Diverse Data Augmentation with Diffusions for Effective Test-time Prompt Tuning

Chun-Mei Feng[1] Kai Yu[1]* Yong Liu[1] Salman Khan[2,3] Wangmeng Zuo[4]
[1]Institute of High Performance Computing (IHPC),
Agency for Science, Technology and Research (A*STAR), Singapore
[2]Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), UAE
[3]Australian National University, Canberra ACT, Australia
[4]Harbin Institute of Technology, Harbin, China
fengcm.ai@gmail.com; yu_kai@ihpc.a-star.edu.sg

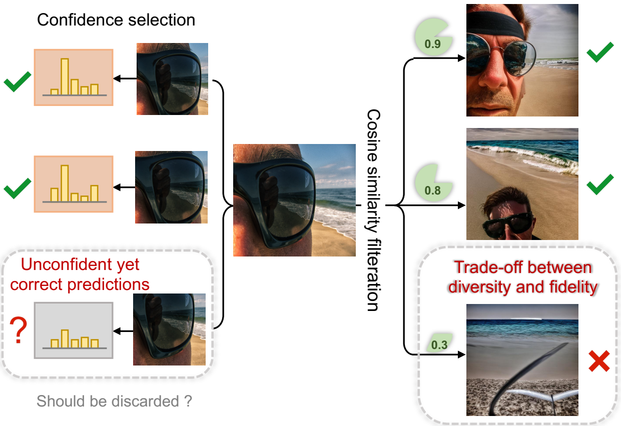
https://github.com/chunmeifeng/DiffTPT

Abstract

Benefiting from prompt tuning, recent years have witnessed the promising performance of pre-trained vision-language models, e.g., CLIP, on versatile downstream tasks. In this paper, we focus on a particular setting of learning adaptive prompts on the fly for each test sample from an unseen new domain, which is known as test-time prompt tuning (TPT). Existing TPT methods typically rely on data augmentation and confidence selection. However, conventional data augmentation techniques, e.g., random resized crops, suffers from the lack of data diversity, while entropy-based confidence selection alone is not sufficient to guarantee prediction fidelity. To address these issues, we propose a novel TPT method, named DiffTPT, which leverages pre-trained diffusion models to generate diverse and informative new data. Specifically, we incorporate augmented data by both conventional method and pre-trained stable diffusion to exploit their respective merits, improving the model's ability to adapt to unknown new test data. Moreover, to ensure the prediction fidelity of generated data, we introduce a cosine similarity-based filtration technique to select the generated data with higher similarity to the single test sample. Our experiments on test datasets with distribution shifts and unseen categories demonstrate that DiffTPT improves the zero-shot accuracy by an average of 5.13% compared to the state-of-the-art TPT method.

1. Introduction

Pre-trained vision-language models, such as CLIP [37], have recently demonstrated promising performance on a variety of downstream tasks without the need for task-specific training data [62, 61, 28, 39].


Confidence selection
Unconfident yet correct predictions
?
Should be discarded ?
Cosine similarity filteration
0.9
0.8
0.3
Trade-off between diversity and fidelity

(a) Augmented views (b) Diverse augmentation

Figure 1: **(a)** Prior TPT method [46] uses different augmented views along with confidence selection, resulting in *overly simplistic variants* in the test data and *unconfident yet correct* predictions being discarded. In comparison, **(b)** our DiffTPT is effective in generating data with ***richer visual appearance variation*** and selecting generated data with higher ***prediction fidelity***.

This is achieved through well-designed prompts, but both hand- crafted prompts and prompt tuning have limitations as they are restricted to the training data distribution of the current domain, making it challenging to generalize to distributions outside the domain, especially in the zero-shot setting [30]. In this context, test-time prompt tuning (TPT) [46] has been proposed to learn adaptive prompts on the fly for each test sample from an unseen new domain, without any training data or annotations. This setting offers a more practical approach for dynamic real-world scenarios where collecting a significant amount of labeled data for an

*Corresponding author.

unseen target distribution is often challenging.

One initial attempt to tackle TPT is to incorporate confidence selection with entropy minimization for prompt tuning by using various augmented views of each test sample [46]. However, the data augmentation method employed in [46] uses simple parametric transformations to alleviate the scarcity of data (see Fig. 1 (a)). Such simplistic transformations are limited in generating diverse data to reflect the rich appearance variation of the test sample [1, 36, 59, 45]. Insufficient diversity in the augmented data can hamper the generalization ability of the learned prompt leading it to overfit on a single data mode. Also, entropy-based confidence selection proposed in [46] is not enough to ensure prediction fidelity, as an augmented sample with low-entropy prediction could still be misclassified into a different class to the test sample, leading to unrepresentative samples in the augmented pool.

Recent advancements in image generation have made it possible to handle the diversity of augmented data better. Early image generation methods, including VAEs [25] and GANs [15], often require a large amount of data for training. Recently, diffusion models have achieved superior performance in text-to-image generation with high-quality photo-realistic details [32, 39, 42, 41]. In comparison to the data augmentation method adopted in [46], the augmented data by diffusion model can exhibit much higher diversity, thereby providing richer visual representations and benefiting the generalization ability of learned prompts. However, diffusion-based data augmentation is prone to producing *spurious* augmentations which are more difficult to be filtered out by entropy-based confidence selection. Thus, further research is required to find the right *balance* between data diversity and prediction fidelity when applying diffusion-based data augmentation for TPT.

In this work, we introduce a new TPT method called DiffTPT, that enhances the ***data diversity*** of test samples through diffusion models while maintaining ***prediction fidelity*** with cosine similarity-based filtration (see Fig. 1 (b)). In terms of data diversity, DiffTPT adopts Stable Diffusion for data augmentation. Stable Diffusion is a text-to-image generation model which synthesizes an image based on the CLIP text feature [41]. Instead, we use the CLIP image feature of the test sample as an alternative to the CLIP text feature and feed it into Stable Diffusion for data augmentation. The diffusion-based augmentation is effective in generating diverse images with *richer visual appearance variation* while ***preserving the key semantics***. Furthermore, we leverage both the augmentation in [46] and diffusion-based augmentation for improving TPT performance. To ensure prediction fidelity, we introduce cosine similarity-based filtration to *remove spurious augmentations*. By incorporating diffusion-based augmentation with cosine similarity-based filtration, our DiffTPT can make a fair trade-off between diversity and fidelity. Moreover, DiffTPT is agnostic to the training data and can be seamlessly integrated into arbitrary CLIP architectures. Experimental results show that DiffTPT achieves a notable improvement of zero-shot accuracy by an average of 5.13% in comparison to the state-of-the-art TPT method [46]. To sum up, our contributions are as follows:

- We present a new test-time prompt tuning method, *i.e.*, DiffTPT, that balances the trade-off between data diversity and prediction fidelity.
- Diffusion-based data augmentation is proposed to generate diverse augmented images with richer visual appearance variations while faithfully preserving the key semantics.
- We introduce cosine similarity filtration to remove spurious augmentations, thereby improving the prediction fidelity of augmented images.
- Experimental results show that our DiffTPT significantly outperforms the state-of-the-art test-time prompt-tuning method [46].

