Efficiently Robustify Pre-Trained Models

Nishant Jain *
IIT Roorkee

Harkirat Behl
Microsoft Research

Yogesh Singh Rawat
CRCV, UCF

Vibhav Vineet
Microsoft Research

Abstract

A recent trend in deep learning algorithms has been towards training large scale models, having high parameter count and trained on big dataset. However, robustness of such large scale models towards real-world settings is still a less-explored topic. In this work, we first benchmark the performance of these models under different perturbations and datasets thereby representing real-world shifts, and highlight their degrading performance under these shifts. We then discuss on how complete model fine-tuning based existing robustification schemes might not be a scalable option given very large scale networks and can also lead them to forget some of the desired characterstics. Finally, we propose a simple and cost-effective method to solve this problem, inspired by knowledge transfer literature. It involves robustifying smaller models, at a lower computation cost, and then use them as teachers to tune a fraction of these large scale networks, reducing the overall computational overhead. We evaluate our proposed method under various vision perturbations including ImageNet-C,R,S,A datasets and also for transfer learning, zero-shot evaluation setups on different datasets. Benchmark results show that our method is able to induce robustness to these large scale models efficiently, requiring significantly lower time and also preserves the transfer learning, zero-shot properties of the original model which none of the existing methods are able to achieve.

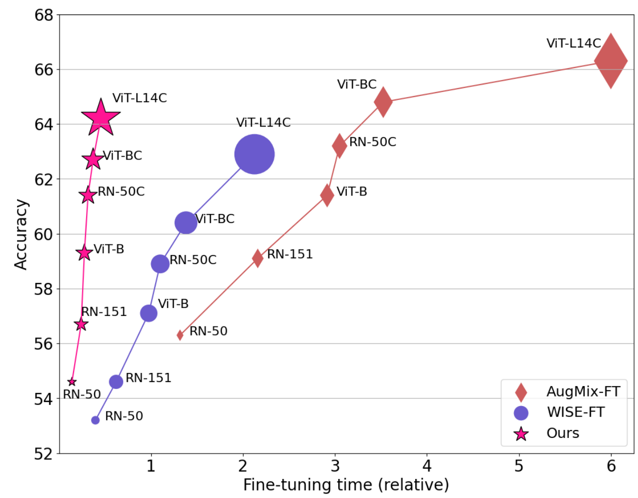

Figure 1. ImageNet-C accuracy v/s training time comparison. Our method is on the pareto-front (achieves better robust accuracy in much lesser time) compared to the state-of-the-art methods Augmix based Complete fine-tuning and WISE-complete fine-tuning. The data points labelled with suffix "C" correspond to CLIP models.

1. Introduction

Large scale deep neural networks trained on large scale data have revolutionized the modern AI era. They are significantly effective in solving practical problems of high importance. These include object detection, zero-shot classification, image segmentation, image generation, and many other applications [18, 25, 27, 39, 5, 29, 35, 9].

Though the large models have shown impressive results on many vision problems [27, 29], their reliability under distribution shift e.g., under illumination changes, geographical variations, camera properties etc., is still under-explored. In this paper, we fist investigate the behavior of large models under distribution shifts. We analyse popular models under synthetic perturbations to images [11], natural distribution shifts [10, 13], differently styled images [31] and dataset shift [2]. Our analysis of models of various sizes, architecture families (transformers or CNNs) and training modalities (uni or multi-modal) establishes their brittleness under distribution shifts.

This analysis begs the question: can we induce robustness to large vision models without sacrificing their original properties? It is critical to simultaneously maintain *clean* accuracy on the original datasets, improve *robust* accuracy on the shifted data and preserve the *transfer learning* capabilities of the large models. Further, computation efficiency during both training and inference is beneficial.

While several prior works can be used to make large-scale models robust, they do not possess the desired prop-

*Correspondence to Nishant Jain at njain@cs.iitr.ac.in.

erties discussed above. One direction involves fine-tuning the model [33, 16]. This generally suffers from either poor performance under synthetic perturbations or requires significant training time. Another line of work could be to use advanced augmentation techniques (e.g., aug-mix, pix-mix) [14, 12, 10, 4]. They are effective under synthetic perturbations and natural shifts in the data. However, they require significantly larger compute time and lead to the large models forgetting their original and transfer learning properties. Figure 1 shows this analysis in a pareto-front plot for two of the recently proposed robustness methods.

To this end, we propose a knowledge transfer method to induce robustness to large models that possesses all the desired properties discussed above. It makes large models robust efficiently (refer Fig.1). We take a *plug-and-play* approach: insert an additional small robust module and update only a very small portion of the existing large models. To achieve robustness, we explore a new direction: a relatively much smaller but robust model inducing robust knowledge to a large model. Though this provides a novel look at the knowledge distillation approach, a straight-forward application leads to the large models forgetting their original properties. For this challenging task of ensuring that clean accuracy is preserved in the clean module, robustness induced into the robust module and correct module selected at test time, we propose a novel uncertainty-aware knowledge distillation technique. This allows us to fulfil all our required objectives. Since our method involves updating only a small chunk of the large network, it achieves low training latency (refer section 5). To the best of our knowledge, this is the first time such a setup has been used involving knowledge transfer from a smaller to a large model. Further, it should be noted that smaller models can be made robust by using prior works like advance augmentation methods [12, 10, 14].

We evaluate our method under various distribution shift on ImageNet data [28] in section 5. It includes ImageNet-C [11], ImageNet-R [10], ImageNet-A [13], ImageNet-sketch [31], ImageNet-V2. We also evaluate on ObjectNet [2] and its perturbed variations ObjectNet-C. We show results for both multi-modal (various CLIP models) and unimodal (various architectures including both ResNets and Vision Transformers). Alongside this, we also test our method on other datasets in the transfer learning setup, to analyze further if the desired properties of the model are restored. In all these cases, our method outperforms prior approaches on robust accuracy while still performing at par on clean accuracy. At the same time, possessing desired characteristics like transfer learning capabilities (refer section 5) and being efficient during training and inference.

