DAFT: Distilling Adversarially Fine-tuned Teachers for OOD Generalization

Anshul Nasery [1] **Sravanti Adepalli** [1 2] **Praneeth Netrapalli** [1] **Prateek Jain** [1]

Abstract

We consider the problem of OOD generalization, where the goal is to train a model that performs well on test distributions that are different from the training distribution. Deep learning models are known to be fragile to such shifts and can suffer large accuracy drops even for slightly different test distributions (Hendrycks & Dietterich, 2019).

We propose a new method – DAFT – based on the intuition that adversarially robust combination of a large number of rich features should provide OOD robustness. Our method carefully distills the model from a powerful teacher that learns several discriminative features using standard training while combining them using adversarial training. The standard adversarial training procedure is modified to produce teachers which can guide the student better. We evaluate DAFT on standard benchmarks in the DomainBed framework (Gulrajani & Lopez-Paz, 2020), and find that DAFT consistently out-performs well-tuned ERM and distillation baselines by up to 6%, with more pronounced gains for smaller networks.

1. Introduction

Several recent works have shown that standard deep learning models trained with stochastic gradient descent (SGD) style methods can be *fragile* and might suffer a large drop in accuracy if the test data distribution (also known as target domain) is even slightly different compared to the training data distribution (also known as source domain) (Hendrycks & Dietterich, 2019; Gulrajani & Lopez-Paz, 2020). However, in practice, it is quite challenging to obtain training data that exactly matches the test distribution. For example, due to privacy restrictions we may not be able to access the data of the actual customers of a web-application. Instead, training data is generated using crowd workers or by seeking volunteers who are willing to donate their data for training. This clearly leads to distribution shift between the training and test data.

Consequently, it is crucial to design models and training mechanisms that are robust to distribution shifts and can perform well on *out-of-distribution* (OOD) data. Even in settings where some amount of data can be collected from the final deployment setting, it should be relatively easier to adapt models with good *OOD generalization*. OOD problems have been studied in a variety of settings where different amounts of source/ target information might be available. We consider the *OOD generalization* setting which is one of the weakest settings, and requires only a labeled training dataset, without any information about the target dataset or even about the sources present in the training dataset. Note that this setting is slightly different and more challenging than the popular *domain generalization* setting which requires the identity of source domain for each training point. Even, in the domain generalization setting, several recent works (Gulrajani & Lopez-Paz, 2020; Vedantam et al., 2021) show that a well-tuned ERM model is still competitive with SOTA methods.

We propose DAFT to mitigate the problem of *OOD generalization*; DAFT is motivated by three key observations as discussed in Sec 3.

Our contributions: In summary, we introduce a novel method – Distillation with Adversarially Finetuned Teacher (DAFT) – that uses the above algorithmic insights to design a robust technique for OOD generalization. DAFT trains a student model by distilling with a standard trained as well as an adversarially finetuned teacher model. Finetuning uses a smooth KL divergence based loss for all logits/prediction values. See Figure 1 for an overview of DAFT.

We conduct extensive experiments in the standard DomainBed framework (Gulrajani & Lopez-Paz, 2020) – using the prescribed methodology for evaluation, hyperparameter tuning – and compare DAFT against various baselines on the five OOD datasets in the testbed. Recall that even for stronger Domain Generalization setting, (Gulrajani & Lopez-Paz, 2020) showed that well-trained ERM is nearly SOTA. In contrast, we demonstrate that DAFT trained in the weaker OOD generalization setting, still consistently out-

[1] Google Research India [2] Indian Institute of Science, Bangalore. Correspondence to: Anshul Nasery <anshulnasery@google.com>.

