APPENDIX

A ADDITIONAL RELATED WORK

For a comprehensive overview of transfer learning, please see the surveys of Zhuang et al. and Pan & Yang. Here, we discuss a few directly works directly relevant to our own.

Recently, Kumar et al. demonstrated that learning probing prior to fine-tuning (e.g., LP+FT) can improve both in-distribution and out-of-distribution performance when transferring to a downstream task given a highly expressive, pretrained model. They demonstrated that FT only modifies features in the ID representation subspace and not in other directions, which can lead higher OOD error as direction outside the ID subspace are necessary for OOD generalization. However, by initializing FTwith a trained linear probe, feature distortion can be decreased since this initialization is closer to optimal model, and thus requires less distortion in ID subspace, preserving the expressiveness of the original model. Concurrently, Kirichenko et al. demonstrated that models are able to learn both core features and spurious features. However, classifiers can rely upon spurious features, harming performance on minority groups. To reduce the reliance on spurious features, they propose to retrain the classifier on a small amount of "re-weighting" data, that allows the model to leverage the core features instead of the spurious features.

Other modifications and heuristics have also been proposed to improve fine-tuning, including side-tuning (Zhang et al., 2019), which tunes a small secondary network that is then combined with the original model, using larger/smaller learning rates for the classifier, as well as regularization-based methods (Jiang et al., 2020). We focus on the LP+FT protocol, as it is principled and achieves strong OOD performance.

Additionally, several works have studied properties of the model that influence the effectiveness of transfer learning (Azizpour et al., 2016; Huh et al., 2016; Kornblith et al., 2019; Lee et al., 2023a; Evci et al., 2022; Lee et al., 2023b; Izmailov et al., 2022; Lubana et al., 2023; Rame et al., 2022), including the robustness of pretrained features (Salman et al., 2020; Utrera et al., 2021). While the connection between adversarial training and improved feature representations (Allen-Zhu & Li, 2021; Kaur et al., 2019) has been studied, we use virtual adversarial training during LP to learn a better classifier that is less reliant upon simple features, and we do not use an adversarially trained feature extractor. Finally, we note that while we are, to the best of our knowledge, the first to consider this holistic evaluation of safety and generalization in the context of transfer learning with highly expressive pretrained models, Hendrycks et al. have considered the trade-offs induced by different data augmentation strategies (Yun et al., 2019; Devries & Taylor, 2017; Hendrycks et al., 2020; Cubuk et al., 2019; 2020) on safety metrics in supervised learning. We emphasize that while our evaluation is similar, that our work focuses on a different context and contains an additional layer of complexity as we consider the interaction between adaptation protocols, generalization behavior and safety performance.

B EXPERIMENTAL DETAILS

Please see the https://github.com/pujacomputes/23-ICLR-Adaptation.git for training details. In brief, we performed grid-search to find the best parameters, which are as follows. For CIFAR-10 and CIFAR-100, we train only the classifier for 200 epochs with LR=30 during LP. For FT, the entire model is trained for 20 epochs with LR=1e-5. For LP+FT, the model's classifier is initialized with the solution found by LP, and then it is fine-tuned for 20 epochs. A grid-search was conducted to determine the LR for LP and FT. For Domain-Net Experiments, we use 200 epochs with LR=30 during LP. For FT, the entire model is trained for 20 epochs with LR=3e-4. For LP+FT, the model's classifier is initialized with the solution found by LP, and then it is fine-tuned for 20 epochs, using LR=3e-7. Furthermore, following Kumar et al., we freeze the batchnorm layers during LP+FT. A CLIP (Radford et al., 2021) pretrained ResNet-50 is used for the DomainNet experiments, while a MoCoV2 (He et al., 2020) is used for all CIFAR experiments. We use augmentation functions from timm (Wightman, 2019) and compute CKA scores using the packaged provided by torch-cka. When using augmented protocols, the same LRs are used. Note, all results were obtained by averaging over 3 seeds. We consider model soups of sizes 5,10,20, tune ϵ in 0.005, 0.01, 0.02 and 0.1 for UDP, and

α in 0.001, 0.01, 0.1 for VAT. For CIFAR-MNIST results, LP is done for 100 epochs, and FT is done for 20 epochs.