2. Related Work

Large scale models. In recent years, studies [38, 36, 8] have shown that training large models such as vision transformers [5] on large datasets can improve accuracies significantly. Several works [26, 3] have evaluated these models for robustness lately . Furthermore, these large models can been trained either in a unimodality setup [26] or multi modality setup [27, 39, 35]. Though they achieve good performance on several downstream tasks, any modification of these large models can lead to forgetting of the knowledge contained in them. In contrast, we propose a method that allows to adapt the model parameters without sacrificing their properties.

Finetuning and Augmentation based methods to achieve robustness. Several advanced augmentation techniques have been proposed to improve robustness of the deep learning based models. Examples includes cut-out, cutmix, mixup, augmix, pixmix and others [4, 37, 12, 40]. Further, a recently proposed method WISE interpolates the fine-tuned and original model parameters [33] for robustness [40]. Generally these techniques are a computation heavy process and also leads to modifying the main network parameters that could lead to these large models forgetting in their original properties. In our approach, we use some of these advanced augmentation technique to make our teacher network robust. We ensure that our robust approach does not sacrifice the large models' properties and is also computationally efficient.

Knowledge distillation It involves transferring knowledge from a large network to a smaller network by minimizing the distance between the predicted logit distribution of the student and teacher networks [15]. It proved to be highly effective on standard downstream datasets [1], [7]. In all these KD applications, knowledge is transferred from a larger network to a smaller network. In contrast, we propose a method to induce robustness to a larger network by transferring knowledge from a smaller (teacher) network.

3. Robustness Analysis

In this section, we analyze the image classification performance of models of different shapes and sizes, with different training settings (unimodal or multimodal). We stress test the models under both synthetic and natural perturbations. Especially, contrasting the behaviour of multimodal models (vision-language) *vs.* unimodal (image only).

Models. In unimodal setting, we analyse Resnet-50, ResNet-101 and ResNet-150 [9], ViT-small, ViT-base and ViT-large models [5] trained on ImageNet [28]. In multimodal setting, we analyse CLIP [27] model with backbones including ResNets: CLIP-RN50 and CLIP-RN101, and transformers: CLiP-ViT B/16, CLiP-ViT B/32. We also analyze self-supervised unimodalities trained on large

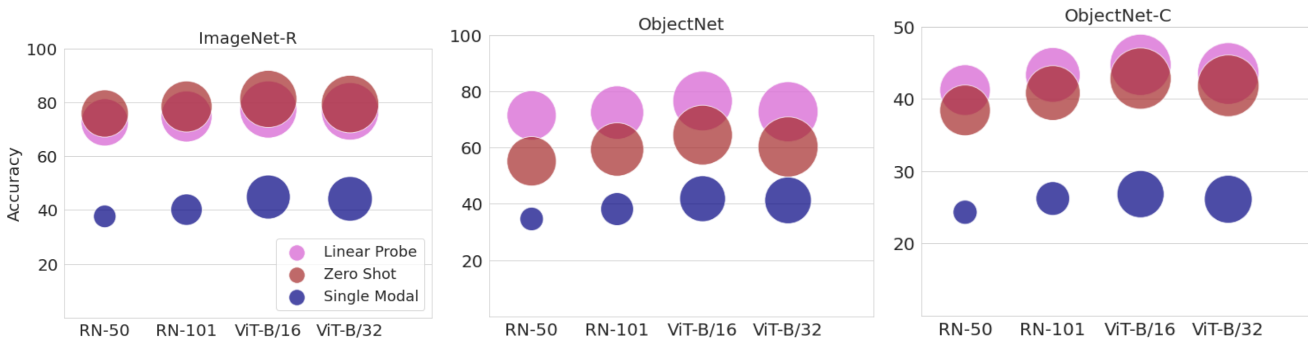

Figure 2. Analysis of multi-modal linear-probe, multi-modal zero-shot and unimodal networks under various distribution shifts including ImageNet-R, ObjectNet, ObjectNet-C. The x-axis denote the model architecture and y-axis denotes the accuracy.

datasets as against only ImageNet pretrained ones. For this, we use the masked autoencoders [8] and DINO V2 [24] models proposed recently, shown to be highly effective in representation learning. We analyze two architecture, ViT-B/16, ViT-B/32 for MAE and ViT-B/14 for DINO V2.

Datasets. We evaluate the models on various shifted version of ImageNet [28]: ImageNet-Corrupted (ImageNet-C) [11] , ImageNet-Rendition (ImageNet-R) [10] and ImageNet-Sketch (ImageNet-S) [31] datasets. For natural shifts, we use ImageNet-Adversarial (ImageNet-A) comprising natural adversarial examples[13] (results in supplementary). Further, we also evaluate the models on ObjectNet [2] and its perturbed version ObjectNet-C, which we generate by applying all the 15 corruptions of ImageNet-C to ObjectNet data.

Experimental Setup. In the unimodal case, we evaluate ImageNet trained models from timm [32] on ImageNet-C, ImageNet-R, ImageNet-S (supplementary), ObjectNet and ObjectNet-C datasets. Further, we evaluate the multimodal models in the linear-probe [27] and zero-shot settings. For the linear probe setup, please refer to the CLIP paper. It is done on the ImageNet dataset and then evaluated on all of these.

Results. Fig. 3 (left) shows analysis of all the architectures on ImageNet-C data with varying severity levels, for ImageNet pretrained unimodalities (solid lines) and Zero-Shot CLIP-models (dashed lines). They all suffer similarly when severity is increased and even though start from different accuracies, converge to similar values at high severity. This implies under high perturbations they break-down equally. However, the robustness of CLIP based models is slightly higher. One possible reason is that they start from lower values of clean accuracy attributed to their zero-shot nature. Fig. 2 shows the analysis of various CLIP model architectures on the ImageNet-R, ObjectNet and ObjectNet-C datasets under both Linear Probe and zero-shot settings along with the unimodal (ImageNet pretrained) counterparts. For the linear probe setting, the models maintain accuracy on ImageNet-R and ObjectNet datasets whereas suffer significantly on ObjectNet-C. On the other hand, zero shot models show better accuracy on the ImageNet-R (compared to ImageNet), slightly lower on the ObjectNet dataset and suffer on the ObjectNet-C dataset similar to linear probe setting. From the results, it can be observed that zero-shot CLIP is more robust on ObjectNet-C than linear probe CLIP based on the relative drop on accuracy. Also, the zero-shot CLIP model outperforms the linear probe one on the ImageNet-R dataset, even though the linear probe was on ImageNet. For unimodal case, all models observe significant drop in accuracy and perform poorly (much worse than CLIP Linear Probe and Zero-Shot) under all the shifts.

