StyleAdv: Meta Style Adversarial Training for Cross-Domain Few-Shot Learning

Yuqian Fu[1], Yu Xie[2], Yanwei Fu[3], Yu-Gang Jiang[1*]

[1]Shanghai Key Lab of Intelligent Information Processing, School of Computer Science, Fudan University
[2]Purple Mountain Laboratories, Nanjing, China. [3]School of Data Science, Fudan University
{fuyq20, yxie18, yanweifu, ygj}@fudan.edu.cn

Abstract

Cross-Domain Few-Shot Learning (CD-FSL) is a recently emerging task that tackles few-shot learning across different domains. It aims at transferring prior knowledge learned on the source dataset to novel target datasets. The CD-FSL task is especially challenged by the huge domain gap between different datasets. Critically, such a domain gap actually comes from the changes of visual styles, and wave-SAN [10] empirically shows that spanning the style distribution of the source data helps alleviate this issue. However, wave-SAN simply swaps styles of two images. Such a vanilla operation makes the generated styles "real" and "easy", which still fall into the original set of the source styles. Thus, inspired by vanilla adversarial learning, a novel model-agnostic meta Style Adversarial training (StyleAdv) method together with a novel style adversarial attack method is proposed for CD-FSL. Particularly, our style attack method synthesizes both "virtual" and "hard" adversarial styles for model training. This is achieved by perturbing the original style with the signed style gradients. By continually attacking styles and forcing the model to recognize these challenging adversarial styles, our model is gradually robust to the visual styles, thus boosting the generalization ability for novel target datasets. Besides the typical CNN-based backbone, we also employ our StyleAdv method on large-scale pretrained vision transformer. Extensive experiments conducted on eight various target datasets show the effectiveness of our method. Whether built upon ResNet or ViT, we achieve the new state of the art for CD-FSL. Code is available at https://github.com/lovelyqian/StyleAdv-CDFSL.

1. Introduction

This paper studies the task of Cross-Domain Few-Shot Learning (CD-FSL) which addresses the Few-Shot Learning (FSL) problem across different domains. As a general recipe for FSL, *episode*-based meta-learning strategy has also been adopted for training CD-FSL models, e.g., FWT [48], LRP [42], ATA [51], and wave-SAN [10]. Generally, to mimic the low-sample regime in testing stage, meta learning samples episodes for training the model. Each episode contains a small labeled *support set* and an unlabeled *query set*. Models learn meta knowledge by predicting the categories of images contained in the query set according to the support set. The learned meta knowledge generalizes the models to novel target classes directly.

Empirically, we find that the changes of visual appearances between source and target data is one of the key causes that leads to the domain gap in CD-FSL. Interestingly, wave-SAN [10], our former work, shows that the domain gap issue can be alleviated by augmenting the visual styles of source images. Particularly, wave-SAN proposes to augment the styles, in the form of Adaptive Instance Normalization (AdaIN) [22], by randomly sampling two source episodes and exchanging their styles. However, despite the efficacy of wave-SAN, such a naïve style generation method suffers from two limitations: 1) The swap operation makes the styles always be limited in the *"real"* style set of the source dataset; 2) The limited real styles further lead to the generated styles too *"easy"* to learn. Therefore, a natural question is whether *we can synthesize "virtual" and "hard" styles for learning a more robust CD-FSL model?* Formally, we use "real/virtual" to indicate whether the styles are originally presented in the set of source styles, and define "easy/hard" as whether the new styles make meta tasks more difficult.

To that end, we draw inspiration from the adversarial training, and propose a novel meta **Style Adv**ersarial training method (**StyleAdv**) for CD-FSL. StyleAdv plays the minimax game in two iterative optimization loops of meta-training. Particularly, *the inner loop* generates adversarial styles from the original source styles by adding perturbations. The synthesized adversarial styles are supposed to be more challenging for the current model to recognize, thus, increasing the loss. Whilst *the outer loop* optimizes the whole network by minimizing the losses of recognizing the images with both original and adversarial styles. Our ultimate goal is to enable learning a model that is robust to various styles, be-

24575

* indicates corresponding author

yond the relatively limited and simple styles from the source data. This can potentially improve the generalization ability on novel target domains with visual appearance shifts.

Formally, we introduce a novel style adversarial attack method to support the inner loop of StyleAdv. Inspired yet different from the previous attack methods [14, 34], our style attack method perturbs and synthesizes the styles rather than image pixels or features. Technically, we first extract the style from the input feature map, and include the extracted style in the forward computation chain to obtain its gradient for each training step. After that, we synthesize the new style by adding a certain ratio of gradient to the original style. Styles synthesized by our style adversarial attack method have the good properties of "hard" and "virtual". Particularly, since we perturb styles in the opposite direction of the training gradients, our generation leads to the "hard" styles. Our attack method results in totally "virtual" styles that are quite different from the original source styles.

Critically, our style attack method makes *progressive style synthesizing*, with *changing style perturbation ratios*, which makes it significantly different from vanilla adversarial attacking methods. Specifically, we propose a novel progressive style synthesizing strategy. The naïve solution of directly plugging-in perturbations is to attack each block of the feature embedding module individually, which however, may results in large deviations of features from the high-level block. Thus, our strategy is to make the synthesizing signal of the current block be accumulated by adversarial styles from previous blocks. On the other hand, rather than attacking the models by fixing the attacking ratio, we synthesize new styles by randomly sampling the perturbation ratio from a candidate pool. This facilitates the diversity of the synthesized adversarial styles. Experimental results have demonstrated the efficacy of our method: 1) our style adversarial attack method does synthesize more challenging styles, thus, pushing the limits of the source visual distribution; 2) our StyleAdv significantly improves the base model and outperforms all other CD-FSL competitors.

We highlight our StyleAdv is model-agnostic and complementary to other existing FSL or CD-FSL models, e.g., GNN [12] and FWT [48]. More importantly, to benefit from the large-scale pretrained models, e.g., DINO [2], we further explore adapting our StyleAdv to improve the Vision Transformer (ViT) [5] backbone in a non-parametric way. Experimentally, we show that StyleAdv not only improves CNN-based FSL/CD-FSL methods, but also improves the large-scale pretrained ViT model.

Finally, we summarize our contributions. 1) A novel meta style adversarial training method, termed StyleAdv, is proposed for CD-FSL. By first perturbing the original styles and then forcing the model to learn from such adversarial styles, StyleAdv improves the robustness of CD-FSL models. 2) We present a novel style attack method with the novel progressive synthesizing strategy in changing attacking ratios. Diverse "virtual" and "hard" styles thus are generated. 3) Our method is complementary to existing FSL and CD-FSL methods; and we validate our idea on both CNN-based and ViT-based backbones. 4) Extensive results on eight unseen target datasets indicate that our StyleAdv outperforms previous CD-FSL methods, building a new SOTA result.

2. Related Work

Cross-Domain Few-Shot Learning. FSL which aims at freeing the model from reliance on massive labeled data has been studied for many years [12, 25, 39, 41, 43, 45, 46, 56, 58]. Particularly, some recent works, e.g., CLIP [38], CoOp [65], CLIP-Adapter [11], Tip-Adapter [59], and PMF [19] explore promoting the FSL with large-scale pretrained models. Particularly, PMF contributes a simple pipeline and builds a SOTA for FSL. As an extended task from FSL, CD-FSL [1, 7–10, 15, 16, 23, 29, 32, 37, 42, 48, 51, 60, 67] mainly solves the FSL across different domains. Typical meta-learning based CD-FSL methods include FWT [48], LRP [42], ATA [51], AFA [20], and wave-SAN [10]. Specifically, FWT and LRP tackle CD-FSL by refining batch normalization layers and using the explanation model to guide training. ATA, AFA, and wave-SAN propose to augment the image pixels, features, and visual styles, respectively. Several transfer-learning based CD-FSL methods, e.g., BSCD-FSL (also known as Fine-tune) [16], BSR [33], and NSAE [32] have also been explored. These methods reveal that finetuning helps improving the performances on target datasets. Other works that introduce extra data or require multiple domain datasets for training include STARTUP [37], Meta-FDMixup [8], Me-D2N [9], TGDM [67], TriAE [15], and DSL [21].

