# Pre-trained Model Guided Fine-Tuning for Zero-Shot Adversarial Robustness

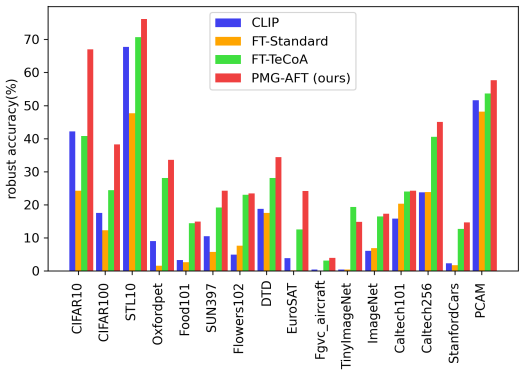
Sibo Wang[1,2] Jie Zhang[1,2] Zheng Yuan[1,2] Shiguang Shan[1,2]
[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
{wangsibo22z, zhangjie, sgshan}@ict.ac.cn , zheng.yuan@vipl.ict.ac.cn

## Abstract

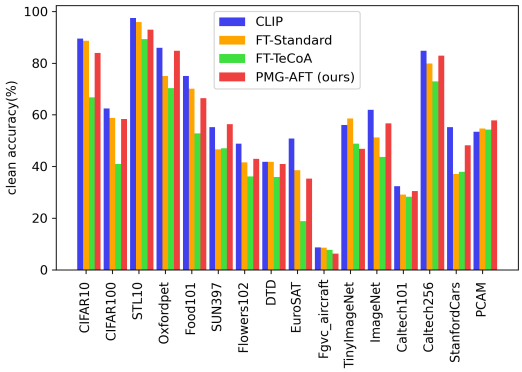
*Large-scale pre-trained vision-language models like CLIP have demonstrated impressive performance across various tasks, and exhibit remarkable zero-shot generalization capability, while they are also vulnerable to imperceptible adversarial examples. Existing works typically employ adversarial training (fine-tuning) as a defense method against adversarial examples. However, direct application to the CLIP model may result in overfitting, compromising the model's capacity for generalization. In this paper, we propose Pre-trained Model Guided Adversarial Fine-Tuning (PMG-AFT) method, which leverages supervision from the original pre-trained model by carefully designing an auxiliary branch, to enhance the model's zero-shot adversarial robustness. Specifically, PMG-AFT minimizes the distance between the features of adversarial examples in the target model and those in the pre-trained model, aiming to preserve the generalization features already captured by the pre-trained model. Extensive Experiments on 15 zero-shot datasets demonstrate that PMG-AFT significantly outperforms the state-of-the-art method, improving the top-1 robust accuracy by an average of 4.99%. Furthermore, our approach consistently improves clean accuracy by an average of 8.72%. Our code is available at here.*[1]

## 1. Introduction

Vision-language models pre-trained on large dataset have achieved impressive success in numerous tasks, such as image classification [22, 36], text-to-image generation [37, 38], image caption [7, 27]. They have also showcased excellent zero-shot generalization ability. As for CLIP [36], given a test image and a set of candidate class labels, it computes the similarity between the image embedding and the embedding of each candidate class label and predicts the class as the one with the highest similarity. While contemporary researches predominantly focus on enhancing their performance [27], comparatively less attention is devoted to investigating their robustness problem [50]. Deep models, including those at a large scale, are well-known to be vulnerable to adversarial examples [18, 30]. By introducing deliberately designed imperceptible perturbations, it is easy to induce model failures. Since an increasing number of large-scale models are deployed in security-related downstream tasks, it is imperative to enhance the robustness of such models. To address the threat of adversarial examples,




(a) Robust Accuracy




(b) Clean Accuracy

Figure 1. Zero-shot robust accuracy (a) and clean accuracy (b) of CLIP and CLIPs fine-tuned on TinyImageNet [12] using various methods across multiple datasets. FT-Standard: CLIP fine-tuned on clean samples. FT-TeCoA: CLIP fine-tuned using [32]. PMG-AFT (ours): CLIP fine-tuned by our method.

[1]https://github.com/serendipity1122/Pre-trained-Model-Guided-Fine-Tuning-for-Zero-Shot-Adversarial-Robustness

numerous defense methods have been proposed. Among them, adversarial training [30, 47] is considered one of the most effective defense strategies. It incorporates adversarial examples into the training dataset during the training phase, significantly enhancing the robustness of deep neural networks.

Undertaking adversarial training from scratch for large-scale models, however, may be an impractical approach. Adversarial training is computation-consuming since it necessitates the concurrent generation of adversarial examples, especially in models with massive parameters. For large-scale models, fine-tuning is often employed to adapt to downstream tasks, significantly reducing the required computational resources. Therefore, applying adversarial training techniques to fine-tuning is an effective method to enhance the robustness of large models. However, if we employ adversarial fine-tuning directly for large-scale models, there may be a tendency for the model to overfit the fine-tuning dataset [32]. This overfitting leads to the model losing the superior generalization capabilities garnered during its pre-training phase.

A recent study [32] explores a similar problem, defining it as zero-shot adversarial robustness. It investigates the zero-shot generalization ability of the CLIP model when dealing with adversarial examples. From the perspective of adversarial examples generation, text supervision is introduced for generating adversarial examples, which are used for adversarial fine-tuning. As illustrated in Fig. 1a, the method with text supervision, *i.e.*, FT-TeCoA [32], indeed improves the zero-shot robust accuracy across numerous unseen datasets, compared with both the original CLIP and the CLIP fine-tuned on clean datasets, which we refer to as FT-Standard. However, this improvement is not sufficient and comes at the cost of a substantial decrease in the accuracy of clean samples, as shown in Fig. 1b. This suggests that adversarial fine-tuning still faces challenges in improving the model's adversarial robustness generalization, *e.g.*, overfitting to the target dataset. Previous methods such as linear interpolation [44] and parameter regularization [25] mitigate overfitting by introducing constraints in the parameter space. Although these methods can apply to adversarial fine-tuning and alleviate overfitting, they do not focus on improving the model's zero-shot robust accuracy as their optimization goal, thereby being limited compared with adding constraints in the feature space.

