On the Importance of Distractors for Few-Shot Classification (Supplementary)

Rajshekhar Das[1] Yu-Xiong Wang[2] José M.F. Moura[1]
[1]Carnegie Mellon University [2]University of Illinois at Urbana-Champaign
rajshekd@andrew.cmu.edu yxw@illinois.edu moura@andrew.cmu.edu

1. Overview of ConFT

Algorithm 1 provides an overview of our distractor-aware contrastive finetuning approach ConFT.

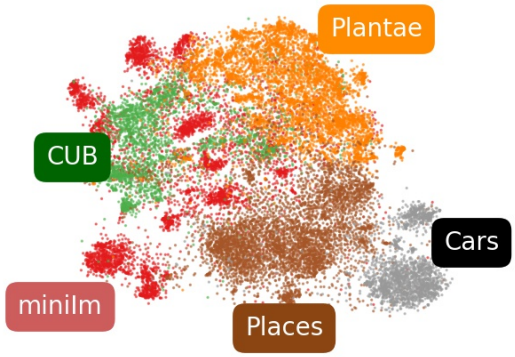
2. Additional Experimental Details

2.1. Data Domains

Problem Setup (Prior Learning)	Domain	Dataset	# categories per split		
			train	val	test
Cross-Domain	Base	*mini*ImageNet	64	16	20
	Novel	CUB	100	50	50
	Novel	Cars	98	49	49
	Novel	Places	183	91	91
	Novel	Plantae	100	50	50
Unsupervised	Base and Novel	*mini*ImageNet	64	16	20

Table 1. **Dataset statistics for both cross-domain [14] and unsupervised prior learning settings [7].** Each dataset is split into *train*, *val*, and *test* categories. For the cross-domain setup, the *train* split of *mini*ImageNet is always used as the base domain whereas the *test* splits of other datasets are used as the novel domain on which few-shot evaluation is performed. For the unsupervised prior learning setup, *train* split of *mini*ImageNet is stripped off its labels to emulate an unlabelled base domain, whereas the *test* split is used as the novel domain. In both setups, *val* splits are used to cross-validate hyperparameters specific to the associated novel domain.

In the main paper, we evaluated our finetuning method on various datasets that serve as base or novel domains in cross-domain as well as unsupervised prior learning settings. Here, in Table 1, we summarize the statistics of these datasets along with their specific use as base or novel domain. Additionally, in Table 2, we visualize these domains, both qualitatively and quantitatively, to provide a reference to their relative proximity in the representation space. This proximity provides a rough estimate of how related two domains are and consequently, the degree of knowledge transfer across domains for cross-domain few-shot classification.



Base Domain	Novel Domain	Proximity Rank	PAD
*mini*ImageNet	Places	1	1.06
	CUB	2	1.57
	Plantae	3	1.76
	Cars	4	1.86

Table 2. **Qualitative and quantitative visualization of the base and novel domains in the cross-domain benchmark [14].** We use t-SNE to visualize the base and novel domains in our cross-domain benchmark. The domain names are presented in boxes with colors that match the corresponding domains in the scatter plot. Here,"miniIm" refers to the *mini*ImageNet domain. We also compute the *Proxy $\mathcal{A}$-distance* (PAD) [1, 5] between the base domain and a novel domain as a measure of their relatedness in the representation space. Smaller the PAD value, closer is the novel domain to the base and hence, more related. The PAD values are also used to rank the novel domains according to their proximity to the base domain with the closest domain ranked the highest.

