{C}$: only leverage CoRA to weight the support images at inference. CoRA*: CoRA w/o out-of-class relevance aggregation. $\mathbb{G}$: Our DETA. "+ MA": Inference with the momentum accumulator. As seen, each component in DETA contributes to the performance. In particular, the results in $\mathbb{F}$ suggest that $\mathcal{L}_g$ and $\mathcal{L}_l$ complement each other to improve the denoising performance. The results in $\mathbb{B}$&$\mathbb{C}$ and $\mathbb{G}$ verify the effectiveness of the out-of-class relevance information for CoRA, and the momentum accumulator for building noise-robust classifier, respectively.

Analysis of the Number of Region, Region Size, β, $\zeta(\cdot)$. In **Sup. Mat. (C)**, we study the impacts of the number of region, region size, β and $\zeta(\cdot)$ on performance. We show that **1)** a too larger number of regions or a too small region size does not result in significant performance gains, and **2)** the DETA framework is in general not sensitive to β and the choice of $\zeta(\cdot)$ within a certain range.

Method (w/ RN-18) ♣	*In-Domain*	*Out-of-Domain*									*Avg*
	ImN-MD	Omglot	Acraft	CUB	DTD	QkDraw	Fungi	Flower	COCO	Sign	
Finetune [51]	45.8 ± 1.1	60.9 ± 1.6	68.7 ± 1.3	57.3 ± 1.3	69.0 ± 0.9	42.6 ± 1.2	38.2 ± 1.0	85.5 ± 0.7	34.9 ± 1.0	66.8 ± 1.3	57.0
ProtoNet [51]	50.5 ± 1.1	60.0 ± 1.4	53.1 ± 1.0	68.8 ± 1.0	66.6 ± 0.8	49.0 ± 1.1	39.7 ± 1.1	85.3 ± 0.8	41.0 ± 1.1	47.1 ± 1.1	56.1
FoProMA [51]	49.5 ± 1.1	63.4 ± 1.3	56.0 ± 1.0	68.7 ± 1.0	66.5 ± 0.8	51.5 ± 1.0	40.0 ± 1.1	87.2 ± 0.7	43.7 ± 1.1	48.8 ± 1.1	57.5
Alfa-FoProMA [51]	52.8 ± 1.1	61.9 ± 1.5	63.4 ± 1.1	69.8 ± 1.1	70.8 ± 0.9	59.2 ± 1.2	41.5 ± 1.2	86.0 ± 0.8	48.1 ± 1.1	60.8 ± 1.3	61.4
BOHB [43]	51.9 ± 1.1	67.6 ± 1.2	54.1 ± 0.9	70.7 ± 0.9	68.3 ± 0.8	50.3 ± 1.0	41.4 ± 1.1	87.3 ± 0.6	48.0 ± 1.0	51.8 ± 1.0	59.1
FLUTE [50]	46.9 ± 1.1	61.6 ± 1.4	48.5 ± 1.0	47.9 ± 1.0	63.8 ± 0.8	57.5 ± 1.0	31.8 ± 1.0	80.1 ± 0.9	41.4 ± 1.0	46.5 ± 1.1	52.6
eTT$^\sharp$ [55]	56.4 ± 1.1	72.5 ± 1.4	72.8 ± 1.0	73.8 ± 1.1	77.6 ± 0.8	68.0 ± 0.9	**51.2 ± 1.1**	**93.3 ± 0.6**	55.7 ± 1.0	84.1 ± 1.0	70.5
URL (*Base Model*) [24]	56.8 ± 1.0	79.5 ± 0.8	49.4 ± 0.8	71.8 ± 0.9	72.7 ± 0.7	53.4 ± 1.0	40.9 ± 0.9	85.3 ± 0.7	52.6 ± 0.9	47.3 ± 1.0	61.1
+ *Beta* [24]	58.4 ± 1.1	81.1 ± 0.8	51.9 ± 0.9	73.6 ± 1.0	74.0 ± 0.7	55.6 ± 1.0	42.2 ± 0.9	86.2 ± 0.8	55.1 ± 1.0	59.0 ± 1.1	63.7
+ *TSA* [25]	59.5 ± 1.1	78.2 ± 1.2	72.2 ± 1.0	74.9 ± 0.9	77.3 ± 0.7	67.6 ± 0.9	44.7 ± 1.0	90.9 ± 0.6	59.0 ± 1.0	82.5 ± 0.8	70.7
+ *TSA* + *DETA*	**60.7 ± 1.0**	**81.6 ± 1.2**	**73.0 ± 1.0**	**77.0 ± 0.9**	**78.3 ± 0.7**	**69.5 ± 0.9**	47.6 ± 1.0	92.6 ± 0.6	**60.3 ± 1.0**	**86.8 ± 0.8**	**72.8**