Inspired by the impressive zero-shot performance of the foundational model (*e.g.*, CLIP), we introduce the Pre-trained Model Guided Adversarial Fine-Tuning (PMG-AFT), a novel method enhancing both the generalizability and adversarial robustness of models. Besides the cross-entropy loss between adversarial examples and true labels used in original adversarial training [30], PMG-AFT incorporates additional constraints from the pre-trained model and clean examples into the objective function, encouraging the target model to retain the generalized information learned during pre-training. Specifically, PMG-AFT conducts an auxiliary generalization information branch, which minimizes the distance between the adversarial example outputs in the target model and the pre-trained model. A regularization loss is also introduced to further enhance the model's adversarial robustness generalization capabilities. We conduct extensive experiments on additional 15 zero-shot datasets to evaluate our method. Our method outperforms the state-of-art methods with a significant improvement of up to 4.99% in terms of average robust accuracy. Moreover, attributed to the effectiveness of the generalization information branch, our method also achieves an 8.72% increase in average accuracy on clean samples, demonstrating the superiority of our PMG-AFT approach again.

Our main contributions are summarized as follows:

- We propose the PMG-AFT method, which learns generalization features from the original pre-trained model, effectively mitigating overfitting and enhancing the zero-shot adversarial robustness of the CLIP model.
- Our method introduces improvements during the parameters update phase of adversarial fine-tuning and can effectively combine with the defense framework that originates from the perspective of adversarial example generation.
- Extensive experiments demonstrate that PMG-AFT consistently outperforms the state-of-the-art in terms of zero-shot robust accuracy and clean accuracy.

## 2. Related Work

**Pre-trained Vision-language Model.** In recent years, the fusion of vision and language understanding has garnered widespread attention. Inspired by the success of pre-trained language models like BERT [13] and GPT [7], the rise of Self-Supervised Learning (SSL) [2] in computer vision has started to produce task-agnostic models with foundational characteristics, which we refer to as "foundational models." For instance, models such as SimCLR [8], VL-bert [42], DINO [6], and DINOv2 [34] learn representations from unlabeled images and have demonstrated impressive flexibility in addressing various downstream tasks. Furthermore, models like CLIP [36], SWAG [41], ALBEF [26], BLIP [27] incorporate a vision-language component during pretraining, further enhancing downstream adaptability via zero-shot learning. Contrary to earlier works that aimed at improving performance on standard benchmarks, our study takes a different direction by focusing on the adversarial robustness problem of such pre-trained frameworks and improving their zero-shot adversarial robustness.

**Adversarial Robustness.** Deep neural networks have been found vulnerable to adversarial attacks [1, 5, 15, 18, 30], which delicately generate imperceptible noises added to the original images leading to model misclassifications. To en-

hance the robustness of neural networks against adversarial examples, a series of defense strategies have been proposed. Among them, adversarial training [30, 40, 45, 47] is considered the most effective defense mechanism, which employs adversarial examples during the training phase and integrates them into the training dataset. It notably enhances the robustness of deep neural networks against adversarial attacks. With the rise of large-scale pre-trained vision-language (VLP) model, the robustness of VLP model has gradually come under investigation, and numerous attack algorithms targeting them have emerged [21, 29, 48, 50]. In our study, we explore the performance of adversarial training on VLP models, aiming to enhance their adversarial robustness for various downstream tasks. Mao et al. [32] investigates this issue from the perspective of adversarial example generation and proposes an adversarial fine-tuning algorithm supervised by text. Li et al. [28] adjusts the embedding of the textual modality, reducing the similarity between labels, thus enhancing robustness. In contrast to their approaches, we tackle the issue from the perspective of the training process itself, encouraging the adversarial fine-tuned model to leverage the generalized features from the original model for improving the adversarial robustness.

**Finetuning and Catastrophic Overfitting.** Fine-tuning is a prevalent strategy aimed at adapting pre-trained models to specific downstream tasks [13, 16]. However, when fine-tuning pre-trained vision-language models, one may encounter overfitting issues. During this process, the target model might deviate significantly from the pre-trained one, leading to overfitting on small-scale fine-tuning datasets [14, 49]. Within the adversarial training framework, the issue of overfitting still exists and is referred to as "adversarial overfitting" [39, 43]. In this situation, the model is influenced by the distribution of adversarial examples, leading to a significant decline in both robust accuracy and clean accuracy. Methods to address overfitting include linear interpolation [44], and parameter regularization [25]. These methods focus on parameter space to constrain the distance between two models while our method addresses overfitting by encouraging memory in the feature space.

# 3. Methodology

We first give background in Sec. 3.1 on adversarial attacks, adversarial training, and zero-shot adversarial robustness. In Sec. 3.2, we discuss that adversarial fine-tuning can lead to model overfitting, damaging the generalization of the target model. Finally, Sec. 3.3 provides a detailed introduction to our Pre-trained Model Guided Adversarial Fine-Tuning (PMG-AFT) method, including its various components and the loss function.

## 3.1. Preliminaries and Problem Setup

In this paper, we select the image classification task to introduce our method, yet it can be also extended to other tasks. For large-scale pre-trained vision-language (VLP) models, we choose the CLIP model, one of the typical VLP models for zero-shot recognition, as our base model. Let $F_\theta(\cdot)$ represent the CLIP image encoder parameterized by $\theta$ and $T_\phi(\cdot)$ represent the CLIP text encoder parameterized by $\phi$. Given an input image $x$ and a textual description about its category, such as "This is a photo of a {}", denoted as $t$, the model will provide an image representation $F_\theta(x)$ and a text representation $T_\phi(t)$. For image classification, we compute the similarity between $F_\theta(x)$ and each candidate category embedding $T_\phi(t_i)$, where $t_i$ represents a certain prompt of one category, to obtain a $c$-dimensional output, where $c$ represents the number of categories, and the category with the highest similarity is selected as the classification result. For training or fine-tuning, the model updates its parameters by minimizing the cross-entropy loss, $L([\text{sim}(F_\theta(x), T_\phi(t_1)), \ldots, \text{sim}(F_\theta(x), T_\phi(t_c))], y)$, where $y$ is the one-hot vector label. For convenience of representation, we denote it as $L(x, t, y)$.

