On the Use of Anchoring for Training Vision Models

Vivek Narayanaswamy
Lawrence Livermore National Laboratory
narayanaswam1@llnl.gov

Kowshik Thopalli
Lawrence Livermore National Laboratory
thopalli1@llnl.gov

Rushil Anirudh
Amazon
rushil15anirudh@gmail.com

Yamen Mubarka
Lawrence Livermore National Laboratory
mubarka1@llnl.gov

Wesam Sakla
Lawrence Livermore National Laboratory
sakla1@llnl.gov

Jayaraman J. Thiagarajan
Lawrence Livermore National Laboratory
jjthiagarajan@gmail.com

Abstract

Anchoring is a recent, architecture-agnostic principle for training deep neural networks that has been shown to significantly improve uncertainty estimation, calibration, and extrapolation capabilities. In this paper, we systematically explore anchoring as a general protocol for training vision models, providing fundamental insights into its training and inference processes and their implications for generalization and safety. Despite its promise, we identify a critical problem in anchored training that can lead to an increased risk of learning undesirable shortcuts, thereby limiting its generalization capabilities. To address this, we introduce a new anchored training protocol that employs a simple regularizer to mitigate this issue and significantly enhances generalization. We empirically evaluate our proposed approach across datasets and architectures of varying scales and complexities, demonstrating substantial performance gains in generalization and safety metrics compared to the standard training protocol. The open-source code is available at - https://software.llnl.gov/anchoring

1 Introduction

Anchoring [1] is a recent architecture-agnostic principle for training deep neural networks. It reparameterizes each input x into a tuple comprising a reference sample $\bar{\mathrm{r}}$ and the *residual* $\mathrm{d} = \mathrm{x} - \bar{\mathrm{r}}$, i.e., $[\bar{\mathrm{r}}, \mathrm{d}]$, $\bar{\mathrm{r}} \sim P_{\mathrm{r}}$ and $\mathrm{d} \sim P_{\Delta}$. Here, P_{r} and P_{Δ} denote the distributions of references and residuals respectively. The resulting tuple is then fed as input to a deep network instead of the original input x, by concatenating the tuple elements along the feature axis for vector-valued data or the channel axis for image data. Although the first layer of the network needs to be modified to accommodate twice the number of input dimensions (due to concatenation), the rest of the model architecture and optimization strategies remain the same as in standard training. This simple re-parameterization of the input forces the neural network to model the joint distribution $P_{(\mathrm{r},\Delta)}$ for predicting the target label y. Formally, the training objective can be written as:

$$\theta^* = \arg\min_{\theta} \; \frac{1}{|\mathcal{D}|} \sum_{(\mathrm{x},\mathrm{y}) \in \mathcal{D}} \mathop{\mathbb{E}}_{\bar{\mathrm{r}} \sim P_{\mathrm{r}}} \mathcal{L}\left[\mathrm{y}, \mathcal{F}_{\theta}\left(\texttt{concat}([\bar{\mathrm{r}}, \mathrm{x} - \bar{\mathrm{r}}])\right)\right], \tag{1}$$

where $\mathcal{L}(.)$ is a loss function such as cross-entropy, $\mathcal{D}$ is the training dataset and $\mathcal{F}$ is the underlying network parameterized by θ. In effect, for a given x and reference samples $\bar{\mathrm{r}}_1, \ldots, \bar{\mathrm{r}}_k$, anchoring

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-3024

ensures that $\mathcal{F}_\theta([\bar{\mathrm{r}}_1, \mathrm{d}_1]) = \cdots = \mathcal{F}_\theta([\bar{\mathrm{r}}_k, \mathrm{d}_k])$, where $\mathrm{d}_k = \mathrm{x} - \bar{\mathrm{r}}_k$. In other words, regardless of the choice of reference the model must arrive at the same prediction for an input. This principle has been shown to produce models with improved calibration and extrapolation properties [2, 3], and to facilitate accurate epistemic uncertainty estimation [1]. In this paper, we systematically explore the utility of anchoring as a generic protocol for building vision models and make a number of fundamental insights on its training and inferencing, applicability to different architecture families (conv-nets, transformers), and most importantly, the implications on model generalization and safety.

Our main contributions in this work can be summarized as follows:

A closer look into anchored training and inferencing: By studying the roles of reference set diversity and the inferencing protocol choice on the behavior of anchored models, we identify a critical limitation in current practice. More specifically, we find that conventional anchored training fails to effectively leverage the reference diversity, thus restricting its generalization capabilities, and that merely adopting sophisticated inference protocols [2] cannot circumvent this limitation.

A new anchored training protocol: We attribute the limited generalization power of anchored models to the increased risk of learning undesirable shortcuts, owing to insufficient sampling of $P_{(\mathrm{r},\Delta)}$ during training, particularly in cases of high reference diversity. To address this, we introduce a new training protocol for anchoring that relies on a novel reference-masking regularizer.

Benchmarking generalization and safety of anchored models: Since anchoring is architecture-agnostic, we benchmark it using a variety of conv-net/transformer architectures on CIFAR-10, CIFAR-100 and Imagenet-1K datasets. We demonstrate significant improvements in OOD generalization, calibration and anomaly resilience over standard training. We also show that, without incurring any additional training or inference overheads, anchoring is synergistic to existing training strategies (e.g., data augmentations, optimizers, schedulers).

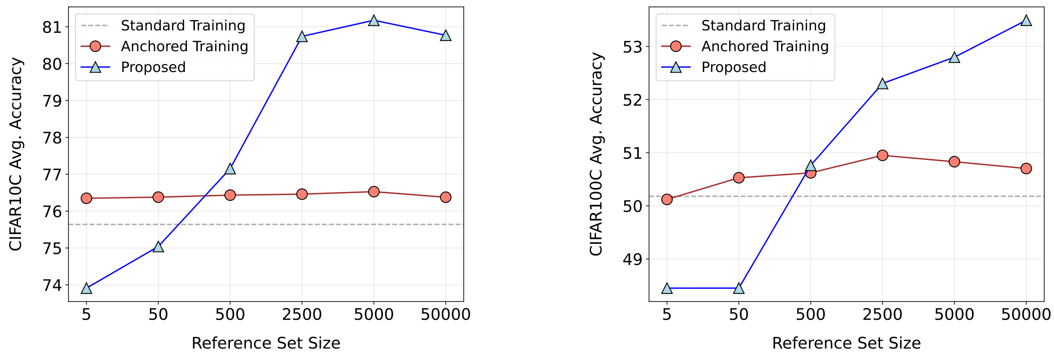
2 A Closer Look into Anchored Training and Inference

2.1 What makes anchoring a promising training protocol?

Anchored training forces the network to learn a mapping between the joint space of (reference, residuals) and the targets, rather than the original input-target pairs. At first glance, anchoring may seem like a trivial reposing of standard training, but it is conceptually very different. Through this reparameterization, anchoring creates different relative representations for a sample with respect to references drawn from P_r, and attempts to marginalize the effect of the reference when making a prediction for that sample. As demonstrated by [1], this process exploits the lack of shift invariance in the neural tangent kernel induced by deep networks [4], and implicitly explores a wider hypothesis class that is potentially more generalizable. Furthermore, anchored models have been found to extrapolate better to unseen data regimes through the use of transductive inferencing [2], i.e., identifying an optimal reference for each sample, such that the resulting residual is likely to have been exposed to the model during training. While anchoring offers promise, its success hinges on effectively leveraging the diversity of the reference-residual pairs and stably converging for the same protocols from standard training (e.g., architectures, data augmentations, optimizers etc.).

