Domain Adaptive Few-Shot Open-Set Learning

Debabrata Pal[1], Deeptej More[2], Sai Bhargav[1], Dipesh Tamboli[3], Vaneet Aggarwal[3], Biplab Banerjee[1]

[1]Indian Institute of Technology, Bombay, [2]Manipal Institute of Technology, India, [3]Purdue University

debabrata.pal@iitb.ac.in, deeptejrane16@gmail.com, sai.bhargav@iitb.ac.in,
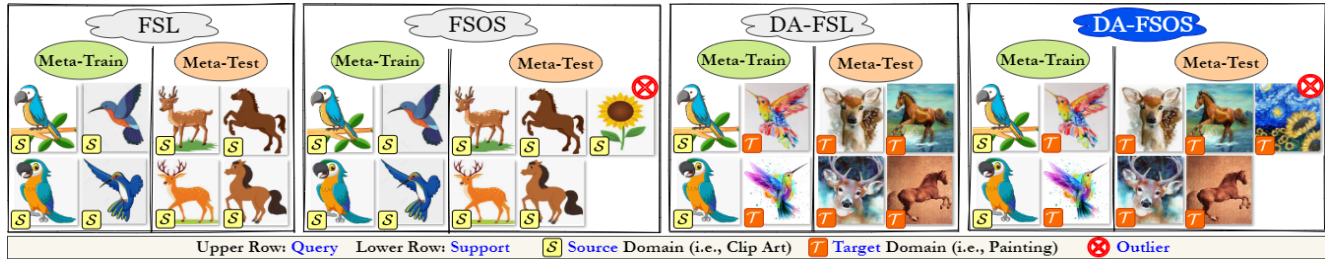dtamboli@purdue.edu, vaneet@purdue.edu, getbiplab@gmail.com

Figure 1. Domain Adaptive Few-Shot Open-Set Learning (DA-FSOS) tackles the challenges of both Domain Adaptive Few-Shot Learning (DA-FSL) and Few-Shot Open-Set Learning (FSOS) by integrating them into a unified framework. In training phase, the model observes a fully-supervised source domain $\mathcal{S}$ and a sparsely-supervised target domain $\mathcal{T}$, where the labels of both domains are disjoint. However, during testing, the model encounters a few-shot support set of new *known* classes from $\mathcal{T}$, while the test query set contains unlabeled samples from both the known and previously unknown classes. **Project page:** https://github.com/DebabrataPal7/DAFOSNET

Abstract

Few-shot learning has made impressive strides in addressing the crucial challenges of recognizing unknown samples from novel classes in target query sets and managing visual shifts between domains. However, existing techniques fall short when it comes to identifying target outliers under domain shifts by learning to reject pseudo-outliers from the source domain, resulting in an incomplete solution to both problems. To address these challenges comprehensively, we propose a novel approach called Domain Adaptive Few-Shot Open Set Recognition (DA-FSOS) and introduce a meta-learning-based architecture named DAFOS-NET. During training, our model learns a shared and discriminative embedding space while creating a pseudo-open-space decision boundary, given a fully-supervised source domain and a label-disjoint few-shot target domain. To enhance data density, we use a pair of conditional adversarial networks with tunable noise variances to augment both domains' closed and pseudo-open spaces. Furthermore, we propose a domain-specific batch-normalized class prototypes alignment strategy to align both domains globally while ensuring class-discriminativeness through novel metric objectives. Our training approach ensures that DAFOS-NET can generalize well to new scenarios in the target domain. We present three benchmarks for DA-FSOS based on the Office-Home, mini-ImageNet/CUB, and DomainNet datasets and demonstrate the efficacy of DAFOS-Net through extensive experimentation.

1. Introduction

The development of deep learning techniques has led to significant advancements in visual recognition tasks by leveraging their ability to learn data-driven features from a vast amount of training examples [14, 21]. However, labeling is a laborious and costly task, which is why few-shot learning (FSL) [33, 22] aims to recognize target classes with limited supervision by learning transferable features from a label-disjoint source domain that has sufficient supervision.

Despite progress [40], FSL models encounter difficulties in two critical scenarios: **i)** when the under-represented target domain classes are drawn from a different distribution than the source domain. One solution to this is *Domain Adaptive Few-Shot Learning* (DA-FSL), which adapts disjoint classes from the fully-supervised source and sparsely-supervised target domains during training [41, 34, 10], and **ii)** *Few-Shot Open-Set Learning* (FSOS) [25], which assumes no domain gap between source and target but combines FSL with *Open-Set Recognition* (OSR) [12, 1] to detect novel class samples as outliers during testing. However, it is desirable for a generic FSL system to handle domain discrepancy and reject outliers during inference in a unified manner for practical applications. To this end, we introduce a novel scenario, called *Domain Adaptive Few-Shot Open-Set Learning* (DA-FSOS), which is illustrated in Fig 1.

DA-FSOS is a method that addresses a challenging problem: *how to learn from supervision in one domain while adapting to a different domain with non-overlapping known*

classes and potential test-time outliers under a few-shot setting. During training, the method uses supervision from the source domain and a few-shot training set from a disjoint set of classes of the target domain. However, during testing, the method is expected to handle unlabelled samples from a new set of *known* and *open-set* classes from the target domain, adding an extra challenge. DA-FSOS can be applied to self-driving cars, where learning to identify known classes and reject outliers from abundant gaming data, the real-world objects of interest (known and unknown) on the road can be identified. Also, to detect novel animal species, unknown land cover objects, unknown viruses in medical imaging, etc., DA-FSOS can become instrumental.

DA-FSOS combines several techniques to tackle these challenges, including domain adaptation, few-shot learning, and open-set recognition. The setting is designed to handle unconstrained domain differences, extremely limited supervision in the target domain, and the absence of prior knowledge regarding the target open space in test time.

While DA-FSL and FSOS techniques can be combined, they may not be sufficient to effectively solve the DA-FSOS problem as they rely on different assumptions individually (see Section 4). Similarly, combining an FSL technique with open-set DA [29] or a DA-FSL model with the OSR approach may not be suitable since DA variants assume label space consistency between domains, which is not the case for DA-FSOS. *Therefore, it is necessary to develop a model specifically designed to solve DA-FSOS.*

Our proposed DAFOS-NET: In this paper, we present a novel model called DAFOS-NET that addresses the DA-FSOS problem by integrating three crucial considerations. Our approach aims to learn a prototype-based classifier that can reject target outliers during testing, and we propose a meta-training method that simulates the test scenario by dividing the available training classes from both domains into known and pseudo-unknown categories in each episode.

To address the issue of overfitting caused by limited target supervision, we propose a ***data augmentation*** technique in DAFOS-NET. Unlike previous FSOS methods that only augment known classes from the source domain [27], our approach involves augmenting both known and pseudo-unknown classes from both source and target domains using a pair of class and domain conditional adversarial networks (cGANs). To maintain feature consistency when generating known samples, we apply a low noise variance for the respective cGAN. However, to increase the scatter when generating pseudo-unknown samples, we use a high noise variance. To prevent mode collapse during training, we introduce a regularizer that ensures the outputs of the two cGANs are consistently different.

We also introduce ***Global Cross-Domain Prototype Alignment (GCDPA)*** strategy by exploiting domain-specific batch-norm statistics to bring the domains closer for the smooth transfer of source knowledge into the target. However, we must ensure that the adaptation does not cause misalignments between the domains. To maintain class discriminativeness, we introduce a class compactness loss for the known class samples. Simultaneously, a novel ***prototype diversification loss*** objective is introduced to maximize the gap between the known-class prototypes and a combination of in-domain pseudo-unknown samples with the across-domain class prototypes.

Finally, we propose a learning-based approach to automatically predict the domain and inlier/outlier class labels for the test samples. This approach eliminates the need for manual threshold selection [25] and improves the ability to solve ***generalized DA-FSOS*** by selectively considering either source or target prototypes based on the predicted domain labels for classifying the test queries.