B.1 Motivation for Hardness-Promoting Variants

We selected UDP (Pagliardini et al., 2022), VAT (Miyato et al., 2017), and model-soups (Wortsman et al., 2022) as simplicity bias mitigation strategies due to their effectiveness and ease of use. We emphasize, however, that our findings are not specific to the choice of a given mitigation strategy and we expect that advancements in such strategies will further improve the effectiveness of our proposed LP+FTvariants. At present, the selected strategies are strong, representative mitigations that we have confirmed are effective at mitigating simplicity bias in the adaptation context using the synthetic dominoes dataset in Sec. 4.

We conceptually justify each strategy here:

- UDP is designed to help mitigate simplicity bias by learning by a large margin classifier, opposed to a narrow margin classifier that relies upon simple features. As noted by Shah et al. (2020), such narrow margin classifiers are sensitive to small perturbations and the simple features supporting the decision boundary may not be discriminative under distribution shifts. By maximizing uncertainty (instead of loss) to create adversarial perturbations, UDP is able to learn a maximum-margin classifier that is better able to handle such shifts. Notably, to create such a maximum-margin classifier, the model will necessarily learn more complex features;
- We use virtual adversarial training (VAT) to help avoid reliance upon simple features, as VAT enforces distribution smoothness so that classifiers become robust in some epsilon neighborhood around the input. We note that we are performing this training in the hidden representation space, so perturbations correspond may be altering high-level semantics. To maintain strong performance under such high-level perturbations, the model should learn to rely upon more complex features, and learn a better margin classifier;
- We use model-soups so that we may learn a set of classifiers that rely upon disjoint sets of features. By learning a set of diverse classifiers, we are able to average classifiers that have learned to rely upon different features, instead of becoming overly reliant upon a single simple feature. In future work, we intend to build a theoretical framework that helps us better justify these interventions and create new ones.

B.2 Applying Simplicity Bias Mitigation Strategies to Fine-Tuning Step.

To demonstrate that simplicity bias mitigation strategies must be applied during the LP step of FT for maximum effectiveness, we conduct the following additional experiment.

Setup. We evaluate two additional protocols where VAT and UDP are applied only during the FT step, (LP+FT(VAT), and LP+FT(UDP)), on the synthetic dominoes dataset. We plot the results for Randomized OOD Accuracy in Fig. 6.

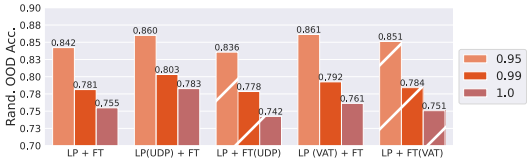

Figure 6: **Applying Mitigation Strategies to FT.** We create FT variants of our LP mitigation strategies and evaluate them on the synthetic dominoes dataset. We see that FT variants lose performance with respect to LP variants, indicating that interventions must be undertaken during the LP step as originally proposed.

Results. Here, we see that, across three different correlation ratios, FT variants lose performance with respect to the LP mitigation variants. Notably, LP+ FT (UDP) loses up to 4% performance with respect to LP(UDP)+ FT. While performance drops are not as large for VAT, we nonetheless see that LP+ FT(VAT) loses performance with respect to LP(VAT)+ FT.

Our results in Fig. 6 support our conceptual argument that mitigation strategies must be undertaken during the LP step to ensure that subsequent FT is in a direction that preserves complex features; applying mitigation strategies during FT may be too late to avoid simplicity bias. We note that

applying mitigation strategies during FT, in addition to LP, may further improve performance, and we will add these variants in the final version. We did not include a FT soup variant as it would be prohibitively expensive to train and average large soups of entire models (instead of classifiers). This highlights the computational efficiency of implementing mitigation strategies in the LP step itself.

C ADDITIONAL RESULTS

Below, we include results corresponding to different hyperparameters (number of souped classifiers, α for vat, and δ for udp).