Adversarial Attack. The adversarial attack aims at misleading models by adding some bespoke perturbations to input data. To generate the perturbations effectively, lots of adversarial attack methods have been proposed [6, 14, 26, 27, 34, 36, 55, 57]. Most of the works [6, 14, 34, 36] attack the image pixels. Specifically, FGSM [14] and PGD [34] are two most classical and famous attack algorithms. Several works [26, 27, 62] attack the feature space. Critically, few methods [57] attack styles. Different from these works that aim to mislead the models, we perturb the styles to tackle the visual shift issue for CD-FSL.

Adversarial Few-Shot Learning. Several attempts [13, 28, 30, 40, 52] that explore adversarial learning for FSL have been made. Among them, MDAT [30], AQ [13], and MetaAdv [52] first attack the input image and then train the model using the attacked images to improve the defense ability against adversarial samples. Shen et al. [40] attacks the feature of the episode to improve the generalization capability of FSL models. Note that ATA [51] and AFA [20], two CD-FSL methods, also adopt the adversarial learning.

However, we are greatly different from them. ATA and AFA perturb image pixels or features, while we aim at bridging the visual gap by generating diverse hard styles.

Style Augmentation for Domain Shift Problem. Augmenting the style distribution for narrowing the domain shift issue has been explored in domain generation [31, 54, 66], image segmentation [3, 63], person re-ID [61], and CD-FSL [10]. Concretely, MixStyle [66], AdvStyle [63], DSU [31], and wave-SAN [10] synthesize styles without extra parameters via mixing, attacking, sampling from a Gaussian distribution, and swapping. MaxStyle [3] and L2D [54] require additional network modules and complex auxiliary tasks to help generate the new styles. Typically, AdvStyle [63] is the most related work to us. Thus, we highlight the key differences: 1) AdvStyle attacks styles on the image, while we attack styles on multiple feature spaces with a progressive attacking method; 2) AdvStyle uses the same task loss (segmentation) for attacking and optimization; in contrast, we use the classical classification loss to attack the styles, while utilize the task loss (FSL) to optimize the whole network.

3. StyleAdv: Meta Style Adversarial Training

Task Formulation. Episode $\mathcal{T} = ((S, Q), Y)$ is randomly sampled as the input of each meta-task, where Y represents the global class labels of the episode images with respect to $\mathcal{C}^{tr}$. Typically, each meta-task is formulated as an N-way K-shot problem. That is, for each episode $\mathcal{T}$, N classes with K labeled images are sampled as the support set S, and the same N classes with another M images are used to constitute the query set Q. The FSL or CD-FSL models predict the probability P that the images in Q belong to N categories according to S. Formally, we have $|S| = NK$, $|Q| = NM$, $|P| = NM \times N$.

FGSM and PGD Attackers. We briefly summarize the algorithms for FGSM [14] and PGD [34], two most famous attacking methods. Given image x with label y, FGSM attacks the x by adding a ratio ϵ of signed gradients with respect to the x resulting in the adversarial image x^{adv} as,

$$x^{adv} = x + \epsilon \cdot sign(\nabla_x J(\theta, x, y)), \qquad (1)$$

where $J(\cdot)$ and θ denote the object function and the learnable parameters of a classification model. PGD can be regarded as a variant of FGSM. Different from the FGSM that only attacks once, PGD attacks the image in an iterative way and sets a random start (abbreviated as RT) for x as,

$$x_0^{adv} = x + k_{RT} \cdot \mathcal{N}(0, I), \qquad (2)$$

$$x_t^{adv} = x_{t-1}^{adv} + \epsilon \cdot sign(\nabla_x J(\theta, x, y)), \qquad (3)$$

where k_{RT}, ϵ are hyper-parameters. $\mathcal{N}$ is Gaussian noises.

3.1. Overview of Meta Style Adversarial Learning

To alleviate the performance degradation caused by the changing visual appearance, we tackle CD-FSL by promoting the robustness of models on recognizing various styles. Thus, we expose our FSL model to some challenging virtual styles beyond the image styles that existed in the source dataset. To that end, we present the novel StyleAdv adversarial training method. Critically, rather than adding perturbations to image pixels, we particularly focus on adversarially perturbing the styles. The overall framework of our StyleAdv is illustrated in Figure 1. Our StyleAdv contains a CNN/ViT backbone E, a global FC classifier f_{cls}, and a FSL classifier f_{fsl} with learnable parameters $\theta_E, \theta_{cls}, \theta_{fsl}$, respectively. Besides, our core style attack method, a novel style extraction module, and the AdaIN are also included.

Overall, we learn the StyleAdv by solving a minimax game. Specifically, the minimax game shall involve two iterative optimization loops in each meta-train step. Particularly,

- **Inner loop:** synthesizing new adversarial styles by attacking the original source styles; the generated styles will increase the loss of the current network.

- **Outer loop:** optimizing the whole network by classifying source images with both original and adversarial styles; this process will decrease the loss.

3.2. Style Extraction from CNNs and ViTs

Adaptive Instance Normalization (AdaIN). We recap the vanilla AdaIN [22] proposed for CNN in style transfer. Particularly, AdaIN reveals that the instance-level mean and standard deviation (abbreviated as *mean* and *std*) convey the style information of the input image. Denoting the *mean* and *std* as μ and σ, AdaIN (denoted as $\mathcal{A}$) reveals that the style of F can be transfered to that of F_{tgt} by replacing the original style (μ, σ) with the target style $(\mu_{tgt}, \sigma_{tgt})$:

$$\mathcal{A}(F, \mu_{tgt}, \sigma_{tgt}) = \sigma_{tgt} \frac{F - \mu(F)}{\sigma(F)} + \mu_{tgt}. \qquad (4)$$

Style Extraction for CNN Features. As shown in the upper part of Figure 1 (b), let $F \in \mathcal{R}^{B \times C \times H \times W}$ indicates the input feature batch, where B, C, H, and W denote the batch size, channel, height, and width of the feature F, respectively. As in AdaIN, the *mean* μ and *std* σ of F are defined as:

$$\mu(\text{F})_{b,c} = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} F_{b,c,h,w}, \qquad (5)$$

$$\sigma(\text{F})_{b,c} = \sqrt{\frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} (F_{b,c,h,w} - \mu_{b,c}(F))^2 + \epsilon}, \quad (6)$$

where $\mu, \sigma \in \mathcal{R}^{B \times C}$.