In summary, our **novel contributions** are as follows:
1. We introduce DA-FSOS, a practical problem setting that generalizes FSOS and DA-FSL, and to solve that, we propose an end-to-end solution called DAFOS-NET.
2. DAFOS-NET integrates four novel ideas: **i**) An episodic training strategy to develop a domain-agnostic open-set classifier. **ii**) A generative feature augmentation scheme that produces diversified known and pseudo-unknown class samples for both domains from few-shot training samples. **iii**) A metric objective that ensures class compactness and discriminativeness for both domains. **iv**) A prototypical batch-norm alignment-based global domain adaptation.
3. We present the *standard* and *generalized* DA-FSOS training and evaluation protocols. Accordingly, we evaluate our approach on DomainNet[31], miniImageNet-CUB [39, 38], and Office-Home [37] datasets and achieve an average gain of 15% closed accuracy and 17% AUROC.

2. Related works

Few-shot learning: Given the significance of FSL, various techniques have been introduced in computer vision and natural language processing. Among them, meta-learning has received substantial attention, owing to the advantages of the episodic training strategy in facilitating *learning to learn*. The metric learning-based meta-learning [33, 38, 19] focus on distance metric learning for nearest neighbour (NN) search, aiming to learn a discriminative embedding space from low-shot support samples. Model-based FSL [32, 9] leverage novel optimization algorithms, rather than gradient descent, to fit in the few-shot regime. Additionally, several FSL models rely on data augmentation [42, 5] and self-supervision [18, 6] to improve performance. For a comprehensive survey of FSL, see [40, 30, 36]. However, *it is essential to note that FSL is typically a closed-set learning problem where the classes in the support and query sets are identical, and it may not account for extreme domain differences between training and testing.*
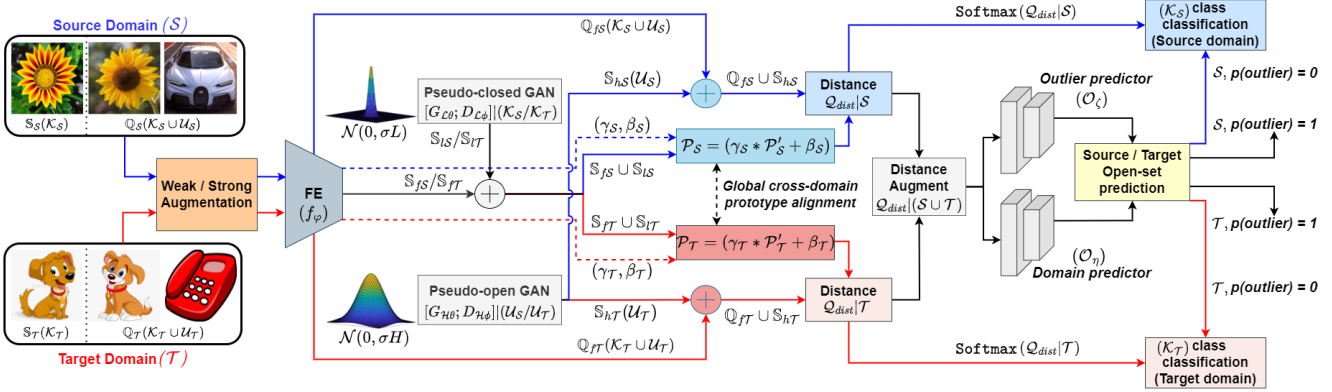
Figure 2. The model architecture of DAFOS-NET. In each episodic training, Support $\mathbb{S}_{\mathcal{S}}(\mathcal{K}_{\mathcal{S}})/\mathbb{S}_{\mathcal{T}}(\mathcal{K}_{\mathcal{T}})$ and Query samples $\mathbb{Q}_{\mathcal{S}}(\mathcal{K}_{\mathcal{S}} \cup \mathcal{U}_{\mathcal{S}})/\mathbb{Q}_{\mathcal{T}}(\mathcal{K}_{\mathcal{T}} \cup \mathcal{U}_{\mathcal{T}})$ from $\mathcal{D}_s$ and $\mathcal{D}_d$ with $\mathcal{K}_{\mathcal{S}}/\mathcal{K}_{\mathcal{T}}$ known and $\mathcal{U}_{\mathcal{S}}/\mathcal{U}_{\mathcal{T}}$ pseudo-unknown classes are considered from $\mathcal{S}$ and $\mathcal{T}$. The domain and class conditional GANs $(G_{\mathcal{L}\theta}, D_{\mathcal{L}\phi})$ and $(G_{\mathcal{H}\theta}, D_{\mathcal{H}\phi})$ separately synthesize features for the known and pseudo-unknown classes. The augmented support sets $\mathbb{S}_{f\mathcal{S}} \cup \mathbb{S}_{l\mathcal{S}}$ and $\mathbb{S}_{f\mathcal{T}} \cup \mathbb{S}_{l\mathcal{T}}$ are used to produce the class prototype embeddings $\mathcal{P}'_{\mathcal{S}}$ and $\mathcal{P}'_{\mathcal{T}}$ for both the domains, which are aligned through the learnable domain-specific batch-norm parameters $(\gamma_{\mathcal{S}}, \beta_{\mathcal{S}})$ and $(\gamma_{\mathcal{T}}, \beta_{\mathcal{T}})$. Finally, a pair of domain-agnostic classifiers $(\mathcal{O}_{\varsigma}, \mathcal{O}_{\eta})$ are deployed for the purpose of domain and inlier/outlier class predictions for the query samples.

Few-shot open-set learning: FSL struggles to handle unknown class samples during testing. A common but suboptimal approach is to use closed-set FSL models for outlier rejection by setting a confidence threshold. However, finding the optimal threshold is challenging. To address this issue, researchers have focused on dedicated FSOS techniques, such as forming Gaussian clusters from a few training samples [25] or optimizing confidence for outlier detection [17]. However, these methods struggle with outliers that resemble known class samples. Recent advances in FSOS, including the binary closed-open discriminator optimized episodically [28] and the use of generative augmentaton [27], have shown promising results in handling outliers. Nonetheless, these methods can not handle domain shifts between the training and test sets. *Our augmentation strategy differs from both of [27, 28], as we seek to generate pseudo-known and unknown samples of both the domains, as opposed to those single domain FSOS methods.*

Domain adaptive few-shot learning: In cross-domain few-shot learning, global or class-conditional domain shifts may occur between the disjoint base and novel classes. One straightforward approach is to combine an off-the-shelf FSL technique with a cross-domain alignment objective [11]. However, direct domain alignment [2] may not always be preferable due to the label shift between the source and target domains. To address this issue, [41, 34] proposed a method to align the domains while preserving global class discrimination. Alternatively, [10] utilized cross-domain mix-up samples to learn generalized features. Lastly, [13] extended this setup beyond natural image classification by using a meta-learning-based approach. *In contrast, we propose a simple and efficient batch-norm-based global cross-domain prototype alignment strategy while maintaining class separation through dedicated metric losses.*

3. Proposed methodology

3.1. Problem definition and notations

Under our DA-FSOS setting, we have access to a large training set $\mathcal{D}_s$ from a set of source classes $\mathcal{C}_s$ in a source domain $\mathcal{S}$, a few-shot sample set $\mathcal{D}_d$ from a set of target classes $\mathcal{C}_d$ in a target domain $\mathcal{T}$ $(P(\mathcal{S}) \neq P(\mathcal{T}))$, and a test set $\mathcal{D}_t$ from another set of $\mathcal{C}_t$ classes in $\mathcal{T}$. It is important to note that $\mathcal{C}_s \cap \mathcal{C}_d = \emptyset$, $\mathcal{C}_t \cap \mathcal{C}_d = \emptyset$, and $\mathcal{C}_s \cap \mathcal{C}_t = \emptyset$. Additionally, the classes in $\mathcal{C}_t$ consist of known and unknown classes, respectively. During testing, the few-shot support set $\mathcal{D}'_t$ contains samples for a subset of $\mathcal{C}'_t \subset \mathcal{C}_t$ known classes, while the remaining $\mathcal{C}_t - \mathcal{C}'_t$ classes in $\mathcal{D}_t$ are unknown during training. Our goal is to use $\mathcal{D}_s$ and $\mathcal{D}_d$ to learn an open-set classifier that can distinguish the samples from $\mathcal{D}_t$ into one of the known classes or an unknown common class with respect to $\mathcal{D}'_t$.