Proceedings of the 39^{th} *International Conference on Machine Learning*, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

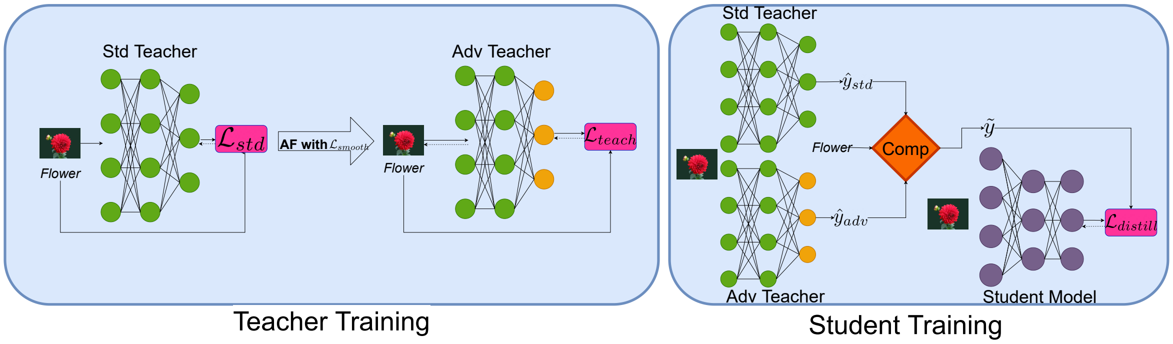

Figure 1. DAFT overview. We pre-train a teacher, followed by adversarial fine-tuning using $\mathcal{L}_{smooth}$. We then distill a student from both standard and adversarial teachers. The Comp operator outputs $\hat{y}_{adv}$ if adversarial teacher's prediction is correct, else it outputs $\hat{y}_{std}$.

performs ERM (trained according to DomainBed approach) and other baselines. DAFT is particularly effective for smaller network architectures. For example, on DomainBed datasets, DAFT trained ResNet-50 models are on an average 4% more accurate than ERM as well as other baselines. In fact, DAFT+ResNet-50 models are more accurate than an ERM trained ResNet-152 as well; see Table 1.

Limitations: In general, different application areas/problems might lead to different forms of distribution shift between training and test distributions. Consequently, there may not be a single algorithm that is optimal under all distribution shifts. To address this, we perform experiments on multiple OOD datasets with different characteristics such as satellite images with temporal/geographical shift in TerraIncognita, and object images with varying sources in OfficeHome, VLCS, PACS and DomainNet datasets. In all of these settings, DAFT gives significant improvements over ERM. However, there may certainly be other forms of distribution shift, beyond those captured in the above datasets, where new ideas might be required to obtain superior performance.

2. Problem Definition

Problem setting: We are given a training dataset of labeled examples $\mathcal{D}_S = \{(x_i, y_i) : i \in [n]\}$, each example drawn independently from a source distribution $\mathcal{S}$. Given a model $\mathcal{M}$, we also assume that we can evaluate its accuracy on validation distribution $\mathcal{V}$, which may be different from $\mathcal{S}$. Note that we do not explicitly access examples from $\mathcal{V}$, but rather use it only for evaluating the accuracy of any given model. Our goal is to train a model $\mathcal{M}^*$ that performs well on data drawn from a target distribution $\mathcal{T}$. Primarily we study the setting of $\mathcal{T} \neq \mathcal{V}$ — especially for the experiments reported in the main paper — but for generality, we report results for $\mathcal{T} = \mathcal{V}$ also in the appendix. We would like to stress that we do not have access to $\mathcal{T}$ during model training (and also model/hyper-parameter selection if $\mathcal{T} \neq \mathcal{V}$).

Motivation: In several settings, there are privacy and proprietary reasons for not having access to or not using validation or target data (i.e., $\mathcal{V}$ and $\mathcal{T}$ respectively) during model training. A common strategy to deal with this is to generate training data through other ways such as using crowd workers, requesting some users to volunteer their data (in the case of user data), using simulation models etc. However, once we train a model, we often have the ability to deploy/evaluate the model for its accuracy on the validation domain $\mathcal{V}$, and in some cases, even on the target domain $\mathcal{T}$.

3. Methodology

Motivation and high level description of algorithm: We begin by describing the key insights that led to our algorithm, and provide a high-level description. Recent works have demonstrated that adversarial training can learn more *robust* features compared to standard training (Ilyas et al., 2019; Yi et al., 2021). However, we find that vanilla adversarial training does not provide substantial improvements on larger datasets. Hence, we hypothesize that standard ERM trained models might already learn features which are good for domain generalization (Kirichenko et al., 2022; Kumar et al., 2022; Rosenfeld et al., 2022), but the final layer is not able to combine these features in a manner robust to domain shifts.