2.2 Does reference diversity play a key role in anchored training ?

A unique property of anchoring is its ability to utilize relative representations w.r.t. a reference distribution P_r (realized using a reference set $\mathcal{R}$), effectively operating in the joint space $P_{(\mathrm{r},\Delta)}$. During implementation, the reference set $\mathcal{R}$ is defined as a subset of the training data itself i.e., $\mathcal{R} \subseteq \mathcal{D}$ [1]. Intuitively, by controlling the construction of $\mathcal{R}$, one can control the diversity of reference-residual combinations that anchored training is exposed to. We hope that with exposure to increasingly large and diverse reference sets, anchoring will explore a wide range of hypotheses, while also ensuring that the model can make consistent predictions for test samples using any randomly drawn reference $\bar{\mathrm{r}} \in \mathcal{R}$. However, when the anchored training does not effectively characterize the joint distribution $P_{(\mathrm{r},\Delta)}$, the generalization can suffer, particularly when tested beyond the regimes of training data. To obtain a deeper understanding of anchored training, we conduct an empirical study on CIFAR10/100 datasets by varying the diversity of $\mathcal{R}$.



(a) (b)

Figure 1: **Impact of reference set size on anchored training performance**. With increase in reference set size, anchoring explores more diverse combinations of reference-residual pairs with the hope of demonstrating improved generalization performance. Surprisingly, the existing anchored training protocol does not effectively leverage this diversity even with increased reference set size albeit providing improvements in accuracy over standard training. We propose reference masking, a simple regularization strategy for training anchored models that recovers the lost performance.

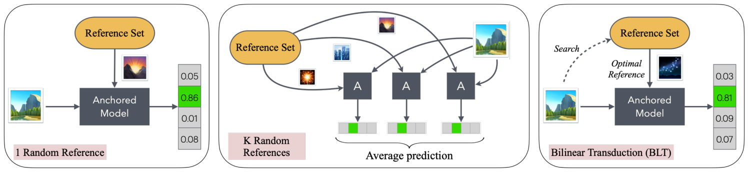
Setup. We first sub-sample $\mathcal{D}$ to construct reference sets of varying sizes ranging between 5 and 50000, where the latter corresponds to the entire training dataset. The construction is such that each set represents an increasing level of sample diversity (i.e., samples from multiple classes). This is followed by anchored training based on the different reference sets with ResNet18 models [5]. All other training specifics and hyper-parameters are fixed across the experiments. Post-training, we evaluate the model performance on the CIFAR10C/100C synthetic corruption benchmarks [6] and report the average corruption accuracy across 5 corruption severity levels.

Observations. Figure 1a and 1b illustrates the performance of CIFAR10/100 anchored training on the respective evaluation benchmarks. Interestingly, we observe that the anchoring performance remains fairly similar (minor improvements in accuracy) even with orders of magnitude growth in the reference set size. While anchoring provides consistent benefits over standard training ($0.5\% - 1\%$ on average), it is clear that the growing diversity of $P_{(\mathrm{r},\Delta)}$ is not fully leveraged. *This observation is in contrary to the insights from existing works, which recommend the use of the entire train data as the reference set for maximal benefits*. It is also worth noting that we utilize a single random reference (from the respective sets) to perform inference. This naturally raises the question if a more sophisticated inference protocol circumvent this limitation that we notice in anchored models.

2.3 Can the choice of inference protocol improve the performance of anchored models?

From existing works on anchoring, we find that different inference protocols can be used to elicit improvements in uncertainty quantification and model extrapolation. For instance, Thiagarajan *et al.* [1] employed a reference marginalization strategy that samples K random references from the reference set to obtain K independent predictions for a given input (similar to MC-dropout or deep ensembles). This is followed by computing the prediction average along with its standard deviation, wherein the latter was interpreted as an estimate of epistemic uncertainty. The intuition is that different reference-residual combinations can lead to slightly different predictions for test sample that has not been observed during training, and marginalizing across references can offer robustness. On the other hand, Netanyahu *et al.* [2] introduced the bilinear transduction (BLT) protocol for performing extrapolation from unseen data regimes in regression tasks. It was found that generalizing to an "out of support" (OOS) sample x_t (i.e., no evidence of observing such a sample in the training data) can be made more tractable by carefully choosing anchors $\tilde{\mathrm{r}} \sim P_{\mathrm{r}}$ such that $\mathrm{x}_t - \tilde{\mathrm{r}} = \tilde{\mathrm{d}} \sim P_\Delta$. It was argued that, even if the specific combination of $[\tilde{\mathrm{r}}, \mathrm{x}_t - \tilde{\mathrm{r}}]$ may not be observed during training, the anchored model can produce better calibrated predictions when $\tilde{\mathrm{r}} \in P_{\mathrm{r}}$ and $\tilde{\mathrm{d}} \sim P_\Delta$. This is in contrast to [1], which hypothesized that when the tuple $[\tilde{\mathrm{r}}, \mathrm{x}_t - \tilde{\mathrm{r}}] \notin P_{(\mathrm{r},\Delta)}$, the inconsistency in the prediction will manifest as epistemic uncertainties. However, neither of these clearly answer the impact of inference protocol choice on generalization performance, particularly when the reference

 https://doi.org/10.52202/079017-3024



	CIFAR-100C Avg. Accuracy		
	Anchoring	Anchoring + RM-Reg.	Time (s) / 1000 samp
1 random	50.70 ± 0.44	53.49 ± 0.38	2.24
K random	50.70 ± 0.43	53.55 ± 0.42	3.51
BLT	50.69 ± 0.45	53.41 ± 0.30	24.24

Figure 2: **Impact of the choice of inference protocol on the performance of anchored models [1, 2].** (Left) A single random reference is chosen for sample prediction; (Middle) Obtaining predictions using K random references followed by averaging; (Right) Bilinear Transduction that identifies the optimal reference for each sample. We find that, while these protocols have varying computational complexities (time (s)/1000 samples), there are no apparent gaps in the performance, indicating that the limitation of anchored training cannot be fixed through sophisticated inference protocols.

set diversity is high. To answer this, we conducted a systematic evaluation of these protocols with anchored models trained on CIFAR100 with the reference set $\mathcal{R} = \mathcal{D}$.

