# Test-Time Low Rank Adaptation via Confidence Maximization for Zero-Shot Generalization of Vision-Language Models

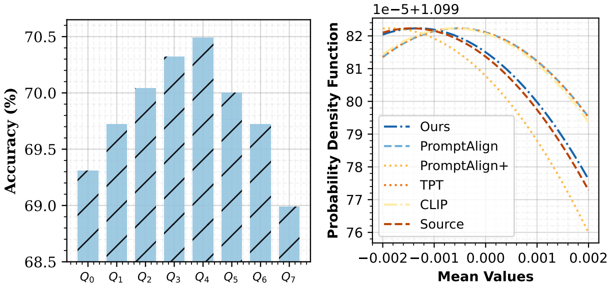
Raza Imam    Hanan Gani    Muhammad Huzaifa    Karthik Nandakumar
Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), UAE
{raza.imam, hanan.ghani, muhammad.huzaifa, karthik.nandakumar}@mbzuai.ac.ae

## Abstract

*The conventional* `modus operandi` *for adapting pre-trained vision-language models (VLMs) during test-time involves tuning learnable prompts,* i.e., *test-time prompt tuning. This paper introduces* ***T**est-**T**ime **L**ow-rank adaptation* (***TTL***) *as an alternative to prompt tuning for zero-shot generalization of large-scale VLMs. Taking inspiration from recent advancements in efficiently fine-tuning large language models, TTL offers a test-time parameter-efficient adaptation approach that updates the attention weights of the transformer encoder by maximizing prediction confidence. The self-supervised confidence maximization objective is specified using a* ***weighted entropy loss*** *that enforces consistency among predictions of augmented samples. TTL introduces only a small amount of trainable parameters for low-rank adapters in the model space while keeping the prompts and backbone frozen. Extensive experiments on a variety of natural distribution and cross-domain tasks show that TTL can outperform other techniques for test-time optimization of VLMs in* `strict zero-shot` *settings. Specifically, TTL outperforms test-time prompt tuning baselines with a significant improvement on average. Our code is available at https://github.com/Razaimam45/TTL-Test-Time-Low-Rank-Adaptation.*

## 1. Introduction

In recent years, foundational vision-language models (VLMs) such as CLIP [5] have significantly transformed the landscape of computer vision by demonstrating remarkable proficiency in encoding diverse tasks and concepts. Trained on extensive datasets comprising millions of image-text pairs, these models exhibit decent generalizability across a spectrum of tasks. However, the process of adapting these models for specific downstream tasks through *fine-tuning* often results in a compromise on their inherent generalization capabilities [29, 47]. To address this challenge, recent works propose the incorporation of learnable prompts into the CLIP model, either in the textual [25, 52, 53] or visual [24] branch, or both [26, 27]. This allows for fine-tuning only the added prompts using a few samples from the target distribution, while keeping the rest of the model frozen. While this approach has been quite effective, fine-tuning on domain-specific data inevitably diminishes the VLM's ability to generalize to unseen domains.



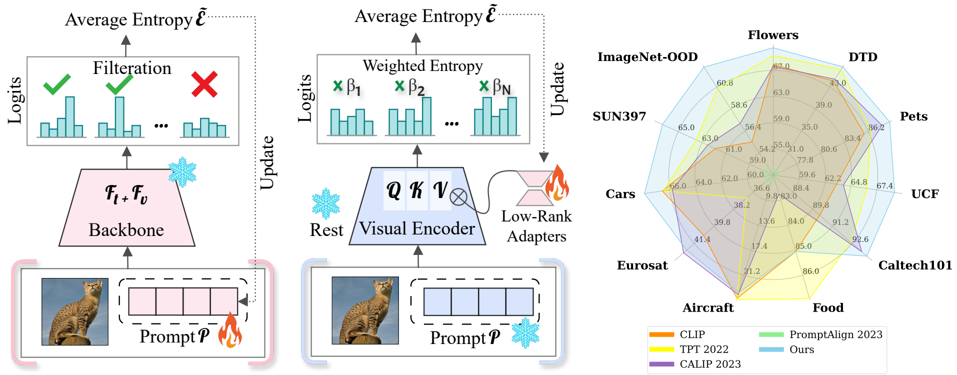
(a) Entropy level *vs.* Accuracy

(b) Visual feature statistics

Figure 1. (a) Entropy corresponding to 8 different octiles result in different performance for Flowers102. (b) TTL implicitly align features such that the mean embeddings of test samples better align with that of source data (LAION) on which CLIP [39] is trained.