**Adversarial Attacks.** Adversarial attacks typically refer to adding an imperceptible, optimizable perturbation to a clean image, misleading the model to produce an incorrect prediction. A well-known white-box attack method is Projected Gradient Descent (PGD) [30], which uses multi-step gradient ascent steps to maximize the cross entropy loss while projecting intermediate perturbation to the specified region constrained by the p-norm:

$$\begin{aligned} x_{k+1} &= x_k + \alpha \nabla_{x_k} L(x_k, t, y), \\ x_{k+1} &= P_{B(x,\varepsilon)}(x_{k+1}), \quad k = 0, \ldots, K-1, \end{aligned} \tag{1}$$

where $x_0 = x + \varepsilon_0$, $\varepsilon_0 \sim U[0, \varepsilon]$, $B(x, \varepsilon) = \{x| \ ||x - x_k||_p < \varepsilon\}$, $\varepsilon$ represents the perturbation bound under p-norm . In the subsequent descriptions of the paper, we denote the adversarial examples as $x_a$.

**Adversarial Fine-Tuning.** Adversarial training formulates the training process into a min-max problem and can be described as:

$$\min_\theta \mathbb{E}_{x,y\sim P_D} \left[ \max_{x_a \in B(x,\varepsilon)} L(x_a, t, y) \right]. \tag{2}$$

The inner maximization problem aims to find a stronger adversarial example, while the outer minimization problem optimizes the model parameters so that the model $F_\theta$ still makes correct predictions on the adversarial examples. It should be noted that here we only adjust the parameters of the image encoder $\theta$, while the parameters of the text encoder $\phi$ remain unchanged. Since fine-tuning is generally used to adapt CLIP to downstream tasks, such an objective function (2) can also be applied to the fine-tuning of CLIP

towards robustness, and we refer to it as adversarial fine-tuning.

**Zero-Shot Adversarial Robustness.** Whether the excellent generalization capabilities of large-scale visual-language models on new tasks and datasets can remain consistent in the field of adversarial defense is a question worth investigating. We use zero-shot robust accuracy to measure this type of generalization. During the training phase, we select a target dataset such as TinyImageNet [12] and perform adversarial fine-tuning on the model. During the testing phase, we generate adversarial examples using white-box attacks to test CLIP, evaluating the zero-shot accuracy of these adversarial examples across multiple datasets.

### 3.2. Adversarial Fine-Tuning is Prone to Overfitting

It is well-known that pre-trained models tend to overfit during fine-tuning, especially when the dataset for the downstream task is small [14]. During the fine-tuning process, a target model continuously deviates from the pre-trained model to adapt to the downstream task. From Fig. 1b, we can easily observe that the model fine-tuned on clean TinyImageNet, compared with the original CLIP, only shows improved accuracy when tested on TinyImageNet itself, while experiencing a decline on other zero-shot datasets. This demonstrates the presence of overfitting.

Adversarial fine-tuning aims to leverage adversarial examples to enhance the robustness of the model. The distribution of these adversarial examples often lies far beyond the manifold of the clean example distribution. To gain the capability of correctly classifying these adversarial examples, the target model is fine-tuned towards a solution that deviates even more from its initial optimization point, namely the pre-trained model. Fig.1 also illustrates this point. The CLIP model, after fine-tuned by FT-TeCoA [32], has indeed improved the adversarial robustness across various datasets. However, its accuracy on clean datasets has significantly decreased. This indicates that adversarial fine-tuning also experiences the issue of overfitting. To more intuitively demonstrate the overfitting phenomenon, we examined the changes in parameters during the training process, as shown in Fig. 2. As training progresses, the parameters of the standard fine-tuning and adversarial fine-tuning CLIP model both deviate from the original CLIP model, with the adversarial fine-tuning deviating to a greater extent. This overfitting phenomenon directly leads to a decrease in the model's generalization capability.

### 3.3. PMG-AFT

To mitigate overfitting and enhance generalization, adversarial fine-tuning should retain generalizable features that the pre-trained model has already captured. For example, adversarial fine-tuning favors the target model extracting features invariant to perturbations for robustness.

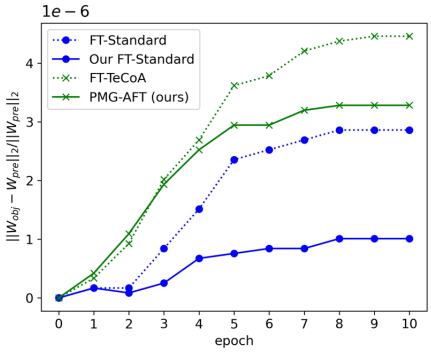



Figure 2. Relative $L_2$ distance between CLIPs fine-tuned on TinyImageNet using various strategies and original CLIP model in the parameter space. Our FT-Standard represents the application of our proposed fine-tuning method on clean target datasets.

Therefore, all the generalizable and robust information captured during pre-training are especially valuable and should be memorized. To achieve this objective, we propose PMG-AFT. PMG-AFT first obtains text embeddings from the frozen CLIP model's text encoder, then uses these text embeddings to supervise the generation of adversarial examples for adversarial fine-tuning. The entire fine-tuning process is divided into two branches: the robustness information branch aims to enhance the model's robustness for a specific task, while the generalization information branch aims to retain the generalization capabilities of the original CLIP model. The pipeline of PMG-AFT is shown in Fig. 3, and we will describe the design of PMG-AFT in detail.