Setup. We consider three evaluation protocols to make predictions for the CIFAR100C benchmark (i) 1 Random, that utilizes a single reference (e.g., average of samples in $\mathcal{R}$) to obtain predictions; (ii) K Random that utilizes K random references followed by reference marginalization ($K = 10$ in our case); (iii) BLT that searches for the optimal reference in $\mathcal{R}$ for each test sample. Since conducting such an exhaustive search can be expensive for bigger datasets, we pick a subset (set to 50 in our experiment).

Observations. The table in Figure 2 provides the average accuracies obtained from these inference protocols. Interestingly, while these protocols incurs varying inference times (column 3) (BLT $>>$ K random $>$ 1 random), their accuracies are statistically similar to each other (averaged across multiple seeds). *This observation implies that that the limitation of anchored training cannot be fixed through sophisticated inference protocols*. This motivates us to revisit anchoring and investigate if its behavior can be systematically improved during training itself.

3 Improving Anchored Training via Reference Masking Regularization

A close examination of anchored training reveals a critical limitation. As the size of the reference set increases, the number of reference-residual pairs grows combinatorially. For example, when $\mathcal{R} = \mathcal{D}$, there are $\binom{|\mathcal{R}|}{2}$ possible pairs, making it impractical to explore all pairs within a fixed number of training iterations. This results in insufficient sampling of $P_{(\mathrm{r},\Delta)}$, increasing the risk that anchored training may overlook the reference and make predictions based solely on the residuals. Such non-generalizable shortcuts are problematic because a sample should not be identifiable without considering the reference. Therefore, it is crucial to enhance anchored training by more effectively utilizing the diversity present in large reference sets.

```
L = CrossEntropy()
for (r, x, y) in loader:
    mask = (bernoulli(alpha) == 1)
    if mask:
        anc = CONCAT([0, x-r])
        y_hat = softmax(model(anc))
        loss = L(U, y_hat)
        # U - Uniform Prior on all classes
    else:
        anc = CONCAT([r, x-r])
        y_hat = softmax(model(anc))
        loss = L(y, y_hat)

    optimizer.zero_grad()
    loss.backward()
    optimizer.step()
```

Figure 3: PyTorch style pseudo code for our proposed approach.

3.1 Reference Masking Regularization

We propose a novel, yet simple regularization strategy for improving anchored training. Formally, for a given tuple $[\bar{\mathrm{r}}, \mathrm{x} - \bar{\mathrm{r}}]$, and a user specified probability α that controls how often the training is regularized, reference masking zeroes out the reference and keeps the residual fixed to obtain $[\mathbf{0}, \mathrm{x} - \bar{\mathrm{r}}]$. For comparison, the tuple for the same sample x but with a "zero" reference (Note: zero vector/image can be a valid reference in our reference distribution) corresponds to $[\mathbf{0}, \mathrm{x} - \mathbf{0}]$. In order to preserve the integrity of the anchoring mechanism, we systematically discourage the model from making meaningful predictions when the reference is masked. This can be implemented by mapping randomly masked tuples to high-entropy predictions (i.e., uniform probabilities). We achieve this by minimizing the cross-entropy loss between the

https://doi.org/10.52202/079017-3024

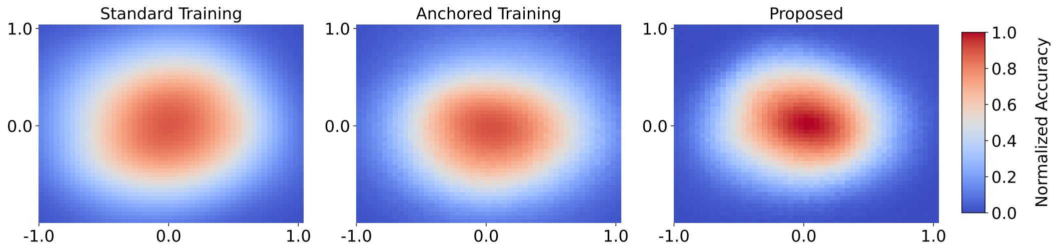


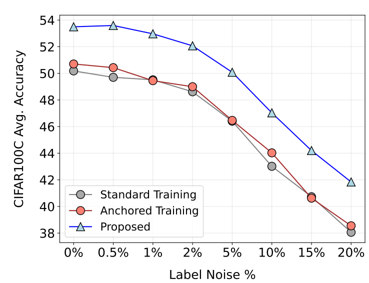
Figure 4: **Impact of the proposed regularizer on anchored training.** Using the CIFAR100C accuracy landscape, i.e., 2D heatmaps of the parameter space, we find that our approach identifies flatter and wider optima, thus leading to improved generalization [7]

predictions from the masked tuple and the uniform prior $\mathcal{U}$ over C classes (i.e, probability of any class $= 1/C$). Figure 3 provides the algorithm our proposed approach.

Circling back to Figure 1, we observe that the proposed regularization significantly improves generalization accuracies compared to standard and original anchored training. This clearly demonstrates our strategy's effectiveness in leveraging the diversity in $P_{\mathrm{r},\Delta}$. Following the insights from the previous section, we use the simple 1 random inferencing protocol to obtain predictions for test samples. At low anchor set sizes ($|\mathcal{R}| \leq 50$), there is high likelihood of exposing the model to all possible combinations of samples and references, and hence the risk of learning such shortcuts is minimal. In such a scenario, overemphasizing the masking-based regularization (i.e., high α) leads to underfitting, as illustrated in Figure 1. Unsurprisingly, reducing the masking probability can circumvent this underfitting behavior, as evidenced by the original anchored training, where $\alpha = 0$. However, the benefits of our regularization become apparent at larger reference set sizes. Additionally, the table in Figure 2 demonstrates that our approach performs similarly to the original anchored training, thereby implying no discernible impact on the inference efficiency.

Augmentations	Standard Training	Anchored Training	Proposed
Geometric	33.98	34.43	**38.06**
RandAug [8]	49.74	50.15	**53.7**
TrivialAug [9]	47.42	47.98	**51.22**
PixMix [10]	58.57	58.38	**59.60**

(a) Across different augmentation protocols, our proposed regularization provides non-trivial gains over standard training. Here, we show the accuracies for the challenging case of highest corruption severity.



(b) Our approach demonstrates improved robustness to label noise in comparison to existing approaches.

Figure 5: **Analysis of Anchored Models**. Using evaluations on the CIFAR100C OOD generalization of ResNet18 models trained on CIFAR100, we study the behavior of the proposed approach when combined with data augmentation protocols (left) and in presence of training label noise (right).