It would be ideal if the pre-trained VLMs could be adapted to the target task at test-time without using any access to target domain data/statistics, few-shot learning, or external model assistance. We refer to this scenario as *strict zero-shot setting*. Test-Time Prompt Tuning (TPT) [41] is an example of this approach, where the prompts are updated dynamically on the fly for each test sample. However, TPT overlooks the *distribution shift* between the training data of the CLIP model and the test samples, resulting in a subpar performance. To address the distribution shift, PromptAlign [17] attempts to align the first-order statistics of test sample with the training data of the CLIP model. However, this approach necessitates access to a proxy dataset mimicking the distribution of CLIP training data and the use of pre-trained prompts for initialization, both of which violate the *strict zero-shot* assumption. Moreover, the token alignment achieved by PromptAlign is not as precise as that of our method, as illustrated in Figure 1b.

To address the above limitations of existing test-time



(a) Existing ZS methods (b) TTL (Ours) (c) Results on visual classification tasks

Figure 2. **TTL *vs.* other zero-shot optimization methods.** (a) Current methods [12, 17, 41] update prompts during inference using self-entropy. (b) TTL introduces low-rank learnable weight matrices at the attention layer of the vision encoder to update the model weights using weighted entropy. (c) TTL outperforms existing baselines across Out-of-Distribution and Cross-Dataset while using less than 0.1% of all model parameters.

adaptation methods, we introduce **T**est-**T**ime **L**ow-rank adaptation (**TTL**), a parameter-efficient test-time adaptation strategy for VLMs like CLIP. TTL *eliminates* the need for source data distribution during adaptation or pre-trained weights for initialization (Figure 2). Originally designed for adapting Large Language Models (LLMs) to new domains, low rank adaptation (LoRA) [22] has been extensively applied in various multi-modal and generative computer vision tasks [2, 4, 7, 10, 15, 30, 32, 38, 46, 49]. LoRA has two main advantages compared to prompt tuning [6]. Firstly, LoRA is generally more effective in low-resource (limited data availability) settings. During test-time adaptation, we have only a single unlabeled test sample available to update the model. Moreover, to minimize the overall time required for inference, only a very limited number of model updates (typically only one) are possible during test-time. Again, LoRA is known to be more stable than prompts in this scenario. It must be highlighted that our work marks *the first exploration of LoRA for test-time adaptation based on a single test sample for zero-shot generalization*.

Additionally, we introduce a confidence maximization objective that replaces the conventional entropy loss used in [17,41] with a new weighted entropy loss. Existing studies [1,13,51] highlight the tendency of deep neural networks to leverage both spurious and semantically meaningful features, leading to diminished performance when spurious correlations are prevalent. Hence, relying solely on entropy for confidence estimation may not be consistently reliable under distribution shifts, as it cannot distinguish whether the model is focusing on spurious features. As shown in Figure 1a, a low entropy value is not a guarantee for correct prediction. Hence, in this work, *we propose a weighted entropy loss that assigns relative weights to the different augmentations, while encouraging consistent high-confidence predictions for these augmentations*. Through empirical validation, we demonstrate the sub-optimality of using standard entropy loss to update parameters at test-time and showcase the advantages of optimizing our weighted entropy loss. In summary, our contributions are as follows:

- We introduce **T**est-**T**ime **L**oRA (**TTL**), a parameter-efficient scheme for low-rank adaptation of VLMs at test-time without relying on source data statistics or pre-trained prompts.
- We propose a weighted entropy loss that introduces a confidence maximization objective for updating parameters at test-time, showcasing its superior performance compared to the conventional entropy loss.
- We conduct extensive experiments and show that TTL achieves 7.49% improvement on average over the baseline CLIP and 2.11% over the best baseline for domain generalization. For cross-dataset transfer, TTL exhibits 1.40% improvement over the baseline.