2. Related Work

Prompt Tuning. Large-scale pre-trained models have improved performance on several tasks in natural language processing [10, 38] and computer vision [23, 7, 24, 12, 27] by learning general representations and transferring the learned knowledge to downstream tasks. For adapting pre-trained models to downstream tasks, several parameter-efficient fine-tuning methods, *e.g.*, prompt tuning and adapters, have been proposed in the recent few years. For example, CoOp [62] and CoCoOp [61] employ continuous prompt optimization strategies and instance-wise conditionalization on prompts to achieve generalization to out-of-distribution data. CLIP-Adapter [13] and Tip-Adapter [57] use adapters and non-parametric key-value cache models to fine-tune the CLIP model, thereby improving its adaptability to the target dataset. UPL alleviated CLIP's reliance on labeled data and trains a prompt representation ensemble to improve transfer performance without the label of target dataset [22]. However, the performance of zero-shot generalization is highly dependent on well-designed prompts.

In another line of work, [46] proposed test-time prompt tuning (TPT) by generating multiple random augmented views of a single test sample that is directly applicable to the zero-shot generalization of the base model [46]. However, the data augmentation in [46] suffers from overly simplistic variants and the entropy-based confidence selection is not sufficient to guarantee prediction fidelity. To improve TPT [46], our work suggests incorporating diffusion-based data augmentation and cosine similarity-based filtration for a better trade-off between data diversity and prediction fidelity.

Test-time Optimization. Adapting machine learning models to test samples is a more challenging and practical setting where no training data is available during inference [51, 50, 6, 44]. This setting alleviates the limitation of inaccessible source data due to privacy concerns and enables training the model once and adapting it to any unknown test distributions [14]. To design an efficient test-time objective, one way is to make the objective independent of a specific training procedure by minimizing the entropy of the batch prediction probability distribution [51] or bypassing the requirement of multi-test samples via data augmentation [56]. Another approach is to explicitly apply the BN layer at test time to constrain a set of parameters for optimization and enhance the robustness of the model to distribution shifts [43].

However, these methods are either limited by the number of test samples required for the model to output non-trivial solutions or by the scalability of the model architecture. Subsequent works moved to large-scale, pre-trained models with parameter-efficient tuning [14, 58]. For example, TPT learned target-specific text prompts while freezing the backbone by generating multiple randomly augmented views during the test phase and filtering out noise augmentations that may lead to misleading predictions through entropy minimization [46]. However, the entropy-based confidence selection [46] is limited in its ability to filter out a misclassified augmented sample with low entropy prediction. Given this, we introduce a cosine similarity-based filtration between augmented and test samples (see Fig.1 (b)) to encourage the augmented samples to preserve consistent class semantics (*i.e.*, *prediction fidelity*) while bringing more *diverse* information.

Image Synthesis. Training models with synthetic images is gaining popularity and undergoing rapid development. In contrast to standard data augmentation methods, such as image manipulation [45], image erasing [60], and image mixup [54, 18], image synthesis offer higher flexibility as these methods augment datasets with pre-defined transformations and cannot provide images with highly diverse content. Early image generation methods, including VAEs [25] and GANs [15], initially provided promising generated images [5], and have been widely applied to various vision tasks. Most recently, diffusion models have been developed to generate higher-quality images with more photo-realistic details than the prior image generation methods [21, 33, 42, 39, 55]. Recent works have shown the outstanding performance of diffusion generative models in many applications, *e.g.*, using the latent space of powerful pretrained autoencoders for high-resolution image synthesis [41], enhancing text-conditional image synthesis [32, 39], learning diffusion-based prior for few-shot conditional image generation [47], and probabilistic model for point cloud generation [20]. These works motivate us to directly augment the test data with the *same semantics but diverse information* through the diffusion model, thereby improving test-time prompt-tuning performance.

3. Methodology

3.1. Test-time Prompt Tuning

The pre-trained vision-language models, *e.g.*, CLIP, consist of two encoders, *i.e.*, image encoder $f(\cdot)$ and the text encoder $g(\cdot)$, providing rich knowledge for various downstream tasks. For zero-shot classification, we can obtain the predication probability by

$$p(y_i \mid \mathbf{x}) = \frac{\exp\left(\cos\left(\boldsymbol{w_i}, \boldsymbol{e}\right)/\tau\right)}{\sum_{j=1}^{K}\exp\left(\cos\left(\boldsymbol{w_j}, \boldsymbol{e}\right)/\tau\right)}, \tag{1}$$

where $\boldsymbol{e}$ is the image features extracted by $f(\cdot)$ for the image $\mathbf{x}$ that together with their paired text feature $\boldsymbol{w}_i$ are used to compute cosine similarity $\cos(\boldsymbol{w}_i, \boldsymbol{e})$ for class i. And τ is the temperature parameter.

However, the performance of CLIP towards zero-shot generalization needs to be improved because the foundation model is often desired to generalize to out-of-distribution samples. Here, we consider test-time prompt tuning since it can modify the context of class names to adapt to new test-time data samples. Specifically, this means that only a test sample is available, and no other labeled training data is available for adaptation. Therefore, we need to optimize prompts based on a single test sample $\mathbf{x}_{\texttt{test}} \in \mathbb{R}^{C\times H\times W}$ during the testing phase. Formally, we have

$$\boldsymbol{v}^* = \arg\min_{\boldsymbol{v}} \mathcal{L}\left(\mathcal{F}, \boldsymbol{v}, \mathbf{x}_{\texttt{test}}\right), \tag{2}$$