In our setting, each episode consists of a set of known and pseudo-unknown classes denoted by $\mathcal{K}_{\mathcal{S}}$ and $\mathcal{U}_{\mathcal{S}}$ from $\mathcal{S}$, where $\mathcal{K}_{\mathcal{S}}, \mathcal{U}_{\mathcal{S}} \in \mathcal{C}_s$. Similarly, from $\mathcal{T}$, the set of known and pseudo-unknown classes are represented as $\mathcal{K}_{\mathcal{T}}$ and $\mathcal{U}_{\mathcal{T}}$, where $\mathcal{K}_{\mathcal{T}}, \mathcal{U}_{\mathcal{T}} \in \mathcal{C}_d$. $\mathcal{K}_{\mathcal{S}} \neq \mathcal{U}_{\mathcal{S}}$ and $\mathcal{K}_{\mathcal{T}} \neq \mathcal{U}_{\mathcal{T}}$

To form the source support, we use $m_{\mathcal{S}}$ samples from each class in $\mathcal{K}_{\mathcal{S}}$, denoted by $\mathbb{S}_{\mathcal{S}} = \{(x_i^s, y_i^s)\}_{i=1}^{|\mathcal{K}_{\mathcal{S}}|*m_{\mathcal{S}}}$. Similarly, we use $m_{\mathcal{T}}$ samples from each class in $\mathcal{K}_{\mathcal{T}}$ to form the target support set, denoted by $\mathbb{S}_{\mathcal{T}} = \{(x_i^t, y_i^t)\}_{i=1}^{|\mathcal{K}_{\mathcal{T}}|*m_{\mathcal{T}}}$. Our support set formation follows the $\mathcal{K}$-way m-shot learning protocol, where $m_{\mathcal{S}} + m_{\mathcal{T}} = m$, $|\mathcal{K}_{\mathcal{S}}| = |\mathcal{K}_{\mathcal{T}}| = |\mathcal{K}|$ and $|\mathcal{U}_{\mathcal{S}}| = |\mathcal{U}_{\mathcal{T}}| = |\mathcal{U}|$. Here, $|\mathcal{K}|$ represents the cardinality operator of $\mathcal{K}$. Also, we employ more supervision in $\mathcal{S}$ and limited samples in $\mathcal{T}$ for episodic training of DA-FSOS. Thereby, $m_{\mathcal{S}} > m_{\mathcal{T}}$. We represent the query sets by $\mathbb{Q}_{\mathcal{S}}(\mathcal{K}_{\mathcal{S}} \cup \mathcal{U}_{\mathcal{S}})$ and $\mathbb{Q}_{\mathcal{T}}(\mathcal{K}_{\mathcal{T}} \cup \mathcal{U}_{\mathcal{T}})$.

3.2. Model and training overview: DAFOS-NET

Architecture overview: Referring to Fig 2, DAFOS-NET is composed of several components. First, there is a pre-trained feature extractor, denoted as f_φ, that encodes the original images along with their augmented versions. In addition, there are two conditional GANs, represented as $(G_{\mathcal{L}\theta}, D_{\mathcal{L}\phi})$ and $(G_{\mathcal{H}\theta}, D_{\mathcal{H}\phi})$, each consisting of a generator and discriminator sub-network. These cGANs further augment the closed classes and the pseudo-open-space for both domains at the feature level, using two noise distributions, $\mathcal{N}(0, \sigma L)$ and $\mathcal{N}(0, \sigma H)$, where $\sigma H >> \sigma L$. The variance for the closed-set classes, σL, is kept low to discourage the generation of sparse features. On the other hand, a high variance, σH, is used to synthesize more diverse pseudo-open-space features.

Finally, we introduce two classifiers: a domain classification network denoted as $\mathcal{O}_\eta$, that predicts whether a sample belongs to the *source* or *target* domain, and an *inlier/outlier* classification network, denoted as $\mathcal{O}_\zeta$, that aids in making better decisions. To ensure that the domains are aligned globally, we introduce the domain-specific batch-norm layer into DAFOS-NET. Specifically, we learn translation and scaling parameters for feature normalization on a per-batch basis for each domain, which are denoted as $(\gamma_\mathcal{S}, \beta_\mathcal{S})$ and $(\gamma_\mathcal{T}, \beta_\mathcal{T})$, respectively.

Training overview: Our training involves loss optimization based on samples from both domains. For 1-shot meta-training, we take each domain data in tandem in each episode. To begin with, we use f_φ to generate feature embeddings for the support and query sets of both domains in $\mathbb{S}_{f\mathcal{S}/f\mathcal{T}}$ and $\mathbb{Q}_{f\mathcal{S}/f\mathcal{T}}$. We train the cGANs for synthesizing domain-specific features according to the $\mathcal{K}_\mathcal{S}/\mathcal{K}_\mathcal{T}$ known classes and the $\mathcal{U}_\mathcal{S}/\mathcal{U}_\mathcal{T}$ pseudo-unknown classes, respectively (Eq. 1). However, training cGANs from limited samples is challenging due to the lack of sufficient gradients to train the generators. To address this issue, we use DAWSON [24], an optimization-based meta-learning method, to bridge GANs' likelihood-invariant training and obtain gradients in the low-shot regime. Specifically, we use first-order optimization-based meta-learning [26].

The discriminator $D_{\mathcal{L}\phi}$ penalizes the generator $G_{\mathcal{L}\theta}$ for producing slightly fake known features $\mathbb{S}_{l\mathcal{S}}/\mathbb{S}_{l\mathcal{T}}$, given the low noise variation σL, such that the union of $\mathbb{S}_{f\mathcal{S}/f\mathcal{T}}$ and $\mathbb{S}_{l\mathcal{S}/l\mathcal{T}}$ better estimates the density for the known classes of $\mathcal{S}$ and $\mathcal{T}$. Similarly, $D_{\mathcal{H}\phi}$ penalizes $G_{\mathcal{H}\theta}$, which considers a higher noise variance σH, for producing highly fake pseudo-unknown features $\mathbb{S}_{h\mathcal{S}}/\mathbb{S}_{h\mathcal{T}}$. To avoid possible mode collapse for the samples from both the cGANs, we introduce a novel regularizer that penalizes the generation of similar features for identical noise vectors (Eq. 2). Moving forward, we calculate the class prototypes $\mathcal{P}'_\mathcal{S}$ and $\mathcal{P}'_\mathcal{T}$ given $\mathbb{S}_{f\mathcal{S}} \cup \mathbb{S}_{l\mathcal{S}}$ and $\mathbb{S}_{f\mathcal{T}} \cup \mathbb{S}_{l\mathcal{T}}$, respectively (Eq. 3). For domain alignment, we propose aligning the styles of $\mathcal{P}_\mathcal{S}$ and $\mathcal{P}_\mathcal{T}$ by

normalizing the prototype features with respect to $(\gamma_\mathcal{S}, \beta_\mathcal{S})$ and $(\gamma_\mathcal{T}, \beta_\mathcal{T})$, and minimizing the global distance between all the normalized source and target prototypes (Eq. 4).

To ensure the discriminativeness of the embedding space, we propose a supervised contrastive loss $\mathcal{L}_{C|\mathcal{S} \cup \mathcal{T}}$ (Eq. 5) to minimize the vectorized euclidean distance ($\mathcal{Q}_{dist}$) between a query sample and its respective prototype, which leads to more compactness, where, $\mathcal{Q}_{dist} = \{(\mathcal{Q}_{dist_1}, \cdots, \mathcal{Q}_{dist_n}), (\mathcal{Q}_{dist} = \|\mathbb{Q}_{f\mathcal{S}/f\mathcal{T}} - \mathcal{P}_{\mathcal{S}/\mathcal{T}}\|_2^2)\}$. Additionally, we introduce a novel prototype diversification objective $\mathcal{L}_{PD}$ (Eq. 6) that ensures the similarity between a prototype and its positive class samples is at least a margin α higher than the similarity between that prototype and all the class prototypes from the other domain and the in-domain pseudo-unknown class samples. Finally, we optimize the classification losses for $(\mathcal{O}_\zeta, \mathcal{O}_\eta)$ given $\mathcal{Q}_{dist}$ (Eq. 7-8). In each episode, we train the model end-to-end concerning all the losses (Eq. 9) and discuss them below.