## 2. Related Work

**Test-Time Adaptation (TTA)**: TTA [33, 43, 44] aims to bridge the distribution gap between the train and test data distributions at test time. While TPT [41] and CALIP [16] first explored zero-shot enhancement of pre-trained VLMs, TPT relies on test-time prompt tuning, struggling with explicit alignment of pre-training and test data distributions. CALIP utilizes a parameter-free attention module for cross-modal features. PromptAlign [17] builds on TPT and aligns distribution statistics by pre-training the learnable prompts using training data, deviating from the *strict zero-shot* assumption. DiffTPT [12] employs an external diffusion model for diverse data augmentation but is impractical due to complexity of dependence on external diffusion model. In contrast, our approach efficiently updates model parameters in a single step, focusing on adapting the visual encoder of CLIP with out-of-distribution samples at test time, without relying on pre-trained weights or external sources.

**Fine-tuning for Large Vision-Language Models**: Having been pre-trained in a self-supervised manner on vast image-text pairs, VLMs like CLIP [39] and ALIGN [23] have demonstrated good generalizability. However, efficiently adapting them to downstream tasks with limited data remains challenging. CoOp [53] proposes to fine-tune

CLIP by learning a set of prompts in the text encoder. CoCoOp [52] highlights the inferior generalization capability of CoOp and conditions the text prompt tokens on image embeddings on the fly. MaPLe [26] jointly learns deep prompts at both vision and text encoders. Despite these advancements, existing methods often rely on pre-trained weights, posing challenges in real-world scenarios where no such training data from the target domain is available. In contrast, our work utilizes LoRA [22], initialized from scratch, to adapt attention layers of the visual encoder at test time for addressing distribution shifts.

**Entropy Minimization**: The primary challenge of TTA is limited access to samples from the test dataset during online updates, which causes error accumulation. To mitigate this issue, TTA methods have utilized the entropy of model predictions as a confidence metric. TPT [41] attempts to select the augmented samples that have minimum entropy. The need to have a batch of samples by generating multiple views via augmentations at test time is eliminated in [50]. Motivated by TENT [44] and EATA [36], recently [31] shows that entropy alone as a measure of confidence is insufficient for TTA, and propose DeYO which leverages a confidence metric called PLPD and entropy together. While effective for natural datasets, the cross-dataset performance is still a unresolved problem, which we attempt to solve using the weighted entropy loss.

# 3. Methodology

## 3.1. Preliminaries

**Contrastive Language-Image Pre-training (CLIP)**: CLIP comprises of two encoders: the visual encoder $\mathcal{F}_{\theta_v}$ which maps visual input $X$ to a fixed-length representation $\boldsymbol{f}_v$, and the text encoder $\mathcal{F}_{\theta_t}$, which processes text inputs and generates latent textual feature $\boldsymbol{f_t}$. The pre-trained parameters for CLIP, represented as $\theta_{\texttt{CLIP}} = \{\theta_v, \theta_t\}$, are associated with the respective encoders. Both the encoders process the input through a sequence of $L$ transformer blocks to produce a latent feature representation. For zero-shot inference, each text feature with class labels $y \in 1, 2, \cdots, C$ is paired with the image feature. The prediction probability on $X$ can be expressed as $p(y_i|X) = \frac{\texttt{exp}(\tau s_i)}{\sum_{j=1}^{C} \texttt{exp}(\tau s_j)}$, where $s$ denotes the cosine similarity and $\tau$ represents the softmax temperature parameter.