Self Supervised Unimodalities. We further analyse another case where the unimodal models are trained in a self supervised fashion on large datasets as against the ImageNet pretrained models. For this, we use the masked autoencoders [8] and DINO V2 [24] models proposed recently, shown to be highly effective in representation learning. Fig. 3 (mid and right) provides their robustness analysis on the ImageNet-C,R datasets alongside the CLIP models. For ImageNet-C, similar to Fig. 3 (left), these models also converge similar to the multi-modal models at the highest severity levels and their relative drop from severity level 1 to 5 is higher. On ImageNet-R, again, multi-modal models perform significantly better than the uni-modal models. These observations are similar to the case seen in Fig.2.

Empirical Conclusion. From all the plots, it can be observed that multi-modal networks are much better than the unimodal counterpart on ImageNet-R, ObjectNet and ObjectNet-C datasets and can be seens as more robust for these settings. However, they also see a significant drop in accuracies under the perturbations in ImageNet-C, especially at higher accuracy levels. Again all of the architectures used show similar drops in accuracy. Also, zero-shot multi-modal networks seem to be more robust than their linear probe counterpart in the presence of distribution shifts (comparing the drop in accuracy from ImageNet to other datasets). On the architectural side, transformer models appear to be more robust than the ResNets for both single and

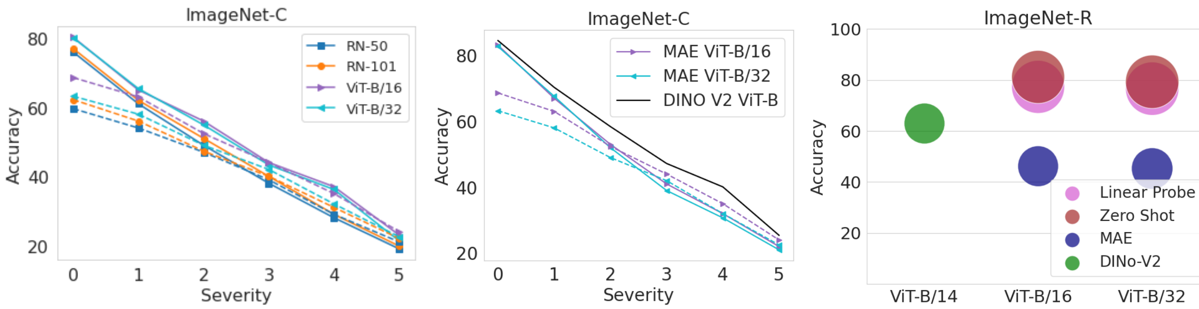

Figure 3. **Left:** Comparison of accuracy score (y-axis) on ImageNet-C dataset against various severity levels (x-axis) of perturbations, including both Unimodal (solid line) and Multi-Modal CLIP (dashed line) architectures. Unimodal architectures are ImageNet-pretrained and Multi Modal architectures correspond to Zero-shot CLIP. **Mid:** Comparison of Self-Supervised unimodalities and CLIP Zero-Shot models on the ImageNet-C benchmark. **Right:** Comparison of self-supervised unimodalities and CLIP Linear Probe and Zero Shot models on the ImageNet-R dataset.

multi-modalities, given their higher accuracy.

4. Methodology

Problem Description. The goal of our work is to make large pre-trained computer vision models robust without sacrificing their original properties. This is critical because we want models to work well both in in-domain and out-of-domain settings. Computational efficiency is also important because pre-training already requires massive amounts of compute, so efficient techniques are more versatile.

A popular technique to robustify a model is to use advanced augmentation techniques like aug-mix [12] or pix-mix [14]. This involves fine-tuning model parameters using the augmented dataset and is very effective at improving robustness. However, such fine-tuning is sub-optimal, as models could forget their original representation properties and at the same time require large computation resources.

Method Overview: To this end, we propose a novel Plug-and-Play method. First, alongside the original clean classification head, we plug a robust and a combined head into the model. Second, we induce robustness from a *small robust teacher* (here the small is relative to the large pretrained model) into the robust head. However, this leaves the challenging task of ensuring that clean accuracy is preserved in the clean head and robustness induced into the robust head. More importantly, we need to ensure that these heads can be correctly selected at test time. Third, we propose a novel uncertainty-aware knowledge distillation technique, which allows us to fulfil all our required objectives. The proposed method is discussed below and also shown in the Fig. 4.

4.1. Augmented Architecture

Let us denote the original model as $\mathcal{M}$. The network can be seen as made of three components: $\mathcal{M}_b$ the *backbone* network spanning from initial layer to the K^{th} layer, $\mathcal{M}_s$ the *shared tunable section* spanning $(K+1)^{th}$ layer to $(N-1)^{th}$ layer, and $\mathcal{M}_h$ the *prediction head section* from N^{th} layer till the end (refer figure 4). Thus the overall network can written as:

$$\mathcal{M} = \mathcal{M}_h \circ \mathcal{M}_s \circ \mathcal{M}_b, \quad (1)$$

where θ, θ_b, θ_s and θ_h denote the respective component parameters.

To address the issue of robustness transfer with preservation of desired characteristics like clean accuracy, transfer learning, etc., we plug-in two more prediction head sections on top of the shared tunable section. This results in a total of three classification heads as shown in figure 4.