Method (w/ RN-18) ♠	*In-Domain*								*Out-of-Domain*		*Avg*
	ImN-MD	Omglot	Acraft	CUB	DTD	QkDraw	Fungi	Flower	COCO	Sign	
CNAPS [42]	50.8 ± 1.1	91.7 ± 0.5	83.7 ± 0.6	73.6 ± 0.9	59.5 ± 0.7	74.7 ± 0.8	50.2 ± 1.1	88.9 ± 0.5	39.4 ± 1.0	56.5 ± 1.1	66.9
SimpCNAPS [3]	58.4 ± 1.1	91.6 ± 0.6	82.0 ± 0.7	74.8 ± 0.9	68.8 ± 0.9	76.5 ± 0.8	46.6 ± 1.0	90.5 ± 0.5	48.9 ± 1.1	57.2 ± 1.0	69.5
TransCNAPS [2]	57.9 ± 1.1	94.3 ± 0.4	84.7 ± 0.5	78.8 ± 0.7	66.2 ± 0.8	77.9 ± 0.6	48.9 ± 1.2	92.3 ± 0.4	42.5 ± 1.1	59.7 ± 1.1	70.3
SUR [8]	56.2 ± 1.0	94.1 ± 0.4	85.5 ± 0.5	71.0 ± 1.0	71.0 ± 0.8	81.8 ± 0.6	64.3 ± 0.9	82.9 ± 0.8	52.0 ± 1.1	51.0 ± 1.1	71.0
URT [29]	56.8 ± 1.1	94.2 ± 0.4	85.8 ± 0.5	76.2 ± 0.8	71.6 ± 0.7	82.4 ± 0.6	64.0 ± 1.0	87.9 ± 0.6	48.2 ± 1.1	51.5 ± 1.1	71.9
FLUTE [50]	58.6 ± 1.0	92.0 ± 0.6	82.8 ± 0.7	75.3 ± 0.8	71.2 ± 0.8	77.3 ± 0.7	48.5 ± 1.0	90.5 ± 0.5	52.8 ± 1.1	63.0 ± 1.0	71.2
Tri-M [32]	51.8 ± 1.1	93.2 ± 0.5	87.2 ± 0.5	79.2 ± 0.8	68.8 ± 0.8	79.5 ± 0.7	58.1 ± 1.1	91.6 ± 0.6	50.0 ± 1.0	58.4 ± 1.1	71.8
URL (*Base Model*) [24]	57.0 ± 1.0	94.4 ± 0.4	88.0 ± 0.5	80.3 ± 0.7	74.6 ± 0.7	81.8 ± 0.6	66.2 ± 0.9	91.5 ± 0.5	54.1 ± 1.0	49.8 ± 1.0	73.8
+ *Beta* [24]	58.8 ± 1.1	94.5 ± 0.4	89.4 ± 0.4	80.7 ± 0.8	77.2 ± 0.7	82.5 ± 0.6	68.1 ± 0.9	92.0 ± 0.5	57.3 ± 1.0	63.3 ± 1.1	76.4
+ *TSA* [25]	59.5 ± 1.0	94.9 ± 0.4	89.9 ± 0.4	81.1 ± 0.8	77.5 ± 0.7	81.7 ± 0.6	66.3 ± 0.8	92.2 ± 0.5	57.6 ± 1.0	82.8 ± 1.0	78.3
+ *TSA* + *DETA*	**61.0 ± 1.0**	**95.6 ± 0.4**	**91.4 ± 0.4**	**82.7 ± 0.7**	**78.9 ± 0.7**	**83.4 ± 0.6**	**68.2 ± 0.8**	**93.4 ± 0.5**	**58.5 ± 1.0**	**86.9 ± 1.0**	**80.1**

Table 5. Comparison with state-of-the-arts on ten MD datasets (84 × 84). ♣ and ♠ indicate the single-domain (trained on ImN-MD only) and multi-domain (trained on 8 datasets) settings in MD, respectively. ♯ means the feature backbone is ViT-T, with results copied from [55].