3.3. Loss functions, training, and inference

The cGAN training losses: In Eq. 1, we present the min-max loss objectives, $\mathcal{L}_h$ and $\mathcal{L}_l$, for the feature-generating cGANs that operate in pseudo-open and closed spaces. The real data for hallucinating closed space is $\mathbb{S}_{f\mathcal{S}/f\mathcal{T}}(\mathcal{K}_\mathcal{S}/\mathcal{K}_\mathcal{T})$, and the same for the open space is $\mathbb{Q}_{f\mathcal{S}/f\mathcal{T}}(\mathcal{U}_\mathcal{S}/\mathcal{U}_\mathcal{T})$. The synthesized features from the closed and open space cGANs for both $\mathcal{S}$ and $\mathcal{T}$ are represented by $\mathbb{S}_{l\mathcal{S}/l\mathcal{T}}$ and $\mathbb{S}_{h\mathcal{S}/h\mathcal{T}}$, respectively. To ensure that the pseudo-open and closed feature synthesis does not overlap, we incorporate a novel regularizer (Eq. 2) in $\mathcal{L}_h$.

$$\mathcal{L}_h = \min_{G_{\mathcal{H}\theta}} \max_{D_{\mathcal{H}\phi}} \mathbb{E}_{s \sim \mathbb{Q}_{f\mathcal{S}/f\mathcal{T}}(\mathcal{U}_\mathcal{S}/\mathcal{U}_\mathcal{T})}[log D_{\mathcal{H}\phi}(s|\mathcal{U}_\mathcal{S}/\mathcal{U}_\mathcal{T})]$$

$$+ \mathbb{E}_{z_h \sim \mathcal{N}(0,\sigma H)}[log(1 - D_{\mathcal{H}\phi}(G_{\mathcal{H}\theta}(z_h|\mathcal{U}_\mathcal{S}/\mathcal{U}_\mathcal{T})) + \mathcal{L}_{AOCMC}$$

$$\mathcal{L}_l = \min_{G_{\mathcal{L}\theta}} \max_{D_{\mathcal{L}\phi}} \mathbb{E}_{s \sim \mathbb{S}_{f\mathcal{S}/f\mathcal{T}}}[log D_{\mathcal{L}\phi}(s|\mathcal{K}_\mathcal{S}/\mathcal{K}_\mathcal{T}))]$$

$$+ \mathbb{E}_{z_l \sim \mathcal{N}(0,\sigma L)}[log(1 - D_{\mathcal{L}\phi}(G_{\mathcal{L}\theta}(z_l|\mathcal{K}_\mathcal{S}/\mathcal{K}_\mathcal{T}))]; \qquad (1)$$

Anti open close mode collapse loss: To ensure a controlled generation of diverse open and closed space features for both the domains by $G_{\mathcal{H}\theta}$ and $G_{\mathcal{L}\theta}$, respectively, using inputs $z_l \in \mathcal{N}(0, \sigma L)$ and $z_h \in \mathcal{N}(0, \sigma H)$, it is essential to avoid generating identical $s_h \in \mathbb{S}_h$ and $s_l \in \mathbb{S}_l$ for similar z_l and z_h inputs to the cGANs. To this end, we propose the anti-open-close mode collapse loss ($\mathcal{L}_{AOCMC}$) during the optimization of the open-space cGAN parameters ($\mathcal{H}\theta, \mathcal{H}\phi$) in $\mathcal{L}_h$, which is as follows,

$$\mathcal{L}_{AOCMC} = \min_{G_{\mathcal{H}\theta}, D_{\mathcal{H}\phi}} 1 + \log \frac{1 - (\cos(z_l, z_h)) + \epsilon}{1 - (\cos(s_l, s_h)) + \epsilon} \qquad (2)$$

Ideally, the set of values for z_l is a subset of those of z_h, which means that there is a high likelihood of generating the same mode for s_h and s_l when (z_h, z_l) is sampled from the overlapping region of $\mathcal{N}(0, \sigma L) \cup \mathcal{N}(0, \sigma H)$, where $\cos(z_l, z_h) \approx 1$. To prevent this, $\mathcal{L}_{AOCMC}$ is used to regulate the generation of such features by adjusting $G_{\mathcal{H}\theta}$ such

that $\cos(s_l, s_h)$ tends to zero instead. A small constant ϵ is used to handle numerical instability. Alternatively, in the mode collapse case, $\cos(s_h, s_l) \to 1$ for $\cos(z_h, z_l) \to 1$ so the error remains large, and needs to be penalized.

Class and domain conditional prototype computation: To compute the known-class prototypes for the classes in $\mathcal{K}_\mathcal{S}$ and $\mathcal{K}_\mathcal{T}$, we use augmented support sets $\mathbb{S}_{f\mathcal{S}/f\mathcal{T}} \cup \mathbb{S}_{l\mathcal{S}/l\mathcal{T}}$ separately for both domains. Specifically, we compute the prototypes $\{\mathcal{P}'^k_{\mathcal{S}/\mathcal{T}}\}^{|\mathcal{K}_\mathcal{S}|/|\mathcal{K}_\mathcal{T}|}_{k=1}$ for each of the kth known classes, where $m^k_{\mathcal{S}/\mathcal{T}}$ is the number of support samples corresponding to the kth class for $(\mathcal{S}, \mathcal{T})$.

$$\mathcal{P}'^k_\mathcal{S} = \frac{1}{m^k_\mathcal{S}} \sum_{k=1}^{|\mathcal{K}_\mathcal{S}|} \mathbb{S}^k_{f\mathcal{S}} \cup \mathbb{S}^k_{l\mathcal{S}}; \ \mathcal{P}'^k_\mathcal{T} = \frac{1}{m^k_\mathcal{T}} \sum_{k=1}^{|\mathcal{K}_\mathcal{T}|} \mathbb{S}^k_{fT} \cup \mathbb{S}^k_{l\mathcal{T}}; \quad (3)$$

Global cross-domain prototype alignment loss: To globally align $\mathcal{S}$ and $\mathcal{T}$, we propose to align the style information of both domains by learning domain-specific batch-norm (DSBN) parameters $(\gamma_\mathcal{S}, \beta_\mathcal{S})$ and $(\gamma_\mathcal{T}, \beta_\mathcal{T})$, respectively. Furthermore, given the rescaled source and target domain prototypes based on the batch-norm parameters, we seek to minimize the global average distance between all $\mathcal{S}$ and $\mathcal{T}$ prototype embeddings, which is expressed as,

$$\mathcal{L}_{Align} = \frac{\sum_{k=1}^{|\mathcal{K}_\mathcal{S}|} (\gamma_\mathcal{S} \times \mathcal{P}'^k_\mathcal{S} + \beta_\mathcal{S})}{|\mathcal{K}_\mathcal{S}|} - \frac{\sum_{k=1}^{|\mathcal{K}_\mathcal{T}|} (\gamma_\mathcal{T} \times \mathcal{P}'^k_\mathcal{T} + \beta_\mathcal{T})}{|\mathcal{K}_\mathcal{T}|}; \quad (4)$$

Our objective is to achieve rapid domain alignment by aligning only the prototype vectors instead of the entire dataset. As the known-class compactness loss brings the query and support samples closer to the prototypes, aligning the prototypes implicitly aligns the domain distributions. In contrast, the original DSBN in [4] generates pseudo-labels from $\mathcal{T}$ and performs a two-stage approach for class-level domain alignment, which is suboptimal for DA-FSOS.

Known-class compactness loss: To maintain a concise and discriminative representation of each known class, minimizing the distance between the known support and query samples and the corresponding class prototypes is crucial. As described in Eq. 5, we enforce each $q \in \mathbb{Q}_{f\mathcal{S}}(\mathcal{K}_\mathcal{S}) \cup \mathbb{S}_{f\mathcal{S}} \cup \mathbb{S}_{l\mathcal{S}}$ to be in close proximity to the respective prototype in $\mathcal{P}_\mathcal{S}$, and similarly for the target domain $\mathcal{T}$. The compactness loss for both domains is denoted as follows,

$$\mathcal{L}_{C|(\mathcal{S}\cup\mathcal{T})} = \mathbb{E}_{q \in \mathbb{Q}_f(\mathcal{K}) \cup \mathbb{S}_f \cup \mathbb{S}_l} \left[-\log \frac{e^{-d(q, \mathcal{P}^l)}}{\sum_{k=1}^{\mathcal{K}} e^{-d(q, \mathcal{P}^k)}} \right] \quad (5)$$

d is the squared Euclidean distance and $\mathcal{P}^l$ is the class prototype for q. We ignore the domain labels in Eq. 5 .