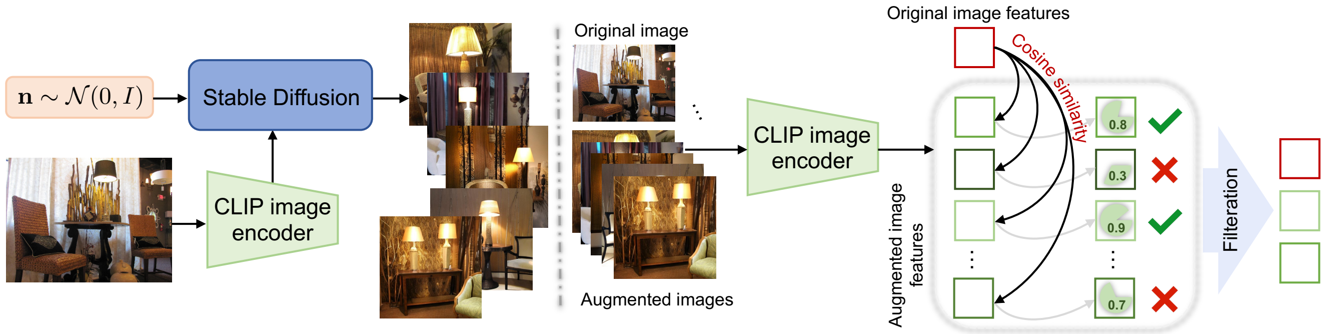
where $\mathcal{F}$ is the CLIP model consist of an image encoder $f(\cdot)$ and a text encoder $g(\cdot)$, $\boldsymbol{v}^*$ denotes the learnable prompts of each class name y_i, which together form the category-specific text inputs $\{\boldsymbol{v}^*; y_i\}$. For the classification task, $\mathcal{L}$ indicates the cross-entropy loss.

To enforce the effectiveness of TPT, Shu *et al.* [46] use multiple (*i.e.*, N) augmented views along with the confidence selection mechanism , which can be expressed as:

$$\boldsymbol{v}^* = \arg\min_{\boldsymbol{v}} -\sum_{i=1}^{K}\tilde{p}_{\boldsymbol{v}}\left(y_i \,|\, \mathbf{x}_{\texttt{test}}\right)\log\tilde{p}_{\boldsymbol{v}}\left(y_i \,|\, \mathbf{x}_{\texttt{test}}\right), \tag{3}$$

$$\tilde{\boldsymbol{p}}_{\boldsymbol{v}} = \frac{1}{\rho_H N}\sum_{n=1}^{N}\mathbb{1}\left[\mathbf{H}\left(\boldsymbol{p}_n\right)\le\tau\right]\boldsymbol{p}_{\boldsymbol{n}}\left(y \,|\, \mathcal{A}_n\left(\mathbf{x}_{\texttt{test}}\right)\right), \tag{4}$$

where K denotes the number of classes. $p_{\boldsymbol{p}}\left(y_i \mid \mathcal{A}_i\left(\mathbf{x}_{\texttt{test}}\right)\right)$ represents the class probabilities for the n-th augmented view and prompt $\boldsymbol{v}$, τ is the confidence selection threshold resulting in a ρ_H percentage on the total N augmented views, $\mathbf{H}$ calculates the self-entropy of the prediction on an augmented view.



(a) Diverse augmentation with diffusion

(b) Cosine similarity filteration

Figure 2: **Overview** of our proposed **DiffTPT**. We first **(a)** use the pre-trained stable diffusion to generate data with *richer visual appearance variation*, then **(b)** uses a cosine similarity based filtration with the single test sample to *remove spurious augmentations*, making our method a ***trade-off between diversity and fidelity***.

3.2. Approach Overview

While the augmentation method in [46] has achieved considerable success in TPT, it is obvious that the solution depends heavily on the diversity of augmented images. Since the augmented views generally have the same object and background appearance content, the model suffers from overly simplistic variants in the test data, which can cause prompt overfitting. Moreover, Shu *et al*. [46] use an entropy-based confidence selection mechanism to discard augmented images with high entropy prediction. That is, most of the retained augmented images are cropped variants of the objects in the original test image (see the augmented views of Fig. 1 (a)). As a result, the augmentation method in [46] will result in trivial variations in augmented images, thereby limiting the generalization ability of the learned text prompt [2].

In this work, we circumvent this problem by leveraging a diffusion model on each test sample to generate diverse new images that capture natural variations in appearance while preserving the key semantics. Thus, diffusion-based data augmentation not only increases the number of original test samples but also achieves semantic consistency in distribution variations. While simply applying diffusion to test-time prompt tuning allows for performance improvements (see Table 1 in Sec. 4.2), it may introduce spurious augmentations and can lead to wrong predictions. Furthermore, we address this issue by introducing a cosine similarity-based filteration. In the following, we will introduce diffusion-based data augmentation and cosine similarity-based filteration in more detail.

3.3. Diffusion-based Diverse Data Augmentation

Diffusion-based data augmentation is presented to generate diverse and informative augmented images. As shown in Fig. 2 (a), from a given single test image $\mathbf{x}_{\texttt{test}}$, we first extract its latent features z_0 from the pre-trained CLIP encoder $f(\mathbf{x}_{\texttt{test}})$, and then use the stable diffusion as the decoder to generate augmented images. In specific, we employ Stable Diffusion-V2 as the generative model, which can generate new image $\mathcal{G}(g(\mathbf{t}), \mathbf{n})$ from the natural language descriptions $\mathbf{t}$. Here, $\mathbf{n} \sim \mathcal{N}(0, I)$ denotes the sampled noise. Because the labels are not available during test time tuning, we use the image encoder of CLIP model $f(\mathbf{x}_{\texttt{test}})$ as an alternative of $g(\mathbf{t})$. Thus, the augmented image can then be generated with

$$\mathcal{D}_n(\mathbf{x}_{\texttt{test}}) = \mathcal{G}(f(\mathbf{x}_{\texttt{test}}), \mathbf{n}_n), \tag{5}$$

where $\mathcal{D}_n(\mathbf{x}_{\texttt{test}})$ denotes the n-th augmented image. Thanks to the ability of CLIP in aligning image and text, diffusion-based data augmentation is effective in generating diverse augmented images. Please refer to the *Suppl.* for visualizations of our diverse and informative augmentations.

Taking the augmented images $\{\mathcal{D}_n(\mathbf{x}_{\texttt{test}})\}$ into account, the test-time prompt tuning in Eq. (4) can be modified as,

$$\tilde{\boldsymbol{p}}_{\boldsymbol{v}} = \frac{1}{\rho_H N} \sum_{n=1}^{N} \mathbb{1}[\mathbf{H}(\boldsymbol{p}_i) \leq \tau]\, p_{\mathbf{n}}\left(y \,|\, \mathcal{D}_n\left(\mathbf{x}_{\texttt{test}}\right)\right). \tag{6}$$

Moreover, we incorporate augmented data by both the method in [46] and diffusion-based one to take advantage of their complementary merits. Then, we can still use Eq. (3) to learn the adaptive prompt for the test sample $\mathbf{x}$.