4.2. Robustness Distillation from a Small Teacher

At the core of our approach lies use of a knowledge distillation (KD) framework to induce robustness to a large network. In standard KD, knowledge is usually transferred from a large network to a small network. *Au contraire*, we provide a novel view of KD. We show that robustness can be transferred from a small robust model to large models. For the small robust teacher (denoted as $\mathcal{M}_t$), we take small image-classification models and robustify them using standard techniques (a combination of augmentation techniques AugMix [12] and DeepAugment [10]). A *small* teacher is essential for an efficient method.

4.3. Uncertainty Aware Knowledge Distillation

While we have introduced a robust head and plan to induce robustness from the small model. It is a challenging task to ensure that clean head preserves clean accuracy, robust head learns robustness and the heads are appropriately selected at test time. We next discuss a novel strategy to achieve these goals.

We update the parameters of shared-tunable θ_s and prediction sections θ_h, keeping the backbone network frozen as shown in figure 4. We use the same augmented training data here as used for robustifying the small (teacher) model. It is denoted as $\mathcal{D}^a$ and contains both clean (x^c, y) and

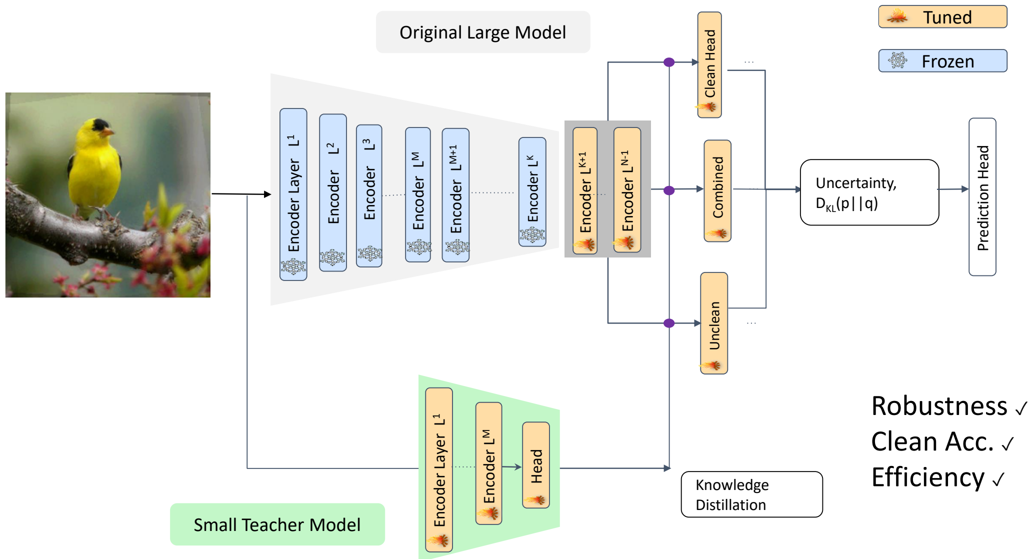

Figure 4. The end-to-end workflow of our proposed method. It involves firstly robustifying a small teacher model using advanced augmentation methods (lower stream). Then using this model along with augmented data, it tunes a small chunk of the large-scale (student) model. As described in sec. 4, we add two more heads to the student model resulting in total three heads. Yellow colored encoder denotes the (only few) tunable layers and blue colored encoders correspond to frozen layers. Finally, at the inference time, the head used for prediction is selected via estimating uncertainty in predictions and analyzing KL divergence between the distributions predicted by each head. For more details, please refer section 4.

augmented samples (x^a, y) samples. Given the augmented training data and the robust network $\mathcal{M}_t$, the parameter estimation for the model can be written as:

$$\{\theta_s, \theta_h\} \sim \mathcal{P}(\{\theta_s, \theta_h\} | \theta, \mathcal{M}_t, \mathcal{D}^a). \tag{2}$$

4.3.1 Generalized distillation

We next discuss our strategy to optimize for both knowledge (clean accuracy) and robustness distillation (robust accuracy). Note that the teacher for the clean head is a copy of the initial large model to preserve the original properties. This estimation is done using a weighted combination of classification loss $\mathcal{L}_c(x, y, \theta)$ and distillation loss [15] $\mathcal{L}_d(\theta_T, \theta_S, x)$, where $(x, y) \in D^a$, θ denotes parameters of the prediction network, θ_T denotes teacher model parameters, θ_S denotes student model parameters.

The head section (with parameters θ_h^c) corresponding to the *clean* head is updated only using the clean examples. The *combined* head (θ_m) uses both clean and unclean examples and the *unclean* head (θ_u) uses only the augmented examples. Thus, for clean examples in a randomly sampled batch of data, the set of updated parameters due to clean head section prediction is $\theta_c^c = \{\theta_s, \theta_h^c\}$ and combined head section is $\theta_m^c = \{\theta_s, \theta_h^m\}$. Similarly for augmented examples, updated parameter set due to unclean head section prediction is $\theta_u^u = \{\theta_s, \theta_h^u\}$ and combined head section is $\theta_l^u = \{\theta_s, \theta_h^m\}$.

Thus, the final loss function to update w.r.t. clean examples is (denoted as $\mathcal{L}_{clean}$):

$$\mathcal{L}_c(x, y, \theta_c^c) + \mathcal{L}_d(x, \theta_c^c, \theta_t) + \mathcal{L}_c(x, y, \theta_m^c) + \mathcal{L}_d(x, \theta_m^c, \theta_t), \tag{3}$$

and similarly for unclean examples (denoted as $\mathcal{L}_{aug}$):

$$\mathcal{L}_c(x, y, \theta_u^u) + \mathcal{L}_d(x, \theta_u^u, \theta_t) + \mathcal{L}_c(x, y, \theta_m^u) + \mathcal{L}_d(x, \theta_m^u, \theta_t). \tag{4}$$

Finally, the cost function $\mathcal{L}$ for a given batch of data can be written as:

$$\mathcal{L} = \beta \mathcal{L}_{clean} + (1 - \beta)\mathcal{L}_{aug}, \tag{5}$$

where $\beta = 1$ for clean and $\beta = 0$ for unclean examples.