Prototype diversification loss: In DAFOS-NET, we strive to achieve not only within-category compactness, as ensured by Eq. 5, but also a margin of separation between open and closed spaces and class-discriminative feature space for both domains, which can be challenging to maintain due to domain adaptation. To address this, we propose a margin-based metric objective that enforces a significant

distance between the prototypes in $\mathcal{P}_\mathcal{S}$ and the samples in $\mathbb{Q}_{f\mathcal{S}}(\mathcal{U}_\mathcal{S}) \cup \mathbb{S}_{h\mathcal{S}}$ from the same domain, as well as the cross-domain prototypes from $\mathcal{P}_\mathcal{T}$. This requirement applies vice versa for $\mathcal{T}$, resulting in highly discriminative feature space for both domains.

$$\mathcal{L}_{PD} = \sum_{k=1}^{|\mathcal{K}_\mathcal{S}|} [\|\mathcal{P}^k_\mathcal{S} - q_{Pos|\mathcal{S}}\|^2_2 - \|\mathcal{P}^k_\mathcal{S} - q_{Neg|\mathcal{S}}\|^2_2 + \alpha]_+$$
$$+ \sum_{k=1}^{|\mathcal{K}_\mathcal{T}|} [\|\mathcal{P}^k_\mathcal{T} - q_{Pos|\mathcal{T}}\|^2_2 - \|\mathcal{P}^k_\mathcal{T} - q_{Neg|\mathcal{T}}\|^2_2 + \alpha]_+; \quad (6)$$

Here, $q_{Pos|\mathcal{S}} \in \mathbb{Q}_{f\mathcal{S}}(\mathcal{K}_\mathcal{S})^k$ refers to the positive queries from the source domain for the kth class prototype anchor $\mathcal{P}^k_\mathcal{S}$, while its negative queries are denoted by $q_{Neg|\mathcal{S}} \in \mathbb{Q}_{f\mathcal{S}}(\mathcal{U}_\mathcal{S}) \cup \mathbb{S}_{h\mathcal{S}} \cup \mathcal{P}_\mathcal{T}$. Ideally, we seek to ensure that the difference between (anchor, +ve) should be less by a margin α than the difference between (anchor, -ve). Also, $[z]_+ = max(z, 0)$.

Outlier detection and domain prediction loss: We subsequently pass each query's distance $(\mathcal{Q}_{dist})$ from the prototypes to Outlier detector, $\mathcal{O}_\zeta$ and domain predictor $\mathcal{O}_\eta$. $\mathcal{O}_\zeta$ in Eq. 7 meta-learns to classify $\mathbb{Q}_{f\mathcal{S}}(\mathcal{K}_\mathcal{S}) \cup \mathbb{Q}_{f\mathcal{T}}(\mathcal{K}_\mathcal{T})$ queries as known and query samples from $\mathbb{Q}_{f\mathcal{S}}(\mathcal{U}_\mathcal{S}) \cup \mathbb{Q}_{f\mathcal{T}}(\mathcal{U}_\mathcal{T}) \cup \mathbb{S}_{h\mathcal{S}}(\mathcal{U}_\mathcal{S}) \cup \mathbb{S}_{h\mathcal{T}}(\mathcal{U}_\mathcal{T})$ as outlier.

$$\mathcal{L}_{OUT} = \mathbb{E}_{q \in \{\mathbb{Q}_{f\mathcal{S}} \cup \mathbb{Q}_{f\mathcal{T}} \cup \mathbb{S}_{h\mathcal{S}} \cup \mathbb{S}_{h\mathcal{T}}\}} \left[-\sum_{i=1}^2 t_i log(\mathcal{O}_\zeta(\mathcal{Q}_{dist})_i) \right] \quad (7)$$

Where, $t_i \in \{0, 1\}$ represents *known* or outlier label.

Similarly, $\mathcal{O}_\eta$ in Eq. 8 learns to classify $\mathbb{Q}_{f\mathcal{S}}(\mathcal{K}_\mathcal{S} \cup \mathcal{U}_\mathcal{S}) \cup \mathbb{S}_{h\mathcal{S}}(\mathcal{U}_\mathcal{S})$ queries as source and $\mathbb{Q}_{f\mathcal{T}}(\mathcal{K}_\mathcal{T} \cup \mathcal{U}_\mathcal{T}) \cup \mathbb{S}_{h\mathcal{T}}(\mathcal{U}_\mathcal{T})$ queries as target.

$$\mathcal{L}_{DC} = \mathbb{E}_{q \in \{\mathbb{Q}_{f\mathcal{S}} \cup \mathbb{S}_{h\mathcal{S}} \cup \mathbb{Q}_{f\mathcal{T}} \cup \mathbb{S}_{h\mathcal{T}}\}} \left[-\sum_{i=1}^2 c_i log(\mathcal{O}_\eta(\mathcal{Q}_{dist})_i) \right] \quad (8)$$

Where, $c_i \in \{0, 1\}$ indicates $\mathcal{S}$ or $\mathcal{T}$ domain label.

Total loss: To the end, we estimate the overall loss function to optimize f_φ in Eq. 9. We follow an alternate optimization strategy between the cGANs and f_φ.

$$\mathcal{L}_{FE} = \lambda_1 \cdot \mathcal{L}_C + \lambda_2 \cdot \mathcal{L}_{PD} + \lambda_3 \cdot \mathcal{L}_{Align} + \lambda_4 \cdot \mathcal{L}_{OUT} + \lambda_5 \cdot \mathcal{L}_{DC} \quad (9)$$

Where, the λs are weight factors of loss components.

Inference: In normal DA-FSOS, we utilize the outputs of $\mathcal{O}_\zeta$ to determine whether a sample is an inlier or outlier. For inlier samples, we estimate the class label based on their similarity with the prototype vectors calculated from $\mathcal{D}'_t$.

However, for generalized DA-FSOS, the query sample may come from any of the classes in $\mathcal{C}_s \cup \mathcal{C}_d \cup \mathcal{C}_t$. To handle this situation, we leverage both $\mathcal{O}_\zeta$ and $\mathcal{O}_\eta$ to determine both the domain and class labels for the test query. Subsequently, we choose the prototypes from the corresponding domain to classify the potential known-class queries.

4. Experimental evaluations

Datasets, preprocessing, and network details. We evaluated the performance of DAFOS-NET on three benchmark datasets: Office-Home [37], DomainNet [31], and mini-imageNet/CUB [38, 39]. Office-Home contains 65 classes and four domains. For our experiments, we selected the `Real-World` and `Clipart` domains, and used the following class split: $\mathcal{C}_s = 25$, $\mathcal{C}_d = 10$, $\mathcal{C}'_t(\text{known}) = 15$, $\mathcal{C}_t - \mathcal{C}'_t(\text{unknown}) = 15$. The DomainNet dataset contains 345 classes and six domains. We considered the following source-target combinations: `Real` to `Clipart`, `Real` to `Painting`, and `Clipart` to `Painting`. The class split we used was $\mathcal{C}_s = 125$, $\mathcal{C}_d = 75$, $\mathcal{C}'_t(\text{known}) = 80$, $\mathcal{C}_t - \mathcal{C}'_t(\text{unknown}) = 65$. For the mini-imageNet and CUB datasets [7], we used $\mathcal{C}_s = 100$, $\mathcal{C}_d = 50$, $\mathcal{C}'_t(\text{known}) = 25$, $\mathcal{C}_t - \mathcal{C}'_t(\text{unknown}) = 25$. Notably, CUB dataset has fine-grained separation among known / unknown classes. For all datasets, we resized the images to 84×84 pixels. We consider two image-level augmentations, i) *Weak* [2] (Flipping the images, rotation, resizing), and ii) *Strong* [8]: intensity transformation with a random magnitude in [0, 0.5].

In DAFOS-NET, we utilize ResNet-18 [14] for f_φ, and construct the generators $G_{\mathcal{H}\theta}$ and $G_{\mathcal{L}\theta}$ with four linear layers. Final batch norm layer in f_φ is replaced by the DSBN layer. For all classifiers and cGAN discriminators, we use four linear layers followed by the respective output layers. In total, DAFOS-NET contains 5.8M parameters, and training involves 0.08 GFLOPS computations. (Fig 3 (d)).