24577

(a) Overall Framework of Our StyleAdv Method **(b) Style Extraction For CNN/ViT Features**
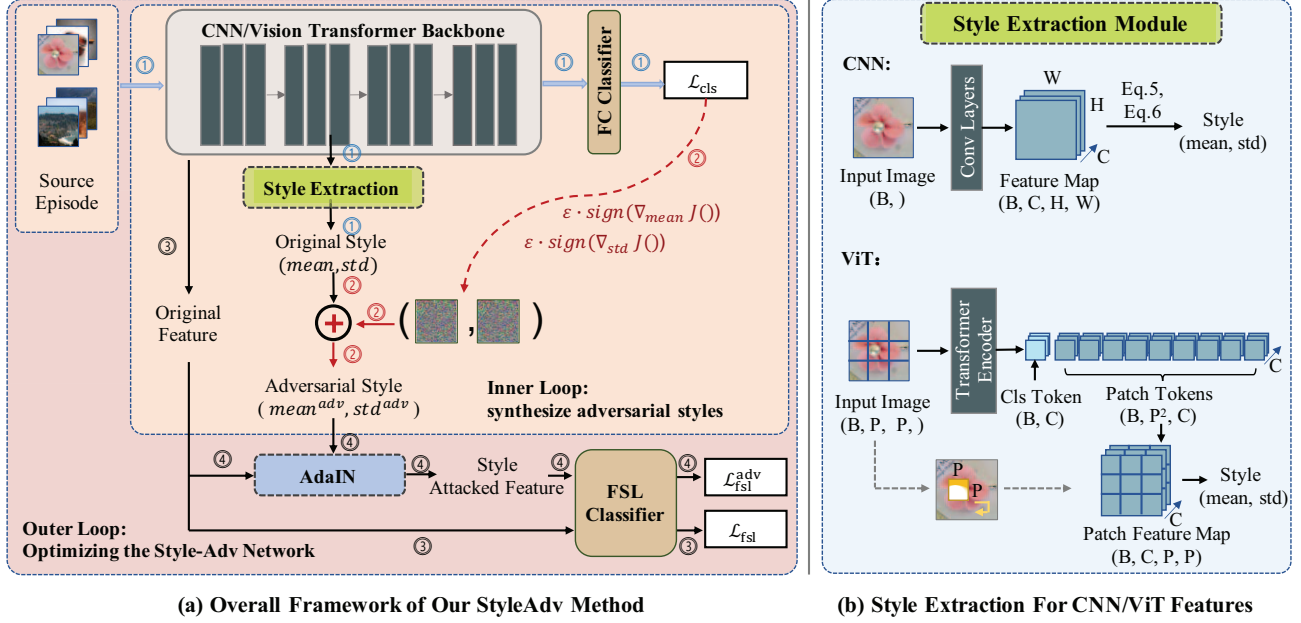
Figure 1. (a): Overview of StyleAdv method. The inner loop synthesizes adversarial styles, while the outer loop optimizes the whole network. (b): Style extraction for CNN-based and ViT-based features (illustration with B=1).

Meta Information Extraction for ViT Features. We explore extracting the meta information of the ViT features as the manner of CNN. Intuitively, such meta information can be regarded as a unique "style" of ViTs. As shown in Figure 1 (b), we take an input batch data with image split into $P \times P$ patches as an example. The ViT encoder will encode the batch patches into a class (cls) token ($F_{cls} \in \mathcal{R}^{B \times C}$) and a patch tokens ($F_0 \in \mathcal{R}^{B \times P^2 \times C}$). To be compatible with AdaIN, we reshape the F_0 as $F \in \mathcal{R}^{B \times C \times P \times P}$. At this point, we can calculate the meta information for patch tokens F as in Eq. 5 and Eq. 6. Essentially, note that the transformer integrates the positional embedding into the patch representation, the spatial relations thus could be considered still hold in the patch tokens. This supports us to reform the patch tokens F_0 as a spatial feature map F. To some extent, this can also be achieved by applying the convolution on the input data via a kernel of size $P \times P$ (as indicated by dashed arrows in Figure 1 (b)).

3.3. Inner Loop: Style Adversarial Attack Method

We propose a novel style adversarial attack method – Fast Style Gradient Sign Method (Style-FGSM) to accomplish the inner loop. As shown in Figure 1, given an input source episode $(\mathcal{T}, Y)$, we first forward it into the backbone E and the FC classifier f_{cls} producing the global classification loss $\mathcal{L}_{cls}$ (as illustrated in the ① paths). During this process, a key step is to make the gradient of the style available. To achieve that, let $F_{\mathcal{T}}$ denotes the features of $\mathcal{T}$, we obtain the style (μ, σ) of $F_{\mathcal{T}}$ as in Sec. 3.2. After that, we reform the original episode feature as $\mathcal{A}(F_{\mathcal{T}}, \mu, \sigma)$. And the reformed

feature is actually used for the forward propagation. In this way, we include μ and σ in our forward computation chain; and thus, we could access the gradients of them.

With the gradients in ② paths, we then attack μ and σ as FGSM does – adding a small ratio ϵ of the signed gradients with respect to μ and σ, respectively.

$$\mu^{adv} = \mu + \epsilon \cdot \text{sign}(\nabla_\mu J(\theta_E, \theta_{f_{cls}}, \mathcal{A}(F_{\mathcal{T}}, \mu, \sigma), Y)), \quad (7)$$

$$\sigma^{adv} = \sigma + \epsilon \cdot \text{sign}(\nabla_\sigma J(\theta_E, \theta_{f_{cls}}, \mathcal{A}(F_{\mathcal{T}}, \mu, \sigma), Y)), \quad (8)$$

where the $J()$ is the cross-entropy loss between classification predictions and ground truth Y, i.e., $\mathcal{L}_{cls}$. Inspired by the random start of PGD, we also add random noises $k_{RT} \cdot \mathcal{N}(0, I)$ to μ and σ before attacking. $\mathcal{N}(0, I)$ refers to Gaussian noises and k_{RT} is a hyper-parameter. Our Style-FGSM enables us to generate both "virtual" and "hard" styles.

Progressive Style Synthesizing Strategy: To prevent the high-level adversarial feature from deviating, we propose to apply our style-FGSM in a progressive strategy. Concretely, the embedding module E has three blocks $E_1, E_2,$ and E_3, with the corresponding features $F_1, F_2,$ and F_3. For the first block, we use (μ_1, σ_1) to denote the original styles of F_1. The adversarial styles ($\mu_1^{adv}, \sigma_1^{adv}$) are obtained directly as in Eq. 7 and Eq. 8. For subsequent blocks, the attack signals on the current block i are those accumulated from the block 1 to block $i - 1$. Take the second block as an example, the block feature F_2 is not simply extracted by $E_2(F_1)$. Instead, we have $F_2' = E_2(F_1^{adv})$, where $F_1^{adv} = \mathcal{A}(F_1, \mu_1^{adv}, \sigma_1^{adv})$. Attacking on F_2' results in the adversarial styles ($\mu_2^{adv}, \sigma_2^{adv}$). Accordingly, we gen-

24578

erate $(\mu_3^{adv}, \sigma_3^{adv})$ for the last block. The illustration of the progressive attacking strategy is attached in the Appendix. **Changing Style Perturbation Ratios:** Different from the vanilla FGSM [14] or PGD [34], our style attacking algorithm is expected to synthesize new styles with diversity. Thus, instead of using a fixed attacking ratio ϵ, we randomly sample ϵ from a candidate list ϵ_{list} as the current attacking ratio. Despite the randomness of ϵ, we still synthesize styles in a more challenging direction, ϵ only affects the extent.

3.4. Outer Loop: Optimize the StyleAdv Network

For each meta-train iteration with clean episode $\mathcal{T}$ as input, our inner loop produces adversarial styles $(\mu_1^{adv}, \sigma_1^{adv})$, $(\mu_2^{adv}, \sigma_2^{adv})$, and $(\mu_3^{adv}, \sigma_3^{adv})$. As in Figure 1, the goal of the outer loop is to optimize the whole StyleAdv with both the clean feature F and the style attacked feature F^{adv} utilized as the training data. Typically, the clean episode feature F can be obtained directly as $E(\mathcal{T})$ as in ③ paths.

In ④ paths, we obtain the F^{adv} by transferring the original style of F to the corresponding adversarial attacked styles. Similar with the progressive style-FGSM, we have $F_1^{adv} = \mathcal{A}(E_1(\mathcal{T}), \mu_1^{adv}, \sigma_1^{adv})$, $F_2^{adv} = \mathcal{A}(E_2(F_1^{adv}), \mu_2^{adv}, \sigma_2^{adv})$, and $F_3^{adv} = \mathcal{A}(E_3(F_2^{adv}), \mu_3^{adv}, \sigma_3^{adv})$. Finally, F^{adv} is obtained by applying an average pooling layer to F_3^{adv}. A skip probability p_{skip} is set to decide whether to skip the current attacking. Conducting FSL tasks for both the clean feature F and style attacked feature F^{adv} results in two FSL predictions P_{fsl}, P_{fsl}^{adv}, and two FSL classification losses $\mathcal{L}_{fsl}$, $\mathcal{L}_{fsl}^{adv}$.