To further illustrate this point, we perform an experiment using a modification of a Fashion-MNIST subset with only two-class: *shoe* and *top*. We superimpose images onto coloured backgrounds, where the colour varies linearly between red $(255, 0, 0)$ and green$(0, 255, 0)$. Training images have a strong correlation between the background colour and label, i.e. color of *tops* images range from $(255, 0, 0)$ to $(123, 132, 0)$, while that of *shoes* images range between $(132, 123, 0)$ and $(0, 255, 0)$. In general, color can easily distinguish between the classes, but there is a small region between $(123, 132, 0)$ and $(132, 123, 0)$, where color cannot distinguish between the classes. During test time, there is no such correlation with colour i.e. the data is OOD w.r.t.

the train data. For models trained on this data, we compute the correlation of each neuron at the output of the feature extractor with the shape and colour of the images. Note that color is a non-robust spurious feature while shape is a robust feature that is strongly correlated with the labels and is useful for prediction despite OOD shifts.

Now, an ERM trained model has an in-domain (ID) test accuracy of 99.9%, and OOD test accuracy of 60%. Furthermore, *only* 2 of the 32 features are highly correlated with the robust shape feature, while the rest are correlated with color. The final class output is dominated by the color features. In contrast, an adversarially trained model has a higher number of shape features (8 out of 32). But the in-domain accuracy is only 98.3%, and the OOD accuracy is 58%, lower than ERM. A probable reason is that even though adversarial training learns *more* shape-correlated robust features, but the average correlation with shape is much smaller (around 0.75) than the similar correlation of features from standard trained model (around 0.8). This is possibly because the shape features learned by adversarially trained models are more suited to the goal of adversarial robustness, while the features learned by standard ERM models are better correlated with the standard classification task. Hence, we introduce the method of adversarial fine-tuning of the *last layer* after standard ERM based pre-training. This encourages the model to give a lower prediction weight to color features, and higher weight to the robust shape features learned by the ERM model. With adversarial fine-tuning, a small ID accuracy drop 99.5% occurs, but the OOD accuracy jumps to 64%.

Next, motivated by the observation that distillation from larger models often helps in-domain performance, we trained our final model through distillation of a larger model which was itself trained using adversarial fine-tuning.

Surprisingly, vanilla distillation failed to transfer the superior performance of adversarially teacher model to the student model. We identify the main reason behind the failure of distillation in transfering the superior OOD performance of teacher to the student to be the following: while the Cross-Entropy loss on adversarial samples ensures that the logit corresponding to the correct class is "robust", it does not put any constraints on the remaining logits. Consequently, the remaining logits do not provide useful information for distillation. In order to tackle this, we add an additional KL divergence term to ensure that all the logits of the teacher model are smooth in the neighborhood of the given input, and are aligned to the logits of the clean image. The loss to be minimized the teacher is hence $\mathcal{L}(W,(x,y)) = \max_{\hat{x}\in B_\epsilon(x)} KL(z||\hat{z}) + CE(\hat{z},y)$, where z and $\hat{z}$ are the logits of the teacher for x and $\hat{x}$, and CE is the cross-entropy loss. We use l_2 constrained perturbations with ϵ being a hyper-parameter.

The final ingredient of our approach is to use two teacher models: the adversarially fine-tuned teacher on inputs where it predicts correctly and the standard trained model on the remaining inputs. We call the resulting algorithm Distillation of Adversarially Fine-Tuned teacher (DAFT).

4. Experimental Results

In this section, we detail our experimental setup, datasets, baselines and results.

4.1. Experimental Setup

Our experimental setup follows the approach and recommendations of (Gulrajani & Lopez-Paz, 2020). For all our experiments, we train models of different sizes from the ResNet (He et al., 2015) family. Hyper-parameters for all the methods are tuned using the *leave-one-domain-out* approach descibed in (Gulrajani & Lopez-Paz, 2020). We use ImageNet pretrained models for comparisons on DomainBed. We report the mean and std deviation of the metrics across five random restarts.