For the qualitative visualization in Table 2, we use t-SNE [15] to embed features of randomly sampled datapoints from each domain onto a 2-dimensional space. These features are obtained from the pretrained ResNet10 model (see §2.2 for training details) and are used for our cross-domain experiments. For quantitative visualization, we compute *Proxy $\mathcal{A}$-distance* [1, 5], or PAD, between the base domain

Algorithm 1 Distractor-Aware Contrastive Finetuning

Input: Distractor Dataset (D), Prior Model ($\mathcal{M}_{\theta_0}$), few-shot task (τ), Number of Finetuning Epochs (J_{ft}), Augmentation Function ($\mathcal{A}$), Temperature Coefficient (γ), Learning Rate (η)

Output: Finetuned Model Parameters (θ_τ)

shuffle D
for j $\leftarrow 1$ to J_{ft} **do**
  From D, randomly sample a fixed size batch S_{dt} without replacement
  Using $\mathcal{A}$ augment each support sample x_i, $\forall i \in I_{\text{supp}}$
  For each augmented support sample, define i) anchor-positive index set $P(i)$; ii) anchor-negative index set $N(i)$ specific to τ; and iii) distractor index set I_{dt}
  For all samples, compute $z_i = \frac{h_i}{||h_i||_2}$, where $h = \mathcal{M}_\theta(\mathcal{A}(x_i))$, $\forall i \in I_{\text{supp}}$ and $h = \mathcal{M}_\theta(x_i)$, $\forall i \in I_{\text{dt}}$
  Evaluate $\mathcal{L}_{\text{conft}}(\theta)$ using the quantities computed in previous steps
  Update model parameters $\theta \leftarrow \theta - \eta \nabla \mathcal{L}_{\text{conft}}(\theta)$
  if $j = |D|$ **then**
    shuffle D
  end if
end for

(here, *mini*ImageNet) and a novel domain as a measure of their closeness in the representation space. To compute PAD, we train a binary classifier over the same ResNet10 model used for t-SNE but with frozen embedding weights. The classifier distinguishes between randomly drawn samples of the base and novel domains. Denoting ϵ as the generalization error of this classifier, the PAD $\in [0, 2]$ is calculated as $2(1 - 2\epsilon)$. Thus, a lower PAD value implies higher generalization error which, in turn, signifies that the base and novel domains are too similar to be distinguished well enough. Finally, the PAD values are used to rank each novel domain, such that the highest rank is assigned to the one closest to the base domain *i.e.*, *mini*ImageNet. These ranks correlate well with the t-SNE visualization as well. For instance, CUB and Places, which are ranked higher than Cars and Plantae, are also closer to *mini*ImageNet in the t-SNE plot.

2.2. Prior Learning

As described in the main paper, we use a ResNet10 model [6] as our prior embedding for cross-domain few-shot classification. To avoid specialized hyperparameter tuning while training the prior model, we simply use the pretrained weights[1] made available by [14]. This model was originally trained on all 64 categories of the *mini*ImageNet *train* split.

For the unsupervised prior learning, we train a modified four-layer convolution neural network (CNN), using the recently proposed self-supervised contrastive learning objective [2]. As proposed in [2], we use a 128-dimensional linear projection head on top of the CNN for better generalizability of learnt representations. We train the model with a batch size of 512, temperature coefficient 0.1, and the same augmentation scheme introduced in [2]. Further, we use ADAM optimizer with initial learning rate of 1e-3, and a weight decay of 1e-5.

[1] https://github.com/hytseng0509/CrossDomainFewShot

2.3. Hyperparameter Details

Our proposed contrastive finetuning involves a few hyperparameters such as temperature, learning rate, early-stopping criteria, distractor batch size, and data augmentation scheme. For early-stopping criteria, we set a predetermined range of epochs up to which the pretrained embedding model is finetuned. Here, one finetuning epoch refers to one pass through all the samples of the few-shot task (exclusive of distractors). The range of these epochs along with other hyperparameters are summarized in Table 3 and Table 4. Additionally, we also show the final hyperparameter values used for finetuning in the cross-domain and unsupervised prior learning settings (the corresponding experiments were reported in the main paper).

3. Additional Ablations

In this section we elucidate the importance of two modifications introduced to the standard contrastive loss, namely, asymmetric construction of similarity pairs and relative weighting of anchor-negative terms.