Training and evaluation protocols. During meta-training, we use the Adam optimizer [20] with a learning rate of 0.0001 and a batch size of 8. After cross validation, we set $\alpha = 0.5$ in Eq. 6 (see `Supplementary`). The weights in Eq. 9 are cross-validated. For evaluation, we use the widely-used 5-way 1-shot and 5-way 5-shot training protocols consistent with previous works in DA-FSL and FSOS [27, 41]. We present the average top-1 accuracy for 500 test episodes to record results over three independent runs.

To comprehensively evaluate DAFOS-NET performance, we utilize standard OSR metrics, which include closed-set classification accuracy (Acc) and Area Under ROC Curve (AUROC) [3]. Specifically, Acc measures the percentage of correctly classified known class samples, while AUROC evaluates model's ability to identify outliers.

4.1. Comparison with the state-of-the-art

In Table 1, we evaluate several state-of-the-art (SOTA) methodologies from various paradigms under the DA-FSOS settings and compare them against DAFOS-NET.

We observe that PrototypicalNet [33] and MetaoptNet [22], which were originally developed for FSL, perform poorly in outlier recognition, especially under domain shift. As a result, their Acc and AUROC values are considerably worse than DAFOS-NET. While OpenMax [1], an OSR method, shows better outlier detection ability in AUROC than its Acc value, it performs poorly overall because its Weibull models are not effective in approximating distribution density from limited samples.

FSOS methods, including PEELER [25], SnaTCHer [17], OCN [28], and MORGAN [27], are moderately effective in rejecting fine-grained outliers, but they lack transferable knowledge in $\mathcal{T}$. AdaMatch [2], a domain adaptation method, and other DAFSL methods, namely, DAPN [41] and NSAE [23], perform better in detecting known classes and generalize well over $\mathcal{T}$, but they fail to recognize fine-grained outliers in $\mathcal{T}$, for example in the CUB dataset consisting of different Birds categories.

Moreover, we incorporate *Domain Adaptive Loss* from DAPN [41] in optimizing MORGAN to adapt an FSOS framework over $\mathcal{T}$, and we observe better closed and open performance compared to their vanilla counterparts. We define this combined framework as MORGAN+DAPN. Nevertheless, overall, DAFOS-NET outperforms all other methodologies for both 1 and 5-shot settings due to its effective margin-aware outlier rejection and distribution alignment strategies. For instance, DAFOS-NET beats its next best alternative by 15.57% Acc and a significant margin of 12.42% AUROC for 1-shot evaluation over the Office-Home dataset. We observe similar dominant performance by DAFOS-NET for both 1 and 5-shot evaluations over Mini-imagenet to CUB and DomainNet datasets, reporting the new state-of-the-art. Finally, we evaluate the sensitivity of DAFOS-NET to the number of shots and the degree of openness and compare it against MORGAN+DAPN in Fig 3 (a, b). The results demonstrate that DAFOS-NET outperforms MORGAN+DAPN in handling a larger set of open classes, with a consistent improvement of 15% AUROC.

4.2. Generalized DA-FSOS comparison

We conducted a thorough comparison of our proposed DAFOS-NET against MORGAN, DAPN, combination of MORGAN+DAPN for the generalized DA-FSOS setting. This setting is evaluated over all the source and target domain classes (refer to Table 2). For all three datasets, DAFOS-NET outperforms MORGAN+DAPN by a significant margin. For instance, for DomainNet, DAFOS-NET has improvements of 14.69% in closed-set accuracy and 21.79% on the AUROC values over MORGAN+DAPN. Similar trends are observed for the other two datasets.

To assess the importance of the domain classifier $\mathcal{O}_\eta$ in classifying $\mathcal{S} \cup \mathcal{T}$ queries, we conducted further evaluations. We found that the model without $\mathcal{O}_\eta$ showed a significant drop in Acc and AUROC performances by almost 6%, averaging over both domains. This finding validates that the inclusion of $\mathcal{O}_\eta$ reduces the training bias of DAFOS-NET, while MORGAN+DAPN is much affected by overfitting.

Table 1. The 5-way ($\mathcal{K}=5$) 1 and 5-shot DA-FSOS performance comparison of the proposed DAFOS-NET and SOTA Methods.

| Model	Venue	Paradigm	Office-Home Real-World to Clipart		MiniImageNet to CUB		DomainNet Real to Clipart		Real to Painting		Clipart to Painting	
			Acc(%)	AUROC(%)	Acc(%)	AUROC(%)	Acc(%)	AUROC(%)	Acc(%)	AUROC(%)	Acc(%)	AUROC(%)
			1-shot Evaluation									
PrototypicalNet [33]	NIPS-17	FSL	25.17	19.23	31.44	25.12	28.15	23.02	29.81	23.51	27.59	22.18
Metaoptnet [22]	CVPR-19	FSL	33.71	26.62	42.46	34.22	35.16	27.19	38.63	26.02	37.21	29.56
OpenMax[1]	CVPR-16	OSR	12.19	14.77	21.88	24.02	14.68	16.25	15.71	16.38	15.66	17.01
PEELER [25]	CVPR-20	FSOS	14.55	16.18	25.47	27.82	18.81	20.28	21.06	22.67	22.64	23.04
SnaTCHer [17]	CVPR-21	FSOS	20.36	22.15	31.33	32.18	23.79	25.17	24.43	25.64	25.18	24.56
OCN [28]	WACV-22	FSOS	22.64	21.32	35.98	33.27	24.11	23.92	26.24	25.17	27.65	26.64
MORGAN [27]	WACV-23	FSOS	33.92	35.15	40.62	37.45	33.15	32.43	34.89	33.45	39.22	38.07
AdaMatch [2]	ICLR-22	DA	30.31	27.29	45.82	36.26	34.53	30.36	33.27	29.71	30.84	26.34
DAPN [41]	WACV-21	DA-FSL	31.86	25.28	47.82	38.76	38.44	29.73	41.65	33.21	43.49	35.27
NSAE [23]	ICCV-21	CDFSL	34.26	28.17	42.67	34.59	36.33	28.25	39.17	31.07	40.14	33.19
MORGAN + DAPN	-	-	36.22	38.12	41.12	40.46	36.58	34.41	38.75	36.19	41.77	39.45
DAFOS-NET [Ours]	-	DA-FSOS	**51.79±0.67**	**50.54±0.54**	**49.25±0.62**	**50.17±0.28**	**52.42±0.37**	**54.84±0.36**	**55.06±0.34**	**54.16±0.33**	**56.21±0.21**	**57.26±0.49**
			5-shot Evaluation									
PrototypicalNet [33]	NIPS-17	FSL	29.61	24.75	35.87	31.27	31.67	23.43	32.28	25.96	31.28	24.98
Metaoptnet [22]	CVPR-19	FSL	36.51	28.63	46.36	35.97	38.45	30.77	41.73	32.29	40.42	32.37
OpenMax[1]	CVPR-16	OSR	15.74	16.28	26.43	29.32	16.26	17.38	17.46	18.23	20.21	19.45
PEELER [25]	CVPR-20	FSOS	20.16	22.24	30.71	33.41	22.52	24.14	23.24	27.08	24.82	25.12
SnaTCHer [17]	CVPR-21	FSOS	23.12	25.12	34.28	35.13	26.13	27.16	26.71	28.16	28.41	26.72
OCN [28]	WACV-22	FSOS	24.28	23.98	37.42	34.21	25.86	25.62	27.87	26.23	31.35	29.89
MORGAN [27]	WACV-23	FSOS	35.31	37.43	42.04	39.82	36.25	34.44	38.32	36.44	42.05	39.86
AdaMatch [2]	ICLR-22	DA	36.67	30.48	46.15	39.19	37.91	31.21	35.37	31.34	33.57	28.38
DAPN [41]	WACV-21	DA-FSL	34.55	27.32	49.55	41.23	40.47	31.42	44.29	35.68	45.57	38.75
NSAE [23]	ICCV-21	CDFSL	37.84	29.65	46.33	36.67	39.11	29.32	42.32	34.42	43.14	35.23
MORGAN + DAPN	-	-	39.52	40.68	45.55	44.26	40.64	41.45	40.27	39.26	45.81	42.26
DAFOS-NET [Ours]	-	DA-FSOS	**54.44±0.29**	**57.72±0.18**	**58.51±0.32**	**59.73±0.43**	**55.09±0.64**	**60.67±0.48**	**59.03±0.25**	**56.08±0.67**	**59.66±0.28**	**60.55±0.31**
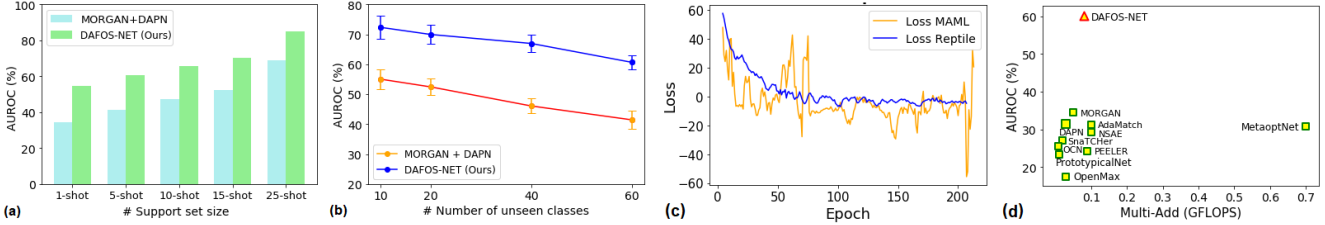
Figure 3. AUROC variation due to a) support size, b) open classes for 5-way evaluation on DomainNet (Real to clipart) (c) Loss comparison due to optimizing DAFOS-NET by Reptile [26], MAML [9] d) Trade-off between AUROC and GFLOPS by SOTA models.