3.4. Filtration with Cosine Similarity

Albeit diffusion-based data augmentation is effective in generating diverse augmented images, some spurious augmentations may be introduced (see Fig. 3), resulting in low data fidelity and collapsing prompt tuning performance. Thus, it is necessary to balance the data diversity and prediction fidelity of the augmented images.

To tackle this issue, we introduce a cosine similarity based filteration approach. In specific, we calculate the cosine similarity between the test sample $\mathbf{x}_{\texttt{test}}$ and each augmented image $\mathcal{D}_n(\mathbf{x}_{\texttt{test}})$. Then, we introduce a mask $\mathcal{M}$ to

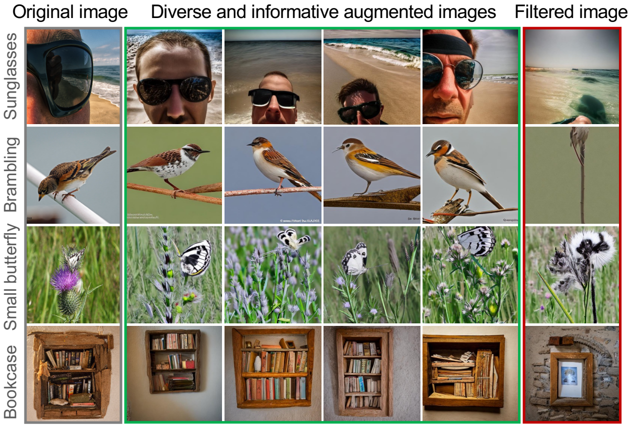


Figure 3: **Visualization** of the diverse and informative diffusion-based augmented images and the filtered image by cosine similarity.

identify the augmented images with higher similarity than ε, *i.e.*, $\mathcal{M}_n = (\cos(\mathcal{D}_n(\mathbf{x}_{\texttt{test}}), \mathbf{x}_{\texttt{test}})) > \varepsilon$. We note that ε is the threshold parameter resulting a ρ_C percentage on the augmented images. Taking both diffusion-based data augmentation and cosine similarity-based filteration into account, the test-time prompt tuning in Eq. (4) can be further modified as,

$$\tilde{\boldsymbol{p}}_{\boldsymbol{v}} = \frac{1}{\rho_H \rho_C N} \sum_{i=1}^{N} \mathbb{1}[\mathbf{H}(p_i) \leq \tau] \cdot \mathbb{1}[\mathcal{M}_n] \, \boldsymbol{p}_{\boldsymbol{n}}(y \,|\, \mathcal{D}_n(\mathbf{x}_{\texttt{test}})), \tag{7}$$

Thus, we can obtain a large number of augmented samples with richer visual appearance variation while preserving key semantics to optimize the prompt during test-time.

4. Experiments

4.1. Experimental Setup

Implementation Details. Our method is implemented on one NVIDIA Tesla V100 GPU and 32GB of memory. Following [46], the initialized prompt is set to a hand-crafted default form, “a photo of a”, and the corresponding 4 tokens are optimized based on a single test image. We augment each test image to produce 63 new images by Stable Diffusion-V2 in addition to the original one, and 64 new images by different augment views [46]. The prompt is optimized with 4 steps during test phase, where Adam is the optimizer. The hyper-parameter of initial learning rate, ρ_H, and ρ_C, are set to 0.005, 0.3, and 0.8, respectively.

Datasets. We evaluate our method in two $\mathcal{S}$cenarios, *i.e.*, $\mathcal{S}_1$: `Natural Distribution Shifts` and $\mathcal{S}_2$: `Cross-Datasets Generalization`. Following [46], for $\mathcal{S}_1$, we use four datasets which are the out-of-distribution (OOD) data for **ImageNet** [9], *i.e.*, **ImageNet-V2** [40], **ImageNet-A** [19], **ImageNet-R** [17], and **ImageNet-Sketch** [52] to investigate the robustness of our method to natural distribution shifts as these datasets differ in image style, data domains, *etc*. For $\mathcal{S}_2$, 10 datasets various in different species of plants or animals, scenes, textures, food, transportation, human actions, satellite images, and general objects are adopted in our experiments, *i.e.*, **Flower102** [34], **OxfordPets** [35], **SUN397** [53], **DTD** [8], **Food101** [4], **StanfordCars** [26], **Aircraft** [29], **UCF101** [49], **EuroSAT** [16], and **Caltech101** [11]. In particular, to investigate the effectiveness of our method with regard to cross-datasets generalization, ImageNet is used as a comprehensive source dataset, and the other 10 datasets are used as target datasets for evaluation. In our experiments, we note that 1,000 test images are randomly selected from all the classes to evaluate all the methods.

Baselines. To evaluate our proposed method, we adopt three groups of methods, **a)** TPT [46], a state-of-the-art test-time prompt tuning method that is optimized upon multiple augmented views, **b)** the classical few-shot prompt tuning methods for CLIP, *i.e.*, CoOp [62], a few-shot prompt tuning baseline that tunes a fixed prompt on each downstream dataset, and CoCoOp [61], a improved few-shot prompt tuning baseline that generate input-conditional prompts by lightweight neural network, as well as **c)** two kinds of zero-shot CLIP, one with the ensemble of 80 specially created prompts [37]. and the other with the default prompt “a picture of a.” Following these works [62, 61, 46], all the baselines are trained on ImageNet with 16-shot and 4 learnable prompt tokens, and finally tested on OOD benchmarks.