**Text Embedding Extraction.** As mentioned in Sec. 3.1, we use the text encoder from the pre-trained CLIP model, denoted as $T_{ori}(\cdot)$ to extract text embeddings of the category prompts we designed, which are used for the generation of adversarial examples and supervision information for adversarial fine-tuning. We organize the text embeddings into the form of a matrix, denoted as $T \in \mathbb{R}^{c \times dim}$, where $c$ is the number of categories, $dim$ is the dimension of embedding, each row represents the embedding of a category. It's worth noting that our primary focus is on attacks in the image modality. Hence, the text modality encoder is frozen and does not participate in the adversarial fine-tuning parameter updates.

**Adversarial Attacks Generation.** Adversarial fine-tuning requires the adversarial examples generated from the training dataset. In the phase of generating adversarial examples, we use the PGD [30] method described in Equ. (1). The specific calculation process is as follows. We pre-set the attack step size $\alpha$, the number of iterations $K$, and the perturbation range $\varepsilon$. We denote the image encoder of the target model as $F_\theta(\cdot)$, the embeddings of a batch of natural images $X$ as $I \in \mathbb{R}^{N \times dim}$, where $N$ is the size of a batch, $dim$ is the dimension of embedding, each row represents the embedding of an input image. Based on the inference process of CLIP, we calculate the similarity matrix through

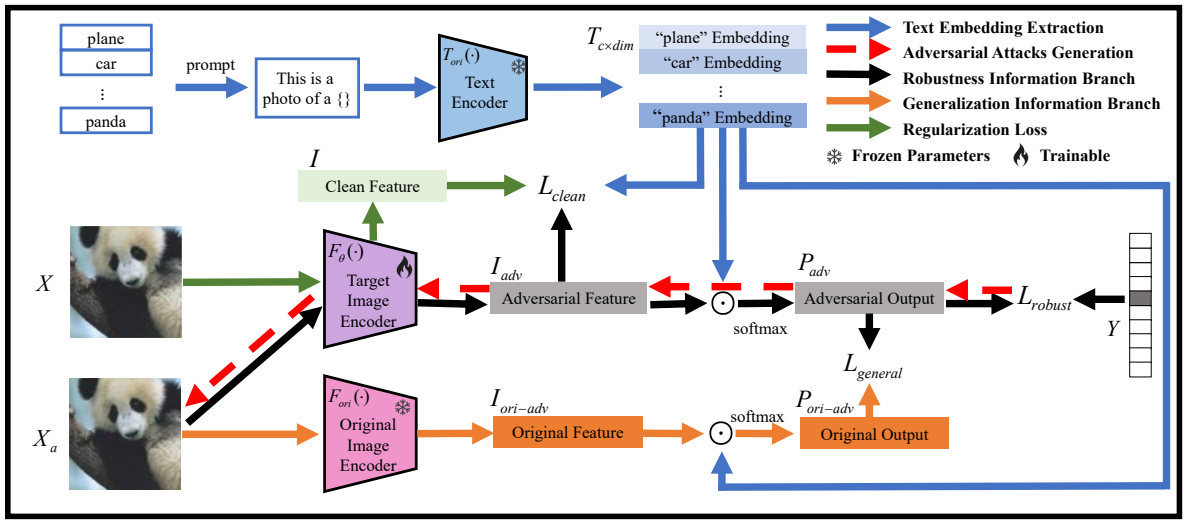



Figure 3. The pipeline of PMG-AFT. PMG-AFT first uses the text encoder from the pre-trained CLIP model to obtain text embeddings, then employs TeCoA[32] loss which is $L_{robust}$ in our method to generate adversarial examples. During the model parameter update phase, we split into two branches: the robustness information branch, which maximizes the similarity between the output of the target model and the GT via $L_{robust}$, generalization information branch maximizes the output of the adversarial samples between the target model and the original model via $L_{general}$. A regularization loss ($L_{clean}$) is applied to the adversarial and clean outputs. Only the image encoder of the target model can be trained and the adversarial examples generation alternates with parameters updating. $\odot$ means matrix inner product.

the matrix inner product:

$$S = I \cdot T^\top. \tag{3}$$

Next, to transform these similarity scores into a probability distribution, we apply the softmax function:

$$P_{ij} = \frac{\exp(S_{ij})}{\sum_{k=1}^{c} \exp(S_{ik})}, \tag{4}$$

where $P_{ij}$ is the predicted probability of the $i$-$th$ image belonging to the $j$-$th$ class. Finally, given the one-hot ground truth label $Y$ (with a size of $N \times c$) of a batch, the cross-entropy loss can be represented as:

$$L_{CE}(X, T, Y) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{c} Y_{ij} \log(P_{ij}). \tag{5}$$

By incorporating the loss function and the aforementioned hyperparameters into Equ. (1), we can obtain the adversarial examples $X_a$.

**Robustness Information Branch.** To enhance the robustness of the target model, we need to update the model parameters through the outer minimization problem in Equ. (2). Its goal is to minimize the model's loss on adversarial examples. The most intuitive and effective loss is the classification cross-entropy loss on the adversarial examples, encouraging the target model to correctly classify adversarial examples to achieve robustness. We take the adversarial examples $X_a$ as input and minimize the loss function in Equ. (5) to optimize the parameters $\theta$ of the image encoder $F_\theta(\cdot)$, obtaining one term of the loss function,

$$L_{robust} = L_{CE}(X_a, T, Y). \tag{6}$$

It encourages the similarity of the adversarial visual feature to be as close as possible to the textual representation of the true class label.