**LoRA Adaptation**: Low Rank Adaptation (LoRA) [22] enables parameter efficient training by freezing the model weights and integration of trainable rank decomposition matrices into each layer of the transformer architecture. This results in a substantial reduction in the number of trainable parameters for the downstream adaptation. We denote the pretrained query $W_Q$, key $W_K$, and value $W_V$ projection matrices in the self-attention module jointly by $W$ such that $W = \{W_Q, W_K, W_V\}$ and $\Delta W$ signifies its accumulated gradient update during adaptation. Assuming a low intrinsic rank, the pre-trained attention weight matrix $W \in \mathbb{R}^{d \times k}$ undergoes a low-rank decomposition, expressed as $W + \Delta W = W + BA$, where $B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$, and the rank $r \ll min(d, k)$. During inference, $W$ is frozen and does not receive gradient updates, while $A$ and $B$ contain trainable parameters. The modified forward pass for any arbitrary input $h$ across the attention module yields $\tilde{h}$ such that,

$$\tilde{h} = Wh + \gamma \cdot \Delta W h = Wh + \gamma \cdot BAh, \tag{1}$$

where $\gamma = \frac{r}{\alpha}$ and $\alpha$ is the scaling factor.

## 3.2. Proposed Approach: TTL

**Overview**: Although current methods [17, 41] for prompt-tuning during test time have demonstrated notable success in enhancing CLIP adaptation, the optimal choice between prompt-tuning and alternative approaches remains *unexplored*. The existing test-time adaptation schemes, as exemplified by [41] and [17], focus on optimizing prompts for each test sample during inference through entropy minimization. While effective, these approaches have certain limitations. 1). Adaptation using standard entropy minimization is sub-optimal at test time [31], 2). Prompt tuning is challenging to optimize and its performance exhibits non-monotonic changes in trainable parameters, as observed by [22] and 3). Some works [17] necessitate access to pre-trained prompt weights and source data statistics, which may not be practical at the test-time scenario. As a solution, illustrated in Figure 3, we propose integrating LoRA (Low-Rank Adaptation) parameters directly inside the CLIP's visual encoder model to account for the domain shift due to out-of-distribution test sample. As opposed to prompts, LoRA parameters are easier to optimize [22] and do not require *pre-trained weights* for initialization or *source data* for alignment, resulting in improved generalization. For confidence maximization, we employ weighted entropy loss, as opposed to the standard self-entropy used by [17, 41]. The proposed weighted entropy objective results in overall higher average prediction confidence which is beneficial for optimal parameter update, resulting in better prediction accuracy. Empirical evidence supporting the advantages of weighted entropy loss over standard self-entropy is presented in Table 3.

**Low-Rank Adaptation at Test-Time**: For parameter efficient adaptation at test-time, we integrate LoRA parameters inside the attention layers of the CLIP's visual encoder. As indicated by [22], over-parameterized models exhibit low intrinsic dimension, and the change in weights during model adaptation has a *low intrinsic rank*. We extend this hypothesis to test-time adaptation, where only a single test

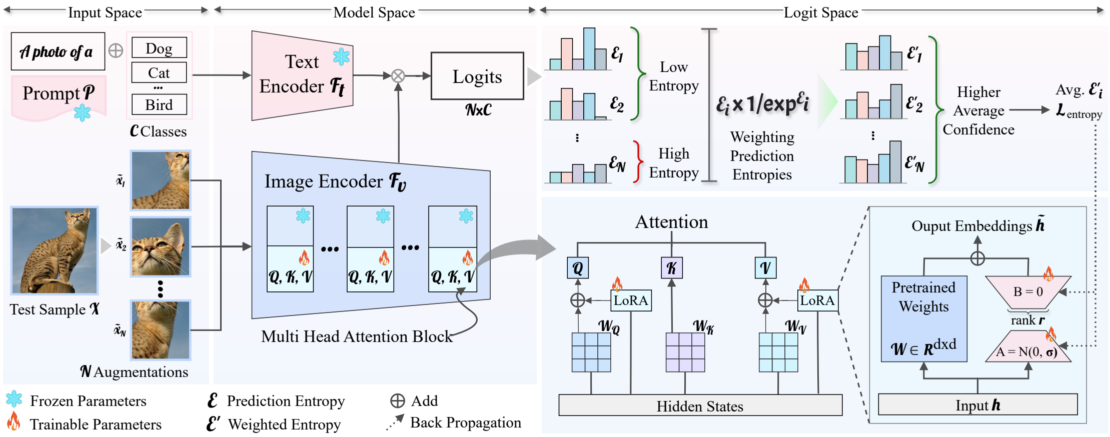


Figure 3. **Working of Test-Time Low-Rank Adaptation (TTL)**. We integrate parameter efficient low rank matrices into the self-attention module of the image encoder. We adapt these low rank weights on the fly given a single test sample, without the need for pre-trained weights or source data. Maximizing confidence via weighted entropy minimization, TTL updates the low rank weights to optimize the VLM to adapt a test sample in a single update step.