3.2 Analysis

How does the accuracy landscape look like? We hypothesize that the improved generalization of anchoring stems from the training process itself, which inherently enables the model to find better solutions in the weight space. To validate this, we follow the analysis in [11], where it was shown that that a well-generalizable solution is typically associated with a wider or flatter local optima in the loss/accuracy landscape. To this end, following the open-source implementation from [12], we obtained 2D heatmaps of accuracy evaluated on the CIFAR100C benchmark over different weight perturbations from the local minima inferring using different training strategies. Figure 4 visualizes the accuracy landscapes, where the x and y axes represent the co-ordinates that correspond to the different weight realizations. It can be observed that our approach produces wider and flatter optima in comparison to the baselines, thus explaining the generalization behavior.

https://doi.org/10.52202/079017-3024

Can anchoring be combined with data augmentations? Using synthetic data augmentations during training is a widely adopted method for improving generalization of vision models. In this study, we investigate if anchoring can be utilized alongside existing augmentation protocols, including state-of-the-art techniques like PixMix [10]), and if the observed generalization improvements persist. Table 5a shows the CIFAR100C accuracies of models trained with different augmentation protocols. Note that, the architecture and the hyper-parameters of the augmentation protocols were fixed to be the same for a fair comparison. Remarkably, our approach consistently provides performance gains regardless of the augmentation protocols used, evidencing its utility as a generic training technique.

Does training label noise impact anchoring? In practice, we construct the reference set $\mathcal{R} \subseteq \mathcal{D}$ for anchored training. However, under label noise, a fraction (or all) noisy samples can be included in the reference set, and get used for obtaining relative representations. A natural question is if this will impact the anchored training; however, we remind that the tuple construction in anchoring does not use the target label of a reference, and the benefits of anchoring will persist even under label noise corruptions. We validate this using the following experiment: We randomly flip the labels of $l\%$ ($l = \{0.5, 1, 2, 5, 10, 15, 20\}$) of training samples before training a ResNet18 model on CIFAR100, and evaluate the generalization performance on CIFAR100C. Figure 5b illustrates that, with increasing levels of label noise, the anchored models do not demonstrate any additional challenges in handling label noise. In fact, it provides superior generalization ($\sim 4\%$ improvements at 20% label noise) when compared to the standard and vanilla anchored training protocols.

4 Experiments

In this section, we empirically demonstrate the effectiveness of our proposed strategy in training models of varying scales (ResNets, Transformers) on datasets of different complexities (CIFAR10, CIFAR100, ImageNet). We systematically evaluate the generalization of these models under natural covariate shifts and synthetic corruptions. Additionally, we perform a comprehensive evaluation of model calibration, anomaly rejection, and robustness of task adapters in an effort to assess the safety of anchored models. For all experiments in this section, we utilize the entire training dataset as the reference set and train both the original and the proposed anchored models. During inference, we randomly select a single reference from the reference set and perform evaluation on the different test datasets.

Training Datasets. (i) CIFAR-10 and (ii) CIFAR-100 [13] datasets contain $50,000$ training samples and $10,000$ test samples each of size 32×32 belonging to 10 and 100 classes, respectively; (iii) ImageNet-1K [14] is a large-scale vision benchmark comprising 1.3 million training images and $50,000$ validation images across 1000 diverse categories.

Architectures. We utilize a suite of vision transformer and CNN architectures with varying levels of structural and parameter complexity. Specifically for training with ImageNet, we consider SWINv2-T (28.4M params), SWINv2-S (49.7M), SWINv2-B (87.8M) [15] and ViT-B-16 (86.6M) [16]. For CIFAR100, we use ResNet-18 (11.7M) [5] and WideResNet40-2 (2.2M) [17] architectures, and ResNet-18 for CIFAR10. We provide the training recipes adopted for our models in Section A.3.

Choice of α. Through extensive empirical studies with multiple architectures, we found using the masking schedule hyper-parameter $\alpha = 0.2$ (corresponds to every 5^{th} batch in an epoch), leads to stable convergence (closely match the top-1 validation accuracy of standard training) on ImageNet and $\alpha = 0.25$ for CIFAR10/100. Note that, our approach performs reference masking for an entire batch as determined by α. We have included our analysis on the impact of choice of α in Section A.1.

4.1 Generalization to Covariate Shifts and Synthetic Corruptions

OOD Datasets and Evaluation Metrics. For models trained on CIFAR10, we evaluate generalization on CIFAR10C and CIFAR10$\bar{\text{C}}$. While the former contains 19 different types of corruptions (e.g., noise, blur, weather, digital), CIFAR10$\bar{\text{C}}$ comprises 10 types of synthetic noise, at 5 different severity levels respectively. Equivalently, for CIFAR100, we use the CIFAR100C and CIFAR100$\bar{\text{C}}$ benchmarks. For ImageNet-1K, we consider (i) ImageNet-C [6] with 19 natural image corruptions across 5 severity levels, (ii) ImageNet-$\bar{\text{C}}$ [18] with 10 noise corruptions across 5 severity levels; (iii) ImageNet-R [19] containing different renditions of 200 classes from ImageNet; (iv) ImageNet-S [20]

Table 1: **Generalization performance of CNNs trained on CIFAR10/100**. We report the ID test and the OOD (CIFAR10 -C/$\bar{C}$, CIFAR100 - C/$\bar{C}$) accuracies of standard and anchored CNNs to evaluate generalization ($\uparrow$). Note, we provide the difference (Δ) between the proposed and the standard model in each case with blue.

Dataset	Model	Method	ID Acc.	CIFAR10/100-C Accuracy %					CIFAR10/100-$\bar{C}$ Accuracy %				
				Sev. 1	Sev. 2	Sev. 3	Sev. 4	Sev. 5	Sev. 1	Sev. 2	Sev. 3	Sev. 4	Sev. 5
CIFAR-10	ResNet-18	Standard	95.15	89.44	83.47	77.91	70.74	58.72	86.86	81.97	74.51	65.94	60.31
		Vanilla Anchoring	94.92	88.99	84.28	79.16	72.09	59.82	87.04	82.79	75.00	66.73	61.52
		Proposed	95.72	90.98	87.15	83.17	77.81	67.26	89.24	85.38	78.34	70.33	65.43
		Δ	+0.57	+1.54	+3.68	+5.26	+7.07	+8.54	+2.38	+3.41	+3.83	+4.40	+5.12
CIFAR-100	ResNet-18	Standard	77.6	65.56	56.77	51.25	44.57	34.13	62.0	54.08	44.89	36.55	32.27
		Vanilla Anchoring	77.21	65.67	57.3	52.02	45.27	34.79	61.69	54.17	44.98	36.90	32.72
		Proposed	77.89	67.0	59.51	54.88	48.78	38.66	64.47	58.10	49.78	41.42	36.81
		Δ	+0.29	+1.44	+2.74	+3.63	+4.21	+4.53	+2.47	+4.02	+4.89	+4.87	+4.54
	WRN 40-2	Standard	75.48	62.26	52.82	46.85	40.12	30.05	60.09	52.89	44.44	35.78	31.06
		Vanilla Anchoring	76.67	64.55	55.47	49.43	42.84	32.75	61.59	54.42	45.50	36.12	31.11
		Proposed	77.03	66.0	57.77	52.33	45.64	35.52	63.83	57.76	49.32	40.26	35.29
		Δ	+1.55	+3.74	+4.95	+5.48	+5.52	+5.47	+3.74	+4.87	+4.88	+4.48	+4.23

Table 2: **Generalization performance of different transformer architectures trained on ImageNet-1K**. We report the ID test and OOD (corruptions and covariate shifts) generalization performance of standard and anchored vision transformers using the top1 accuracy. For calibration performance, we report the mean and standard deviation of the Smoothed ECE ($\downarrow$) metric across all ImageNet OOD datasets. Note, we provide the difference (Δ) between the proposed and the standard model in each case with blue.