Further, despite the styles of F^{adv} shifts from that of F, we encourage that the semantic content should be still consistent as in wave-SAN [10]. Thus we add a consistent constraint to the predictions of P_{fsl} and P_{fsl}^{adv} resulting in the consistent loss $\mathcal{L}_{cons}$ as,

$$\mathcal{L}_{cons} = \text{KL}(P_{fsl}, P_{fsl}^{adv}), \tag{9}$$

where $\text{KL}()$ is Kullback–Leibler divergence loss. In addition, we have the global classification loss $\mathcal{L}_{cls}$. This ensures that θ_{cls} is optimized to provide correct gradients for style-FGSM. The final meta-objective of StyleAdv is as,

$$\mathcal{L} = \mathcal{L}_{fsl} + \mathcal{L}_{fsl}^{adv} + \mathcal{L}_{cons} + \mathcal{L}_{cls}. \tag{10}$$

Note that our StyleAdv is model-agnostic and orthogonal to existing FSL and CD-FSL methods.

3.5. Network Inference

Applying StyleAdv Directly for Inference. Our StyleAdv facilitates making CD-FSL model more robust to style shifts. Once the model is meta-trained, we can employ it for inference directly by feeding the testing episode into the E and the f_{cls}. The class with the highest probability will be taken as the predicted result.

Finetuning StyleAdv Using Target Examples. As indicated in previous works [16, 32, 33, 51], finetuning CD-FSL models on target examples helps improve the model performance. Thus, to further promote the performance of StyleAdv, we also equip it with the fintuning strategy forming an upgraded version ("StyleAdv-FT"). Specifically, as in ATA-FT [51], for each novel testing episode, we augment the novel support set to form pseudo episodes as training data for tuning the meta-trained model.

4. Experiments

Datasets. We take two CD-FSL benchmarks proposed in BSCD-FSL [16] and FWT [48]. Both of them take mini-Imagenet [39] as the source dataset. Two disjoint sets split from mini-Imagenet form $\mathcal{D}^{tr}$ and $\mathcal{D}^{eval}$. Totally eight datasets including ChestX [53], ISIC [4,47], EuroSAT [18], CropDisease [35], CUB [50], Cars [24], Places [64], and Plantae [49] are taken as novel target datasets. The former four datasets included in BSCD-FSL's benchmark cover medical images varying from X-ray to dermoscopic skin lesions, and natural images from satellite pictures to plant disease photos. While the latter four datasets that focus on more fine-grained concepts such as birds and cars are contained in FWT. These eight target datasets serve as testing set $\mathcal{D}^{te}$, respectively.

Network Modules. For typical CNN based network, following previous CD-FSL methods [10, 42, 48, 51], ResNet-10 [17] is selected as the embedding module while GNN [12] is selected as the FSL classifier; For the emerging ViT based network, following PMF [19], we use the ViT-small [5] and the ProtoNet [41] as the embedding module and the FSL classifier, respectively. Note that, the ViT-small is pretrained on ImageNet1K by DINO [2] as in PMF. The f_{cls} is built by a fully connected layer.

Implementation Details. The 5-way 1-shot and 5-way 5-shot settings are conducted. Taking ResNet10 as backbone, we meta train the network for 200 epochs, each epoch contains 120 meta tasks. Adam with a learning rate of 0.001 is utilized as the optimizer. Taking ViT-small as backbone, the meta train stage takes 20 epoch, each epoch contains 2000 meta tasks. The SGD with a initial learning rate of 5e-5 and 0.001 are used for optimize the $E()$ and the f_{cls}, respectively. The ϵ_{list}, k_{RT} of Style-FGSM attacker are set as $[0.8, 0.08, 0.008]$, $\frac{16}{255}$. The probability p_{skip} of random skipping the attacking is chosen from $\{0.2, 0.4\}$. We evaluate our network with 1000 randomly sampled episodes and report average accuracy (%) with a 95% confidence interval. Both the results of our "StyleAdv" and "StyleAdv-FT" are reported. The details of the finetuning are attached in Appendix. ResNet-10 based models are trained and tested on a single GeForce GTX 1080, while ViT-small based models require a single NVIDIA GeForce RTX 3090.

24579

Table columns: 1-shot | Backbone | FT | LargeP | ChestX | ISIC | EuroSAT | CropDisease | CUB | Cars | Places | Plantae | Average

1-shot	Backbone	FT	LargeP	ChestX	ISIC	EuroSAT	CropDisease	CUB	Cars	Places	Plantae	Average
GNN [12]	RN10	-	-	$22.00_{\pm0.46}$	$32.02_{\pm0.66}$	$63.69_{\pm1.03}$	$64.48_{\pm1.08}$	$45.69_{\pm0.68}$	$31.79_{\pm0.51}$	$53.10_{\pm0.80}$	$35.60_{\pm0.56}$	43.55
FWT [48]	RN10	-	-	$22.04_{\pm0.44}$	$31.58_{\pm0.67}$	$62.36_{\pm1.05}$	$66.36_{\pm1.04}$	$47.47_{\pm0.75}$	$31.61_{\pm0.53}$	$55.77_{\pm0.79}$	$35.95_{\pm0.58}$	44.14
LRP [42]	RN10	-	-	$22.11_{\pm0.20}$	$30.94_{\pm0.30}$	$54.99_{\pm0.50}$	$59.23_{\pm0.50}$	$48.29_{\pm0.51}$	$32.78_{\pm0.39}$	$54.83_{\pm0.56}$	$37.49_{\pm0.43}$	42.58
ATA [51]	RN10	-	-	$22.10_{\pm0.20}$	$33.21_{\pm0.40}$	$61.35_{\pm0.50}$	$67.47_{\pm0.50}$	$45.00_{\pm0.50}$	$33.61_{\pm0.40}$	$53.57_{\pm0.50}$	$34.42_{\pm0.40}$	43.84
AFA [20]	RN10	-	-	$22.92_{\pm0.20}$	$33.21_{\pm0.30}$	$63.12_{\pm0.50}$	$67.61_{\pm0.50}$	$46.86_{\pm0.50}$	$34.25_{\pm0.40}$	$54.04_{\pm0.60}$	$36.76_{\pm0.40}$	44.85
wave-SAN [10]	RN10	-	-	$\mathbf{22.93}_{\pm0.49}$	$33.35_{\pm0.71}$	$69.64_{\pm1.09}$	$70.80_{\pm1.06}$	$\mathbf{50.25}_{\pm0.74}$	$33.55_{\pm0.61}$	$57.75_{\pm0.82}$	$40.71_{\pm0.66}$	47.37
StyleAdv (ours)	RN10	-	-	$22.64_{\pm0.35}$	$\mathbf{33.96}_{\pm0.57}$	$\mathbf{70.94}_{\pm0.82}$	$\mathbf{74.13}_{\pm0.78}$	$48.49_{\pm0.72}$	$\mathbf{34.64}_{\pm0.57}$	$\mathbf{58.58}_{\pm0.83}$	$\mathbf{41.13}_{\pm0.67}$	48.06
ATA-FT [51]	RN10	Y	-	$22.15_{\pm0.20}$	$34.94_{\pm0.40}$	$68.62_{\pm0.50}$	$75.41_{\pm0.50}$	$46.23_{\pm0.50}$	$\mathbf{37.15}_{\pm0.40}$	$54.18_{\pm0.50}$	$37.38_{\pm0.40}$	47.01
StyleAdv-FT (ours)	RN10	Y	-	$\mathbf{22.64}_{\pm0.35}$	$\mathbf{35.76}_{\pm0.52}$	$\mathbf{72.92}_{\pm0.75}$	$\mathbf{80.69}_{\pm0.28}$	$\mathbf{48.49}_{\pm0.72}$	$35.09_{\pm0.55}$	$\mathbf{58.58}_{\pm0.83}$	$\mathbf{41.13}_{\pm0.67}$	49.41
PMF* [19]	ViT-small	Y	DINO/IN1K	$21.73_{\pm0.30}$	$30.36_{\pm0.36}$	$70.74_{\pm0.63}$	$80.79_{\pm0.62}$	$78.13_{\pm0.66}$	$37.24_{\pm0.57}$	$71.11_{\pm0.71}$	$53.60_{\pm0.66}$	55.46
StyleAdv (ours)	ViT-small	-	DINO/IN1K	$\mathbf{22.92}_{\pm0.32}$	$\mathbf{33.05}_{\pm0.44}$	$72.15_{\pm0.65}$	$\mathbf{81.22}_{\pm0.61}$	$\mathbf{84.01}_{\pm0.58}$	$\mathbf{40.48}_{\pm0.57}$	$\mathbf{72.64}_{\pm0.67}$	$\mathbf{55.52}_{\pm0.66}$	57.75
StyleAdv-FT (ours)	ViT-small	Y	DINO/IN1K	$\mathbf{22.92}_{\pm0.32}$	$\mathbf{33.99}_{\pm0.46}$	$\mathbf{74.93}_{\pm0.58}$	$\mathbf{84.11}_{\pm0.57}$	$\mathbf{84.01}_{\pm0.58}$	$\mathbf{40.48}_{\pm0.57}$	$\mathbf{72.64}_{\pm0.67}$	$\mathbf{55.52}_{\pm0.66}$	58.57