sample is available, implying that updating only a few parameters is sufficient for efficient and effective adaptation. Given a test sample $X \in \mathcal{D}^{test}$, we take $N$ randomly augmented views using transformation function $\mathcal{H}$ such that we get a batch of images denoted as $\mathcal{H}(X)$. Low rank weight matrices are introduced in the query $(W_Q)$ and value $(W_V)$ projection layers inside the self-attention module and are jointly parameterised by $\Phi$. Let $h^l_*$ be the input features to the self-attention module of $l^{th}$ encoder block and $\tilde{h}^l_*$ be the corresponding output, then the forward pass (Eq. 1) can be expressed as,

$$\tilde{h}^l_Q = W^l_Q h^l_Q + \gamma \cdot \Delta W^l_{\Phi;Q} h^l_Q = W^l_Q h^l_Q + \gamma \cdot (B^l_Q A^l_Q)_\Phi h^l_Q$$

$$\tilde{h}^l_V = W^l_V h^l_V + \gamma \cdot \Delta W^l_{\Phi;V} h^l_V = W^l_V h^l_V + \gamma \cdot (B^l_V A^l_V)_\Phi h^l_V$$

At the test-time, we optimize the rank decomposition matrices corresponding to query and value projection matrices in the self-attention module while keeping the pre-trained weights of CLIP frozen. In general, the optimization objective for a randomly augmented view $\tilde{x} \in \mathcal{H}(X)$ can be constructed as,

$$\Phi^* = \arg\min_{\Phi} \mathcal{L}(\mathcal{F}_{\theta_{\texttt{CLIP}}}, \Phi, \tilde{x}) \qquad (2)$$

Since LoRA parameters directly influence the model attention, it leads to better predictions by concentrating the model attention on the object of interest. Standard optimization objective such as self-entropy loss gives equal weight to all the augmented views leading towards sub-optimal optimization, while as confidence selection ignores certain views which may be beneficial for the correct prediction. To this end, we propose to update the parameters with weighted entropy objective which gives variable weight to each view.

**Weighted Entropy Minimization**: Instead of discarding majority of crops of test sample based on confidence selection as done in previous works [17, 41], we take a slightly different approach and utilize all the crops for optimization. As discussed in Sec. 1, relying solely on entropy for confidence selection is not consistently reliable due to model focusing on unwanted elements in the input. Our analysis in Figure 1a reinforces this observation, highlighting a weaker correlation between confidence selection and the true model prediction during test-time. This signifies that predictions from augmented views in the highest confidence quartile may not consistently contribute favorably to model updates (Figure 1a) as further validated by [31]. Therefore, we introduce weighted entropy loss which encompasses all the augmented views at test-time and assigns variable weights to model's prediction of each view resulting in overall higher average confidence. For each augmented view $\tilde{x} \in \mathcal{H}(X)$, we map its visual features to the class labels and compute a standard self-entropy loss across $L$ encoder blocks represented as,

$$\mathcal{L}_\Phi(\tilde{x}) = \sum_{i=1}^{C} \tilde{p}_\Phi(y_i|\tilde{x}) \log \tilde{p}_\Phi(y_i|\tilde{x}), \qquad (3)$$

where $\tilde{p}_\Phi(y_i|\tilde{x})$ represents the vector class probabilities produced by the model. The final objective is thus the weighted sum of the individual entropy losses corresponding to each augmented view $\tilde{x}$. The final objective function for parameter update is given as,

$$\arg\min_{\Phi} -\frac{1}{N} \sum_{\tilde{x} \in \mathcal{H}(X)} \beta_\Phi(\tilde{x}) \cdot \mathcal{L}_\Phi(\tilde{x}) \qquad (4)$$

Here $\beta_\Phi(\tilde{x})$ is the weight coefficient of the augmented view corresponding to $\tilde{x}$ and is expressed as,

$$\beta_{\Phi}(\tilde{x}) = \frac{1}{\exp(\mathcal{L}_{\Phi}(\tilde{x}) - \varepsilon)}, \tag{5}$$

where $\varepsilon$ is a normalization factor. The resulting objective function in Eq. 4 maximizes the average confidence and enhances the model's prediction of test sample.