Dataset	**SWINv2-T** (28.4M)			**SWINv2-S** (49.7M)			**VITb16** (86.6M)			**SWINv2-B** (87.8M)		
	Standard	Proposed	Δ	Standard	Proposed	Δ	Standard	Proposed	Δ	Standard	Proposed	Δ
ImageNet (val)	82.07	82.03	−0.04	83.71	83.68	−0.03	81.07	80.76	−0.31	84.11	84.09	−0.02
ImageNet-R	40.84	41.17	+0.33	45.17	46.63	+1.46	44.06	46.39	+2.33	45.7	48.16	+2.46
ImageNet-S	27.08	27.68	+0.60	32.25	33.3	+1.05	29.4	33.0	+3.60	31.91	33.34	+1.43
ImageNet-C (Sev. 1)	71.63	72.13	+0.50	74.48	74.7	+0.22	72.37	72.52	+0.15	74.45	75.24	+0.79
ImageNet-C (Sev. 2)	64.89	65.71	+0.82	68.8	69.12	+0.32	66.57	67.38	+0.81	68.55	69.63	+1.08
ImageNet-C (Sev. 3)	57.77	59.21	+1.44	62.84	63.65	+0.81	61.6	62.87	+1.27	62.34	64.05	+1.71
ImageNet-C (Sev. 4)	47.77	50.01	+2.24	54.32	55.5	+1.18	52.88	55.13	+2.25	53.66	56.08	+2.42
ImageNet-C (Sev. 5)	35.66	38.58	+2.92	42.85	44.33	+1.48	41.09	44.52	+3.43	41.87	45.19	+3.32
ImageNet-$\bar{C}$ (Sev. 1)	71.37	73.51	+2.14	75.39	76.59	+1.20	72.75	73.65	+0.90	75.12	77.1	+1.98
ImageNet-$\bar{C}$ (Sev. 2)	67.12	70.45	+3.33	72.26	74.24	+1.98	69.01	70.91	+1.90	72.15	74.69	+2.54
ImageNet-$\bar{C}$ (Sev. 3)	61.2	65.77	+4.57	67.14	70.17	+3.03	63.47	66.87	+3.39	67.16	70.81	+3.65
ImageNet-$\bar{C}$ (Sev. 4)	52.01	57.31	+5.30	58.73	62.93	+4.20	54.7	59.29	+4.59	58.66	63.53	+4.87
ImageNet-$\bar{C}$ (Sev. 5)	46.54	51.76	+5.22	53.7	58.25	+4.55	50.07	54.94	+4.86	53.75	58.77	+5.02

comprising black and white sketch images from each class of ImageNet. We use the top@1 accuracy to evaluate generalization performance.

Results and Discussions. First, in Table 1, we report the averaged accuracy over all corruptions for every severity level on the CIFAR10C/$\bar{C}$, CIFAR100C/$\bar{C}$ datasets, for the conv-nets trained on CIFAR10/100 respectively. We make a key finding that our proposed approach leads to significant gains in corruption accuracies across all severity levels over standard training ($1.54\% - 8.54\%$) on an average. When compared to CIFAR10, the improvements of anchoring are apparent even at lower severity levels, for e.g., $+3.74$ improvement with WRN 40-2 at CIFAR100C severity level 1.

Second, as shown in Table 2, we investigated the efficacy of anchored transformers trained on the large-scale ImageNet-1K dataset in terms of OOD generalization. It can be observed that our proposed approach consistently yields improvements in corruption accuracies over standard training across all architectures. A striking observation is that network capacity plays a significant role in effectively leveraging the increased diversity produced by anchored training (we used the entire ImageNet-1K as the reference set). For example, as we move from SWINv2-T (28.4M) to SWINv2-B (88M), we observe increasingly larger performance gains over standard training. Importantly, our proposed strategy handles high noise severity better, achieving improvements of $2\% - 7\%$ at severity 5 for both

https://doi.org/10.52202/079017-3024

Imagenet-C and $\bar{\text{C}}$. All these observations clearly evidence the importance of leveraging the diversity of $P_{(\mathrm{r},\Delta)}$ for enhanced generalization. Finally, we observe from Tables 1 and 2 that anchored training maintains competitive, and in a few cases, improved ID accuracies compared to standard training.

4.2 Assessing Safety of Anchored Models

Calibration and Anomaly Rejection. While generalization is key to improve model utility, it must be ensured that the models are not over-confident on unknown inputs and produce well-calibrated prediction probabilities that match the likelihood of correctness. Hence, measuring calibration [21] is vital to understand how tempered the model predictions are under distribution shifts. On the other hand, when the inputs are semantically disconnected and do not share the same label space as the training data, we require the models to appropriately flag them as anomalies. To that end, we also conduct an extensive evaluation of model calibration under distribution shifts and anomaly rejection. For the former, we use the ImageNet-C/$\bar{\text{C}}$/R/S variants, and for the latter, we consider two benchmarks: (a) Vision OOD, comprising commonly used anomaly rejection datasets - *iSUN* [22], *Textures* [23], and *Places365* [24]; and (b) *NINCO* [25], a recent benchmark containing images with semantic overlap with ImageNet but with no class overlap. Following standard practice [26], we use the Smoothed ECE metric [27] to assess calibration. For anomaly rejection, we obtain the energy scores [26] for both ID validation and OOD data, and report the AUROC metric.

We report the anomaly rejection and calibration performance of of transformer models trained with ImageNet-1K in Table 3. The results demonstrate notable improvements in anomaly rejection across architectures, highlighting the ability of our approach to better recognize residuals $\mathrm{x}_t - \bar{\mathrm{r}} = \bar{\mathrm{d}} \notin P_\Delta$ for an anomalous input sample x_t and a reference $\bar{\mathrm{r}}$ observed during training. This is evidenced by substantial gains on both vision OOD and the challenging NINCO anomaly detection benchmarks. For instance, ViTb16 trained with the proposed approach achieves a gain of $+4.34\%$ on AUROC over non-anchored variant on the NINCO benchmark. In addition, our approach produces consistently lower calibration errors irrespective of the choice of architecture, showcasing our ability to produce tempered predictions under OOD shifts.

Table 3: **Anomaly rejection and calibration performance of transformers trained on ImageNet-1K**. We compare the anomaly rejection performance against standard training using common vision OOD benchmarks (Textures, Places365, and iSUN datasets) and the more recent NINCO dataset. For evaluation, we consider the AUROC (↑) metric. Moreover, we also provide Smoothed ECE scores (↓) (mean, std) across different Imagenet corruption benchmarks. We highlight the best performing model in each case with blue.