5-shot	Backbone	FT	LargeP	ChestX	ISIC	EuroSAT	CropDisease	CUB	Cars	Places	Plantae	Average
GNN [12]	RN10	-	-	$25.27_{\pm0.46}$	$43.94_{\pm0.67}$	$83.64_{\pm0.77}$	$87.96_{\pm0.67}$	$62.25_{\pm0.65}$	$44.28_{\pm0.63}$	$70.84_{\pm0.65}$	$52.53_{\pm0.59}$	58.84
FWT [48]	RN10	-	-	$25.18_{\pm0.45}$	$43.17_{\pm0.70}$	$83.01_{\pm0.79}$	$87.11_{\pm0.67}$	$66.98_{\pm0.68}$	$44.90_{\pm0.64}$	$73.94_{\pm0.67}$	$53.85_{\pm0.62}$	59.77
LRP [42]	RN10	-	-	$24.53_{\pm0.30}$	$44.14_{\pm0.40}$	$77.14_{\pm0.40}$	$86.15_{\pm0.40}$	$64.44_{\pm0.48}$	$46.20_{\pm0.46}$	$74.45_{\pm0.47}$	$54.46_{\pm0.46}$	58.94
ATA [51]	RN10	-	-	$24.32_{\pm0.40}$	$44.91_{\pm0.40}$	$83.75_{\pm0.40}$	$90.59_{\pm0.30}$	$66.22_{\pm0.50}$	$49.14_{\pm0.40}$	$75.48_{\pm0.40}$	$52.69_{\pm0.40}$	60.89
AFA [20]	RN10	-	-	$25.02_{\pm0.50}$	$\mathbf{46.01}_{\pm0.40}$	$85.58_{\pm0.40}$	$88.06_{\pm0.30}$	$68.25_{\pm0.50}$	$49.28_{\pm0.50}$	$76.21_{\pm0.50}$	$54.26_{\pm0.40}$	61.58
wave-SAN [10]	RN10	-	-	$25.63_{\pm0.49}$	$44.93_{\pm0.67}$	$85.22_{\pm0.71}$	$89.70_{\pm0.64}$	$\mathbf{70.31}_{\pm0.67}$	$46.11_{\pm0.66}$	$76.88_{\pm0.63}$	$57.72_{\pm0.64}$	62.06
StyleAdv (ours)	RN10	-	-	$\mathbf{26.07}_{\pm0.37}$	$45.77_{\pm0.51}$	$\mathbf{86.58}_{\pm0.54}$	$\mathbf{93.65}_{\pm0.39}$	$68.72_{\pm0.67}$	$\mathbf{50.13}_{\pm0.68}$	$\mathbf{77.73}_{\pm0.62}$	$\mathbf{61.52}_{\pm0.68}$	63.77
Fine-tune [16]	RN10	Y	-	$25.97_{\pm0.41}$	$48.11_{\pm0.64}$	$79.08_{\pm0.61}$	$89.25_{\pm0.51}$	$64.14_{\pm0.77}$	$52.08_{\pm0.74}$	$70.06_{\pm0.74}$	$59.27_{\pm0.70}$	61.00
ATA-FT [51]	RN10	Y	-	$25.08_{\pm0.20}$	$49.79_{\pm0.40}$	$89.64_{\pm0.30}$	$95.44_{\pm0.20}$	$69.83_{\pm0.50}$	$54.28_{\pm0.50}$	$76.64_{\pm0.40}$	$58.08_{\pm0.40}$	64.85
NSAE [32]	RN10	Y	-	$\mathbf{27.10}_{\pm0.44}$	$54.05_{\pm0.63}$	$83.96_{\pm0.57}$	$93.14_{\pm0.47}$	$68.51_{\pm0.76}$	$54.91_{\pm0.74}$	$71.02_{\pm0.72}$	$59.55_{\pm0.74}$	64.03
BSR [33]	RN10	Y	-	$26.84_{\pm0.44}$	$\mathbf{54.42}_{\pm0.66}$	$80.89_{\pm0.61}$	$92.17_{\pm0.45}$	$69.38_{\pm0.76}$	$\mathbf{57.49}_{\pm0.72}$	$71.09_{\pm0.68}$	$61.07_{\pm0.76}$	64.17
StyleAdv-FT (ours)	RN10	Y	-	$26.24_{\pm0.35}$	$53.05_{\pm0.54}$	$\mathbf{91.64}_{\pm0.43}$	$\mathbf{96.51}_{\pm0.28}$	$\mathbf{70.90}_{\pm0.63}$	$56.44_{\pm0.68}$	$\mathbf{79.35}_{\pm0.61}$	$\mathbf{64.10}_{\pm0.64}$	67.28
PMF [19]	ViT-small	Y	DINO/IN1K	$\mathbf{27.27}$	50.12	85.98	92.96	-	-	-	-	-
StyleAdv (ours)	ViT-small	-	DINO/IN1K	$26.97_{\pm0.33}$	$47.73_{\pm0.44}$	$\mathbf{88.57}_{\pm0.34}$	$94.85_{\pm0.31}$	$\mathbf{95.82}_{\pm0.27}$	$61.73_{\pm0.62}$	$\mathbf{88.33}_{\pm0.40}$	$75.55_{\pm0.54}$	72.44
StyleAdv-FT (ours)	ViT-small	Y	DINO/IN1K	$26.97_{\pm0.33}$	$\mathbf{51.23}_{\pm0.51}$	$90.12_{\pm0.33}$	$\mathbf{95.99}_{\pm0.27}$	$\mathbf{95.82}_{\pm0.27}$	$\mathbf{66.02}_{\pm0.64}$	$\mathbf{88.33}_{\pm0.40}$	$78.01_{\pm0.54}$	74.06

Table 1. Results of 5-way 1-shot/5-shot tasks. "FT" means whether the finetuning stage is employed. "LargeP" represents if large pretrained models are used for model initialization. "RN10" is short for "ResNet-10". ∗ denotes results are reported by us. Results perform best are bolded. Whether based on ResNet-10 or ViT-small, our method outperforms other competitors significantly.