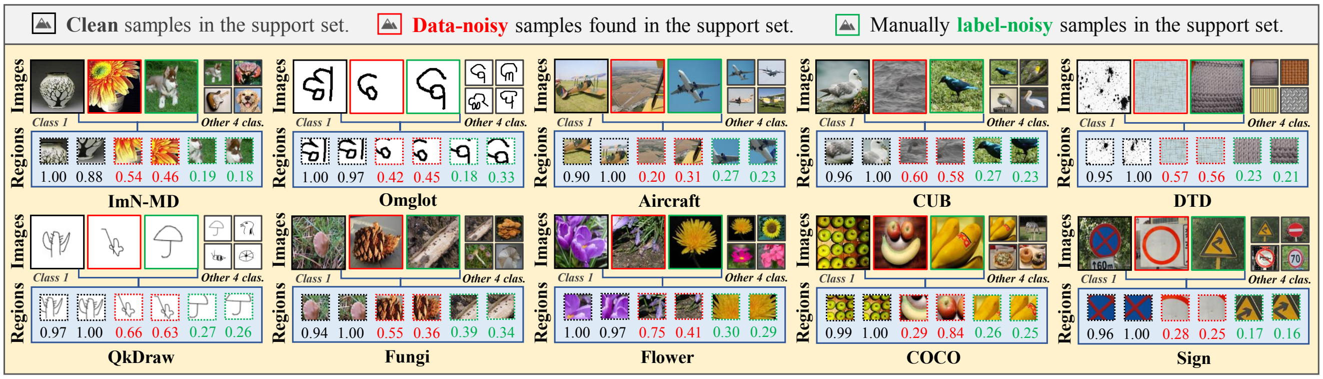


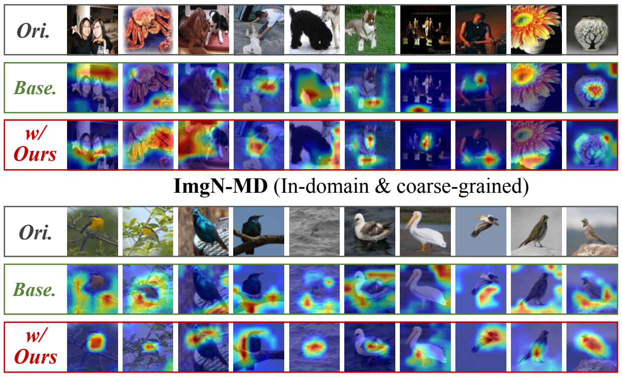
Figure 4. Visualizations of the cropped regions and calculated weights for ten 5-way 10-shot tasks sampled from the 10 MD datasets. We record the region weights after the last iteration. To facilitate comparison, the weights in each class are divided by their maximum value.

4.3. Qualitative Results

Here, we provide some visualization results to qualitatively see how our method works. In Figure 4, we present the visualization of the cropped regions and the calculated weights of CoRA for few-shot tasks from MD. As observed, CoRA successfully assigns larger (resp. smaller) weights to task-specific clean (resp. task-irrelevant noisy) regions for each task. In Figure 5, we show the CAM [44] visualization of the activation maps for two tasks from the representative ImgN-MD and CUB. As shown, our method helps the baseline method accurately locate the task-specific discriminative regions in label-clean but image-noisy samples. For example, on CUB, our method yields more attention on birds rather than cluttered backgrounds.

5. Conclusions

In this work, we propose DETA, a first, unified and plug-and-play framework to tackle the joint (image, label)-noise issue in test-time task adaptation. Without extra supervision, DETA filters out task-irrelevant, noisy representations by taking advantage of both global visual information and local region details of support sample. We evaluate DETA on the challenging Meta-Dataset and demonstrate that it consistently improves the performance of a wide range of baseline methods applied to various pre-trained models. We also uncover the overlooked image noise in Meta-Dataset, by tackling this issue DETA establishes new state-of-the-art results. We hope this work can bring new inspiration to few-shot learning as well as other related fields.



ImgN-MD (In-domain & coarse-grained)

CUB (Out-of-domain & fine-grained)

Figure 5. CAM visualizations on two 5-way 10-shot tasks sampled from ImgN-MD and CUB, respectively. Two images per class are listed for each task. *Please zoom in for details.*

Acknowledgements. This study is supported by grants from National Key R&D Program of China (2022YFC2009903/2022YFC2009900), the National Natural Science Foundation of China (Grant No. 62122018, No. 62020106008, No. 61772116, No. 61872064), Fok Ying-Tong Education Foundation(171106), and SongShan Laboratory YYJC012022019.