Model	Method	Vision OOD	NINCO	Calibration
SWINv2-T	Standard	76.54	77.46	0.121 ± 0.034
	Proposed	**77.65**	**78.49**	$\mathbf{0.117 \pm 0.027}$
SWINv2-S	Standard	77.13	74.73	0.126 ± 0.039
	Proposed	**79.56**	**78.47**	$\mathbf{0.119 \pm 0.041}$
VITb16	Standard	**77.29**	65.98	0.109 ± 0.037
	Proposed	76.88	**70.32**	$\mathbf{0.105 \pm 0.028}$
SWINv2-B	Standard	75.89	72.13	0.132 ± 0.055
	Proposed	**78.91**	**74.53**	$\mathbf{0.124 \pm 0.051}$

Robustness to Task Adaptation. Evaluating model adaptation under task shifts [28] becomes important to shed light onto the quality and re-usability of features inferred in a backbone network. To that end we employ two evaluation protocols: Adaptation(`ID Eval`) and Adaptation (`OOD Eval`). In the former, we assume that the distribution of the dataset used for linear probing is the same as that of the test set. In the latter, we first train the linear probe (LP) with our anchored training approach using a probing dataset but evaluate the same with data drawn from a shifted w.r.t the probing dataset. Note, for both evaluation protocols, we fix the ViTb16 architecture as the Imagenet pre-trained feature extractor backbone. Note, we set $\alpha = 0.4$, a higher value than the original task model training as we observed stable convergence.

Adaptation (`ID Eval`): We consider the following target datasets: (i) CIFAR-10 [13] ; (ii) CIFAR-100 [29] ; (iii) UCF101 [30]; (iv) Flowers102 [31]; (v) StanfordCars [32]. The results in Figure 6(a) demonstrate that the proposed approach achieves substantial performance gains over the baseline (0.81% - 2.68%). These findings suggest that the reference masking regularizer yields feature representations that are transferable even under complex task shifts.

Dataset	VITb16 (86.6M)		
	Standard	**Proposed**	Δ
CIFAR-10	95.48	96.29	+0.81
CIFAR-100	80.1	82.78	+2.68
UCF101	75.55	77.01	+1.46
Flowers102	94.68	95.7	+1.02
StandfordCars	58.54	61.15	+2.61

(a) **LP-based adaptation for ViTb16 architecture pre-trained on Imagenet-1K on downstream tasks**. We measure the accuracy (↑) of the adapted model using the validation split of the target dataset.

Evaluation Domain	Train Domain: **Real**			Train Domain: **Sketch**		
	Standard	**Proposed**	Δ	**Standard**	**Proposed**	Δ
Real	–	–	–	41.35	44.81	+3.46
Sketch	25.85	28.02	+2.17	–	–	–
Clipart	37.38	38.98	+1.6	35.4	37.76	+2.36
Painting	46.3	46.97	+0.67	31.42	32.7	+1.28

(b) **OOD Evaluation of LP Adaptation**. Using the ViTb16 backbone we train two LPs for the *Real* and *Sketch* domains from the Domainnet dataset respectively. We then assess their zero-shot accuracies on three held-out test domains. Our findings show that the proposed approach consistently outperforms the non-anchored baselines.

Figure 6: Assessing anchored and standard pre-trained ImageNet backbones on robustness to task shifts.

Adaptation (`OOD Eval`): For training linear probes, we use the DomainNet [33], comprising of images from 345 categories across six diverse domains. Specifically, we pick four domains, namely *real*, *sketch*, *clipart*, and *painting* and train probes on (i) images from the *real* domain, and (ii) images from the *sketch* domain respectively. We then evaluate the LPs on the remaining three held-out domains. As Figure 6(b) illustrates, our proposed reference masking continues to substantially outperform standard training baseline on all held-out domains under both configurations. We attribute this behavior to our approach being able to effectively leverage the diversity in the reference-residual space to produce robust and better generalizable features supporting transferability.

5 Related Work

Anchoring in Predictive Models. Our work is based on the principle of anchoring first introduced in [1] where it was used to achieve stochastic data centering for epistemic uncertainty estimation. Since then, the anchoring has been extended to a variety of use-cases and applications. For e.g, Netanyahu *et al.* [2] utilized anchoring for extrapolating to unseen data regimes [2] in regression settings and Trivedi *et al.* [34] employed the same for graph neural network calibration. In contrast, our paper is the first to explore and facilitate the utility of anchoring as a viable training protocol for large scale vision models.

Data Augmentations. Augmentation strategies enforce models to be robust under different pixel-space manipulations improving generalization. For e.g., strategies such as Augmix [35] or random convolutions (RandConv) [36] are known to improve generalization. Recent advancements in the field include strategies such as PixMix [10], which utilizes an external dataset with complex image patterns to augment the training data, and ALT [37], which learns adversarially robust augmentations. While the idea of enforcing prediction consistency in anchoring might appear similar to training with synthetic data augmentations, we emphasize that anchoring does not alter the data (e.g., with perturbations or geometric transformations) but only creates relative representations for each sample with respect to different reference choices. Furthermore, it can be combined with data augmentations to achieve further gains in generalization (Table 5a).

Model Safety. As models are being increasingly adopted in a variety of sensitive applications [38, 39], safe model deployment has become critical [40, 41]. In this context, generalization to data beyond the training distribution [42, 6], ability to accurately detect anomalies in the input data [43, 26, 44] as well producing calibrated prediction probabilities [21, 3] are all important facets of safety evaluation. Hendrycks *et al.* [10] argued that most existing training strategies compromise for one safety objective to satisfy another objective, thus limiting their real-world utility. We find from our experiments that anchoring jointly produces better generalization, calibration and anomaly rejection properties, which makes it a promising choice for practical deployment.

https://doi.org/10.52202/079017-3024

6 Conclusion

Through this work, we showed that anchoring leads to significant performance gains in generalization and other safety metrics, including calibration, anomaly rejection, and task adaptation, across varying dataset sizes (CIFAR-10 to ImageNet) and model architectures (Conv-Nets to Transformers). Notably, when the training recipe includes high-capacity architectures or advanced mechanisms, our method yields even greater performance gains over the base models. Our observations suggest that anchored training with larger reference sets requires reference masking regularization to control the risk of learning undesirable shortcuts while making predictions. However, we realize that state-of-the-art results in OOD generalization are often obtained using model souping [45] or by fine-tuning large scale pre-trained models [46]. Hence, we believe it will be valuable to integrate anchoring into these approaches. While we have not theoretically characterized the generalization of anchored models, our hypothesis is rooted in existing theory and our empirical results provide evidence for the hypothesis. Finally, it must be noted that anchoring is a domain-agnostic, architecture-agnostic, and task-agnostic training strategy for deep neural networks. However, developing a theoretical understanding of anchored models and understanding its benefits in domain-specific applications is crucial and forms an important future direction.

Acknowledgements

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. 22-ERD-006. LLNL-CONF-864979.

References

[1] Jayaraman J. Thiagarajan, Rushil Anirudh, Vivek Narayanaswamy, and Peer timo Bremer. Single model uncertainty estimation via stochastic data centering. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=j0J9upqN5va`.