3.1. Asymmetric Construction of Similarity Pairs

Our proposed finetuning approach is a general contrastive learning framework for incorporating additional *unlabelled* data in the form of distractors. While construction of positive distractor pairs (that share the same class) is difficult in the absence of distractor labels, constructing anchor-negatives, with anchors being task-specific samples,

Hyperparameter	Range	CUB		Cars		Places		Plantae	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
learning rate	{5e-4, 5e-3}	5e-3	5e-3	5e-3	5e-3	5e-4	5e-4	5e-3	5e-3
temperature, γ	{0.05, 0.1, 0.5}	0.1	0.1	0.05	0.05	0.1	0.05	0.1	0.1
distractor batch size, $\lvert S_{\text{dt}}\rvert$	{64, 128}	64	128	128	128	64	64	128	128
early stopping epoch	{50, 100, 200, 300, 400}	100	100	400	300	200	50	100	100

Table 3. **Hyperparameter details for ConFT with cross-domain prior learning.** This table summarizes the range of various hyperparameters used for finetuning. Additionally, we report the cross-validated values used for the cross-domain prior learning setup. The input image resolution used in this setup is 224×224.

Hyperparameter	Range	*mini*ImageNet	
		1-shot	5-shot
learning rate	{5e-4, 5e-3}	5e-4	5e-4
temperature, γ	{0.05, 0.1, 0.5}	0.05	0.05
distractor batch size, $\lvert S_{\text{dt}}\rvert$	{64, 128}	64	64
early-stopping epoch	{50, 100, 200, 300, 400, 500}	400	400

Table 4. **Hyperparameter details for ConFT with unsupervised prior learning.** This table summarizes the range of various hyperparameters used for finetuning. Additionally, we report the cross-validated values used for the unsupervised prior learning setup. The input image resolution used in this setup is 84×84.

Similarity-Pair Construction	Cars	
	1-shot	5-shot
Standard	37.09 ± 0.76	60.72 ± 0.74
Assymetric (ours)	$\mathbf{39.11 \pm 0.77}$	$\mathbf{61.53 \pm 0.75}$

Table 5. **Comparing our proposed assymetric construction of similarity pairs against standard construction:**. Results are shown for both 1-shot and 5-shot tasks sampled from the Cars domain with *mini*Imagenet as the base domain. These results are averaged over 600 random novel tasks and are reported with $(\pm)$ 95% confidence intervals. Despite using more supervision in the form of distractor labels, the standard pair construction underperforms our (distractor) label-agnostic asymmetric pair construction.

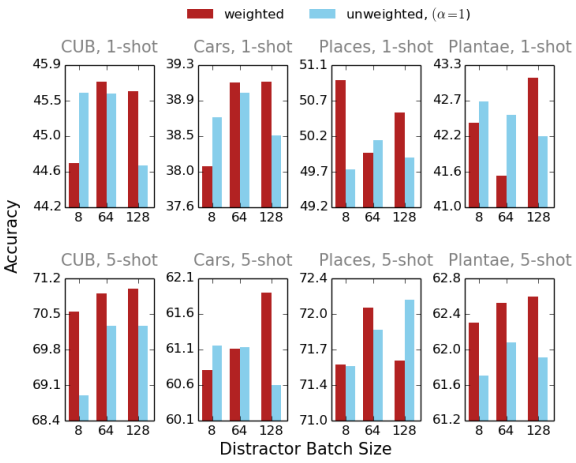


Figure 1. **Comparing the effect of distractor batch size, $\lvert S_{\text{dt}}\rvert$, on the weighted and unweighted versions of $\mathcal{L}_{\text{conft}}$.** The *red* and *blue* bars represent weighted and unweighted versions of $\mathcal{L}_{\text{conft}}$, respectively, where α represents the parameter used to relatively weigh task-specific and task-exclusive (distractors) anchor-negative terms. For each novel domain and shot setting per domain, we compare the performance of two versions in terms of the classification accuracy of unseen samples given a novel task at various distractor batch sizes. These accuracies, as in all other cross-domain experiments, are averaged over 600 randomly chosen novel tasks.