# 4. Experiments and Results

## 4.1. Experimental Setup

**Implementation Details**: We initialize trainable LoRA matrices with random $Xavier$ initialization [14] and set rank $r = 16$ and $\alpha = 32$. Optimization of the LoRA weights occurs in layers 10 to 12 within the vision branch, involving a single-step update using a single test sample. We obtain 63 augmented views of input sample using random resized crops and horizontal flip augmentations to construct a batch of 64 images including the original image to mimic the setting of TPT. We utilize all the 64 crops to compute the average prediction probability and optimize the LoRA parameters to minimize the weighted version of combined average prediction entropy loss using the AdamW optimizer. We use a learning rate of 5e-3 for all the cases and set the normalization constant $\mathcal{E}$ equal to 0.4. A fixed prompt template ”a photo of a” is used with the classnames. The entire setup runs on a single NVIDIA A100 40GB GPU.

**Datasets**: We assess the performance of our proposed method in two cases *i.e.* `Natural Distribution Shifts` ($\mathcal{C}_1$) and `Cross-Datasets Generalization` ($\mathcal{C}_2$) in accordance with [17, 41]. For $\mathcal{C}_1$, we utilize four datasets—ImageNet-V2 [40], ImageNet-A [21], ImageNet-R [20], and ImageNet-Sketch [45]—as out-of-distribution (OOD) data for ImageNet [9], to assess the effectiveness of our method. For $\mathcal{C}_2$, we utilize 10 image classification datasets covering a diverse range of visual recognition tasks. This set includes the generic-objects dataset Caltech101 [11] and five fine-grained datasets: OxfordPets [37], StanfordCars [28], Flower102 [35], Food101 [3], and FGVC-Aircraft [34]. These fine-grained datasets encompass images of animals, flowers, and transportation. Additionally, four datasets covering scenes, textures, satellite imagery, and human actions are considered: SUN397 [48], DTD [8], EuroSAT [19], and UCF101 [42].

**Baselines**: To evaluate our proposed approach, we adopt two groups of VLM methods: $\mathcal{M}_1$, which perform strict zero-shot classification, *i.e.* without any form of few-shot pre-training or external model support, for fair comparison; and $\mathcal{M}_2$, standard baselines followed in the recent state-of-the-art methods.

- For $\mathcal{M}_1$: TPT [41], a state-of-the-art test-time prompt tuning method optimizing learnable prompts across multiple augmented views; CALIP [16], introduces parameter-free attention to enhance the exchange of informative features between images and text in CLIP; and standard zero-shot CLIP is included with default configurations.
- For $\mathcal{M}_2$: CoOp [53], a few-shot prompt tuning method that adjusts a template prompt for each downstream task; CoCoOp [52]: an enhanced method for few-shot prompt-tuning that generates input-conditional prompts using a lightweight neural network; and zero-shot CLIP with an Ensemble [39] of 80 specially crafted prompts; and PromptAlign [17], an extension of TPT that incorporates multi-modal prompt learning for explicit alignment of feature distributions.

**Reproducibility**: All baselines are reproduced on our system to ensure fairness. PromptAlign [17] does not use pre-trained prompts and PromptAlign$^{\dagger}$ uses pre-trained prompts. DiffTPT [12] is not considered due to impracticality in replicating the method, given the extensive time required for generating diffusion-based samples during inference. Additionally, their reported scores with 4 update steps would not offer a fair comparison.