Datasets We report OOD accuracy results on 5 different datasets, and the average accuracy across them. We use all the datasets from the DomainBed (Gulrajani & Lopez-Paz, 2020) benchmark.

Baselines We compare DAFT against the standard ERM method trained on training data $\mathcal{D}_S$. We also compare against AT which trains (*not finetunes*) models using PGD for l_2-norm constrained input adversarial perturbations (Madry et al., 2019), as well as TRADES (Zhang et al., 2019) which is a variant. Since DAFT uses larger models to train a smaller student network, we also compare it against the performance of distilling directly from an ERM trained teacher model. The teacher in all cases is ResNet-152.

4.2. Results

We compare the OOD accuracy of DAFT against baselines in Table 1. We notice that our method provides significant improvements over standard ERM across all datasets and model sizes. For example, on OfficeHome, we show gains of almost 5% over ERM on all model sizes. We also note that smaller models trained with DAFT outperform ERM trained larger models; ResNet-34 trained with DAFT is close in performance to ResNet-152 on an average, while ResNet-50 can beat it.

We also notice that our method outperforms standard logit distillation on all benchmark datasets. This demonstrates that our method leverages the information provided by larger models in a more efficient manner. Furthermore, the KL-regularization of teachers in DAFT helps improve the transfer, as we demonstrate in the ablation experiments (sec 4.3).

Model Size	Method	PACS	VLCS	OfficeHome	DomainNet	TerraIncognita	Avg
ResNet-18	ERM	80.2± 1.0	71.4± 0.6	57.4± 0.4	31.2± 0.0	40.8± 1.3	56.2
	AT	79.6± 0.9	68.6± 0.3	56.5± 0.8	30.6± 0.6	56.5± 0.7	58.4
	TRADES	79.4± 0.6	70.4± 0.8	56.7± 0.7	29.8± 0.1	39.6± 0.9	55.2
	Distillation	83.1± 0.3	76.6± 0.3	62.8± 0.3	33.8± 0.2	48.3± 0.5	60.9
	DAFT	**84.7± 1.1**	**78.2± 0.1**	**63.2± 0.2**	**36.4± 0.2**	**50.2± 0.8**	**62.5**
ResNet-34	ERM	83.2± 0.8	73.5± 1.0	60.8± 0.6	32.5± 0.0	41.0± 0.7	58.2
	AT	82.2± 1.0	72.9± 0.5	60.5± 0.5	30.7± 0.3	40.6± 0.4	57.4
	TRADES	82.5± 0.6	72.2± 0.7	60.7± 0.8	31.4± 0.3	41.3± 0.2	57.6
	Distillation	84.0± 1.7	76.0± 0.7	66.3± 0.1	36.7± 0.1	48.5± 0.9	62.3
	DAFT	**87.4± 0.3**	**79.1± 0.9**	**67.2± 0.5**	**38.5± 0.3**	**51.4± 1.1**	**64.7**
ResNet-50	ERM	83.3± 1.7	75.2± 1.2	67.0± 0.6	41.1± 0.1	46.2± 0.7	62.6
	AT	82.6± 1.2	72.0± 1.2	67.0± 0.3	40.3± 0.2	45.3± 1.1	61.4
	TRADES	82.6± 0.9	72.3± 0.9	66.1± 0.8	40.4± 0.1	45.1± 0.4	61.3
	Distillation	85.9± 0.9	76.5± 0.9	67.7± 0.4	41.9± 0.2	50.7± 0.7	64.5
	DAFT	**88.0± 0.1**	**80.0± 0.2**	**71.0± 0.2**	**42.6± 0.2**	**52.8± 0.1**	**66.9**
ResNet-101	ERM	85.0± 0.0	76.9± 0.4	67.6± 0.5	42.6± 0.1	49.5± 0.0	64.3
	AT	72.6± 0.1	75.9± 0.4	67.5± 0.4	42.3± 0.1	47.9± 0.1	61.2
	TRADES	83.7± 0.5	76.3± 0.4	68.0± 0.3	42.2± 0.1	49.5± 0.8	63.9
	Distillation	86.9± 0.7	77.1± 0.4	69.1± 0.2	43.2± 0.1	50.3± 0.3	65.3
	DAFT	**88.8± 0.5**	**79.1± 0.5**	**72.2± 0.8**	**43.7± 0.5**	**54.1± 0.9**	**67.6**
ResNet-152	ERM	87.0± 0.4	79.2± 0.1	69.0± 0.5	43.2± 0.0	50.4± 0.2	65.7
	AT	87.1± 0.1	78.8± 0.1	69.6± 0.3	42.8± 0.0	49.6± 0.5	65.6
	TRADES	87.3± 0.1	78.8± 0.1	69.7± 0.1	42.7± 0.0	49.8± 0.2	65.7
	Distillation	**88.8± 1.6**	**80.4± 1.3**	**71.3± 0.2**	43.6± 0.1	**55.1± 1.2**	67.8
	DAFT	**88.7± 2.0**	**80.7± 1.7**	**71.9± 1.2**	**44.1± 0.0**	**55.9± 1.0**	**68.3**