**Generalization Information Branch.** As adversarial examples are generated based on a specific dataset, the robustness features acquired by the target model are confined and may excessively specialize to a particular downstream task dataset, consequently compromising generalizability. Our method introduces a generalization information branch to improve the generalization of adversarial robustness. The main body of this branch is an image encoder of the original pre-trained CLIP, which is denoted as $F_{ori}(\cdot)$. $I_{ori-adv}$ denotes the embeddings of adversarial images it produces. Since the pre-trained model is a fixed deterministic function, the supervision objective of this branch essentially encourages the target model $F_\theta(\cdot)$ to output features that can predict the information of the original foundational model as much as possible, thereby helping to mitigate the problem of overfitting. We feed the adversarial examples $X_a$ into both the target model and the original pre-trained model, obtaining both prediction results based on the text embeddings $T$.

$$\begin{aligned} P_{adv} &= \text{softmax}(I_{adv} \cdot T^\top), \\ P_{ori-adv} &= \text{softmax}(I_{ori-adv} \cdot T^\top). \end{aligned} \tag{7}$$

We measure the distance between them using the KL divergence, aiming to minimize this distance as much as possible

in order to learn knowledge from the original model.

$$D_{KL}(P||Q) = \sum_i P(i) \log \left( \frac{P(i)}{Q(i)} \right), \tag{8}$$

$$L_{general} = \frac{1}{N} \sum_{j=1}^{N} D_{KL}(P_{adv_j}||P_{ori-adv_j}), \tag{9}$$

where $P(i)$ represents the $i\text{-}th$ element in the output probability tensor $P$, and $j$ represents the $j\text{-}th$ sample in a batch.

**Loss Function.** In addition to the two terms of loss mentioned above, we further introduce a regularization loss. Specifically, it encourages the features of adversarial images to be similar to those of clean images in the target model:

$$\begin{aligned} P_{clean} &= \text{softmax}(I \cdot T^{\top}), \\ L_{clean} &= \frac{1}{N} \sum_{j=1}^{N} D_{KL}(P_{adv_j}||P_{clean_j}). \end{aligned} \tag{10}$$

The regularization loss is independent of the labels of the training examples and helps to maintain the generalization ability of the original CLIP model. Combined with generalization information branch, the regularization loss can further enhance the model's adversarial robustness on unseen categories. See Sec. 4.6 for detailed results.

In summary, the loss function during the training process is $L = L_{robust} + \alpha L_{general} + \beta L_{clean}$, where $\alpha$ and $\beta$ are hyper-parameters.

Our algorithm iteratively alternates between generating adversarial examples and updating the model parameters. As PMG-AFT utilizes additional information from the original model to correct the fine-tuning process of the target model, it helps the model maintain adversarial robustness in terms of zero-shot generalization capability.

# 4. Experiments

## 4.1. Experimental Setup

**Datasets.** We fine-tune the CLIP model on the TinyImageNet [12] dataset, and conduct evaluations on TinyImageNet [12] as well as additional 15 zero-shot datasets, reporting their robust accuracy and clean accuracy. Specifically, these 15 datasets fall into 5 categories: general object recognition including CIFAR10 [24], CIFAR100 [24], STL10 [10], ImageNet [12], Caltech101 [17], and Caltech256 [19]; fine-grained recognition such as OxfordPets [35], Flowers102 [33], FGVCAircraft [31], and StanfordCars [23]; scene recognition represented by SUN397 [46]; domain-specific data which includes Food101 [4], EuroSAT [20], and DTD [9]; medical image, which in this case is PCAM [3]. We also conduct experiments with the same settings on the CIFAR100 dataset and evaluations on the same 16 datasets. To accommodate the input requirements of the CLIP model, they have all been preprocessed to a size of $3 \times 224 \times 224$.

**Baseline.** Considering that research on zero-shot adversarial robustness is in its nascent stages with a limited number of available methods, our primary comparisons are focused on evaluating our approach against the current SOTA method named FT-TeCoA [32]. Following FT-TeCoA [32], we also employ two adaptation methods, *i.e.*, fine-tuning and visual prompt for comprehensive evaluations.

**Implementation Details.** We utilize the ViT-B/32 architecture of the CLIP model as the backbone and use the SGD optimizer to adversarially finetune the target model for 10 epochs. For fine-tuning, we update all parameters of the image encoder with a learning rate of 5e-5. For visual prompt, we introduce learnable parameters as prompts, adding them to both the image layer with the same dimensions as the original image and the token layer with a dimension of 100. The learning rate for the visual prompt is set at 40. We use $l_\infty$ norm PGD-2 [30] and PGD-10 [30] attacks with perturbation bounds of $\varepsilon = 1/255, 2/255$ and $4/255$ for both adversarial training and evaluation, respectively. For hyper-parameters, we set $\alpha$ to 1 and $\beta$ to 1. Each model in this paper is trained on two NVIDIA GeForce RTX 3090 GPUs.

## 4.2. Main Result

We fine-tune the model on TinyImageNet and subsequently evaluate it on all 16 datasets. It's noteworthy that, on datasets other than TinyImageNet, the evaluation is conducted in a zero-shot manner. During training and evaluation, we use the PGD-10 [30] attack with a perturbation bound $\varepsilon = 1/255$. The robust accuracy results are shown in Tab. 1, and the clean accuracy results are displayed in Tab. 2. We bold the best robust accuracy results for each dataset.

From Tab. 1, we can see that our method shows an average improvement in robust accuracy of 14.57% compared with the original CLIP model. Compared with FT-TeCoA, the current state-of-the-art, our method shows an average improvement in robust accuracy of 4.99%, and achieves improvement on the most of datasets except TinyImageNet. However, the test on TinyImageNet is not a strict zero-shot test, which also indicates that our method effectively mitigates overfitting on specific downstream tasks. Moreover, Tab. 2 demonstrates that the improvement in robust accuracy brought by the FT-TeCoA comes at the cost of a 12.99% decrease in average clean accuracy compared with the original CLIP. However, our method outperforms FT-TeCoA in terms of clean accuracy, achieving a clean accuracy comparable to the CLIP model, slightly higher than fine-tuning on clean datasets *i.e.* FT-Standard.