is much easier following the non-overlapping assumption of task and distractor categories. This results in an asymmetric construction of similarity pairs where distractors, unlike task-specific samples, can meaningfully participate only as anchor-negatives. In fact, this asymmetry is critical in the unsupervised prior learning setup, where distractors are sampled from an unlabelled base domain. In the case of cross-domain prior learning, however, we have a labelled base data as a source for distractors. To motivate our asymmetric pair construction in this case, we compare it to a standard construction that allows distractors to additionally participate as anchor-positives. To form such an anchor-positive, a distractor is paired with another distractor sharing the same class. Here, anchor-negatives with respect to a distractor include all the datapoints that do not share the class with it. This includes samples from both the novel few-shot task and other distractors. Overall, the resulting form of the contrastive loss can be viewed as applying supervised contrastive objective [8] (without augmentation-based positives) to the union set of task samples and distractors within a training batch. In Table 5, we evaluate these two types of pair constructions on the cross-domain setting, *mini*ImageNet $\rightarrow$ Cars. Interestingly enough, our formulation of the contrastive loss with asymmetric pair construction yields superior performance despite using less supervision than the supervised contrastive loss.

3.2. Importance of Weighted Negatives

Another important component of our loss is the relative weighting parameter α that balances the effect of task-specific and task-exclusive (distractor based) anchor-negative terms. To validate the utility of such a weighting scheme, we compare the weighted version of $\mathcal{L}_{\text{conft}}$ to its unweighted version *i.e.*, $\alpha = 1$. Following the results for various novel domains and shot settings in Fig-

ure 1, we make the following observations. The weighted loss (*red* bars) performance improves with larger distractor batch sizes in most cases (5 out of 8). The improvement is more pronounced for domains like Cars and Plantae that are farther away from the base dataset - *mini*ImageNet (see Table 2). For closer domains like CUB or Places, we sometimes notice a sweet spot at batch size $= 64$. In contrast, the unweighted version (*blue* bars) experiences a performance drop with increasing batch sizes, when the novel domains are farther from the base domain. In other cases, the trends are inconclusive. The most important observation, however, comes from comparing the two versions of the loss. Specifically, the weighted version not only outperforms the unweighted loss at higher batch sizes but also results in the best performance in almost every setting. The only exception is Places, 5-shot where the unweighted loss yields the best performance. A possible explanation is as follows: due to the similarity of Places (novel domain) and *mini*ImageNet (base domain) in the embedding space (see Table 2), distractor samples from Places may serve as hard negatives that are important for effective contrastive learning [10]. Thus, down-weighting their contribution at higher batch sizes would degrade the final performance.

3.3. Data Augmentation

Augmentation		CUB	Cars
Task Samples	Distractors	5-shot	5-shot
-	-	69.90 ± 0.75	58.64 ± 0.88
✓	-	70.53 ± 0.75	61.53 ± 0.75

Table 6. In this ablation we compare the few-shot performance when a prior embedding is finetuned (using ConFT) with or without augmentation to task-specific samples. Note that, we never use augmentation for distractors in our experiments.

Yet another important component of our contrastive finetuning objective is the data augmentation function $\mathcal{A}$. To avoid extensive tuning of large hyperparameter space associated with $\mathcal{A}$, we adopt a fixed augmentation strategy introduced in [3]. In Table 6, we show the benefit of using this strategy to augment samples specific to the novel task. Following preliminary investigations, we found that augmenting distractors did not make much difference. Hence, we never apply data augmentation to distractors in our experiments.

3.4. Loss Type

In Table 7, we compare contrastive and cross-entropy finetuning in conjunction with the auxiliary cross-entropy objective (MT). While the two objectives yield similar performance for the CUB case, contrastive finetuning outperforms cross-entropy loss based finetuning in Cars. These results show that the contrastive loss could be a better choice for few-shot classification.