Table 1. OOD accuracy on various datasets with different ResNet (RN) architectures.

Model	Algorithm	PACS	VLCS	OfficeHome	Avg
ResNet-101	ERM	85.0± 0.0	76.9± 0.4	67.6± 0.5	76.5
	AT	72.6± 0.1	75.9± 0.4	67.5± 0.4	72.0
	AF	**86.4± 0.1**	**77.9± 0.1**	**69.6± 0.2**	**77.9**
	TRADES	83.7± 0.5	76.3± 0.4	68.0± 0.3	76.0
	AF+$\mathcal{L}_{smooth}$	**86.5± 0.1**	**77.6± 0.3**	**69.5± 0.3**	**77.9**

Table 2. **Effect of adversarial finetuning (AF)**: Accuracy achieved by ERM, adversarial training (AT) and adversarial finetuning (AF) for different architectures. Note that AF performs better than ERM, while AT is often worse than or similar to ERM due to poor in-domain accuracy of AT.

We also notice that DAFT is able to outperform standard baselines by larger margins for datasets like TerraIncognita where the domains are significantly different from ImageNet. This means that features learnt from ImageNet pretraining would be less useful in this scenario. The better performance of DAFT on this dataset implies that it is able to transfer generalizable features better. We also verify the gains of DAFT without using ImageNet pretrained models, but do not report them due to lack of space. We also performed several ablations of our method to verify the effectiveness of each component.

4.3. Ablations

Does adversarial finetuning work? To study the effect of our teacher training paradigms, we compare performance of various teacher models on the PACS, VLCS and OfficeHome dataset in Table 2. We show that it is much better to pre-train a model and finetune the final layer adversarially (AF), rather than training the full model adversarially (AT). Note that AT is competitive to ERM only on the OfficeHome dataset, since there is a high inter-domain similarity in three of the four domains of this dataset, and the images are also similar to ImageNet, on which the models were originally pretrained. This is consistent with the findings of (Yi et al., 2021).

Effect of $\mathcal{L}_{smooth}$ on distillation: To verify the effect of using teachers trained with $\mathcal{L}_{\text{smooth}}$, we present results on three datasets in Fig 2. For each split, we compute the average gain in OOD accuracy for the teacher (which is a ResNet-152) when trained with adversarial finetuning with ($\Delta^{\text{Teach}}_{\mathcal{L}_{\text{smooth}}}$) and without ($\Delta^{\text{Teach}}_{\text{AF}}$) $\mathcal{L}_{\text{smooth}}$. We then compare the gains over standard distillation observed in students distilled from these teachers ($\Delta^{\text{Dist}}_{\mathcal{L}_{\text{smooth}}}$ and $\Delta^{\text{Dist}}_{\text{AF}}$ respectively). Note that gain here refers to the difference in the OOD accuracy of the modified teacher (resp. distilled student of the modified