References

[1] Devansh Arpit, Stanisław Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. *ICML*, pages 233–242, 2017. 3

[2] Peyman Bateni, Jarred Barber, Jan-Willem van de Meent, and Frank Wood. Enhancing few-shot image classification with unlabelled examples. In *WACVW*, pages 2796–2805, 2022. 8

[3] Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved few-shot visual classification. In *CVPR*, pages 14493–14502, 2020. 2, 8

[4] Jonathan Bragg, Arman Cohan, Kyle Lo, and Iz Beltagy. Flex: Unifying evaluation for few-shot nlp. *NeurIPS*, 34:15787–15800, 2021. 1

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *ICCV*, pages 9650–9660, 2021. 6

[6] Guneet Singh Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. In *ICLR*, 2020. 1

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2020. 2, 4

[8] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Selecting relevant features from a multi-domain representation for few-shot classification. In *ECCV*, pages 769–786, 2020. 8

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *ICML*, page 1126–1135, 2017. 1, 4

[10] Benoît Frénay and Michel Verleysen. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):845–869, 2013. 5

[11] Yuqian Fu, Yanwei Fu, Jingjing Chen, and Yu-Gang Jiang. Generalized meta-fdmixup: Cross-domain few-shot learning guided by labeled target data. *IEEE Transactions on Image Processing*, 31:7078–7090, 2022. 2

[12] Yuqian Fu, Yu Xie, Yanwei Fu, and Yu-Gang Jiang. Styleadv: Meta style adversarial training for cross-domain few-shot learning. In *CVPR*, pages 24575–24584, 2023. 2

[13] Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. In *CVPR*, pages 21–30, 2019. 1

[14] Kai Han, An Xiao, Enhua Wu, Jianyuan Guo, Chunjing Xu, and Yunhe Wang. Transformer in transformer. *NeurIPS*, 34:15908–15919, 2021. 1

[15] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 16000–16009, 2022. 1, 2, 4

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 1, 6

[17] Jie Hong, Pengfei Fang, Weihao Li, Tong Zhang, Christian Simon, Mehrtash Harandi, and Lars Petersson. Reinforced attention for few-shot learning and beyond. In *CVPR*, pages 913–923, 2021. 1

[18] Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. *NeurIPS*, 32, 2019. 2, 3

[19] Shell Xu Hu, Da Li, Jan Stühmer, Minyoung Kim, and Timothy M Hospedales. Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference. In *CVPR*, pages 9068–9077, 2022. 1, 3

[20] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, Ondřej Chum, and Cordelia Schmid. Graph convolutional networks for learning with few clean and many noisy labels. In *ECCV*, pages 286–302. Springer, 2020. 3

[21] Shuming Kong, Yanyan Shen, and Linpeng Huang. Resolving training biases via influence-based data relabeling. In *ICLR*, 2021. 2, 5

[22] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *CVPR*, pages 2661–2671, 2019. 4

[23] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. Revisiting local descriptor based image-to-class measure for few-shot learning. In *CVPR*, pages 7260–7268, 2019. 3

[24] Wei-Hong Li, Xialei Liu, and Hakan Bilen. Universal representation learning from multiple domains for few-shot classification. In *ICCV*, pages 9526–9535, 2021. 1, 4, 6, 8

[25] Wei-Hong Li, Xialei Liu, and Hakan Bilen. Cross-domain few-shot learning with task-specific adapters. In *CVPR*, pages 7161–7170, 2022. 1, 2, 4, 6, 7, 8

[26] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*, 2017. 1

[27] Hanwen Liang, Qiong Zhang, Peng Dai, and Juwei Lu. Boosting the generalization capability in cross-domain few-shot learning via noise-enhanced supervised autoencoder. In *ICCV*, pages 9424–9434, 2021. 2

[28] Kevin J Liang, Samrudhdhi B Rangrej, Vladan Petrovic, and Tal Hassner. Few-shot learning with noisy labels. In *CVPR*, pages 9089–9098, 2022. 3, 5

[29] Lu Liu, William Hamilton, Guodong Long, Jing Jiang, and Hugo Larochelle. A universal representation transformer layer for few-shot image classification. *ICLR*, 2021. 8

[30] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. In *ICCV*, pages 10551–10560, 2019. 1

[31] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(3):447–461, 2015. 2

[32] Yanbin Liu, Juho Lee, Linchao Zhu, Ling Chen, Humphrey Shi, and Yi Yang. A multi-mode modulator for multi-domain few-shot classification. In *ICCV*, pages 8453–8462, 2021. 8