Method		CUB	Cars
Prior Learning	Task Specific Finetuning	5-shot	5-shot
CE Training	MT-ceFT ($\beta = 1$)	71.35 ± 0.70	58.97 ± 0.76
CE Training	MT-ceFT ($\beta = 10$)	74.32 ± 0.69	60.01 ± 0.74
CE Training	MT-ConFT ($\beta = 1$)	71.65 ± 0.74	61.25 ± 0.70
CE Training	MT-ConFT ($\beta = 10$)	74.45 ± 0.71	62.54 ± 0.72

Table 7. **Ablation.** Cross-entropy/contrastive finetuning with a multi-task (MT) cross entropy objective. Here, all cross entropy objectives are based on cosine classifier with a multiplying factor, β

4. ConFT as a General Finetuning Approach

In Table 8, we validate the complementary effect of our finetuning approach to a variety of prior learning schemes. Specifically, we compare our simple cross-entropy objective with ProtoNet [11] and ProtoNet with auxiliary self-supervision [12]. Both of these approaches are based on meta-learning, and were originally proposed for in-domain few-shot classification where base and novel tasks follow the same distribution. Nevertheless, the embeddings thus learnt are readily applicable to cross-domain tasks as well. For the auxiliary self-supervision, we use image rotation as our pretext task. While previous work [12] has demonstrated the improvement in in-domain few-shot generalization resulting from rotation based self-supervision, we found that the improvement is marginal in our cross-domain setting (see ProtoNet without finetuning vs. ProtoNet + Rot. without finetuning in Table 8), except for when the novel domain is Cars. To obtain these results, we use the official implementation[2] of [12] with the same hyperparameters (such as loss weighting term) but different backbone. As our pretrained embedding, we trained a ProtoNet model (with auxiliary self-supervision) based on ResNet10 [6] architecture. Our main observation from Table 8 is as follows: while better prior learning objectives such as those with auxiliary self-supervision can improve few-shot classification in the novel domains, finetuning with ConFT consistently leads to large improvements over the prior embeddings.

5. Additional Comparison with Prior Work

In Table 9, we report additional comparison with a concurrent work SCL [9] that introduces attention-based spatial contrastive objective in the prior-learning phase. For a fair comparison to SCL, we adopt the same backbone based on the ResNet12 architecture which was originally proposed in [13]. While the spatial contrastive objective benefits from larger image resolution (224×224), we found it significantly increases the time for finetuning in our case, especially given the larger backbone. So, in this case, we conduct our experiments with a smaller resolution of 84×84 . Despite the drop in resolution, our finetuning based

[2] https://github.com/cvl-umass/fsl_ssl

Method			5-shot			
Prior Learning	Task Specific Finetuning	Backbone	CUB	Cars	Places	Plantae
ProtoNet [11]	-	ResNet10	58.80 ± 0.77	44.07 ± 0.69	71.03 ± 0.72	51.33 ± 0.72
ProtoNet [11]	ConFT (ours)	ResNet10	$\mathbf{66.63 \pm 0.69}$	$\mathbf{59.27 \pm 0.73}$	$\mathbf{72.05 \pm 0.71}$	$\mathbf{58.83 \pm 0.76}$
ProtoNet + Rot. [12]	-	ResNet10	58.68 ± 0.75	46.48 ± 0.71	71.20 ± 0.75	51.93 ± 0.67
ProtoNet + Rot. [12]	ConFT (ours)	ResNet10	$\mathbf{66.75 \pm 0.71}$	$\mathbf{61.67 \pm 0.75}$	$\mathbf{73.91 \pm 0.70}$	$\mathbf{60.38 \pm 0.75}$
CE Training	-	ResNet10	62.80 ± 0.76	51.41 ± 0.72	70.71 ± 0.68	55.54 ± 0.69
CE Training	ConFT (ours)	ResNet10	$\mathbf{70.53 \pm 0.75}$	$\mathbf{61.53 \pm 0.75}$	$\mathbf{72.09 \pm 0.68}$	$\mathbf{62.54 \pm 0.76}$

Table 8. **Combining ConFT with different pretraining schemes for cross-domain prior learning.** We present the results for 5-way 5-shot tasks averaged over 600 such tasks with $(\pm)$ 95% confidence intervals. The highlighted numbers demonstrate that ConFT consistently improves the few-shot performance of prior embeddings across data domains.