4.1. Comparison with the SOTAs

We compare our StyleAdv/StyleAdv-FT against several most representative and competitive CD-FSL methods. Concretely, with the ResNet-10 (abbreviated as RN10) as backbone, totally nine methods including GNN [12], FWT [48], LRP [42], ATA [51], AFA [20], wave-SAN [10], Fine-tune [16], NSAE [32], and BSR [33] are introduced as our competitors. Among them, the former six competitors are meta-learning based method that used for inference directly, thus we compare our "StyleAdv" against them for a fair comparison. Typically, the GNN [12] works as a base model. The Fine-tune [16], NSAE [32], BSR [33], and ATA-FT [51] (formed by finetuning ATA) all require finetuning model during inference, thus our "StyleAdv-FT" is used. With the ViT as backbone, the most recent and competitive PMF (SOTA method for FSL) is compared. For fair comparisons, we follow the same pipeline proposed in PMF [19]. Note that we promote CD-FSL models with only one single source domain. Those methods that use extra training datasets, e.g., STARTUP [37], meta-FDMixup [8], and DSL [21] are not considered. The comparison results are given in Table 1.

For all results, our method outperforms all the listed CD-FSL competitors significantly and builds a new state of the art. Our StyleAdv-FT (ViT-small) on average achieves 58.57% and 74.06% on 5-way 1-shot and 5-shot, respectively. Our StyleAdv (RN10) and StyleAdv-FT (RN10) also

beats all the meta-learning based or transfer-learning (finetuning) based methods. Besides of the state-of-the-art accuracy, we also have other worth-mentioning observations. 1) We show that our StyleAdv method is a general solution for both CNN-based models and ViT-based models. Typically, based on ResNet10, our StyleAdv and StyleAdv-FT improve the base GNN by up to 4.93% and 8.44% on 5-shot setting. Based on ViT-small, at most cases, our StyleAdv-FT outperforms the PMF by a clear margin. More results of building StyleAdv upon other FSL or CD-FSL methods can be found in the Appendix. 2) Comparing FWT, LRP, ATA, AFA, waveSAN, and our StyleAdv, we find that StyleAdv performs best, followed by wave-SAN, then comes the AFA, ATA, FWT, and LRP. This phenomenon indicates that tackling CD-FSL by solving the visual shift problem is indeed more effective than other perspectives, e.g., adversarial training by perturbing the image features (AFA) or image pixels (ATA), transforms the normalization layers in FWT, and explanation guided training in LRP. 3) For the comparison between StyleAdv and wave-SAN that both tackles the visual styles, we notice that StyleAdv outperforms the wave-SAN in most cases. This demonstrates that the styles generated by our StyleAdv are more conducive to learning robust CD-FSL models than the style augmentation method proposed in wave-SAN. This justifies our idea of synthesizing more challenging ("hard and virtual") styles.

4) Overall, the large-scale pretrained model promotes the CD-FSL obviously. Take 1-shot as an example, StyleAdv-FT (ViT-small) boosts the StyleAdv-FT (RN10) by 9.16% on average. However, we show that the performance improvement varies greatly on different target domains. Generally, for target datasets with relative small domain gap, e.g., CUB and Plantae, models benefit a lot; otherwise, the improvement is limited. 5) We also find that under the cross-domain scenarios, finetuning model on target domain, e.g., NSAE, BSR do show an advantange over purely meta-learning based methods, e.g., FWT, LRP, and wave-SAN. However, to finetune model using extremely few examples, e.g., 5-way 1-shot is much harder than on relatively larger shots. This may explain why those finetune-based methods do not conduct experiments on 1-shot setting.

Effectiveness of Style-FGSM Attacker. To show the advantages of our progressive style synthesizing strategy and attacking with changing perturbation ratios, we compare our Style-FGSM against several variants and report the results in Figure 2. Specifically, for Figure 2 (a), we compare our style-FGSM against the variant that attacks the blocks individually. Results show that attacking in a progressive way exceeds the naive individual strategy in most cases. For Figure 2 (b), to demonstrate how the performance will be affected by fixed attacking ratios, we also conduct experiments with different ϵ_{list}. Since we set the ϵ_{list} as $[0.8, 0.08, 0.008]$, three different choices including $[0.8]$, $[0.08]$, and $[0.008]$ are selected. From the results, we first notice that the best result can be reached by a single fixed ratio. However, sampling the attacking ratio from a pool of candidates achieves the best result in most cases.
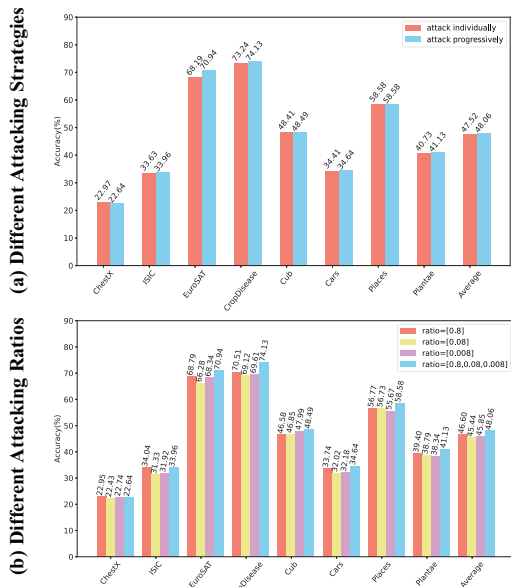

Figure 2. Effectiveness of the progressive style synthesizing strategy and the changing style perturbation ratios. The 5-way 1-shot results are reported. Models are built on ResNet10 and GNN.
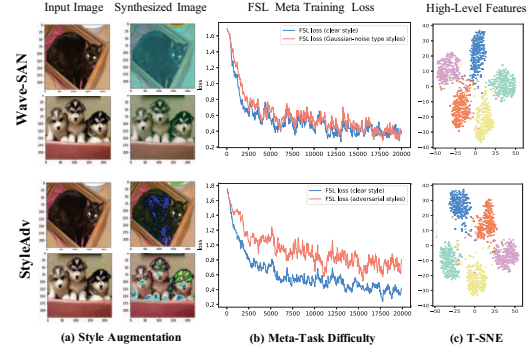

Figure 3. **Visualization of wave-SAN and StyleAdv.** (a): synthesized images; (b): meta-training losses; (c): T-SNE results.

4.2. More Analysis

Visualization of Hard Style Generation. To help understand the "hard" style generation of our method intuitively, as in Figure 3, we make several visualizations comparing StyleAdv against the wave-SAN. 1) As in Figure 3 (a), we show the stylized images generated by wave-SAN and our StyleAdv. The visualization is achieved by applying the style augmentation methods to input images. Specifically, for wave-SAN, the style is swapped with another randomly sampled source image; for StyleAdv, the results of attacking style with $\epsilon = 0.08$ are given. We observe that wave-SAN tends to exchange the global visual appearance, e.g., the color of the input image randomly. By contrast, StyleAdv prefers to disturb the important regions that are key to recognizing the image category. For example, the fur of the cat and the key parts (face and feet) of the dogs. These observations intuitively support our claim that our StyleAdv synthesize more harder styles than wave-SAN. 2) To quantitatively evaluate whether our StyleAdv introduces more challenging styles into the training stage, as in Figure 3 (b), we visualize the meta-training loss. Results reveal that the perturbed losses of wave-SAN oscillate around the original loss, while StyleAdv increases the original loss obviously. These phenomenons further validate that we perturb data towards a more difficult direction thus pushing the limits of style generation to a large extent. 3) To further show the advantages of StyleAdv over wave-SAN, as shown in Figure 3 (c), we visualize the high-level features extracted by the meta-trained wave-SAN and StyleAdv. Five classes (denoted by different colors) of mini-Imagenet are selected. T-SNE is used for reducing the feature dimensions. Results demonstrate that StyleAdv enlarges the inter-class distances making classes more distinguishable.