4.2. Comparison with State-of-the-arts

Natural Distribution Shifts. Table 1 summarizes the evaluation of competing methods under $\mathcal{S}$cenario 1 and different backbones, *i.e.*, ResNet-50 and ViT-B/16, where ensemble refers to zero-shot CLIP performance under an ensemble of 80 hand-crafted prompts and CLIP refers to the zero-shot CLIP performance by default prompt “a photo of a”. “—&CoOp” and “—&CoCoOp” indicate applying the test-time prompt tuning method to the CoOp [62] or CoCoOp [61], which are fine-tuned with 16 shot training data per category on ImageNet. As can be seen from this table, DiffTPT outperforms all the other methods on the five datasets. Applying DiffTPT to prompts learned by CoOp [62] or CoCoOp [61] can both improve the accuracy of their in-domain **ImageNet** data as well as their generalization ability to OOD data, *i.e.*, based on a ResNet-50, the average of in-domain **ImageNet** data is improved from 47.12 to **50.87** and 46.71 to **49.61**, respectively, and the average for OOD generation is improved from 43.08 to **47.42** and 43.01 to **46.14**, respectively. Similar improvements of our method are also obtained on ViT-B/16, *i.e.*, 61.27 → **64.12**, and 61.01 → **61.57** on ResNet-50, 58.51 → **61.97**, and 58.41 → **59.64** on ViT-B/16. Compared with TPT, TPT&CoOp, and TPT&CoCoOp, our proposed methods, DiffTPT, DiffTPT&CoOp, and DiffTPT&CoCoOp, in-

Table 1: **Top 1 accuracy** % of state-of-the-art baselines under $\mathcal{S}_1$, where **ImageNet-Sk.** indicates the ImageNet-Sketch dataset, **OOD Avg.** indicates the OOD average results. $bs.$ indicates the baseline of each group, *i.e.*, CLIP-RN50 / CLIP-ViT-B-16, CoOp, and CoCoOp. The arrow ↑ and ↓ indicate **improvements** and **decrements** compared with $bs.$. Detailed analyses are provided in Sec. 4.2.

Method	ImageNet	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-Sk.	Average	OOD Avg.
CLIP-RN50	58.10($bs.$)	22.81($bs.$)	53.00($bs.$)	53.90($bs.$)	33.50($bs.$)	42.26($bs.$)	40.80($bs.$)
Ensemble	59.90(1.80) ↑	24.12(1.31) ↑	53.50(0.50) ↑	58.00(4.10) ↑	35.20(1.70) ↑	46.14(3.88) ↑	42.70(1.90) ↑
TPT	59.40(1.30) ↑	27.34(4.53) ↑	55.20(2.20) ↑	56.80(2.90) ↑	34.50(1.00) ↑	46.65(4.39) ↑	43.46(2.66) ↑
DiffTPT	**60.80**(2.70) ↑	**31.06**(8.25) ↑	**55.80**(2.80) ↑	**58.80**(4.90) ↑	**37.10**(3.60) ↑	**48.71**(6.45) ↑	**45.69**(4.89) ↑
CoOp	63.30($bs.$)	24.52($bs.$)	57.90($bs.$)	55.10($bs.$)	34.80($bs.$)	47.12($bs.$)	43.08($bs.$)
TPT&CoOp	63.70(0.40) ↑	29.75(5.23) ↑	60.90(3.00) ↑	57.80(2.70) ↑	36.50(1.70) ↑	49.73(2.61) ↑	46.24(3.16) ↑
DiffTPT&CoOp	**64.70**(1.40) ↑	**32.96**(8.44) ↑	**61.70**(3.80) ↑	**58.20**(3.10) ↑	**36.80**(2.00) ↑	**50.87**(3.75) ↑	**47.42**(4.34) ↑
CoCoOp	61.50($bs.$)	25.73($bs.$)	54.80($bs.$)	56.00($bs.$)	35.50($bs.$)	46.71($bs.$)	43.01($bs.$)
TPT&CoCoOp	62.40(0.90) ↑	26.43(0.70) ↑	56.10(1.30) ↑	56.50(0.50) ↑	35.60(0.10) ↑	47.41(0.70) ↑	43.66(0.65) ↑
DiffTPT&CoCoOp	**63.50**(2.00) ↑	**30.45**(0.72) ↑	**57.70**(2.90) ↑	**58.50**(2.50) ↑	**37.90**(2.40) ↑	**49.61**(2.90) ↑	**46.14**(3.13) ↑
CLIP-ViT-B/16	67.30($bs.$)	47.14($bs.$)	59.90($bs.$)	71.20($bs.$)	43.00($bs.$)	57.71($bs.$)	55.31($bs.$)
Ensemble	68.50(1.20) ↑	48.44(1.30) ↑	62.70(2.80) ↑	73.50(2.30) ↑	45.50(2.20) ↑	59.73(2.02) ↑	57.53(2.22) ↑
TPT	69.70(2.40) ↑	53.67(6.53) ↑	64.30(4.40) ↑	73.90(2.70) ↑	46.40(3.40) ↑	61.59(3.88) ↑	59.57(4.26) ↑
DiffTPT	**70.30**(3.00) ↑	**55.68**(8.54) ↑	**65.10**(5.20) ↑	**75.00**(3.80) ↑	**46.80**(3.80) ↑	**62.28**(4.57) ↑	**60.52**(5.21) ↑
CoOp	72.30($bs.$)	49.25($bs.$)	65.70($bs.$)	71.50($bs.$)	47.60($bs.$)	61.27($bs.$)	58.51($bs.$)
TPT&CoOp	73.30(1.00) ↑	56.88(7.63) ↑	66.60(0.90) ↑	73.80(2.30) ↑	49.40(1.80) ↑	64.00(2.73) ↑	61.67(3.16) ↑
DiffTPT&CoOp	**75.00**(2.70) ↑	**58.09**(8.84) ↑	**66.80**(1.10) ↑	**73.90**(2.40) ↑	**49.50**(1.90) ↑	**64.12**(2.85) ↑	**61.97**(3.46) ↑
CoCoOp	71.40($bs.$)	50.05($bs.$)	63.80($bs.$)	73.10($bs.$)	46.70($bs.$)	61.01($bs.$)	58.41($bs.$)
TPT&CoCoOp	67.30(4.10) ↓	50.25(0.20) ↑	62.30(1.50) ↓	73.90(0.80) ↑	47.10(0.40) ↑	60.17(0.84) ↓	58.39(0.02) ↓
DiffTPT&CoCoOp	**69.30**(2.10) ↓	**52.56**(2.51) ↑	**63.20**(0.60) ↓	**75.30**(2.20) ↑	**47.50**(0.80) ↑	**61.57**(0.56) ↑	**59.64**(1.23) ↑

crease classification accuracy in-domain **ImageNet** data from 46.65, 49.73, 47.41 to **48.71**, **50.87**, **49.61**, respectively. Since the TPT-based method uses random resized crops to augment test image, making it limited in generalization ability. In particular, we discover that DiffTPT significantly improves the generalization test of OOD data. This supports our conclusion that DiffTPT increases the ***data diversity*** of the test samples while maintaining semantic consistency (*i.e.*, ***prediction fidelity***), resulting in better robustness. On the contrary, the performance of the few-shot prompt tuning method differs significantly across the five datasets; *e.g.*, the accuracy gains are prominent on the **ImageNet** validation set and **ImageNet-V2**, while on the datasets with more significant distribution shift, even prompt tuning methods become brittle compared to handcrafted prompts.