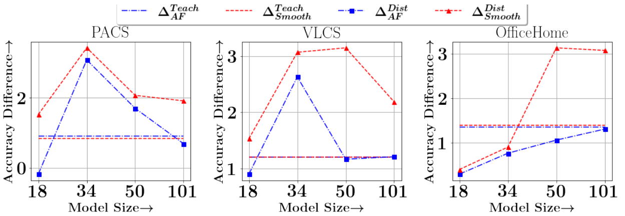

Figure 2. **Importance of smoothness term $\mathcal{L}_{\text{smooth}}$ in teacher training for student performance**: $(\Delta^{\text{Dist}}_{\text{Smooth}})$ and $(\Delta^{\text{Dist}}_{\text{AF}})$ denote accuracy increment for student models compared to std distillation, when distilled from a teacher with and without the smoothness term, respectively. Accuracy improvements for teacher over ERM are $\Delta^{\text{Teach}}_{\text{AF}}$ and $\Delta^{\text{Teach}}_{\text{Smooth}}$ when trained with and without smoothness term, respectively.

teacher) model over a standard ERM (resp. distilled student of a standard teacher) model. We observe that students distilled from a teacher trained with $\mathcal{L}_{\text{smooth}}$ obtain similar or even better accuracy gains compared to those achieved by the teacher. In contrast, students of teachers trained without this term do not even consistently achieve similar accuracy gains as their teachers.

5. Conclusion

Summary: In this paper, we considered the problem of out of distribution (OOD) generalization, where we are given training examples from a source distribution and are required to output a model which will be evaluated on test examples sampled from a different target distribution. We first observed that the non-robustness of standard trained models on OOD data is primarily due to a non-robust combination of learned features in the final linear layer and that the features themselves are capable of obtaining high OOD accuracy. Inspired by this observation, we designed adversarial finetuning (AF) which first trains the model using standard training and then finetunes the final linear layer using adversarial training.

Motivated by the in-domain accuracy improvements obtained by distillation in prior works, we attempted to train a student model by distilling a teacher model that is trained by AF. However, we observed that standard distillation does not yield large improvements for AF trained teachers. We identified the reason for this to be the instability of logit values around the input and to address this, we incorporated an additional loss term in AF to encourage the logit values of teacher to be smooth. Finally, to tackle the suboptimal in-domain accuracy of AF trained teacher, we distilled both standard trained and AF trained teachers into the student giving our final algorithm DAFT.

On five benchmark datasets, with diverse kinds of distribution shifts, we showed that DAFT provides significantly higher OOD accuracy when compared to ERM as well as standard baselines like adversarial training. We also presented ablation studies showing the importance of various components of DAFT.

Limitations & Future work: An avenue for future work is devising the optimal way of performing AF – in this work, we only consider finetuning the final linear layer but have not explored if this can be further improved by finetuning last few layers or other subsets of parameters. Finally, models that are fragile to OOD shifts and depend on spurious correlation can significantly amplify biases in data. So, further investigation of DAFT for mitigating biases in data is highly interesting.

References

Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition, 2015.

Hendrycks, D. and Dietterich, T. G. Benchmarking neural network robustness to common corruptions and perturbations. *CoRR*, abs/1903.12261, 2019. URL http://arxiv.org/abs/1903.12261.

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. *arXiv preprint arXiv:1905.02175*, 2019.

Kirichenko, P., Izmailov, P., and Wilson, A. G. Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*, 2022.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pp. 5637–5664. PMLR, 2021.

Kumar, A., Raghunathan, A., Jones, R., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks, 2019.

Rosenfeld, E., Ravikumar, P., and Risteski, A. Domain-adjusted regression or: Erm may already learn features sufficient for out-of-distribution generalization. *arXiv preprint arXiv:2202.06856*, 2022.

Vedantam, R., Lopez-Paz, D., and Schwab, D. J. An empirical investigation of domain generalization with empirical risk minimizers. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 28131–28143. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/file/ecf9902e0f61677c8de25ae60b654669-Paper.pdf.

Yi, M., Hou, L., Sun, J., Shang, L., Jiang, X., Liu, Q., and Ma, Z.-M. Improved ood generalization via adversarial training and pre-training. *arXiv preprint arXiv:2105.11144*, 2021.

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy, 2019.