[2] Aviv Netanyahu, Abhishek Gupta, Max Simchowitz, Kaiqing Zhang, and Pulkit Agrawal. Learning to extrapolate: A transductive approach. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=lid14UkLPd4`.

[3] Rushil Anirudh and Jayaraman J Thiagarajan. Out of distribution detection via neural network anchoring. In *Asian Conference on Machine Learning*, pages 32–47. PMLR, 2023.

[4] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

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

[6] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=HJz6tiCqYm`.

[7] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in neural information processing systems*, 31, 2018.

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

[9] Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 774–782, 2021.

[10] Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Bo Li, Dawn Song, and Jacob Steinhardt. Pixmix: Dreamlike pictures comprehensively improve safety measures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16783–16792, 2022.

[11] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

[12] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Neural Information Processing Systems*, 2018.

[13] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[14] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[15] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, Furu Wei, and Baining Guo. Swin transformer v2: Scaling up capacity and resolution. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

[17] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference 2016*. British Machine Vision Association, 2016.

[18] Eric Mintun, Alexander Kirillov, and Saining Xie. On interaction between augmentations and corruptions in natural corruption robustness. *Advances in Neural Information Processing Systems*, 34:3571–3583, 2021.

[19] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021.

[20] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

[21] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[22] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[23] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

[24] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.

https://doi.org/10.52202/079017-3024

[25] Julian Bitterwolf, Maximilian Mueller, and Matthias Hein. In or out? fixing imagenet out-of-distribution detection evaluation. In *ICML*, 2023. URL `https://proceedings.mlr.press/v202/bitterwolf23a.html`.

[26] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 2020.

[27] Jarosław Błasiok and Preetum Nakkiran. Smooth ece: Principled reliability diagrams via kernel smoothing. *arXiv preprint arXiv:2309.12236*, 2023.

[28] Anders Andreassen, Yasaman Bahri, Behnam Neyshabur, and Rebecca Roelofs. The evolution of out-of-distribution robustness throughout fine-tuning. *arXiv preprint arXiv:2106.15831*, 2021.

[29] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55:5, 2014.

[30] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[31] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[32] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013.

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

[34] Puja Trivedi, Mark Heimann, Rushil Anirudh, Danai Koutra, and Jayaraman J. Thiagarajan. Estimating epistemic uncertainty of graph neural networks. In *Data Centric Machine Learning Workshop @ ICML*, 2023.

[35] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. AugMix: A simple data processing method to improve robustness and uncertainty. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[36] Zhenlin Xu, Deyi Liu, Junlin Yang, Colin Raffel, and Marc Niethammer. Robust and generalizable visual representation learning via random convolutions. In *International Conference on Learning Representations*, 2021.

[37] Tejas Gokhale, Rushil Anirudh, Jayaraman J Thiagarajan, Bhavya Kailkhura, Chitta Baral, and Yezhou Yang. Improving diversity with adversarially learned transformations for domain generalization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 434–443, 2023.

[38] Thomas Davenport and Ravi Kalakota. The potential for artificial intelligence in healthcare. *Future healthcare journal*, 6(2):94, 2019.

[39] Daniel Bogdoll, Maximilian Nitsche, and J Marius Zöllner. Anomaly detection in autonomous driving: A survey. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4488–4499, 2022.

[40] Clark Barrett, Brad Boyd, Elie Bursztein, Nicholas Carlini, Brad Chen, Jihye Choi, Amrita Roy Chowdhury, Mihai Christodorescu, Anupam Datta, Soheil Feizi, et al. Identifying and mitigating the security risks of generative ai. *Foundations and Trends® in Privacy and Security*, 6(1):1–52, 2023.

[41] Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt. Unsolved problems in ml safety. *arXiv preprint arXiv:2109.13916*, 2021.

[42] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*, 2021.

[43] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017.

[44] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2018.

[45] Mitchell Wortsman, Gabriel Ilharco, Samir Ya 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 Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 23965–23998. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/wortsman22a.html`.

[46] Sachin Goyal, Ananya Kumar, Sankalp Garg, Zico Kolter, and Aditi Raghunathan. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19338–19347, June 2023.

[47] Rushil Anirudh and Jayaraman J Thiagarajan. Out of distribution detection via neural network anchoring. In *Asian Conference on Machine Learning (ACML)*. PMLR, 2022.

https://doi.org/10.52202/079017-3024

A Appendix

A.1 How does the choice of α impact training?

The parameter α controls the frequency of the regularization applied to anchored training. Under the assumptions of operating with wide reference sets, through Table 4 we note that moderate to small values of α enable better regularization of anchored training. Notably, setting $\alpha = 0.25$ i.e. masking references for one in four samples, yields impressive gains in ID and OOD performance. Conversely, over-regularizing by setting α to a large value (e.g., 1.0) entails masking every reference, unsurprisingly results in models that generalize poorly, as they are tasked with learning solely from residuals.

Table 4: **Impact of α on anchored training**. As we gradually increase α, there is a risk of over-regularization which can lead to severe underfits. Note, we consider $\mathcal{R} = \mathcal{D}$ in this study.

$\alpha \rightarrow$	0.0	0.25	0.5	0.75	1.0
ID Test Acc. %	77.21	**77.89**	76.97	75.4	57.90
OOD Acc. %	51.01	**53.77**	52.61	52.30	35.40

A.1.1 Choosing α in practice

At low reference set sizes, there is a high likelihood of exposing the model to all possible combinations of samples and references, and hence the risk of learning shortcuts is minimal. In this case, overemphasizing the reference masking probability (i.e., increasing α) can significantly inhibit this exposure. Consequently, this leads to underfitting as the model is tasked with learning solely from the residuals which is undesirable in practice (Blue curve for reference set size $\leq$50 in Fig. 1). Reducing α can combat this behavior, as evidenced by the original anchored training (special case of reference masking with $\alpha = 0$ namely the red curves for reference set size $\leq$50 in Fig. 1).

Now, with larger reference sets (e.g., datasets in the scale of ImageNet1K), the number of reference-residual pairs grows combinatorially, making it impractical to expose the model to all diverse pairs in a fixed number of training iterations. In such a scenario, reducing α can increase the risk of learning shortcuts and lead to suboptimal performance. Increasing α on the other hand can in fact aid training as it systematically avoids these shortcuts and improves generalization. In summary, the optimal α value depends both on the reference set size and the convergence behavior of model training.

A.2 Does Training for Additional Epochs Alleviate the Reference Set Size Problem?

One possible way of alleviating this problem is by reducing the reference set size. However, this reduces the diversity of the reference-residual pairs exposed during training and can lead to a poor solution. While the issue of diversity can be combated with large reference set sizes, increasing the number of epochs alone does not solve the problem as there exists a combinatorially large number of reference-residual pairs which cannot be practically explored, and the model will still be vulnerable to shortcuts. Moreover, modifying the number of training epochs results in non-trivial modifications in the training hyper-parameters (e.g., learning rate schedules) and can lead to poorly convergent models if the hyper-parameters are chosen incorrectly. Hence, our reference masking regularizer for anchored training, helps mitigate shortcut decision rules while also being computationally efficient.