For visual prompt, since it involves fewer parameters update than fine-tuning, both robust accuracy and clean accuracy are reduced across different methods. Neverthe-

Table 1. Adversarial zero-shot robust accuracies under PGD-10 [30] attack. We fine-tune the model on TinyImageNet [12] and evaluate six methods (rows) on 16 datasets (columns), presenting the accuracy for each dataset as well as the average accuracy, with the best results shown in bold. The accuracy here is percentage.

| Method | CIFAR10 | CIFAR100 | STL10 | SUN397 | Food101 | Oxfordpet | Flowers102 | DTD | EuroSAT | Fgvc_Aircraft | TinyImageNet | ImageNet | Caltech101 | Caltech256 | StanfordCars | PCAM | Average | Time (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CLIP | 42.18 | 17.57 | 67.77 | 10.49 | 3.28 | 8.98 | 4.88 | 18.75 | 3.84 | 0.39 | 0.39 | 6.09 | 15.82 | 23.76 | 2.34 | 51.61 | 17.38 | 0 |
| FT-Standard | 24.21 | 12.30 | 47.65 | 5.71 | 2.65 | 1.56 | 7.61 | 17.57 | 0.13 | 0.00 | 0.39 | 6.87 | 20.31 | 23.82 | 1.75 | 48.15 | 13.79 | 234 |
| FT-TeCoA [32] | 40.82 | 24.41 | 70.70 | 19.21 | 14.45 | 28.13 | 23.05 | 28.13 | 12.57 | 3.13 | **19.33** | 16.48 | 24.02 | 40.56 | 12.69 | 53.68 | 26.96 | 551 |
| PMG-AFT (ours) | **66.99** | **38.28** | **76.17** | **24.23** | **14.92** | **33.59** | **23.43** | **34.38** | **24.15** | **3.91** | 14.84 | **17.26** | **24.02** | **45.05** | **14.64** | **57.64** | **31.95** | 849 |
| VP-TeCoA [32] | 36.71 | 16.21 | 57.61 | 6.05 | 4.45 | 13.67 | 13.02 | **10.15** | 9.73 | 0.00 | 18.55 | 5.56 | 18.16 | 25.32 | 2.14 | 54.92 | 18.27 | 364 |
| VPT-PMG-AFT (ours) | **50.19** | **22.07** | **64.45** | **12.79** | **8.35** | **18.55** | **16.85** | 3.45 | **11.08** | **2.73** | **20.70** | **9.92** | **19.25** | **38.99** | **5.46** | **58.71** | **22.74** | 661 |

Table 2. Zero-shot clean accuracies. After fine-tuning or adversarial fine-tuning on TinyImageNet [12], the zero-shot accuracy of the CLIP model on clean images generally decreases. Compared with other fine-tuning methods, our approach achieves the best clean accuracy.

| Method | CIFAR10 | CIFAR100 | STL10 | SUN397 | Food101 | Oxfordpet | Flowers102 | DTD | EuroSAT | Fgvc_Aircraft | TinyImageNet | ImageNet | Caltech101 | Caltech256 | StanfordCars | PCAM | Average | Time (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CLIP | 89.52 | 62.50 | 97.46 | 55.25 | 75.00 | 85.93 | 48.82 | 41.79 | 50.84 | 8.70 | 56.03 | 61.99 | 32.42 | 84.76 | 55.27 | 53.40 | 59.98 | 0 |
| FT-Standard | 88.67 | 58.78 | 95.89 | 46.61 | 70.07 | 75.00 | 41.60 | 41.79 | 38.60 | 8.57 | 58.59 | 51.28 | 29.10 | 79.88 | 37.10 | 54.74 | 54.76 | 234 |
| FT-TeCoA [32] | 66.79 | 41.01 | 89.25 | 47.01 | 52.81 | 70.31 | 36.13 | 35.94 | 18.88 | 7.81 | 48.83 | 43.67 | 28.32 | 72.98 | 37.89 | 37.89 | 46.99 | 551 |
| PMG-AFT (ours) | 83.98 | 58.39 | 92.97 | 56.41 | 66.40 | 84.76 | 42.96 | 41.02 | 35.28 | 6.25 | 46.87 | 56.75 | 30.46 | 82.94 | 48.24 | 48.24 | **55.71** | 849 |
| VPT-TeCoA [32] | 53.51 | 26.36 | 74.41 | 17.22 | 15.85 | 33.20 | 22.39 | 12.50 | 13.77 | 0.39 | 44.92 | 15.82 | 21.26 | 42.38 | 3.32 | 58.71 | 28.50 | 364 |
| VPT-PMG-AFT (ours) | 67.38 | 33.78 | 81.64 | 29.19 | 28.51 | 49.60 | 17.18 | 13.92 | 13.92 | 7.81 | 52.14 | 26.83 | 21.31 | 61.19 | 16.99 | 58.95 | **36.03** | 661 |

less, our method consistently outperforms FT-TeCoA, further proving the effectiveness of our approach.

We also conduct experiments with the same setting on the CIFAR100 dataset, where our method achieves an average robust accuracy of 26.06%, which is an improvement of 4.71% over FT-TeCoA. Detailed experimental results can be found in the supplementary materials. Additionally, since we have introduced an extra branch, the computational overhead is increased compared to FT-TeCoA, resulting in an extra 298 seconds per training epoch. However, our PMG-AFT achieves much better results than FT-TeCoA in terms of both robustness and clean accuracies.

### 4.3. Performance against AutoAttack

AutoAttack [11], as a strong attack method, is usually used to further verify the robustness of different models. It provides a more comprehensive evaluation of robustness than a single attack algorithm. We conduct evaluations using the standard version AutoAttack [11], and present the robust accuracy at a perturbation bound $\varepsilon$ of $1/255$ in Tab. 3. Under the AutoAttack, as compared with the PGD-10 [30] attack, the robustness of the original CLIP model drops drastically, with an average robust accuracy dropping from 17.38% to 1.74% across various datasets. Our method also experiences a decline to some extent, but it still enhances the adversarial accuracy of the model compared with both CLIP and FT-TeCoA, demonstrating the effectiveness of our method again.