A. Experimental Setup

A.1. Hyperparameters

We use the Adam optimizer for all our experiments, with a batch size of 64. We run ERM, adversarial training, and distillation for 10000 steps each, while fine-tuning is run for 5000 steps. We tune the following hyper-parameters for our methods and baselines -

- Learning Rate - Selected from the range $[10^{-6}, 10^{-3}]$
- Norm of adversarial perturbation ϵ - Selected from the range $[0.05, 0.5]$
- Number of adversarial perturbation steps k - An integer selected from the range $[3, 7]$
- LR for PGD - Selected from the range $[10^{-3}, 10^{-1}]$
- Distillation temperature τ - An integer selected from the range $[2, 8]$
- Weight α for $\mathcal{L}_{smooth}$ - Selected from the range $[10^{-6}, 10^{-2}]$

The hyperparameters were tuned using random search over the intervals, with 32 configurations being considered for each algorithm.

A.2. Datasets

The data can be downloaded using the DomainBed repo

A.3. Hardware Setup

We conducted all experiments on a single A100 GPU. The experimental code was using the DomainBed framework, in PyTorch.

B. Additional results

B.1. Results on WILDS benchmark

We perform experiments on the WILDS benchmark (Koh et al., 2021) with non-ImageNet pre-trained models to verify the efficacy of our approach in settings where pre-training on large datasets is not possible. We present results on iWildCams and FMoW-WILDS datasets in tables **??**. We notice that DAFT consistently gives gains over most of the baselines.

Table 3. iWildCam dataset: OOD accuracy for different student architectures. RN refers to the ResNet architecture family. For all the considered RN models, DAFT is significantly more accurate than ERM and DIST, while vanilla ADV/TRADES training leads to worse OOD accuracy than ERM.

Model	ERM	Dist	Adv	Trades	DAFT
RN-200	56.1 ± 0.4	56.7 ± 0.4	51.2 ± 1.2	53.0 ± 1.2	**58.7 ± 0.6**
RN-101	50.9 ± 0.3	55.8 ± 1.5	47.6 ± 1.0	50.2 ± 0.8	**58.1 ± 0.4**
RN-50	49.1 ± 0.5	55.4 ± 1.1	45.4 ± 1.1	47.9 ± 1.7	**57.2 ± 0.5**
RN-34	47.6 ± 0.9	52.5 ± 0.9	42.3 ± 1.7	45.5 ± 1.6	**55.2 ± 1.5**
RN-18	44.9 ± 0.6	50.7 ± 1.3	41.2 ± 1.8	42.7 ± 1.1	**54.3 ± 1.3**

Table 4. FMoW-WILDS dataset. OOD Accuracy with different ResNet (RN) architectures and training methods.

Model	ERM	Dist	Adv	Trades	DAFT
RN-200	54.1 ± 0.2	54.9 ± 0.2	53.0 ± 0.2	50.6 ± 0.3	**55.5 ± 0.1**
RN-101	50.7 ± 0.4	54.8 ± 0.3	43.9 ± 0.4	42.2 ± 0.6	**55.2 ± 0.2**
RN-50	41.8 ± 0.3	**51.3 ± 0.3**	41.3 ± 0.6	41.6 ± 0.5	**51.2 ± 0.2**
RN-34	47.9 ± 0.4	**53.7 ± 0.3**	42.9 ± 0.8	42.6 ± 0.4	**53.9 ± 0.2**
RN-18	40.1 ± 0.4	**49.5 ± 0.2**	38.4 ± 0.6	37.7 ± 0.1	**49.3 ± 0.3**

C. Verifying our Design Choices

C.1. Perturbing features instead of the input

We also experiment with a variant of adversarial fine-tuning where we perform perturbations in the feature space of the model rather than the input space. The comparison with adversarial finetuning on four datasets is reported in table 5. We notice that the difference in the performance obtained is not consistent across datasets or sizes. While the performance of finetuning in feature space is slightly more for ImageNet pre-trained models, we notice that the range over which ϵ needs to be fine-tuned is larger for this variant, and the obtained best ϵ differs quite a bit between different models. On the contrary, for input space perturbations, using the same ϵ across models does not degrade performance noticeably. Note that the last column contains results using non-ImageNet pretrained models, where perturbing in the input space seems to have an edge.