A.3 Additional Details on Training Protocols

Table 5 outlines the recipes (augmentations, epochs, optimizers) leveraged for model training. Note that, the other hyper-parameters can be found in [47] for CIFAR10/100 and `https://pytorch.org/vision/stable/models.html` for ImageNet. We emphasize that, anchoring can be used as a generic model training wrapper, allows integration with any data augmentation or training strategy, and is not restricted to the recipes considered.

Table 5: **Protocols adopted for training anchored models across different datasets and architectures.** While we adopt standard training recipes for training our models, we note that anchoring can serve as a generic wrapper that can be applied on top of any other existing recipe.

Model	Dataset	Training Recipes	Number of Epochs		Optimizer
			Non-Anchored	Anchored	
ResNet-18, WRN-40-2	CIFAR-10/100	`Horizontal & Vertical Flips`	200	200	SGD with Multi-Step
SWINv2-T, SWINv2-S, SWINv2-B	ImageNet	`Mixup`, `CutMix`, `AutoAugment`, `Random Erase`, `AugMix`, `Label Smoothing`	300	330	AdamW with Cosine Annealing
VITb16	ImageNet	`Mixup`, `CutMix`, `AutoAugment`, `AugMix`, `Label Smoothing`	300	330	AdamW with Cosine Annealing

https://doi.org/10.52202/079017-3024

A.4 Expanded ImageNet Generalization Results

We provide an expanded version of Table 2 that includes the anchored training protocol without the reference-masking regularizer.

Table 6: **Generalization performance of models trained on ImageNet-1K**. We compare the generalization performance of different training strategies under both ID and OOD (corruptions and distribution shifts) test settings. For evaluating the prediction performance on each of the benchmarks, we consider the widely adopted Top1 accuracy metric. For calibration performance, we report the mean and standard deviation of the Smoothed ECE (↓) metric across all ImageNet OOD datasets. Note, we highlight the best performing model in each case with blue.

Model	Method	ID Acc.	ImageNet-R	ImageNet-S	ImageNet-C					ImageNet-$\bar{C}$					Calibration
					Sev. 1	Sev. 2	Sev. 3	Sev. 4	Sev. 5	Sev. 1	Sev. 2	Sev. 3	Sev. 4	Sev. 5	
SWINv2-T (28.4M)	Standard	82.07	40.84	27.08	71.63	64.89	57.77	47.77	35.66	71.37	67.12	61.2	52.01	46.54	0.121 ± 0.034
	Anchoring	**82.26**	40.36	27.56	**72.32**	**65.85**	58.95	49.51	37.41	72.68	68.96	63.29	53.74	48.14	0.121 ± 0.032
	Proposed	82.03	**41.17**	**27.68**	72.13	65.71	**59.21**	**50.01**	**38.58**	**73.51**	**70.45**	**65.77**	**57.31**	**51.76**	$\mathbf{0.117 \pm 0.027}$
SWINv2-S (49.7M)	Standard	83.71	45.17	32.25	74.48	68.8	62.84	54.32	42.85	75.39	72.26	67.14	58.73	53.7	0.126 ± 0.039
	Anchoring	**84.0**	45.95	32.08	**74.75**	68.87	63.12	54.7	43.14	76.07	73.33	68.79	60.49	55.19	0.122 ± 0.045
	Proposed	83.68	**46.63**	**33.3**	74.7	**69.12**	**63.65**	**55.5**	**44.33**	**76.59**	**74.24**	**70.17**	**62.93**	**58.25**	$\mathbf{0.119 \pm 0.041}$
VITb16 (86.6M)	Standard	**81.07**	44.06	29.4	72.37	66.57	61.6	52.88	41.09	72.75	69.01	63.47	54.7	50.07	0.109 ± 0.037
	Anchoring	80.57	45.56	32.32	**72.64**	67.14	62.33	54.46	43.48	73.21	69.74	64.57	56.03	51.46	0.106 ± 0.035
	Proposed	80.76	**46.39**	**33.0**	72.52	**67.38**	**62.87**	**55.13**	**44.52**	**73.65**	**70.91**	**66.87**	**59.29**	**54.94**	$\mathbf{0.105 \pm 0.028}$
SWINv2-B (87.8M)	Standard	**84.11**	45.7	31.91	74.45	68.55	62.34	53.66	41.87	75.12	72.15	67.16	58.66	53.75	0.132 ± 0.055
	Anchoring	84.06	47.6	**33.42**	74.95	69.28	63.43	55.08	43.8	76.36	73.3	68.49	60.05	54.81	0.129 ± 0.058
	Proposed	84.09	**48.16**	33.34	**75.24**	**69.63**	**64.05**	**56.08**	**45.19**	**77.1**	**74.69**	**70.81**	**63.53**	**58.77**	$\mathbf{0.124 \pm 0.051}$

A.5 Expanded Anomaly Rejection Results for Vision OOD Datasets

While Table 3 in the main paper provided anomaly rejection results averaged over all Vision OOD datasets, we expand and present metrics for each dataset in Table 7

Table 7: Measuring anomaly rejection performance on Imagenet-1K. We report the AUROC (↑) scores

Architecture	Method	Anomaly Rejection (AUROC)		
		iSUN	Textures	Places365
SWINv2-T	Standard	**80.25**	76.83	72.53
	Anchored Training	78.68	76.64	74.75
	Proposed	77.69	**78.09**	**77.16**
SWINv2-S	Standard	82.89	77.87	70.63
	Anchored Training	**87.73**	**80.83**	**76.67**
	Proposed	84.18	79.66	74.85
VITb16	Standard	**86.92**	**79.24**	65.72
	Anchored Training	85.17	76.88	66.16
	Proposed	84.55	78.91	**67.18**
SWINv2-B	Standard	85.32	76.35	65.99
	Anchored Training	85.98	**77.88**	70.75
	Proposed	**87.34**	75.74	**73.66**

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: [NA]

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In a few cases of training transformers on ImageNet, our approach produces slightly lower in-distribution accuracies (for e.g, 0.3% reduction in VitB16). While this comes at a significant gain in OOD accuracy, the question remains of how to improve the anchored training to prevent this performance drop. We also provide additional limitations section in conclusion section.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: [NA]

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have provided extensive details regarding data pre-processing pipelines, hyper-parameters, training protocols in 4. We have also provided PyTorch code snippets for easy implementation of our approach.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: [NA]

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We used standard train, validation and test splits that are made available with datasets. Training protocols, hyperparameters and the sensitivity of model for hyperparameters are provided in the appendix.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Experiments were conducted using multiple seeds and error bars are reported.

8. **Experiments Compute Resources**

https://doi.org/10.52202/079017-3024

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: [NA] .

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: [NA]

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We have discussed them in the related work as well conclusion section. In our opinion, there are no negative societal impacts.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: [NA]

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer:[NA]

    Justification: [NA]

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: [NA]

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: [NA]

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: [NA]