	Generalization		*Robustness*			*Calibration*				*Anomaly Detection*	*Rep. Similarity*
Protocol	ID Acc.	OOD Acc.	C Acc.	$\overline{\text{C}}$ Acc.	Adv. Acc.	ID 1-RMS	C 1-RMS	$\overline{\text{C}}$ 1-RMS	OOD. 1-RMS	Out-of-Class AUROC	ID CKA
LP	0.9138	0.8190	0.6912	0.6553	0.0003	0.9595	0.8303	0.8142	0.8696	0.6206	1.0000
LP+ soup-5	0.9108	0.8348	0.7007	0.6678	0.0002	0.9748	0.8943	0.8835	0.9108	0.8463	1.0000
LP+ soup-10	0.9129	0.8359	0.6985	0.6652	0.0003	0.9669	0.9104	0.8956	0.9205	0.8713	1.0000
LP+ soup-20	0.9052	0.8353	0.6917	0.6588	0.0003	0.9605	0.9205	0.9037	0.9364	0.8859	1.0000
LP+ udp-0.005	0.9129	0.8332	0.7015	0.6702	0.0003	0.9729	0.8879	0.8817	0.9017	0.8708	1.0000
LP+ udp-0.01	0.9033	0.8356	0.6948	0.6643	0.0003	0.9689	0.9111	0.9023	0.9277	0.9033	1.0000
LP+ udp-0.02	0.8885	0.8281	0.6796	0.6492	0.0004	0.9655	0.9259	0.9142	0.9473	0.9217	1.0000
LP+ udp-0.1	0.8573	0.8005	0.6290	0.6064	0.0007	0.9245	0.9235	0.9143	0.9531	0.8570	1.0000
LP+ vat-0.001	0.9189	0.8276	0.6945	0.6606	0.0006	0.9714	0.8564	0.8442	0.8927	0.7159	1.0000
LP+ vat-0.01	0.8977	0.8251	0.6742	0.6483	0.0002	0.9265	0.9255	0.9139	0.9375	0.7200	1.0000
FT	0.9539	0.8754	0.7434	0.7553	0.0231	0.9668	0.8364	0.8453	0.9232	1.0000	0.6831
LP+FT	0.9442	0.8678	0.6921	0.6790	0.0018	0.9521	0.7849	0.7721	0.8864	0.6511	0.7853
(LP+soup-5) +FT	0.9466	0.8832	0.6997	0.6861	0.0001	0.9639	0.8197	0.8051	0.9155	0.9020	0.7603
(LP+soup-10) +FT	0.9467	0.8857	0.7022	0.6907	0.0001	0.9660	0.8307	0.8182	0.9184	0.9161	0.7671
(LP+soup-20) +FT	0.9466	0.8892	0.7031	0.6931	0.0001	0.9678	0.8390	0.8287	0.9216	0.9265	0.7806
(LP+udp-0.005) +FT	0.9458	0.8864	0.6962	0.6893	0.0005	0.9643	0.8127	0.8110	0.9119	0.9180	0.7742
(LP+udp-0.01) +FT	0.9450	0.8869	0.7048	0.6977	0.0004	0.9642	0.8335	0.8311	0.9209	0.9419	0.7746
(LP+udp-0.02) +FT	0.9440	0.8848	0.7028	0.6986	0.0004	0.9670	0.8472	0.8476	0.9237	0.9559	0.7764
(LP+udp-0.1) + FT	0.9435	0.8836	0.6959	0.6952	0.0000	0.9676	0.8449	0.8525	0.9355	0.9651	0.7382
(LP+vat)+FT	0.9611	0.8900	0.7442	0.7321	0.0027	0.9294	0.8355	0.8281	0.9178	0.8276	0.7839