4.3.2 Uncertainity aware head selection

We need a reliable head selection, such that clean head is selected for clean examples (to preserve clean accuracy) and unclean head for shifted examples (robustness).

Uncertainty. Modelling uncertainty in predictions corresponding to each of the heads can be a way to select the final head as the most certain head. Clean head should be the most certain on clean examples and similarly unclean for augmented examples. For this, we use Dropout [6] as it gives allows *Bayesian* approximation. At training time, we update the tunable portion of $\mathcal{M}$, starting from encoder L^{K+1} in Fig.4 using *dropout* regularization. This can be

done by just setting the dropout rate to some non-zero fraction in the existing implementation. This dropout is a part of all the three heads.
At the inference time, we activate the dropout, for each of the heads, to estimate a predictive distribution from the model as against a point estimate [6, 17]. We take K forward passes through each of these heads and then for each head we calculate mean and std of the outputs and use them as the mean and std of the predicted distribution from the model. This is referred as Monte Carlo Dropout. Finally, we use std directly as the measure of uncertainty ($\mathcal{U}_{mc}$).

KL Divergence. Now, there can be a case where the clean model completely breaks down for a noisy input and predicts a random output with very low $\mathcal{U}_{mc}$ (highly certain). Given the test-distribution is unknown, this can be a case with significant probability. To handle this, we also calculate the distance between the predicted distributions of each of the clean and unclean head with the combined head using KL divergence only at Inference time. This results in the following objective for selecting the final prediction head h_f:

$$h_f = \arg\min_{k \in \{c,u\}} \mathcal{U}_{mc} \cdot \mathrm{KL}(\phi_l^m(x) || \phi_l^k(x)), \tag{6}$$

where h_c, h_m, h_u correspond to clean, combined, unclean heads respectively and ϕ_l^c, ϕ_l^m, ϕ_l^u are the corresponding prediction functions. Thus, the desired head out of clean/unclean is selected using eq. 6. Note, the third head is here just to select the correct desired head for the input from the clean and unclean head via the KL divergence term in the head selection metric in eq. 6. In supplementary, we have provided a detailed ablation on the utility of each of these components and also the comparison against a naive confidence based head selection baseline.

4.4. Zero-Shot Multi-Modal scenario

The method described above can be directly applicable to uni-modalities and vision encoders of multi-modalities by attaching a classification head, similar to the Linear Probe setup [27]. We further adapt our scheme to zero-shot setup for these multi-modal networks which comprise both vision and text encoders. For this, we apply our scheme in the presence of text encoder and use the dot products between the vision encoder embeddings ($\phi_v(x)$) and the prompt embedding obtained from the text encoder ($\phi_{text}(Y)$, where Y denotes the set of prompts corresponding all classes present in the data) , $\phi_l(x) = \phi_v(x) \cdot \phi_{text}(Y)$, as the model prediction for both classification and distillation losses.

5. Experiments

We evaluate the presented approach in making large models robust on the benchmark datasets and perturbations described in the section 3 for image classification task. We show performance on clean accuracy, robust accuracy and transfer learning properties on downstream datasets.

Experimental Setup. We demonstrate results of our approach under two settings. First corresponds to using visual encoders of uni or multi-modal models (as described in linear probing approach in the Sec. 5.1) and attaching a classification head to the visual encoders. We term this as the Visual Evaluation or *VE* setup. Next, we also provide results (Sec. 5.1) in multi-modal models settings where both text and vision are used in the zero-shot (or *ZS*) setting under dataset shift. Along with clean and robust accuracy, we also compare different methods on whether they can preserve the transfer learning capabilities of the large models. The robust teacher for both settings is a single modal network trained by finetuning complete model using augmentation based techniques using a combination of augmix and deep augmentation techniques (as described in Sec. 4).

Datasets. We evaluate the presented approach on the ImageNet validation set and its perturbed variations, ImageNet-C [11], ImageNet-R [10], ImageNet-S [31] and ImageNet-A [13] for robust analysis in the VE setup. Further, we use the ObjectNet data [2] and its perturbed variation ObjectNet-C (Corrupted) version for the zero-shot evaluation tasks or ZS setup. For transfer learning experiment, we show results five datasets (Tiny-ImageNet [21], Flowers [23], PLACES025 [22], iNaturalist2021 [30] and SUN397 [34]) in the VE setup. For the ZS setup, we instead show results on dataset shift on six datasets (Cars [19], Flowers [23], CIFAR100 [20], SUN397 [34], ObjectNet [2]), where the zero-shot model is directly evaluated on these datasets. More information about these datasets have been provided in the supplementary material.

Baselines and metrics. We now describe baselines used for the VE and ZS setups. For the VE experiments involving only the visual encoders, we compare against five prior approaches. First approach involves adapting the the same number of parameters as in the presented linear probe approach (Sec. 3). Second baseline involves naive finetuning full network using the current dataset. Third baseline is the visual prompt tuning approach [16] that involves adapting input prompt parameters while fixing the feature network parameters. We also compare against the recently proposed WISE [33] framework for finetuning large networks. Finally, we define a new baseline, Augmentation based Partial Fine-Tuning or *APT* to show the effectiveness of multi-headed scheme. It involves directly updating the small part of the large network (same number of tunable layers as ours), using the augmentation based technique.

We further define two more baselines, as variations of our proposed scheme, to further highlight its importance.