[33] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pages 10012–10022, 2021. 1, 6

[34] Xu Luo, Longhui Wei, Liangjian Wen, Jinrong Yang, Lingxi Xie, Zenglin Xu, and Qi Tian. Rectifying the shortcut learning of background for few-shot learning. *NeurIPS*, 34:13073–13085, 2021. 2, 3

[35] Xu Luo, Hao Wu, Ji Zhang, Lianli Gao, Jing Xu, and Jingkuan Song. A closer look at few-shot classification again. *arXiv preprint arXiv:2301.12246*, 2023. 1, 2, 3

[36] Pratik Mazumder, Pravendra Singh, and Vinay P Namboodiri. Rnnp: A robust few-shot learning approach. In *WACV*, pages 2664–2673, 2021. 3

[37] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2624–2637, 2013. 4

[38] Yujie Mo, Yajie Lei, Jialie Shen, Xiaoshuang Shi, Heng Tao Shen, and Xiaofeng Zhu. Disentangled multiplex graph representation learning. In *ICML*, 2023. 3

[39] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018. 1

[40] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. *NeurIPS*, 31, 2018. 2

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021. 6

[42] James Requeima, Jonathan Gordon, John Bronskill, Sebastian Nowozin, and Richard E Turner. Fast and flexible multi-task classification using conditional neural adaptive processes. *NeurIPS*, 32, 2019. 1, 2, 8

[43] Tonmoy Saikia, Thomas Brox, and Cordelia Schmid. Optimized generic feature learning for few-shot classification across domains. *arXiv preprint arXiv:2001.07926*, 2020. 8

[44] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017. 8

[45] Shuai Shao, Yan Wang, Bin Liu, Weifeng Liu, Yanjiang Wang, and Baodi Liu. Fads: Fourier-augmentation based data-shunting for few-shot classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023. 2

[46] Zhiqiang Shen, Zechun Liu, Jie Qin, Marios Savvides, and Kwang-Ting Cheng. Partial is better than all: Revisiting fine-tuning strategy for few-shot learning. In *AAAI*, volume 35, pages 9594–9602, 2021. 1

[47] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, pages 4077–4087, 2017. 1, 5

[48] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022. 2, 3

[49] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers-distillation through attention. In *ICML*, pages 10347–10357, 2021. 6

[50] Eleni Triantafillou, Hugo Larochelle, Richard Zemel, and Vincent Dumoulin. Learning a universal template for few-shot dataset generalization. In *ICML*, pages 10424–10433, 2021. 6, 8

[51] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. *ICLR*, 2020. 2, 4, 5, 8

[52] Davis Wertheimer, Luming Tang, and Bharath Hariharan. Few-shot classification with feature map reconstruction networks. In *CVPR*, pages 8012–8021, 2021. 3

[53] Ziyang Wu, Yuwei Li, Lihua Guo, and Kui Jia. Parn: Position-aware relation networks for few-shot learning. In *ICCV*, pages 6659–6667, 2019. 3

[54] Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro O O Pinheiro. Adaptive cross-modal few-shot learning. *NeurIPS*, 32, 2019. 1

[55] Chengming Xu, Siqian Yang, Yabiao Wang, Zhanxiong Wang, Yanwei Fu, and Xiangyang Xue. Exploring efficient few-shot adaptation for vision transformers. *Transactions on Machine Learning Research*, 2022. 1, 2, 4, 6, 7, 8

[56] Jing Xu, Xu Luo, Xinglin Pan, Wenjie Pei, Yanan Li, and Zenglin Xu. Alleviating the sample selection bias in few-shot learning by removing projection to the centroid. *NeurIPS*, 2022. 3

[57] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *CVPR*, pages 8808–8817, 2020. 2

[58] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *CVPR*, pages 12203–12213, 2020. 3

[59] Ji Zhang, Jingkuan Song, Lianli Gao, Ye Liu, and Heng Tao Shen. Progressive meta-learning with curriculum. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(9):5916–5930, 2022. 2

[60] Ji Zhang, Jingkuan Song, Lianli Gao, and Hengtao Shen. Free-lunch for cross-domain few-shot learning: Style-aware episodic training with robust contrastive learning. In *ACM MM*, pages 2586–2594, 2022. 2

[61] Ji Zhang, Jingkuan Song, Yazhou Yao, and Lianli Gao. Curriculum-based meta-learning. In *ACM MM*, pages 1838–1846, 2021. 2