Table 6: **CIFAR10, Hardness-Promoting Augmentations.**

	Generalization		*Robustness*			*Calibration*				*Anomaly Detection*	*Rep. Similarity*
Protocol	ID Acc.	OOD Acc.	C Acc.	$\overline{\text{C}}$ Acc.	Adv. Acc.	ID 1-RMS	C 1-RMS	$\overline{\text{C}}$ 1-RMS	OOD. 1-RMS	Out-of-Class AUROC	ID CKA
LP	0.9521	0.8124	0.7010	0.7378	0.2350	0.9313	0.8693	0.8802	0.9117	0.9907	1.0000
LP+ udp-0.005	0.9524	0.8114	0.7012	0.7379	0.2337	0.9304	0.8699	0.8806	0.9108	0.9907	1.000
LP+ udp-0.01	0.9524	0.8110	0.7017	0.7382	0.2353	0.9308	0.8691	0.8801	0.9118	0.9908	1.000
LP+ udp-0.02	0.9500	0.8126	0.7036	0.7387	0.2373	0.9343	0.8621	0.8763	0.9135	0.9913	1.000
LP+ udp-0.1	0.9459	0.8165	0.6840	0.7220	0.2339	0.9032	0.8243	0.8427	0.8990	0.9882	1.000
LP+ soup-5	0.9439	0.7996	0.6874	0.7290	0.2451	0.8806	0.7868	0.8094	0.9064	0.9897	1.0000
LP+ soup-10	0.9373	0.7904	0.6767	0.7220	0.2547	0.8496	0.7478	0.7709	0.8841	0.9887	1.0000
LP+ soup-20	0.9298	0.7841	0.6601	0.7082	0.2575	0.8056	0.7084	0.7305	0.8274	0.9867	1.0000
LP+ vat-0.001	0.9524	0.8122	0.7010	0.7379	0.2345	0.9299	0.8682	0.8791	0.9103	0.9907	1.0000
FT	0.9518	0.7168	0.7011	0.7164	0.1563	0.8873	0.9019	0.8604	0.9295	0.9794	0.7847
LP+FT	0.9643	0.8261	0.7426	0.7671	0.2135	0.9782	0.9472	0.9451	0.8742	0.9924	0.9887
(LP+udp-0.005) +FT	0.9627	0.8243	0.7434	0.7666	0.2153	0.9811	0.9456	0.9445	0.8736	0.9922	0.98950
(LP+udp-0.01) +FT	0.9627	0.8253	0.7436	0.7669	0.2133	0.9812	0.9454	0.9447	0.8737	0.9923	0.98957
(LP+udp-0.02) +FT	0.9637	0.8265	0.7448	0.7681	0.2157	0.9768	0.9464	0.9467	0.8757	0.9927	0.98927
(LP+udp-0.1) +FT	0.9614	0.8249	0.7499	0.7689	0.2165	0.9808	0.9441	0.9420	0.8711	0.9912	0.9861
(LP+soup-5) + FT	0.9608	0.8163	0.7456	0.7684	0.1855	0.9760	0.9498	0.9492	0.8678	0.9936	0.98540
(LP+soup-10) + FT	0.9580	0.8114	0.7445	0.7678	0.1753	0.9838	0.9503	0.9488	0.8748	0.9938	0.98360
(LP+soup-20) + FT	0.9594	0.8165	0.7450	0.7684	0.1782	0.9893	0.9503	0.9490	0.8609	0.9936	0.98190
(LP+vat-0.001) +FT	0.9647	0.8247	0.7425	0.7650	0.2224	0.9727	0.9521	0.9463	0.8775	0.9925	0.9370