Why Attack Styles Instead of Images or Features? A natural question may be why we choose to attack styles instead of other targets, e.g., the input image as in AQ [13], MDAT [30], and ATA [51] or the features as in Shen et al. [40] and AFA [20]? To answer this question, we compare our StyleAdv which attacks styles against attacking images

	Attack Target	ChestX	ISIC	EuroSAT	CropDisease	CUB	Cars	Places	Plantae	Average
1-shot	Image	**22.71**±0.35	33.00±0.53	67.00±0.82	72.65±0.75	48.15±0.72	34.40±0.60	57.89±0.83	39.85±0.64	46.96
	Feature	22.55±0.35	32.95±0.53	68.71±0.81	70.86±0.78	46.52±0.70	34.07±0.54	56.68±0.81	39.62±0.62	46.50
	Style (ours)	22.64±0.35	**33.96**±0.57	**70.94**±0.82	**74.13**±0.78	**48.49**±0.72	**34.64**±0.57	**58.58**±0.83	**41.13**±0.67	**48.06**
5-shot	Image	24.92±0.36	42.63±0.47	84.18±0.54	90.31±0.47	66.37±0.65	47.46±0.67	75.94±0.62	57.33±0.65	61.14
	Feature	25.55±0.37	43.71±0.50	84.22±0.55	91.71±0.44	67.31±0.67	**50.26**±0.67	76.46±0.65	57.39±0.63	62.08
	Style (ours)	**26.07**±0.37	**45.77**±0.51	**86.58**±0.54	**93.65**±0.39	**68.72**±0.67	50.13±0.68	**77.73**±0.62	**61.52**±0.68	**63.77**

Table 2. Comparison results (%) of attacking image, feature, and styles. Models build upon ResNet10 and GNN classifier.

	Augment Method	ChestX	ISIC	EuroSAT	CropDisease	CUB	Cars	Places	Plantae	Average
1-shot	StyleGaus†	22.37±0.35	31.48±0.52	65.71±0.82	69.25±0.80	46.32±0.72	32.69±0.54	55.48±0.79	37.27±0.61	45.07
	MixStyle [66]	22.43±0.35	33.21±0.53	67.35±0.80	68.80±0.82	47.08±0.73	33.39±0.58	56.12±0.78	38.03±0.62	45.80
	AdvStyle [63]	22.04±0.36	30.83±0.52	65.19±0.82	64.96±0.81	47.43±0.72	31.90±0.52	53.95±0.79	35.81±0.59	44.01
	DSU [31]	22.35±0.36	31.43±0.51	64.55±0.83	64.73±0.81	47.74±0.72	31.61±0.53	54.81±0.81	37.19±0.61	44.30
	Style-FGSM (ours)	**22.64**±0.35	**33.96**±0.57	**70.94**±0.82	**74.13**±0.78	**48.49**±0.72	**34.64**±0.57	**58.58**±0.83	**41.13**±0.67	**48.06**
5-shot	StyleGaus†	24.97±0.37	41.74±0.48	81.88±0.61	89.71±0.49	65.98±0.67	45.03±0.64	72.66±0.68	56.66±0.65	59.83
	MixStyle [66]	25.04±0.36	43.77±0.53	82.67±0.58	88.90±0.52	65.73±0.66	45.91±0.63	75.90±0.63	56.59±0.62	60.56
	AdvStyle [63]	25.03±0.35	43.15±0.50	83.09±0.57	88.44±0.52	66.42±0.67	44.85±0.64	74.14±0.65	54.89±0.64	60.00
	DSU [31]	25.02±0.36	45.19±0.52	80.30±0.63	86.30±0.56	67.94±0.66	45.65±0.63	75.17±0.64	54.31±0.62	59.99
	Style-FGSM (ours)	**26.07**±0.37	**45.77**±0.51	**86.58**±0.54	**93.65**±0.39	**68.72**±0.67	**50.13**±0.68	**77.73**±0.62	**61.52**±0.68	**63.77**

Table 3. Different style augmentation methods are compared. "StyleGaus†" means adding random Gaussian noises to the styles, where † represents it is proposed by us. "MixStyle [66]", "AdvStyle [63]" and "DSU [31]" are adapted from other tasks, e.g., domain generation. Results (%) conducted under 5-way 1-shot/5-shot settings. Methods are built upon the ResNet10 and GNN.

and features by modifying the attack targets of our method. The 5-way 1-shot/5-shot results are given in Table 2. We highlight several points. 1) We notice that attacking image, feature, and style all improve the base GNN model (given in Table 1) which shows that all of them boost the generalization ability of the model by adversarial attacks. Interestingly, the results of our "Attack Image"/"Attack Feature" even outperform the well-designed CD-FSL methods ATA [51] and AFA [20] (shown in Table 1); 2) Our method has clear advantages over attacking images and features. This again indicates the superiority of tackling visual styles for narrowing the domain gap issue for CD-FSL.

Is Style-FGSM Better than Other Style Augmentation Methods? To show the advantages of our Style-FGSM against other style augmentation methods, we introduce several competitors including "StyleGaus", MixStyle [66], AdvStyle [63], and DSU [31]. Typically, "StyleGaus" that adds random Gaussian noises into the styles is introduced as a simple but reasonable baseline. MixStyle [66], AdvStyle [63], and DSU [31] which are initially designed for other tasks, e.g., segmentation and domain generation are also adapted. The results are reported in Table 3. Comparing the results of StyleGuas with that reported in Table 1, we find that perturbing the styles on the feature level by simply adding random noises also improves the base GNN and even surpasses a few CD-FSL competitors on some target datasets. This phenomenon is consistent with the insight that augmenting the style distributions helps boost the CD-FSL methods. As for the comparison between our Style-FGSM and other advanced style augmentation competitors, we find that Style-FGSM performs better than all the MixStyle, AdvStyle, and

DSU on both 1-shot and 5-shot settings. Typically, MixStyle and DSU both generate virtual styles, but their new styles are still relatively easy. This shows that our hard styles boost the model to a larger extent. AdvStyle generates both virtual and hard (adversarial) styles. However, it is still inferior to us. This indicates the advantages of our method that attacks in latent feature space and adopts two individual tasks for attacking and optimization.

5. Conclusion

This paper presents a novel model-agnostic StyleAdv for CD-FSL. Critically, to narrow the domain gap which is typically in the form of visual shifts, StyleAdv solves the minimax game of style adversarial learning: first adds perturbations to the source styles increasing the loss of the current model, then optimizes the model by forcing it to recognize both the clean and style perturbed data. Besides, a novel progressive style adversarial attack method termed style-FGSM is presented by us. Style-FGSM synthesizes diverse "hard" and "virtual" styles via adding the signed gradients to original clean styles. These generated styles support the max step of StyleAdv. Intuitively, by exposing the CD-FSL to adversarial styles which are more challenging than those limited real styles that exist in the source dataset, the generalization ability of the model is boosted. Our StyleAdv improves both CNN-based and ViT-based models. Extensive experiments indicate that our StyleAdv build new SOTAs.