		Distribution Shifts						Transfer Learning					Avg.	Latency
	IN	IN-V2	IN-R	IN-Sketch	ObjectNet	IN-A	IN-C	Tiny-IN	Flowers	Places	iNaT	SUN	shifts	(ms/img)
CLIP ViT-L/14@336px														
Zero-Shot [27]	76.2	70.1	88.9	60.2	70.0	77.2	60.6	**85.2**	**98.8**	**74.87**	**68.20**	**82.20**	71.7	-
Fine-Tuning (LP) [27]	85.4	75.8	84.1	57.4	66.3	75.3	57.9	**85.2**	**98.8**	-	-	-	69.5	320
Fine-Tuning	86.1	76.6	79.7	57.7	63.4	65.4	52.1	83.5	97.6	72.12	65.70	78.89	65.8	2500
VPT [16]	85.8	74.2	80.1	56.9	63.9	66.1	52.6	85.2	98.6	-	-	-	65.5	500
WISE-FT (LC) [33]	85.1	76.6	85.2	63.0	71.0	79.5	62.1	85.1	98.4	73.98	67.45	81.12	72.9	570
WISE-FT(E2E) [33]	**86.9**	**79.5**	**90.1**	65.0	72.1	80.6	62.9	83.4	97.4	72.94	66.28	80.23	75.0	3950
K.D. from ViT-B/16 teacher														
Single-Teacher	84.7	76.2	88.2	61.1	68.7	76.3	60.9	83.3	97.1	-	-	-	71.9	530
Only K.D.	85.1	77.3	89.3	63.9	70.6	78.6	61.7	85.1	**98.8**	-	-	-	73.5	620
Ours	85.4	79.1	89.9	**65.8**	**73.2**	**80.9**	**64.9**	**85.2**	98.7	74.69	**68.20**	82.10	**75.6**	750

Table 1. **Visual Evaluations results.** Comparison with existing robustification methods and complete fine-tuning on various distribution shift and transfer learning benchmarks for CLIP model. The column *Avg. shifts* shows the average accuracy overall the six distribution shifts. Last column shows training cost per image for all the models.

The first baseline, *Only K.D.*, involves doing knowledge distillation directly from the Small Teacher Network (STN) teacher to Large Learner Network (LLN) student without using the proposed multi-headed architecture and copy of initial LLN as teacher for clean examples. The second baseline, *combined head*, involves using copy of initial LLN as teacher for clean examples and STN for augmented, but without the multi-headed architecture, *i.e.* requiring only the combined head.

For the ZS setting, the set of baselines involves the existing zero-shot multi-modal network, along with the APT baseline (ZS-APT) and complete fine-tuning baseline (ZS-Full tuning). Both these baselines are tuned similar to our method, in presence of the text encoder, as described in section 4 (Zero Shot Multi-Modal scenario).

We use the accuracy metric on clean and perturbed dataset to evaluate each of the methods. Furthermore, we also calculate the robustness metrics for evaluating under perturbations, proposed for the ImageNet-C dataset in the supplementary.

5.1. Evaluating Multi Modality Trained Methods

Visual Evaluation. Table 1 shows the comparison of our method with the baselines under various distribution shifts for ImageNet dataset along with the transfer learning capabilities and training time per image. It uses the CLIP ViT-L/14@333px model for all the methods. Our method uses ViT-B/16 as the teacher model. It can be observed that even though WISE-E2E performs best for two shift scenarios, it suffers from high training time and poor performance under transfer learning tasks (accuracy drop of 1.8 and 1.4) which is a bottleneck with E2E fine-tuning methods. On the other hand, the methods like VPT which require low training time perform poorly under distribution shift (average accuracy drop greater than 5% when compared with Zero-Shot model). On the other hand, performs best on four distribution shift tasks boosting up accuracy upto 2% (ImageNet-C) and is able to achieve the same accuracy as the zero-shot model on the transfer learning experiments and also in a significantly lesser time compared to WISE-E2E (approximately 5 times faster).

We further evaluate our method against the APT baseline, we defined, for various model architectures as Teachers and Learners. Figure 5 (top left and bottom left) shows this analysis, with visual encoder of four CLIP models (RN-50, RN101, ViT-B/16 and ViT-B/32) as students and single modal networks as teacher evaluated on ImageNet-R and ImageNet-C datasets. The rows with teacher as *None* correspond to the APT baseline. For ImageNet-C, accuracy is improved by more than 3% for most cases, and is atleast 2.5 % for all cases. This knowledge transfer doesn't rely much on the architectural family of student or teacher as only marginal difference is there in the improvements offered by ViT or ResNet architecturs as teachers on CLIP ViT/ResNet students (less than 0.3% difference observed when updating CLIP RN-50 with ResNet-101 or ViT-Small).

For ImageNet-R, our method provides gains 2.0% for most cases with maximum as 2.9%, compared to APT baseline. For accuracy of the teacher models, please refer supplementary.

Zero Shot. We now apply our scheme on multi-modal CLIP networks using both text and visual encoder for a complete zero-shot setup, as discussed in section 4. Table 2 shows the results for our method and baselines under this setup, for various distribution shifts to the ImageNet data and also for zero-shot evaluation on new datasets (dataset shift). Here again, ViT-B/16 is used as the teacher for our method. It can be observed that our method shows best performance under all the distribution shifts with minimum difference of 1% (IN-R) and maximum of 4.5% (IN-C). Also, it is the best performing zero-shot method for four out of five dataset shifts with the maximum improvement being on ObjectNet-C 3.9%). This shows that it improves zero-shot properties of the model as compared to the complete fine-tuning which instead degrades the performance under dataset shift as observed in table 2. Table 5 (top mid

	IN	Distrbution Shifts						Dataset Shift						Avg.
		IN-V2	IN-R	IN-S	ON	IN-A	IN-C	Cars	Flowers	CIFAR100	SUN397	ON-C		shifts
CLIP ViT-L/14@336px														
Zero-Shot	76.2	70.1	88.9	60.2	70.0	77.2	60.6	78.8	**78.3**	77.5	68.4	52.2		71.1
ZS-Full Tuning	**86.5**	**77.1**	88.2	58.9	65.5	78.2	53.3	76.3	76.8	77.1	67.2	51.5		70.0
ZS-APT	84.8	76.3	89.2	60.7	68.8	77.9	62.3	77.2	77.9	77.3	67.9	54.3		71.8
Ours (ViT-B/16 teacher)	86.3	76.8	**90.2**	**62.9**	**71.6**	**74.6**	**78.9**	**78.9**	78.1	**77.6**	**69.3**	**58.2**		**73.6**

Table 2. **Zero Shot results.** Comparison with existing robustification methods and complete fine-tuning on various distribution and dataset shifts under the zero shot setup using CLIP model. Last column shows the average accuracy including all the dataset and distribution shifts. All these results are a result of zero-shot evaluation of the robustly tuned classifier model using only the ImageNet-1K dataset. No further tuning on any of the other datasets.