Table 7: **Living17, Hardness-Promoting Augmentations**

Protocol	*Generalization* ID Acc.	*Generalization* OOD Acc.	*Robustness* Sketch-C Acc.	*Robustness* Real-C Acc.	*Robustness* Adv. Acc.	*Calibration* ID 1-RMS	*Calibration* Sketch-C 1-RMS	*Calibration* Real-C 1-RMS	*Calibration* OOD. 1-RMS	*Anomaly Detection* Out-of-Class AUROC	*Rep. Similarity* ID CKA
LP	0.8913	0.8013	0.6019	0.6020	0.1768	0.9638	0.9264	0.9045	0.9014	0.8679	1.0000
LP+augmix	0.8897	0.7998	0.6336	0.6104	0.1872	0.9718	0.9230	0.9263	0.9083	0.8818	1.0000
LP+autoaug	0.8944	0.8057	0.6419	0.6257	0.1857	0.9614	0.9357	0.9309	0.9022	0.8849	1.0000
LP+randaug	0.8971	0.8090	0.6392	0.6232	0.1877	0.9559	0.9321	0.9312	0.9036	0.8875	1.0000
LP+vat	0.8836	0.7914	0.5893	0.5963	0.1687	0.8897	0.9552	0.8905	0.9178	0.8735	1.0000
FT	0.7613	0.4522	0.5186	0.2744	0.4164	0.8368	0.6379	0.7234	0.5597	0.8841	0.6092
FT+augmix	0.8246	0.5233	0.5911	0.3408	0.4802	0.9308	0.8042	0.8665	0.6761	0.9255	0.5272
FT+autoaug	0.7786	0.5161	0.5561	0.3160	0.4313	0.9157	0.7485	0.8246	0.7324	0.9231	0.7025
FT+randaug	0.7823	0.5370	0.5610	0.3298	0.4551	0.9160	0.7970	0.8682	0.7444	0.9318	0.6477
LP+FT	0.8985	0.7990	0.6343	0.5979	0.1927	0.9566	0.8899	0.8445	0.8024	0.9022	0.9222
LP+(FT+augmix)	0.9047	0.8081	0.6673	0.5980	0.2597	0.9768	0.9200	0.9067	0.8443	0.9155	0.8811
LP+(FT+autoaug)	0.9023	0.8028	0.6571	0.5851	0.2354	0.9830	0.9249	0.8990	0.8484	0.9034	0.9096
LP+(FT+randaug)	0.9054	0.8099	0.6703	0.6152	0.2489	0.9786	0.9194	0.9044	0.8598	0.9252	0.9000
(LP+vat) +FT	0.9048	0.8009	0.6466	0.6131	0.1942	0.9686	0.8911	0.8428	0.7985	0.9204	0.9370
(LP+vat) +(FT+augmix)	0.9032	0.8024	0.6589	0.5896	0.2525	0.9769	0.9169	0.8929	0.8384	0.9212	0.8673
(LP+vat)+(FT+autoaug)	0.9003	0.8049	0.6600	0.5862	0.2331	0.9783	0.9178	0.9000	0.8381	0.9149	0.9244
(LP+vat)+(FT+randaug)	0.9006	0.8060	0.6651	0.5894	0.2622	0.9762	0.9197	0.8993	0.8414	0.9238	0.8956

Table 8: **DomainNet, Diversity Promoting Augmentations and Generalization Trade-offs.**