Model Size	Method	PACS	VLCS	OfficeHome	FMoW
ResNet-101	AF	86.4	77.9	69.6	51.8
	AFLast	87.0	77.1	70.3	49.6
ResNet-152	AF	88.3	80.4	70.9	55.0
	AFLast	88.9	80.1	71.1	54.7

Table 5. Comparison between input perturbations and perturbations in the feature space.

C.2. Fine-tuning multiple layers

In table 6, we fine-tune the last three layers instead of just the final layer. We find that this leads to slightly improved performance on two datasets, while slightly degrading performance on one. The optimal parameters to fine-tune remains an open question.

Model Size	Method	PACS	VLCS	OfficeHome
ResNet-101	AF	86.4	77.9	69.6
	AFMulti	87.1	77.7	70.5

Table 6. Comparison between input perturbations and perturbations in the feature space.

C.3. Do adversarial perturbations lead to unstable logits?

In order to verify the effect of adversarial finetuning with and without the $\mathcal{L}_{smooth}$ loss, we compute the logits of an ERM trained model, an adversarially finetuned model and a KL-regularized finetuned model on the ”Clipart” split of the OfficeHome dataset. The training data were the ”Real”, ”Product” and ”Painting” splits of the dataset. We find that the mean rank correlation of the KL-regularized model with the ERM trained model is higher (0.52 v/s 0.49). In table 7, we list the average precision@k for k between 1-5 of the logits with the ERM model. Here prec@k is defined as $|topk\_predictions(model) \cap topk\_predictions(ERMModel)|/k$. As we can see, $\mathcal{L}_{smooth}$ encourages the order of logits to be maintained, while having a similar target accuracy as the AF model.

Method	prec@1	prec@2	prec@3	prec@4	prec@5	MAP
AF with $\mathcal{L}_{smooth}$	94.1%	63.1%	53.2%	49.7%	48.8%	72.6%
AF	94.1%	59.3%	50.1%	48.1%	48.0%	69.4%

Table 7. Comparison between teachers trained with and without $\mathcal{L}_{smooth}$. We show the overlap in the order of the predictions here, and note that the OOD predictions of smooth teacher are better aligned with the ERM teacher.

C.4. Additional results on Colored-FashionMNIST

In fig 3, we show examples of images from the FashionMNIST dataset, as well as the l_2-norm constrained adversarial perturbations. We find that the perturbations mainly change the colour of the images. For each feature, we compute the relative average perturbation (RAP, i.e. $\mathbb{E}[\frac{|f_i(x+\delta)-f_i(x)|}{|f_i(x)|}]$, where δ is the adversarial perturbation, and f_i denotes the i^{th} feature) when the input is perturbed adversarially. We call this RAP-input. We also compute the relative average perturbation when the adversarial perturbations are in the *feature* space, denoted as RAP-feature. We notice that the maximum perturbation for colour features is much more than that of shape features (11x v/s 0.4x). This is expected since the adversarially perturbations only change the colour of the image. Further, we also notice that there is a high correlation between RAP-input and RAP-feature. In fact, RAP-feature also follows a similar trend, with the maximum RAP-feature for shape features being 0.3x, while the maximum RAP-feature for colour features is 13x. This means that fine-tuning the last layer with either feature perturbations or input perturbations would lead the model to similar classification weights.

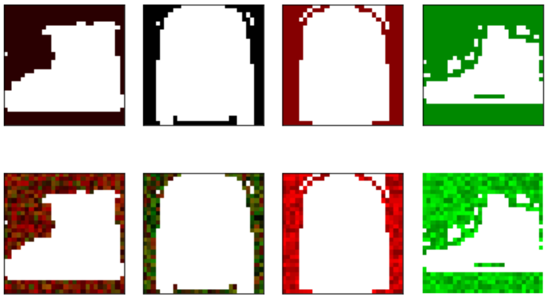

Figure 3. Sample images and their adversarially perturbed versions from Colored-FashionMNIST dataset. We notice that the color of the image is perturbed, while the shape remains constant.