	IN	Distrbution Shifts						Transfer Learning		Avg.	Latency
		IN-V2	IN-R	IN-Sketch	ObjectNet	IN-A	IN-C	Tiny-IN	Flowers	shifts	(ms/img)
ViT-L/14											
Standard Training	82.8	75.3	49.4	40.4	45.2	51.9	51.5	84.5	98.1	52.3	2200
VPT [16]	82.1	74.4	47.2	38.1	45.3	51.2	50.3	84.1	97.9	51	420
WISE-FT (LC, optimal α) [33]	82.6	76.2	54.2	48.3	49.2	55.3	54.2	84.0	98.1	56.2	500
WISE-FT (E2E, optimal α) [33]	**83.6**	**78.4**	58.3	52.1	**52.3**	57.7	55.1	83.8	97.5	58.9	3400
K.D. from ViT-Small teacher											
Single-Teacher	81.9	75.9	52.7	46.3	47.4	53.2	53.3	83.8	97.2	54.8	470
Only K.D.	82.2	76.7	53.4	46.9	48.1	54.2	53.2	84.3	97.9	55.4	550
Ours	82.7	78.2	**59.1**	**52.7**	51.6	**58.5**	**58.3**	**84.5**	**98.1**	**59.7**	700

Table 3. **Unimodal results.** Comparison with existing robustification methods and complete fine-tuning on various distribution shift and transfer learning benchmarks for single-modal ViT-L/14 model, pretrained on JFT dataset. Second last column shows average accuracy over the distribution shifts and the last one shows training latency per image.

and bottom mid) shows the further accuracy analysis under this setting for various student (single modal) and teacher (CLIP) pairs on ImageNet-R, ImageNet-C datasets. The evaluation is done on the ObjectNet dataset and its perturbed version ObjectNet-C. Here, the baseline (rows with teacher as *None*) is just the zero-shot large-scale network (using text and visual encoder) and doesn't use the tuning set. Again, our scheme improves accuracy by a significant amount, atleast 2.6 percent under zero-shot setting for the dataset shift. Again it works irrespective of the architecture or modality of the teacher network (eg. CLIP RN-50 student and RN-101, ViT-S teachers in the figure).
We further analyze the effect of our method and the APT baseline on the transfer learning setup on tiny-ImageNet and Flowers dataset for various teacher-student pairs in the supplementary.

Inference time Latency. Since our method attaches extra heads, although quite small, on top of the existing model, we also analysed the inference time overhead it adds on top of the naive model. We observed that this overhead was quite small. For instance, CLIP ViT-L/14 model used in table 1, GFLOPs for baselines is 81.17 and ours is 83.7, (3% overhead).

5.2. Evaluating Unimodally Trained Methods

We now analyze our method for a unimodal *Learner* network scenario, comparing it with all the baselines on different distribution shifts/transfer learning tasks, as done for the multi-modal VE setup. Table 3 shows this comparative results. Here again, our method emerges as the winner for four out of the six distribution shifts with minimum gain of 0.8% and maximum gain of 3.2%. Furthermore, it is also able to preserve transfer learning capabilities, as shown in table 3 for Tiny-IN, Flowers, PLACES205, iNaturalist2021, SUN397 datasets, of the original model whereas other baselines (VPT and WISE) suffer.

Figure 5 (last column) shows this comparison for various Leaner-Teacher pairs consisting both ViT and ResNet architectures. Again, the rows with Teacher *None* correspond to the APT baseline. For majority cases on ImageNet-C, our method improves accuracy by greater than 3 percent, when compared to the baseline. Similarly, on the ImageNet-R dataset, it shows greater than 5% for most cases with maximum going to 3.2% for RN-50 teacher and RN-101 student. Furthermore, increasing the size of teacher model (from RN-34 to 50) results in improved performance. Finally, both our method and the baseline we compare to, make significant improvement in performance on the perturbed datasets (especially ImageNet-C), compared to the

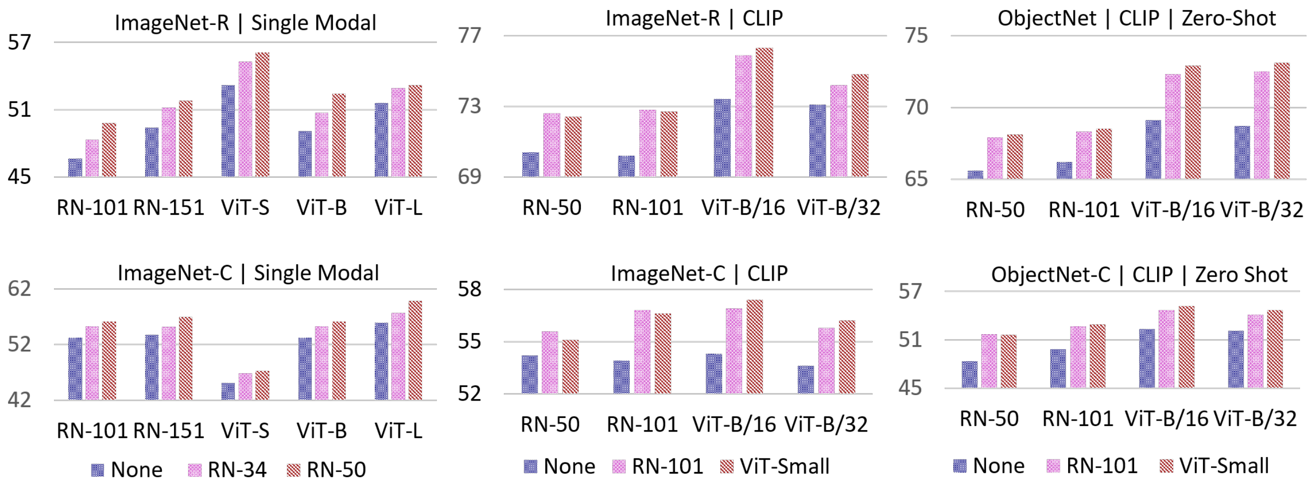

Figure 5. Analysis of our methods for various Teacher (RN-34, RN-50)-Student pairs and the APT baseline (None) applied to students. For Zero-Shot, None corresponds to the given zero-shot model without any tuning. x-axis: Students, y-axis: Accuracy, vertical bars: Teacher.