Protocol	*Generalization* ID Acc.	*Generalization* OOD Acc.	*Robustness* C Acc.	*Robustness* $\overline{C}$ Acc.	*Robustness* Adv. Acc.	*Calibration* ID 1-RMS	*Calibration* C 1-RMS	*Calibration* $\overline{C}$ 1-RMS	*Calibration* OOD. 1-RMS	*Anomaly Det.* Out-of-Class AUROC	*Rep. Similarity* ID CKA
LP	0.9297	0.9083	0.8532	0.7491	0.7077	0.9794	0.9006	0.9007	0.9301	0.9623	0.0668
LP+ soup-5	0.9220	0.9151	0.8315	0.7432	0.7050	0.9598	0.9232	0.9279	0.9623	0.9665	0.1399
LP+ soup-10	0.9156	0.9135	0.8183	0.7344	0.6985	0.9476	0.9221	0.9271	0.9732	0.9602	0.1778
LP+ soup-20	0.9069	0.9064	0.8065	0.7216	0.6885	0.9279	0.9129	0.9191	0.9714	0.9484	0.2617
LP+ udp-0.005	0.9299	0.9092	0.8533	0.7494	0.7079	0.9794	0.9009	0.9003	0.9312	0.9614	0.0822
LP+ udp-0.01	0.9298	0.9097	0.8535	0.7495	0.7083	0.9795	0.9007	0.9006	0.9316	0.9616	0.0880
LP+ udp-0.02	0.9294	0.9108	0.8538	0.7497	0.7088	0.9789	0.9012	0.9014	0.9335	0.9631	0.1017
LP+ udp-0.1	0.9238	0.9218	0.8377	0.7488	0.7111	0.9801	0.9154	0.9216	0.9517	0.9645	0.1478
LP+ vat-0.001	0.9298	0.9091	0.8533	0.7493	0.7078	0.9801	0.9014	0.9012	0.9325	0.9614	0.0784
LP+ vat-0.01	0.9295	0.9094	0.8531	0.7494	0.7080	0.9800	0.9039	0.9040	0.9342	0.9632	0.0837
LP+ vat-0.1	0.9275	0.9106	0.8493	0.7481	0.7087	0.9581	0.9191	0.9246	0.9589	0.9598	0.1528
FT	0.9724	0.8761	0.9218	0.8131	0.8074	0.9577	0.8429	0.8418	0.8855	0.9138	0.9317
LP+FT	0.9692	0.9387	0.9195	0.8106	0.7736	0.9451	0.8034	0.7743	0.9026	0.8949	0.5349
(LP+soup-5) +FT	0.9685	0.9417	0.9210	0.8136	0.7787	0.9385	0.8079	0.7765	0.9102	0.8974	0.5315
(LP+soup-10) +FT	0.9681	0.9411	0.9220	0.8178	0.7824	0.9382	0.8119	0.7796	0.9072	0.8933	0.5521
(LP+soup-20) +FT	0.9677	0.9395	0.9213	0.8164	0.7837	0.9385	0.8107	0.7817	0.9070	0.8964	0.5411
(LP+udp-0.005) +FT	0.9677	0.9297	0.9142	0.8104	0.7710	0.9422	0.8024	0.7718	0.8942	0.8916	0.6428
(LP+udp-0.01) +FT	0.9677	0.9359	0.9195	0.8098	0.7721	0.9417	0.8029	0.7732	0.9019	0.8999	0.4239
(LP+udp-0.02) +FT	0.9687	0.9349	0.9195	0.8136	0.7724	0.9437	0.8067	0.7736	0.8994	0.8981	0.5015
(LP+udp-0.1) +FT	0.9688	0.9423	0.9242	0.8174	0.7811	0.9408	0.8130	0.7815	0.9072	0.9064	0.4496
(LP+vat-0.001)+FT	0.9681	0.9366	0.9180	0.8111	0.7727	0.9422	0.8033	0.7732	0.9013	0.8962	0.5904
(LP+vat-0.01)+FT	0.9689	0.9366	0.9168	0.8121	0.7766	0.9455	0.8062	0.7791	0.9013	0.8918	0.5687
(LP+vat-0.1)+FT	0.9692	0.9402	0.9207	0.8127	0.7743	0.9420	0.8068	0.7734	0.9083	0.8978	0.4398

Table 9: **CIFAR10 with Resnet101/SimCLR Pretrained Model.** We see that with a larger model, and different pretraining method, our proposed variants still have some benefits. We note that the baseline performance is also improved as a result of a more larger pretrained model.

ACKNOWLEDGMENTS

We thank Ekdeep Singh Lubana for several helpful discussions during the course of this project. This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344, Lawrence Livermore National Security, LLC.and was supported by the LLNL-LDRD Program under Project No. 21-ERD-012. It was also partially supported by the National Science Foundation under CAREER Grant No. IIS 1845491. PT began this work as an intern at Lawrence Livermore National Laboratory.

REFERENCES

Zeyuan Allen-Zhu and Yuanzhi Li. Feature purification: How adversarial training performs robust deep learning. In *Proc. Symposium on Foundations of Computer Science, FOCS*, 2021.

Anders Andreassen, Yasaman Bahri, Behnam Neyshabur, and Rebecca Roelofs. The evolution of out-of-distribution robustness throughout fine-tuning, 2021.

Hossein Azizpour, Ali Sharif Razavian, Josephine Sullivan, Atsuto Maki, and Stefan Carlsson. Factors of transferability for a generic convnet representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(9):1790–1802, 2016.

Alon Brutzkus, Amir Globerson, Eran Malach, and Shai Shalev-Shwartz. SGD learns over-parameterized networks that provably generalize on linearly separable data. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2017.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proc. Int. Conf. on Computer Vision (ICCV)*, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2020.

Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform resnets without pretraining or strong data augmentations. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2022.

Ekin Dogus Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

Mohammad Reza Davari, Stefan Horoi, Amine Natik, Guillaume Lajoie, Guy Wolf, and Eugene Belilovsky. Reliability of CKA as a similarity measure in deep learning. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2023.

Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017.

Frances Ding, Jean-Stanislas Denain, and Jacob Steinhardt. Grounding representation similarity with statistical testing. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2021.