Method			1-shot			
Prior Learning	Task Specific Finetuning	Backbone	CUB	Cars	Places	Plantae
SCL [9]	-	ResNet12	50.09 ± 0.7	34.93 ± 0.6	$\mathbf{60.32 \pm 0.8}$	40.23 ± 0.6
CE Training	-	ResNet12	50.00 ± 0.77	34.88 ± 0.64	55.62 ± 0.91	38.47 ± 0.72
CE Training	ConFT (ours)	ResNet12	$\mathbf{52.01 \pm 0.82}$	$\mathbf{39.54 \pm 0.68}$	56.66 ± 0.85	$\mathbf{40.90 \pm 0.73}$

Method			5-shot			
Prior Learning	Task Specific Finetuning	Backbone	CUB	Cars	Places	Plantae
SCL [9]	-	ResNet12	68.81 ± 0.6	52.22 ± 0.7	$\mathbf{76.51 \pm 0.6}$	$\mathbf{59.91 \pm 0.6}$
CE Training	-	ResNet12	69.75 ± 0.73	49.92 ± 0.74	73.79 ± 0.67	54.66 ± 0.77
CE Training	ConFT (ours)	ResNet12	$\mathbf{76.49 \pm 0.63}$	$\mathbf{64.87 \pm 0.70}$	74.22 ± 0.71	$\mathbf{59.23 \pm 0.77}$

Table 9. **Additional Prior Work Comparison.** SCL introduces a novel attention-based spatial contrastive objective for prior learning. While we employ a much simpler cross-entropy objective for prior learning (see CE training *without* ConFT), finetuning the prior embedding with ConFT outperforms SCL significantly in two (CUB and Cars) out of four domains. Our approach yields competitive results for Plantae as well. Further, due to the complementary nature of finetuning, the best performance might be achieved by combining SCL with our ConFT.

approach over simple cross-entropy prior learning outperforms the more sophisticated SCL by significant margins in CUB (7 points) and Cars (13 points). While we attain similar performance in the case of Plantae, we underperform in Places domain. This gap can be understood as a consequence of a stronger SCL based prior embedding for *mini*ImageNet and greater similarity of the *mini*ImageNet domain to Places as opposed to other novel domains (see Table 2). Nonetheless, our finetuning is complimentary to SCL, and hence we suspect that the best performance could be achieved by combining it with our ConFT.

6. Meta-Dataset Results

In this section, we present the results of our ConFT approach on Meta-Dataset (see Table 10). Here, we use an off-the-shelf ResNet18 model[3] pretrained on ImageNet-train-split of Meta-Dataset using just cross-entropy objective. In order to maintain consistency with pretraining, our finetuning operates at a small image resolution of 84×84. In this experiments, we keep most of the hyperparameters fixed across all datasets. In particular, we use a temperature of 0.1, a distractor batch size of 128, and a learning rate of $5e-5$. The early stopping epoch is cross-validated using the meta-validation splits of respective datasets. We obeserve that our approach outperforms the state of the art in 7 out of 10 datasets and sometimes by a significant margin. This is despite the fact that our input resolution is much smaller compared to 224×224 in the state of the art and our approach does *not* benefit from a tansductive setting. Finally, our results reinforce the superiority of simple finetuning over more complex meta-learning frameworks (*e.g.* cross-attention based) even when the domain gap is large.