Method	ImageNet	ImageNet-C	ImageNet-R
Augmix [12]	78.3	55.2	49.1
Augmix+DA [10]	78.9	56.1	49.8
PixMix [14]	79.1	56.8	49.9
AugMix+PixMix	79.0	57.1	50.9

Table 4. Ablation of various robustification schemes used to robustify the teacher used in our approach. Here, a ResNet-101 is the student and ResNet-50 is the teacher.

Fraction-tuned	ImageNet	ImageNet-C	ImageNet-R
0	77.5	39.2	40.1
0.05	77.4	52.9	46.5
0.10	77.2	55.1	47.7
0.20	76.4	57.2	48.8
0.25	76.2	57.4	49.2
0.30	75.9	57.5	50.1

Table 5. Ablation showing variation in model performance as a function of its proportion of tuned parameters using our method, Here, ResNet-101 is the student and ResNet-50 is the teacher.

ImageNet pretrained models (refer section 3 for performance of these ImageNet-pretrained models).

5.3. Ablations

Robustification schemes. We now compare the effect of using a different robustification schemes for the teacher model, used in our method. We limit ourselves to augmentation based robustification schemes such as AugMix, PixMix, DeepAugment *etc.*. Table 4 shows the results for this comparison for the single modal setup when a ResNet-101 student model is using ResNet-50 teacher. Using PixMix or combining with AugMix does improves the accuracy by around 1-1.5 over our current technique (Augmix+DeepAugment). Our scheme shows that that gains in teacher model can be efficently transferred to the student.

Amount of parameters tuned. We further analyze the effect of tuning different proportions of the LLN using our scheme. It can be observed that tuning more parameters increases the accuracy on ImageNet-C and R datasets at the cost of clean accuracy on ImageNet dataset.

Please refer to the supp. material for more description about the ablations for understanding model design choices (KL Div., Uncertainty, Multiple-heads) and other details.

6. Conclusion

We first benchmark and discuss existing large pre-trained models under various shifts to the data. Following this, we proposed an effective method to distill robustness to these networks via small robust models, at the same time preserving the characterstics of the large models. Results on various distribution shift settings showed that our method is effective and efficient in making large pretrained models robust to several distribution shifts, and also retaining their transfer learning properties.

Limitations. Though we have provided extensive empirical evidence to demonstrate the benefit of our approach, a theoretical underpinning is missing. We leave theoretical analysis as an interesting future work.

References

[1] Abdolmaged Alkhulaifi, Fahad Alsahli, and Irfan Ahmad. Knowledge distillation in deep learning and its applications. *PeerJ Computer Science*, 7:e474, 2021. 2

[2] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset

for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019. 1, 2, 3, 6

[3] Srinadh Bhojanapalli, Ayan Chakrabarti, Daniel Glasner, Daliang Li, Thomas Unterthiner, and Andreas Veit. Understanding robustness of transformers for image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10231–10241, 2021. 2

[4] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 2

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arxiv 2020. *arXiv preprint arXiv:2010.11929*, 2010. 1, 2

[6] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR, 2016. 5, 6

[7] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021. 2

[8] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 2, 3

[9] 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. 1, 2

[10] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021. 1, 2, 3, 4, 6, 9

[11] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 1, 2, 3, 6

[12] 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. *arXiv preprint arXiv:1912.02781*, 2019. 2, 4, 9

[13] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021. 1, 2, 3, 6

[14] 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. 2, 4, 9

[15] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015. 2, 5

[16] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXIII*, pages 709–727. Springer, 2022. 2, 6, 7, 8

[17] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems*, 30, 2017. 6

[18] Samir Khan and Takehisa Yairi. A review on the application of deep learning in system health management. *Mechanical Systems and Signal Processing*, 107:241–265, 2018. 1

[19] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013. 6

[20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[21] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 6

[22] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 6

[23] M-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008. 6

[24] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 3

[25] Ajeet Ram Pathak, Manjusha Pandey, and Siddharth Rautaray. Application of deep learning for object detection. *Procedia computer science*, 132:1706–1717, 2018. 1

[26] Sayak Paul and Pin-Yu Chen. Vision transformers are robust learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 2071–2081, 2022. 2

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 6, 7

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015. 2, 3

[29] Dustin Tran, Jeremiah Liu, Michael W Dusenberry, Du Phan, Mark Collier, Jie Ren, Kehang Han, Zi Wang, Zelda Mariet, Huiyi Hu, et al. Plex: Towards reliability using pretrained large model extensions. *arXiv preprint arXiv:2207.07411*, 2022. 1

[30] Grant Van Horn, Elijah Cole, Sara Beery, Kimberly Wilber, Serge Belongie, and Oisin Mac Aodha. Benchmarking representation learning for natural world image collections. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12884–12893, 2021. 6

[31] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 2, 3, 6

[32] Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019. 3

[33] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7959–7971, 2022. 2, 6, 7, 8

[34] 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. 6

[35] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Bin Xiao, Ce Liu, Lu Yuan, and Jianfeng Gao. Unified contrastive learning in image-text-label space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19163–19173, 2022. 1, 2

[36] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Zi-Hang Jiang, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 558–567, 2021. 2

[37] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019. 2

[38] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12104–12113, 2022. 2

[39] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18123–18133, 2022. 1, 2

[40] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 2