Linus Ericsson, Henry Gouk, and Timothy M. Hospedales. How well do self-supervised models transfer? In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Utku Evci, Vincent Dumoulin, Hugo Larochelle, and Michael C Mozer. Head2Toe: Utilizing intermediate representations for better transfer learning. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2022.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2019.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

Suriya Gunasekar, Jason D. Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2018.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proc. of the Int. Conf. on Machine Learning, (ICML)*, 2017.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *Proc. Int. Conf. on Learning Representations, (ICLR)*, 2019.

Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2019.

Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2020.

Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ML safety. *CoRR*, abs/2109.13916, 2021.

Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Bo Li, Dawn Song, and Jacob Steinhardt. Pixmix: Dreamlike pictures comprehensively improve safety measures. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Katherine L. Hermann, Ting Chen, and Simon Kornblith. The origins and prevalence of texture bias in convolutional neural networks. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

Mi-Young Huh, Pulkit Agrawal, and Alexei A. Efros. What makes imagenet good for transfer learning? *CoRR*, abs/1608.08614, 2016.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2019.

Pavel Izmailov, Polina Kirichenko, Nate Gruver, and Andrew Gordon Wilson. On feature learning in the presence of spurious correlations. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2022.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. SMART: robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. In *Proc. Assn. for Computational Linguistics, ACL*, 2020.

Simran Kaur, Jeremy M. Cohen, and Zachary C. Lipton. Are perceptually-aligned gradients a general property of robust classifiers? *CoRR*, abs/1910.08640, 2019.

Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. *CoRR*, abs/2204.02937, 2022.

Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2022.

Yoonho Lee, Annie S Chen, Fahim Tajwar, Ananya Kumar, Huaxiu Yao, Percy Liang, and Chelsea Finn. Surgical fine-tuning improves adaptation to distribution shifts. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2023a.

Yoonho Lee, Huaxiu Yao, and Chelsea Finn. Diversify and disambiguate: Out-of-distribution robustness via disagreement. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2023b.

Hong Liu, Jeff Z. HaoChen, Adrien Gaidon, and Tengyu Ma. Self-supervised learning is more robust to dataset imbalance. *CoRR*, abs/2110.05025, 2021.

Ekdeep Singh Lubana, Eric J. Bigelow, Robert P. Dick, David Krueger, and Hidenori Tanaka. Mechanistic mode connectivity, 2023.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2018.

Eric Mintun, Alexander Kirillov, and Saining Xie. On interaction between augmentations and corruptions in natural corruption robustness. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2021.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

Thao Nguyen, Maithra Raghu, and Simon Kornblith. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2021.

Matteo Pagliardini, Gilberto Manunza, Martin Jaggi, Michael I. Jordan, and Tatjana Chavdarova. Improving generalization via uncertainty driven perturbations, 2022.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22 (10):1345–1359, 2010.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2021.

Alexandre Rame, Matthieu Kirchmeyer, Thibaud Rahier, Alain Rakotomamonjy, patrick gallinari, and Matthieu Cord. Diverse weight averaging for out-of-distribution generalization. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2022.

Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2020.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *J. Mach. Learn. Res.*, 19:70:1–70:57, 2018.

Andreas Peter Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *Transactions on Machine Learning Research*, 2022.

Damien Teney, Ehsan Abbasnejad, Simon Lucey, and Anton van den Hengel. Evading the simplicity bias: Training a diverse set of models discovers solutions with superior OOD generalization. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *Proc. Adv. in Neural Information Processing Systems (NeurIPS)*, 2021.

Francisco Utrera, Evan Kravitz, N. Benjamin Erichson, Rajiv Khanna, and Michael W. Mahoney. Adversarially-trained deep nets transfer better: Illustration on image classification. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 2021.

Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proc. Int. Conf. on Machine Learning (ICML)*, 2022.

I. Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. *CoRR*, abs/1905.00546, 2019.

Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proc. of Int. Conf. on Computer Vision, ICCV*, 2019.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proc. Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Jeffrey O. Zhang, Alexander Sax, Amir Zamir, Leonidas J. Guibas, and Jitendra Malik. Side-tuning: Network adaptation via additive side networks. In *Proc. Euro. Conf. on Computer Vision (ECCV)*, 2019.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proc. IEEE*, 109(1):43–76, 2021.