[3] https://github.com/peymanbateni/simple-cnaps

References

[1] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *NeurIPS*. 2007. 1

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv*, 2020. 2

[3] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019. 4

[4] C. Doersch, A. Gupta, and Andrew Zisserman. Crosstransformers: Spatially-aware few-shot transfer. In *NeurIPS*, 2020. 6

[5] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *CVPR*, 2017. 1

Method	Target Datasets				
	ILSVRC	Omni	Aircraft	Birds	DTD
PN [4]	$41.87_{\pm 0.89}$	$61.33_{\pm 1.13}$	$39.40_{\pm 0.78}$	$65.57_{\pm 0.73}$	$59.06_{\pm 0.60}$
CTX [4]	$51.70_{\pm 0.90}$	$84.24_{\pm 0.79}$	$62.29_{\pm 0.73}$	$\mathbf{79.38_{\pm 0.54}}$	$\mathbf{65.86_{\pm 0.58}}$
CTX+SC [4]	$51.29_{\pm 0.89}$	$86.14_{\pm 0.74}$	$\mathbf{69.74_{\pm 0.67}}$	$74.85_{\pm 0.62}$	$63.84_{\pm 0.62}$
CTX+SC+Aug [4]	$52.56_{\pm 0.86}$	$87.53_{\pm 0.61}$	$64.28_{\pm 0.71}$	$73.27_{\pm 0.63}$	$64.72_{\pm 0.63}$
ConFT (ours)	$\mathbf{72.07_{\pm 0.71}}$	$\mathbf{98.22_{\pm 0.17}}$	$68.44_{\pm 0.70}$	$74.93_{\pm 0.67}$	$63.11_{\pm 0.70}$
Method	Target Dataset				
	QDraw	Fungi	Flower	Sign	COCO
PN [4]	$47.86_{\pm 0.80}$	$41.64_{\pm 1.02}$	$83.88_{\pm 0.48}$	$44.84_{\pm 0.88}$	$41.14_{\pm 0.82}$
CTX [4]	$63.36_{\pm 0.73}$	$49.43_{\pm 0.98}$	$92.74_{\pm 0.29}$	$68.31_{\pm 0.71}$	$48.63_{\pm 0.79}$
CTX+SC [4]	$64.11_{\pm 0.67}$	$48.87_{\pm 0.91}$	$93.00_{\pm 0.30}$	$70.62_{\pm 0.68}$	$48.45_{\pm 0.83}$
CTX+SC+Aug [4]	$66.90_{\pm 0.66}$	$48.22_{\pm 0.94}$	$93.23_{\pm 0.28}$	$78.45_{\pm 0.60}$	$56.61_{\pm 0.78}$
ConFT (ours)	$\mathbf{80.02_{\pm 0.6}}$	$\mathbf{50.16_{\pm 0.80}}$	$\mathbf{94.52_{\pm 0.29}}$	$\mathbf{88.22_{\pm 0.59}}$	$\mathbf{70.73_{\pm 0.79}}$

Table 10. **Meta-Dataset Results (5-shot).** Cross-domain results of our distractor-aware contrastive finetuning (ConFT) on transfer from ImageNet-only are presented here. The accuracies are averaged over 600 evaluation tasks with 95% confidence intervals. PN: Prototypical Net, SC: SimCLR Episodes.

[6] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2, 4

[7] Kyle Hsu, Sergey Levine, and Chelsea Finn. Unsupervised learning via meta-learning. In *ICLR*, 2019. 1

[8] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *NeurIPS*, 2020. 3

[9] Yassine Ouali, Céline Hudelot, and Myriam Tami. Spatial contrastive learning for few-shot classification. *arXiv*, 2020. 4, 5

[10] Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *ICLR*, 2021. 4

[11] Jake Snell, Kevin Swersky, and Richard S. Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017. 4, 5

[12] Jong-Chyi Su, Subhransu Maji, and Bharath Hariharan. When does self-supervision improve few-shot learning? In *ECCV*, 2020. 4, 5

[13] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: A good embedding is all you need? *arXiv*, 2020. 4

[14] 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

[15] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9, 2008. 1