### 4.4. Performance against Different Attack Strength

To verify the impact of changes in adversarial perturbation bounds on our method, we increase the perturbation bound for PMG-AFT from $1/255$ to $4/255$. We sampled several datasets to conduct experiments on our method and

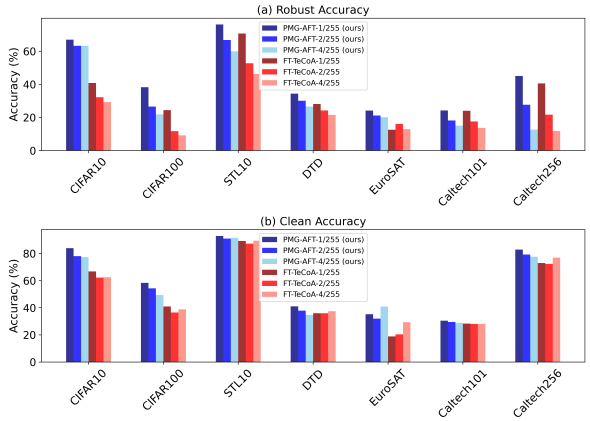



Figure 4. Zero-shot robust accuracy (a) and clean accuracy (b) under different perturbation bounds ($\varepsilon = 1/255, 2/255$ and $4/255$). We employ the same perturbation bound for training and testing.

FT-TeCoA facing different attack bounds. The results, as shown in Fig. 4 indicate that with the increase in adversarial perturbation, both our method and FT-TeCoA experience varying degrees of decline in robust accuracy. However, our defense consistently outperforms FT-TeCoA with different attack intensities. Besides, as the size of adversarial perturbation continues to increase, the clean accuracy is almost unaffected. We show detail results on all datasets in the supplementary materials.

### 4.5. Trade-off between Robust and Clean Accuracy

Adversarial training often involves a trade-off between clean accuracy and robust accuracy. Existing work has approached the trade-off from the perspective of model parameters interpolation [44]. In our experiments, we refer to this method as "interpolation". We plotted the relationship between the robust accuracy and the clean accuracy of different methods, as shown in Fig. 5. The $x$-axis repre-

Table 3. Zero-shot robust accuracies under AutoAttack [11] with $\varepsilon = 1/255$. The dataset used for fine-tuning remains TinyImageNet [12].

| Method | CIFAR10 | CIFAR100 | STL10 | SUN397 | Food101 | Oxfordpet | Flowers102 | DTD | EuroSAT | Fgvc_Aircraft | TinyImageNet | ImageNet | Caltech101 | Caltech256 | StanfordCars | PCAM | Average | Time (s) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CLIP | 8.98 | 3.90 | 13.47 | 0.07 | 0.39 | 0.00 | 0.00 | 0.00 | 0.00 | 0.39 | 0.00 | 0.00 | 0.41 | 0.15 | 0.09 | 0.16 | 1.74 | 0 |
| FT-TeCoA [32] | 9.96 | 4.68 | 23.24 | 0.21 | 2.73 | 5.44 | 5.17 | 14.85 | 9.73 | 0.39 | **16.99** | 8.85 | 16.21 | 27.08 | 4.49 | 15.26 | 10.33 | 551 |
| PMG-AFT (ours) | **31.05** | **13.28** | **44.33** | **14.47** | **10.54** | **12.78** | **8.75** | **23.56** | **15.13** | **0.78** | 13.39 | **9.07** | **16.40** | **43.16** | **12.10** | **17.84** | **17.91** | 849 |

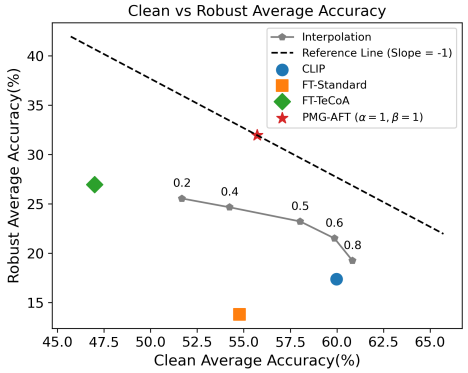



Figure 5. Trade-off between robust and clean accuracy for different fine-tuning methods. Each scatter point represents one method, and the dashed line represents a reference slope line, used to examine the optimal performance of the trade-off.

sents the average clean accuracy tested on 16 datasets for the original CLIP and various fine-tuned models, while the $y$-axis represents the average robust accuracy. It can be observed that our method achieves the most optimal balance among all the methods compared, and it outperforms all other methods in both robust accuracy and clean accuracy.

### 4.6. Ablation Study

**Contribution of Loss Function Term.** To demonstrate the effectiveness of the auxiliary branch and regularization loss introduced by our method, we incrementally added loss terms and adjusted hyper-parameters. The experimental results, as shown in 4, indicated that after introducing the generalization information branch ($L_{general}$), the model's adversarial robust generalization and clean sample generalization both improved compared with FT-TeCoA, with an average increase of 1.48% and 11.29% respectively. With the incorporation of the regularization loss, the model's robust generalization experienced a further significant increase, from 28.44% to 32.04%, which underscores the effectiveness and considerable potential of the branch we proposed. By adjusting different hyper-parameters, we found that the performance of the method is optimal when $\alpha = 1$ and $\beta = 1$. For detail experimental results, please refer to the supplementary materials.

**Impact of Feature Layer and Distance Metric.** Our PMG-AFT involves minimizing the feature-level distance, as in $L_{general}$. Therefore, the choice of which feature layer to use and how to measure the distance is crucial to our method. As shown in Tab. 5, we compared the results of PMG-AFT when applied to the output layer and penultimate feature layer using KL divergence, $L_2$ distance, and cosine distance. Due to space limitations, we only show their average robust and clean accuracies. For the detailed results on each dataset, please refer to the supplementary materials. From the experimental results, we can see that using KL divergence at the output layer yields the best results in terms of both robust accuracy and clean accuracy.