Table 2. Generalized DA-FSOS comparison, where a query is sampled from $\mathcal{S} + \mathcal{T}$ in meta-testing ($\mathcal{K} = 5, m = 5$)

Dataset	Acc(%)	AUROC(%)	Acc(%)	AUROC(%)
	MORGAN		DAPN	
Office-Home (R → C)*	32.37±0.71	30.65±0.09	30.09±0.28	27.16±0.67
MiniImageNet to CUB	36.18±0.37	34.89±0.82	31.54±0.43	26.08±0.58
DomainNet (R → C)*	32.25±0.52	35.53±0.71	29.25±0.31	28.67±0.45
	MORGAN+DAPN		DAFOS-NET	
Office-Home (R → C)*	36.23±0.17	38.47±0.47	**49.24±0.59**	**52.25±0.83**
MiniImageNet to CUB	42.26±0.15	41.06±0.31	**52.79±0.62**	**52.35±0.35**
DomainNet (R → C)*	38.23±0.06	37.38±0.29	**52.92±0.46**	**59.17±0.18**

*R: RealWorld domain, C: Clipart domain

Table 3. Performance comparison due to loss components for 5-way 5-shot evaluation on DomainNet (Real to clipart).

Loss function	Acc(%)	AUROC(%)
$\mathcal{L}_C$	45.38±0.56	41.64±0.38
$\mathcal{L}_{PD}$	43.48±0.47	52.09±0.35
$\mathcal{L}_C + \mathcal{L}_{PD}$	47.76±0.71	56.27±0.59
$\mathcal{L}_C + \mathcal{L}_{PD} + \mathcal{L}_{Align}$	52.46±0.48	58.29±0.31
$\mathcal{L}_C + \mathcal{L}_{PD} + \mathcal{L}_{Align} + \mathcal{L}_{OUT}$	53.79±0.43	59.38±0.35
Without $\mathcal{L}_{AOCMC}$ regularizer	54.32±0.49	58.45±0.36
Intermediate DSBN layers	51.56±0.29	53.09±0.64
$\mathcal{L}_C + \mathcal{L}_{PD} + \mathcal{L}_{Align} + \mathcal{L}_{OUT} + \mathcal{L}_{DC} + \mathcal{L}_{AOCMC}$	**55.09±0.64**	**60.67±0.48**

4.3. Ablation analysis

i. Ablation on the loss components. We conducted experiments to assess the individual contributions of the loss components in DAFOS-NET and report their performance in Table 3. Optimizing solely with $\mathcal{L}_C$, resulted in poor open-set performance since the model failed to distinguish outliers from known samples. Conversely, optimizing solely with $\mathcal{L}_{PD}$, leading to suboptimal Acc as the queries were not heavily pulled towards their respective prototype. However, combining $\mathcal{L}_C$ and $\mathcal{L}_{PD}$ resulted in improved Acc and AUROC compared to their individual presence. $\mathcal{L}_{Align}$, significantly improved Acc by aligning $\mathcal{S}$ and $\mathcal{T}$, enabling the model to estimate $\mathcal{T}$ domain known samples better following the $\mathcal{S}$ domain compaction. Furthermore, the addition of $\mathcal{L}_{OUT}$, enhanced AUROC performance due to its

ability to distinguish pseudo-unknown queries. We have already discussed the importance of $\mathcal{L}_{DC}$ for generalized DA-FSOS. Finally, regularizing with $\mathcal{L}_{AOCMC}$ helped generate discriminative adversarial open samples, thereby boosting AUROC by almost 2%. Also, applying DSBN at the final layer of f_φ instead of all intermediate layers (at all the batch norm layer positions of ResNet-18) helps to boost performance due to reduced $\mathcal{S}$ and $\mathcal{T}$ domain alignment.

ii. Optimization stability of cGANs. DAFOS-NET leverages a dual-GAN approach to generate adversarial known and outlier samples concurrently. Second-order optimization based on MAML [9] for this dual-GAN results in oscillations and longer convergence time compared to first-order optimization based on Reptile [26], as demonstrated in Fig. 3 (c). The first-order optimization method promotes training stability while dealing with limited training samples.

Table 4. Performanace comparison due to different augmentations for 5-way 5-shot evaluation on DomainNet (`Real` to `clipArt`)

Augmentation type	Acc(%)	AUROC(%)
W/o augmentation	50.32±0.48	43.06±0.41
Image level augmentation	51.88±0.76	48.62±0.31
Image + Feature level augmentation	**55.09±0.64**	**60.67±0.48**

Table 5. Analysis of noise variance $(\sigma L, \sigma H)$ values on $G_{\mathcal{L}\theta}, G_{\mathcal{H}\theta}$ over DomainNet (`Real` to `clipArt`) [5-way 5-shot]

σL	σH	Acc(%)	AUROC(%)	KL divergence
0.4	0.8	56.05±0.16	57.38±0.29	0.57
0.5	0.8	56.11±0.51	56.43±0.51	0.59
0.7	0.8	57.02±0.48	54.15±0.23	0.58
0.3	0.9	**55.09±0.64**	**60.67±0.48**	**0.45**
0.1	1.0	53.37±0.68	62.61±0.17	0.31
0.4	1.2	58.68±0.24	55.54±0.34	0.58

iii. Importance of the proposed augmentations: In Table 4, we examine the impact of image space enhancement using weak [2] and strong augmentations [8] and find that it results in a notable 5.6% improvement in AUROC. Additionally, enhancing data density through dual-GAN generated adversarial closed-open features leads to a boost of 4.78% in Acc and 17.61% in AUROC compared to the model without any augmentation. Jointly maximizing image and feature space augmentation schemes results in a better approximation of the known class distribution, thereby minimizing the few-shot risk.

iv. Sensitivity to $(\sigma H, \sigma L)$: Table 5 presents the impact of varying the standard deviations of two noise vectors on the quality of samples generated by both the cGANs. To measure the similarity between adversarial known samples and their real counterparts, we report the Kullback-Leibler (KL) divergence. Samples with slightly higher KL divergence [15] produce greater variety and enhance the density of the known classes. In this regard, we observed three cases, which are also illustrated in the `Supplementary`: a) increasing σL and keeping σH constant resulted in higher Acc and KL divergence, as many discriminative adversarial known features were generated. However, AUROC decreased due to some limited pseudo-unknown queries being misclassified as known samples. b) decreasing σL and increasing σH reduced Acc and KL divergence, as adversarial known samples were generated with a distribution similar to that of real known samples. AUROC increased due to an enriched open space. c) extending σH beyond 1 with constant σL created a potential risk of mixing pseudo-unknown samples with other known class samples, resulting in false positives for Acc. The best performance was achieved by combining $\sigma L = 0.3$ and $\sigma H = 0.9$.

v. Comparison among domain adaptation losses: We compared three methods, *Domain Adaptive Loss* [41], vanilla *Batch Normalization* [16] and *Domain-Specific Batch Normalization (DSBN)* in [4] against our GCDPA loss. Distribution alignment through [41] enhances model robustness, but batch-norm [16] helps to converge faster. However, due to globally aligning domains through prototypes by *GCDPA* loss, DAFOS-NET converges faster

Table 6. Analysis of different domain adaptation losses on DomainNet (`Real` to `clipArt`) for 5-way 5-shot evaluation.