Naturally, CLIP yields the lowest results, and testing directly on the new dataset will be significantly affected by domain shift. Despite the fact that they benefit from the learnable prompts and the complement of TPT, CoOp [62] and CoCoOp [61] perform better against CLIP; however, these methods require the training set and are not test-time prompt tuning methods. That is, they did not consider the zero-shot generalization in real world scenarios, resulting in less effective performance. The results support our primal hypothesis that augmenting test data with diverse synthetic data can improve zero-shot generalization performance.

Cross-Datasets Generalization. To investigate how our proposed method and baselines generalize from ImageNet to the 10 fine-grained datasets, we record the quantitative performance of various baselines of $\mathcal{S}_2$ in Table 2. TPT [46] works in a zero-shot manner, while CoOp [62] and CoCoOp [61] are tuned on **ImageNet** using 16-shot training data per category. From this table, since the distribution of these fine-grained datasets varies widely, these methods perform differently on each dataset. However, our method still achieves the best performance, *i.e.*, increasing the **Avg.** accuracy from 55.12 to **59.85**, and from 63.03 to **65.47**. It is worth noting that our method still achieves 5.1% and 2.3% performance gain against TPT [46] on the backbone of both ResNet50 and Vit-B/16. This indicates that among all competing methods, even without training data, our method is robust to natural distributional variations and significantly outperforms those few-shot prompt-tuning methods, *i.e.*, CoOp [62] and CoCoOp [61].

4.3. Ablation Studies

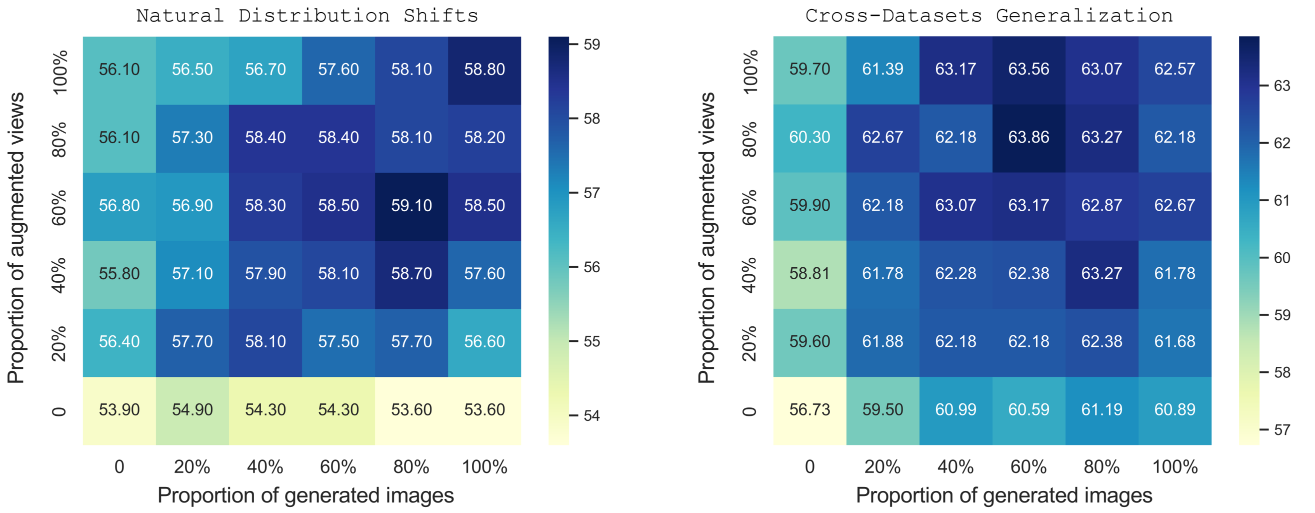
Balancing Synthetic Data *vs.* Standard Augmentation. Since our method absorbs the complementary merits of both standard augmentation [46] and diffusion-based one, it is necessary to investigate how these two methods help train classifiers. To this end, we evaluate the average performance of ResNet50 in the two scenarios, *i.e.*, $\mathcal{S}_1$ and $\mathcal{S}_2$. For better visualization, we plot the mixed com-

Table 2: **Top 1 accuracy** % of state-of-the-art baselines under $\mathcal{S}_2$, where **Avg.** indicates average accuracies of the `Cross-Datasets Generalization`. The arrow ↑ and ↓ indicate **improvements** and **decrements** of our method against the CLIP method, *i.e.*, CLIP-RN50 and CLIP-ViT-B/16. Detailed analyses are provided in Sec. 4.2.

Method	Flower [34]	DTD [8]	Pets [35]	Cars [26]	UCF101 [49]	Caltech11 [11]	Food101 [4]	SUN397 [53]	Aircraft [29]	EuroSAT [16]	Avg.
CLIP-RN50	62.45(*bs.*)	39.65(*bs.*)	80.50(*bs.*)	57.48(*bs.*)	56.73(*bs.*)	81.58(*bs.*)	74.85(*bs.*)	57.43(*bs.*)	16.20(*bs.*)	24.30(*bs.*)	55.12(*bs.*)
Ensemble	63.14	41.68	80.79	58.33	55.74	83.68	74.95	59.53	17.40	27.69	56.29
CoOp$_{2022}$ [62]	62.25	37.33	86.00	56.29	59.01	82.38	78.81	57.18	15.40	26.99	56.16
CoCoOp$_{2022}$ [61]	63.53	38.49	86.29	55.70	60.40	83.38	77.43	59.28	15.70	27.39	56.76
TPT$_{2022}$ [46]	62.25(0.20) ↓	40.04(0.39) ↑	82.82(2.32) ↑	60.54(3.06) ↑	60.79(4.06) ↑	84.58(3.00) ↑	77.23(2.38) ↑	61.80(4.37) ↑	17.50(1.30) ↑	22.21(2.09) ↓	56.98(1.86) ↑
DiffTPT	**63.53**(1.08) ↑	**40.72**(1.07) ↑	**83.40**(3.35) ↑	**60.71**(3.23) ↑	**62.67**(5.94) ↑	**86.89**(5.31) ↑	**79.21**(4.36) ↑	**62.72**(5.29) ↑	**17.60**(1.40) ↑	**41.04**(16.74) ↑	**59.85**(4.68) ↑
CLIP-ViT-B/16	67.94(*bs.*)	44.10(*bs.*)	85.71(*bs.*)	66.58(*bs.*)	63.37(*bs.*)	90.29(*bs.*)	85.05(*bs.*)	61.88(*bs.*)	24.70(*bs.*)	40.64(*bs.*)	63.03(*bs.*)
Ensemble	67.65	44.87	86.20	67.60	64.36	90.89	85.35	64.65	24.40	47.01	64.30
CoOp$_{2022}$ [62]	66.08	42.17	89.00	63.44	66.04	91.69	85.15	61.54	18.00	35.36	61.85
CoCoOp$_{2022}$ [61]	70.88	44.78	88.71	65.22	68.42	92.49	86.53	64.65	24.20	46.22	65.21
TPT$_{2022}$ [46]	69.31(1.37) ↑	46.23(2.13) ↑	86.49(0.78) ↑	66.50(0.08) ↓	66.44(3.07) ↑	92.49(2.20) ↑	86.93(1.88) ↑	63.48(1.60) ↑	24.90(0.20) ↑	37.15(3.49) ↓	63.99(0.96) ↑
DiffTPT	**70.10**(2.16) ↑	**47.00**(2.90) ↑	**88.22**(2.51) ↑	**67.01**(0.43) ↑	**68.22**(4.85) ↑	**92.49**(2.20) ↑	**87.23**(2.18) ↑	**65.74**(3.86) ↑	**25.60**(0.90) ↑	**43.13**(3.49) ↑	**65.47**(2.44) ↑