Table 4. Contribution of the loss function we propose. we incrementally added loss terms and report the average robust accuracy and clean accuracy of the model after fine-tuning.

| Method | Avg Clean Acc (%) | Avg Robust Acc (%) |
|---|---|---|
| FT-TeCoA [32] ($\alpha = 0, \beta = 0$) | 46.99 | 26.96 |
| +$L_{general}$ ($\alpha = 1, \beta = 0$) (ours) | **58.28** | 28.44 |
| +$L_{general}$+$L_{clean}$ ($\alpha = 1, \beta = 1$) (ours) | 55.72 | **32.04** |

Table 5. Average robust and clean accuracy under the selection of different feature layers and distance metric.

| Method | Avg Clean Acc (%) | Avg Robust Acc (%) |
|---|---|---|
| Output + KL | **55.71** | **31.95** |
| Output + $L_2$ | 48.21 | 27.23 |
| Feature + $L_2$ | 38.32 | 22.90 |
| Feature + COS | 52.98 | 24.53 |

## 5. Conclusion

Inspired by the powerful generalization capabilities of pre-trained models, we propose a novel adversarial fine-tuning method PMG-AFT to enhance the CLIP's zero-shot adversarial robustness. By incorporating constraints from the pre-trained model and clean examples, generalized features are learned by the target model. Our method makes improvement from the perspective of the outer minimization problem in adversarial training and can combine with adversarial sample generation algorithms. Extensive experiments demonstrate that our method has strong adversarial robustness across multiple zero-shot datasets, and achieves a better trade-off between robust and clean accuracy.

## 6. Acknowledgement

This work is partially supported by National Key R&D Program of China (No. 2021YFC3310100), Strategic Priority Research Program of the Chinese Academy of Sciences (No.XDB0680101), Beijing Nova Program (20230484368), Suzhou Frontier Technology Research Project (No. SYG202325), and Youth Innovation Promotion Association CAS.

## References


[1] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pages 274–283. PMLR, 2018. 2

[2] Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Gregoire Mialon, Yuandong Tian, et al. A cookbook of self-supervised learning. *arXiv preprint arXiv:2304.12210*, 2023. 2

[3] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes Van Diest, Bram Van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen AWM Van Der Laak, Meyke Hermsen, Quirine F Manson, Maschenka Balkenhol, et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *Jama*, 318(22):2199–2210, 2017. 6

[4] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 446–461. Springer, 2014. 6

[5] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. Ieee, 2017. 2

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 2

[7] Jun Chen, Han Guo, Kai Yi, Boyang Li, and Mohamed Elhoseiny. Visualgpt: Data-efficient adaptation of pretrained language models for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18030–18040, 2022. 1, 2

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 2

[9] 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. 6

[10] Adam Coates, Andrew Y Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. *AISTATS*, 15:215–223, 2011. 6

[11] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020. 7, 8

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1, 4, 6, 7, 8

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 3

[14] Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*, 2020. 3, 4

[15] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193, 2018. 2

[16] 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 preprint arXiv:2010.11929*, 2020. 3

[17] Li Fei-Fei, Robert Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006. 6

[18] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 2

[19] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007. 6

[20] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019. 6

[21] Nathan Inkawhich, Gwendolyn McDonald, and Ryan Luley. Adversarial attacks on foundational vision models. *arXiv preprint arXiv:2308.14597*, 2023. 3

[22] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1

[23] 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

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

[25] Cheolhyoung Lee, Kyunghyun Cho, and Wanmo Kang. Mixout: Effective regularization to finetune large-scale pretrained language models. *arXiv preprint arXiv:1909.11299*, 2019. 2, 3

[26] Dongxu Li, Junnan Li, Hongdong Li, Juan Carlos Niebles, and Steven CH Hoi. Align and prompt: Video-and-language pre-training with entity prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4953–4963, 2022. 2

[27] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for uni-

fied vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022. 1, 2

[28] Xiao Li, Wei Zhang, Yining Liu, Zhanhao Hu, Bo Zhang, and Xiaolin Hu. Anchor-based adversarially robust zero-shot learning driven by language. *arXiv preprint arXiv:2301.13096*, 2023. 3

[29] Dong Lu, Zhiqiang Wang, Teng Wang, Weili Guan, Hongchang Gao, and Feng Zheng. Set-level guidance attack: Boosting adversarial transferability of vision-language pre-training models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 102–111, 2023. 3

[30] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 1, 2, 3, 4, 6, 7

[31] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 6

[32] Chengzhi Mao, Scott Geng, Junfeng Yang, Xin Wang, and Carl Vondrick. Understanding zero-shot adversarial robustness for large-scale models. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 2, 3, 4, 5, 6, 7, 8

[33] 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. 6

[34] 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. 2

[35] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012. 6

[36] 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

[37] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 1

[38] Mr D Murahari Reddy, Mr Sk Masthan Basha, Mr M Chinnaiahgari Hari, and Mr N Penchalaiah. Dall-e: Creating images from text. *UGC Care Group I Journal*, 8(14):71–75, 2021. 1

[39] Leslie Rice, Eric Wong, and Zico Kolter. Overfitting in adversarially robust deep learning. In *International Conference on Machine Learning*, pages 8093–8104. PMLR, 2020. 3

[40] Ali Shafahi, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! *Advances in Neural Information Processing Systems*, 32, 2019. 3

[41] Mannat Singh, Laura Gustafson, Aaron Adcock, Vinicius de Freitas Reis, Bugra Gedik, Raj Prateek Kosaraju, Dhruv Mahajan, Ross Girshick, Piotr Dollár, and Laurens Van Der Maaten. Revisiting weakly supervised pre-training of visual perception models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 804–814, 2022. 2

[42] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019. 2

[43] Yifei Wang, Liangchen Li, Jiansheng Yang, Zhouchen Lin, and Yisen Wang. Balance, imbalance, and rebalance: Understanding robust overfitting from a minimax game perspective. *Advances in neural information processing systems*, 36, 2023. 3

[44] 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, 3, 7

[45] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. *Advances in Neural Information Processing Systems*, 33:2958–2969, 2020. 3

[46] 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

[47] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019. 2, 3

[48] Jiaming Zhang, Qi Yi, and Jitao Sang. Towards adversarial attack on vision-language pre-training models. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 5005–5013, 2022. 3

[49] Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q Weinberger, and Yoav Artzi. Revisiting few-sample bert fine-tuning. *arXiv preprint arXiv:2006.05987*, 2020. 3

[50] Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Chongxuan Li, Ngai-Man Cheung, and Min Lin. On evaluating adversarial robustness of large vision-language models. *arXiv preprint arXiv:2305.16934*, 2023. 1, 3