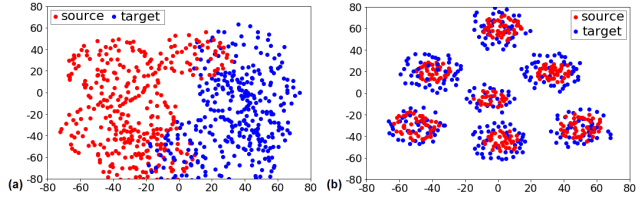| Adaptation Method	Acc(%)	AUROC(%)	Convergence
Domain Adaptive Loss [41]	49.37±0.23	50.28±0.35	450 epochs
Batch Normalization [16]	49.02±0.33	48.45±0.23	265 epochs
Domain-Specific Batch Norm [4]	48.18±0.36	46.76±0.51	288 epochs
Global Cross-Domain Prototype Alignment	**55.09±0.64**	**60.67±0.48**	**180 epochs**

Figure 4. t-SNE visualization a) without and b) with adaptation.

and generalizes better than DSBN-based variant [4], which aligns domains based on individual query samples. As per Table 6, the distribution alignment [41] resulted in a 5.72% lower Acc and 10% lower AUROC compared to our *GCDPA*-based adaptation. Time analysis also indicates that our approach achieves convergence quicker than [41, 4].

vi. t-SNE visualization of metric space. The t-SNE [35] visualization in Fig 4 illustrates the learning of DAFOS-NET with and without adaptation. For visual clarity, we presented seven categories from the `Real` and `Clipart` domains of DomainNet: Axe, Bus, Bucket, Clock, Flower, Foot, and Strawberry. The visualization reveals that the categories in $\mathcal{S}$ form compact clusters due to optimizing by $\mathcal{L}_C$. Furthermore, thanks to $\mathcal{L}_{Align}$, $\mathcal{T}$ categories maintain comparable semantic relationships while reducing domain shifts for each known class. For illustration simplicity, we did not evaluate the impact of $\mathcal{L}_{PD}$ in Fig 4 since we are considering the same set of classes from $\mathcal{S}$ and $\mathcal{T}$.

5. Takeaways

In this paper, we introduce a new problem setting for domain adaptive few-shot open-set learning and propose a novel architecture called DAFOS-NET to address it. Our approach ensures the modeling of a discriminative and domain-invariant embedding space while being able to reject outliers during testing. To achieve this, we propose a generative augmentation strategy for both the known classes and pseudo-open space for both domains, a domain-centric batch-norm parameter learning-based GCDPA loss, and a prototype diversification objective to ensure discriminativeness, in our meta-learning algorithm. We evaluate our proposed method on benchmark datasets from the domain adaptation literature in both the standard and generalized DA-FSOS settings. Our results demonstrate that DAFOS-NET significantly outperforms all competitors in both settings. We hope that our research can be applied to various domains such as edge AI, medical imaging, remote sensing, and other safety-critical areas that deal with domain shifts, low-shot learning, and open-set learning in a joint fashion.

References

[1] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572, 2016. 1, 6, 7

[2] David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alex Kurakin. Adamatch: A unified approach to semi-supervised learning and domain adaptation. *arXiv preprint arXiv:2106.04732*, 2021. 3, 6, 7, 8

[3] Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997. 6

[4] Woong-Gi Chang, Tackgeun You, Seonguk Seo, Suha Kwak, and Bohyung Han. Domain-specific batch normalization for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 7354–7362, 2019. 5, 8

[5] Xuewei Chao and Lixin Zhang. Few-shot imbalanced classification based on data augmentation. *Multimedia Systems*, pages 1–9, 2021. 2

[6] Da Chen, Yuefeng Chen, Yuhong Li, Feng Mao, Yuan He, and Hui Xue. Self-supervised learning for few-shot image classification. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1745–1749. IEEE, 2021. 2

[7] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *arXiv preprint arXiv:1904.04232*, 2019. 6

[8] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 702–703, 2020. 6, 8

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017. 2, 7

[10] Yuqian Fu, Yanwei Fu, and Yu-Gang Jiang. Meta-fdmixup: Cross-domain few-shot learning guided by labeled target data. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 5326–5334, 2021. 1, 3

[11] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015. 3

[12] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3614–3631, 2020. 1

[13] Yunhui Guo, Noel C Codella, Leonid Karlinsky, James V Codella, John R Smith, Kate Saenko, Tajana Rosing, and Rogerio Feris. A broader study of cross-domain few-shot learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 124–141. Springer, 2020. 3

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceed-ings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 6

[15] John R Hershey and Peder A Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, pages IV–317. IEEE, 2007. 8

[16] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015. 8

[17] Minki Jeong, Seokeon Choi, and Changick Kim. Few-shot open-set recognition by transformation consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12566–12575, 2021. 3, 6, 7

[18] Zilong Ji, Xiaolong Zou, Tiejun Huang, and Si Wu. Unsupervised few-shot learning via self-supervised training. *arXiv preprint arXiv:1912.12178*, 2019. 2

[19] Dahyun Kang, Heeseung Kwon, Juhong Min, and Minsu Cho. Relational embedding for few-shot classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8822–8833, 2021. 2

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. iclr. 2015. *arXiv preprint arXiv:1412.6980*, 9, 2015. 6

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017. 1

[22] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10657–10665, 2019. 1, 6, 7

[23] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9424–9434, 2021. 6, 7

[24] Weixin Liang, Zixuan Liu, and Can Liu. Dawson: A domain adaptive few shot generation framework. *arXiv preprint arXiv:2001.00576*, 2020. 4

[25] Bo Liu, Hao Kang, Haoxiang Li, Gang Hua, and Nuno Vasconcelos. Few-shot open-set recognition using meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2020. 1, 2, 3, 6, 7

[26] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018. 4, 7

[27] Debabrata Pal, Shirsha Bose, Biplab Banerjee, and Yogananda Jeppu. Morgan: Meta-learning-based few-shot open-set recognition via generative adversarial network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6295–6304, 2023. 2, 3, 6, 7

[28] Debabrata Pal, Valay Bundele, Renuka Sharma, Biplab Banerjee, and Yogananda Jeppu. Few-shot open-set recognition of hyperspectral images with outlier calibration network.

In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3801–3810, 2022. 3, 6, 7

[29] Pau Panareda Busto and Juergen Gall. Open set domain adaptation. In *Proceedings of the IEEE international conference on computer vision*, pages 754–763, 2017. 2

[30] Archit Parnami and Minwoo Lee. Learning from few examples: A summary of approaches to few-shot learning. *arXiv preprint arXiv:2203.04291*, 2022. 2

[31] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019. 2, 6

[32] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018. 2

[33] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017. 1, 2, 6, 7

[34] Hung-Yu Tseng, Hsin-Ying Lee, Jia-Bin Huang, and Ming-Hsuan Yang. Cross-domain few-shot classification via learned feature-wise transformation. *arXiv preprint arXiv:2001.08735*, 2020. 1, 3

[35] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 8

[36] Joaquin Vanschoren. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548*, 2018. 2

[37] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017. 2, 6

[38] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016. 2, 6

[39] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 2, 6

[40] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34, 2020. 1, 2

[41] An Zhao, Mingyu Ding, Zhiwu Lu, Tao Xiang, Yulei Niu, Jiechao Guan, and Ji-Rong Wen. Domain-adaptive few-shot learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1390–1399, 2021. 1, 3, 6, 7, 8

[42] Jing Zhou, Yanan Zheng, Jie Tang, Jian Li, and Zhilin Yang. Flipda: Effective and robust data augmentation for few-shot learning. *arXiv preprint arXiv:2108.06332*, 2021. 2