binations of various ratios in Fig. 4, where the abscissa and ordinate represent the proportions of synthetic data obtained by diffusion-based augmentation and data obtained by standard augmentation, respectively. In the matrix of this figure, each element $\mathcal{M}_{ij}$ represents the classification performance of DiffTPT on $i\%$ of synthetic data and $j\%$ of standard augmented data. As shown in Fig. 4 (a), one can observe an improvement in accuracy of the `Natural Distribution Shifts` as the size of the standard augmented data grows, while maintaining a constant amount of synthetic data. However, similar results are more obvious when the proportion of synthetic data is increased while keeping the proportion of standard augmented data fixed. Overall, increasing the amount of synthetic data leads to an improved performance in $\mathcal{S}_1$. In Fig. 4 (b), we show the performance of the classifier for $\mathcal{S}_2$, *i.e.*, `Cross-Datasets Generalization`. We observe that, while keeping the amount of synthetic data fixed, the effectiveness of the classifier increases significantly as the proportion of standard augmented data increases. These conclusions are supported by the results of most datasets, see the **`Suppl.`** for more details.

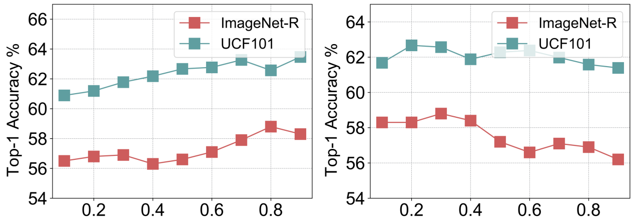
Analysis of Ratio ρ_H and ρ_C. As mentioned in Sec. 3.4, ρ_H and ρ_C filter out the less informative "noisy" augmented view in standard augmentation by entropy and spurious augmentations in diffusion-based augmentation by cosine similarity. We examine the classification accuracy for various values of ρ_H and ρ_C for the two scenarios in Fig. 5 to evaluate the diverse information that a good test augmentation should retain. It can be seen from Fig. 5 (a) that as the value of ρ_C increases, the classification accuracy gradually improves. This is because a high threshold will result in too much effective information being filtered out and reduce the effect of test set augmentation. However, from Fig. 5 (b), we find that when $\rho_H > 0.5$, it will lead to a decrease in classification accuracy. In summary, when the value of



(a)

(b)

Figure 4: **Variation of the top 1 accuracy** versus the varied proportion of the *standard* augmented views and the *diffusion-based* augmented images under (a) $\mathcal{S}_1$ and (b) $\mathcal{S}_2$.



(a) ρ_H

(b) ρ_C

Figure 5: Top-1 accuracy analysis of the **ratios** ρ_H and ρ_C with regard to $\mathcal{S}_1$ and $\mathcal{S}_2$.

ρ_H is too large and ρ_C is too small, the model goes to an extreme of overemphasizing the augmented data, giving rise to unsatisfactory performance. To sum up, our method achieves the highest results on $\rho_H = 0.3$ and $\rho_C = 0.8$. This means that only a small amount of data is filtered out, indicating the high fidelity of the synthetic data by diffusion-based augmentation.

Effect of the Generated Dataset Size. The results on a variety of datasets in both two scenarios demonstrate that our approach is effective for all images regardless of their category and content (see Table 1 and 2). Considering the effect of augmented data on model performance, we recorded the classification accuracy of different numbers of augmented samples in two scenarios in Fig. 6. It can be seen that as the number of augmented views increases, the accuracy gradually increases until reaching a plateau around $N = 64$. Additionally, even under the more complex scenario, *i.e.*, $\mathcal{S}_2$, our method can still preserve similar results. Notably, even when $N = 8$, DiffTPT still brings more than 3.5% and 4.6% accuracy gain for zero-shot CLIP, respectively. When $N = 128$, the performance of DiffTPT did not appear to be degraded in accuracy due to overemphasizing synthetic data. These positive experimental results support that our cosine similarity filteration is effective in preserving prediction fidelity between synthetic and original data.

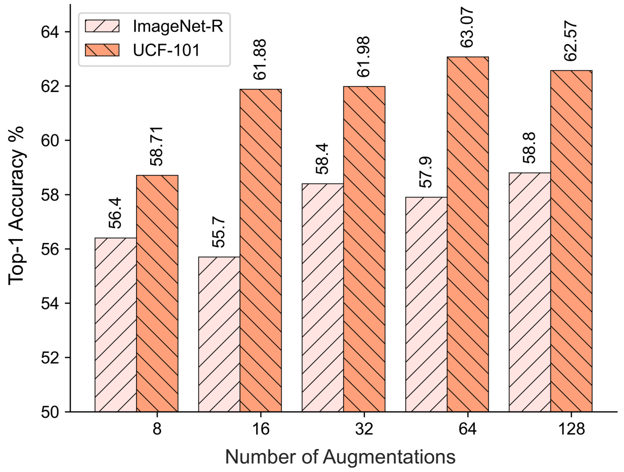
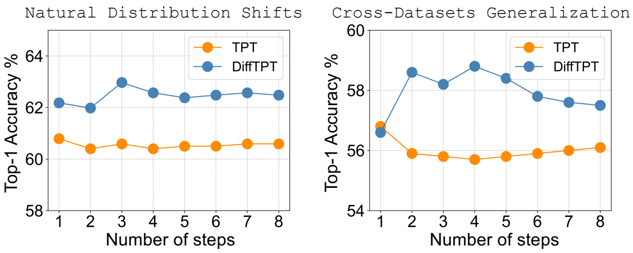


Figure 6: **Ablation studies** on the size of augmented images with regard to $\mathcal{S}_1$ and $\mathcal{S}_2$.