References

[1] John Cai, Bill Cai, and Shen Sheng Mei. Damsl: Domain agnostic meta score-based learning. In *CVPR*, 2021. 2

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 2, 5

[3] Chen Chen, Zeju Li, Cheng Ouyang, Matt Sinclair, Wenjia Bai, and Daniel Rueckert. Maxstyle: Adversarial style composition for robust medical image segmentation. *arXiv preprint*, 2022. 3

[4] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint*, 2019. 5

[5] 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. *arXiv preprint*, 2020. 2, 5

[6] Ranjie Duan, Xingjun Ma, Yisen Wang, James Bailey, A Kai Qin, and Yun Yang. Adversarial camouflage: Hiding physical-world attacks with natural styles. In *CVPR*, 2020. 2

[7] Yuqian Fu, Yanwei Fu, Jingjing Chen, and Yu-Gang Jiang. Generalized meta-fdmixup: Cross-domain few-shot learning guided by labeled target data. *TIP*, 2022. 2

[8] Yuqian Fu, Yanwei Fu, and Yu-Gang Jiang. Meta-fdmixup: Cross-domain few-shot learning guided by labeled target data. In *ACM Multimedia*, 2021. 2, 6

[9] Yuqian Fu, Yu Xie, Yanwei Fu, Jingjing Chen, and Yu-Gang Jiang. Me-d2n: Multi-expert domain decompositional network for cross-domain few-shot learning. In *ACM Multimedia*, 2022. 2

[10] Yuqian Fu, Yu Xie, Yanwei Fu, Jingjing Chen, and Yu-Gang Jiang. Wave-san: Wavelet based style augmentation network for cross-domain few-shot learning. *arXiv preprint*, 2022. 1, 2, 3, 5, 6, 12

[11] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint*, 2021. 2

[12] Victor Garcia and Joan Bruna. Few-shot learning with graph neural networks. *arXiv preprint*, 2017. 2, 5, 6, 12, 13

[13] Micah Goldblum, Liam Fowl, and Tom Goldstein. Adversarially robust few-shot learning: A meta-learning approach. *NeurIPS*, 2020. 2, 7

[14] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint*, 2014. 2, 3, 5

[15] Jiechao Guan, Manli Zhang, and Zhiwu Lu. Large-scale cross-domain few-shot learning. In *ACCV*, 2020. 2

[16] 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 *ECCV*, 2020. 2, 5, 6

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

[18] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, 2019. 5

[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*, 2022. 2, 5, 6, 12, 13

[20] Yanxu Hu and Andy J Ma. Adversarial feature augmentation for cross-domain few-shot classification. In *ECCV*, 2022. 2, 6, 7, 8

[21] Zhengdong Hu, Yifan Sun, and Yi Yang. Switch to generalize: Domain-switch learning for cross-domain few-shot classification. In *ICLR*, 2021. 2, 6

[22] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, 2017. 1, 3

[23] Ashraful Islam, Chun-Fu Richard Chen, Rameswar Panda, Leonid Karlinsky, Rogerio Feris, and Richard Radke. Dynamic distillation network for cross-domain few-shot recognition with unlabeled data. *NeurIPS*, 2021. 2

[24] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV Workshop*, 2013. 5

[25] Manoj Kumar, Varun Kumar, Hadrien Glaude, Cyprien de Lichy, Aman Alok, and Rahul Gupta. Protoda: Efficient transfer learning for few-shot intent classification. In *SLT Workshop*, 2021. 2

[26] Cassidy Laidlaw and Soheil Feizi. Functional adversarial attacks. *NeurIPS*, 2019. 2

[27] Bo Li and Yevgeniy Vorobeychik. Feature cross-substitution in adversarial classification. *NeurIPS*, 2014. 2

[28] Kai Li, Yulun Zhang, Kunpeng Li, and Yun Fu. Adversarial feature hallucination networks for few-shot learning. In *CVPR*, 2020. 2

[29] Pan Li, Shaogang Gong, Yanwei Fu, and Chengjie Wang. Ranking distance calibration for cross-domain few-shot learning. *arXiv preprint*, 2021. 2

[30] Wenbin Li, Lei Wang, Xingxing Zhang, Jing Huo, Yang Gao, and Jiebo Luo. Defensive few-shot adversarial learning. *arXiv preprint*, 2019. 2, 7

[31] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and Ling-Yu Duan. Uncertainty modeling for out-of-distribution generalization. *arXiv preprint*, 2022. 3, 8

[32] 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*, 2021. 2, 5, 6

[33] Bingyu Liu, Zhen Zhao, Zhenpeng Li, Jianan Jiang, Yuhong Guo, and Jieping Ye. Feature transformation ensemble model with batch spectral regularization for cross-domain few-shot classification. *arXiv preprint*, 2020. 2, 5, 6

24583

[34] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint*, 2017. 2, 3, 5

[35] Sharada P Mohanty, David P Hughes, and Marcel Salathé. Using deep learning for image-based plant disease detection. *Frontiers in plant science*, 2016. 5

[36] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *CVPR*, 2016. 2

[37] Cheng Perng Phoo and Bharath Hariharan. Self-training for few-shot transfer across extreme task differences. *arXiv preprint*, 2020. 2, 6

[38] 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*, 2021. 2

[39] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017. 2, 5

[40] Wei Shen, Ziqiang Shi, and Jun Sun. Learning from adversarial features for few-shot classification. *arXiv preprint*, 2019. 2, 7

[41] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017. 2, 5

[42] Jiamei Sun, Sebastian Lapuschkin, Wojciech Samek, Yunqing Zhao, Ngai-Man Cheung, and Alexander Binder. Explanation-guided training for cross-domain few-shot classification. *arXiv preprint*, 2020. 1, 2, 5, 6

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

[44] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, 2018. 12, 13

[45] Hao Tang, Zechao Li, Zhimao Peng, and Jinhui Tang. Block-mix: meta regularization and self-calibrated inference for metric-based meta-learning. In *ACM Multimedia*, 2020. 2

[46] Hao Tang, Chengcheng Yuan, Zechao Li, and Jinhui Tang. Learning attention-guided pyramidal features for few-shot fine-grained recognition. *PR*. 2

[47] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 2018. 5

[48] Hung-Yu Tseng, Hsin-Ying Lee, Jia-Bin Huang, and Ming-Hsuan Yang. Cross-domain few-shot classification via learned feature-wise transformation. In *ICLR*, 2020. 1, 2, 5, 6, 12, 13

[49] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *CVPR*, 2018. 5

[50] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 5

[51] Haoqing Wang and Zhi-Hong Deng. Cross-domain few-shot classification via adversarial task augmentation. *arXiv preprint*, 2021. 1, 2, 5, 6, 7, 8

[52] Ren Wang, Kaidi Xu, Sijia Liu, Pin-Yu Chen, Tsui-Wei Weng, Chuang Gan, and Meng Wang. On fast adversarial robustness adaptation in model-agnostic meta-learning. *arXiv preprint*, 2021. 2

[53] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *CVPR*, 2017. 5

[54] Zijian Wang, Yadan Luo, Ruihong Qiu, Zi Huang, and Mahsa Baktashmotlagh. Learning to diversify for single domain generalization. In *ICCV*, 2021. 3

[55] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *CVPR*, 2020. 2

[56] Chengming Xu, Yanwei Fu, Chen Liu, Chengjie Wang, Jilin Li, Feiyue Huang, Li Zhang, and Xiangyang Xue. Learning dynamic alignment via meta-filter for few-shot learning. In *CVPR*, 2021. 2

[57] Qiuling Xu, Guanhong Tao, Siyuan Cheng, and Xiangyu Zhang. Towards feature space adversarial attack by style perturbation. In *AAAI*, 2021. 2

[58] Ji Zhang, Jingkuan Song, Lianli Gao, Ye Liu, and Heng Tao Shen. Progressive meta-learning with curriculum. *TCSVT*, 2022. 2

[59] Renrui Zhang, Rongyao Fang, Peng Gao, Wei Zhang, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint*, 2021. 2

[60] Hao ZHENG, Runqi Wang, Jianzhuang Liu, and Asako Kanezaki. Cross-level distillation and feature denoising for cross-domain few-shot classification. In *ICLR*. 2

[61] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *CVPR*, 2019. 3

[62] Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *IJCV*, 2021. 2

[63] Zhun Zhong, Yuyang Zhao, Gim Hee Lee, and Nicu Sebe. Adversarial style augmentation for domain generalized urban-scene segmentation. *arXiv preprint*, 2022. 3, 8

[64] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2017. 5

[65] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022. 2

[66] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *ICLR*, 2021. 3, 8, 12

[67] Linhai Zhuo, Yuqian Fu, Jingjing Chen, Yixin Cao, and Yu-Gang Jiang. Tgdm: Target guided dynamic mixup for cross-domain few-shot learning. In *ACM Multimedia*, 2022. 2