(a)

(b)

Figure 7: **Ablation studies** on the *steps of prompt updating* under $\mathcal{S}_1$ and $\mathcal{S}_2$.

Steps of Prompt Updating. To assess the effect of prompt updating, we record the accuracy of different optimization steps under the two scenarios in Fig. 7. As can be seen from the figure, the accuracy can be improved from 56.6 to

58.8 by increasing the number of optimization steps from 1 to 4 on $\mathcal{S}_2$, while the performance will decrease slightly when the optimization is continued (*i.e*., optimization steps greater than 5). This shows that more optimization steps cannot bring gains to the classifier; on the contrary, a few updates can make the prompt learn more information about the test samples. For TPT [46], more update steps did not make the classifier optimize better, while step $= 1$ achieved the best performance. Nevertheless, our method always outperforms TPT [46] in terms of the classification accuracies on the two scenarios. Additionally, since more update steps will increase the inference time, we set optimization step to 4 as the default setting in our experiments.

Inference cost. Albeit the original SD is time-consuming, *e.g*., it takes 6s to inference 10 test images for TPT and with standard SD it is 36min, with the advancements in this field, faster SD models have emerged, *e.g*., ToMe [3], two-stage distillation [31], and Consistency Model [48]. The later only need 0.5s for generating 10 images, while the original SD needs 70s. Additionally, there are a few strategies to boost the SD efficiency, *e.g*., TensorRT and Memory Efficient Attention[1]. These approaches lead to a further 25% and 100% gains in inference speed.

5. Conclusion

This work proposed a simple but effective test-time prompt tuning method, based on a pre-trained diffusion model, DiffTPT, by incorporating diffusion-based augmentation with cosine similarity-based filteration. In specific, with the pre-trained diffusion model, diffusion-based augmentation can generate diverse but semantically consistent augmented images. And cosine similarity-based filteration can enforce the prediction fidelity of the generated samples. Extensive experiments demonstrated the effectiveness of DiffTPT in various zero-shot generalization tasks, *e.g*., DiffTPT improves zero-shot accuracy by an average of 5.13% in comparison to the state-of-the-art test-time prompt-tuning method, TPT.

Acknowledgements: This work was supported by the National Research Foundation, Singapore under its AI Singapore Programme (AISG Award No: AISG2-TC-2021-003), Agency for Science, Technology and Research (A*STAR) through its AME Programmatic Funding Scheme Under Project A20H4b0141, A*STAR Central Research Fund "A Secure and Privacy Preserving AI Platform for Digital Health", and Agency for Science, Technology and Research (A*STAR) through its RIE2020 Health and Biomedical Sciences (HBMS) Industry Alignment Fund Pre-Positioning (IAF-PP) (grant no. H20C6a0032).

[1] https://www.photoroom.com/tech/stable-diffusion-100-percent-faster-with-memory-efficient-attention

References

[1] Antreas Antoniou, Amos Storkey, and Harrison Edwards. Data augmentation generative adversarial networks. *arXiv preprint arXiv:1711.04340*, 2017.

[2] Hritik Bansal and Aditya Grover. Leaving reality to imagination: Robust classification via generated datasets. *arXiv preprint arXiv:2302.02503*, 2023.

[3] Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4598–4602, 2023.

[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.

[5] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[6] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 295–305, 2022.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[12] Chun-Mei Feng, Bangjun Li, Xinxing Xu, Yong Liu, Huazhu Fu, and Wangmeng Zuo. Learning federated visual prompt in null space for mri reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8064–8073, 2023.

[13] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021.

[14] Yunhe Gao, Xingjian Shi, Yi Zhu, Hao Wang, Zhiqiang Tang, Xiong Zhou, Mu Li, and Dimitris N Metaxas. Visual prompt tuning for test-time domain adaptation. *arXiv preprint arXiv:2210.04831*, 2022.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and

Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[22] Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022.

[23] 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.

[24] 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.

[25] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[26] 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.

[27] Haoran Li, Chun-Mei Feng, Tao Zhou, Yong Xu, and Xiaojun Chang. Prompt-driven efficient open-set semi-supervised learning. *arXiv preprint arXiv:2209.14205*, 2022.

[28] Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022.

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

[30] Devraj Mandal, Sanath Narayan, Sai Kumar Dwivedi, Vikram Gupta, Shuaib Ahmed, Fahad Shahbaz Khan, and Ling Shao. Out-of-distribution detection for generalized zero-shot action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9985–9993, 2019.

[31] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.

[32] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[33] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021.

[34] 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.

[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.

[36] Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.

[37] 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.

[38] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[39] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[40] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[42] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image

diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

[43] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *Advances in Neural Information Processing Systems*, 33:11539–11551, 2020.

[44] Divya Shanmugam, Davis Blalock, Guha Balakrishnan, and John Guttag. Better aggregation in test-time augmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1214–1223, 2021.

[45] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):1–48, 2019.

[46] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *arXiv preprint arXiv:2209.07511*, 2022.

[47] Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2c: Diffusion-decoding models for few-shot conditional generation. *Advances in Neural Information Processing Systems*, 34:12533–12548, 2021.

[48] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. 2023.

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

[50] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pages 9229–9248. PMLR, 2020.

[51] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

[52] 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.

[53] 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.

[54] Linjun Zhang, Zhun Deng, Kenji Kawaguchi, Amirata Ghorbani, and James Zou. How does mixup help with robustness and generalization? *arXiv preprint arXiv:2010.04819*, 2020.

[55] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022.

[56] Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. *arXiv preprint arXiv:2110.09506*, 2021.

[57] Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021.

[58] Tianjun Zhang, Xuezhi Wang, Denny Zhou, Dale Schuurmans, and Joseph E Gonzalez. Tempera: Test-time prompting via reinforcement learning. *arXiv preprint arXiv:2211.11890*, 2022.

[59] Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient gan training. *Advances in Neural Information Processing Systems*, 33:7559–7570, 2020.

[60] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 13001–13008, 2020.

[61] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022.

[62] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.