## 4.2. Main Results

**Natural Distribution Shifts**: Table 1 summarizes the evaluation of our method comparing $\mathcal{M}_1$ and $\mathcal{M}_2$ baseline methods under $\mathcal{C}$ase 1 with ViT-B/16 backbone under `strict zero-shot` settings. We can see that: **Our approach outperforms all $\mathcal{M}_1$ and $\mathcal{M}_2$ baseline methods**, across all four out-of-distribution (OOD) datasets, demonstrating a substantial increase in OOD generalization performance. TTL achieves significant in-domain performance gain compared to $\mathcal{M}_1$ methods like CLIP, TPT, CALIP. However, its in-domain performance is suboptimal compared to $\mathcal{M}_2$ methods like CoOp and CoCoOp. This is due to TTL operating in strict zero-shot settings without external weights, while CoOp and CoCoOp benefit from pre-training on ImageNet with few-shot prompt tuning, providing better initialization and a degree of overfitting to the ImageNet distribution. Across the average OOD dataset, TTL shows consistent performance gain in handling natural distribution shifts compared to CLIP, TPT, CALIP ($\mathcal{M}_1$ methods), Ensemble, CoOp, CoCoOp, and PromptAlign ($\mathcal{M}_2$ methods) improving from 55.31, 60.69, 57.16, 57.53, 58.51, 58.41, and 52.65 to **62.80**, respectively. Supporting our initial hypothesis that low-rank attention weight adaptation for a single test sample improves the generalization, this highlights the superior adaptability of our method in handling OOD domain shifts. This establishes TTL as a *go-to* choice over SOTA $\mathcal{M}_1$ methods like TPT and PromptAlign, which update learnable prompts during inference.

**Generalization to Cross-Dataset Transfer**: To evaluate the generalization performance of our proposed method and

Table 1. **Top 1 accuracy** % of state-of-the-art baselines (*i.e.*, $\mathcal{M}_1$ and $\mathcal{M}_2$) under `strict zero-shot settings`, where **ImageNet-Sk.** indicates the ImageNet-Sketch dataset, **OOD Avg.** indicates the OOD average results. $bs.$ indicates the baseline, *i.e.*,CLIP-ViT-B-16. 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-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(2.50) ↑ | 59.73(2.02) ↑ | 57.53(2.22) ↑ |
| CoOp$_{2021}$ | **72.30**(5.00) ↑ | 49.25(2.11) ↑ | 65.70(5.80) ↑ | 71.50(0.30) ↑ | 47.60(4.60) ↑ | 61.27(3.56) ↑ | 58.51(3.20) ↑ |
| CoCoOp$_{2022}$ | 71.40(4.10) ↑ | 50.05(2.91) ↑ | 63.80(3.90) ↑ | 73.10(1.90) ↑ | 46.70(3.70) ↑ | 61.01(3.30) ↑ | 58.41(3.10) ↑ |
| PromptAlign$_{2023}$ | 60.02(7.28) ↓ | 45.52(1.62) ↓ | 54.53(5.37) ↓ | 72.84(1.64) ↑ | 37.72(5.28) ↓ | 54.13(3.58) ↓ | 52.65(2.66) ↓ |
| TPT$_{2022}$ | 68.90(1.60) ↑ | 54.59(7.45) ↑ | 63.13(3.23) ↑ | 77.05(5.85) ↑ | 47.99(4.99) ↑ | 62.33(4.62) ↑ | 60.69(5.38) ↑ |
| CALIP$_{2023}$ | 66.74(0.56) ↓ | 47.76(0.62) ↑ | 60.76(0.86) ↑ | 73.99(2.79) ↑ | 46.12(3.12) ↑ | 59.07(1.36) ↑ | 57.16(1.85) ↑ |
| **TTL (Ours)** | 70.23(2.93) ↑ | **60.51**(13.37) ↑ | **64.55**(4.65) ↑ | **77.54**(6.34) ↑ | **48.61**(5.61) ↑ | **64.29**(6.58) ↑ | **62.80**(7.49) ↑ |

Table 2. **Top 1 accuracy** % of state-of-the-art baselines (*i.e.*, $\mathcal{M}_1$ and $\mathcal{M}_2$) under `strict zero-shot settings`, where **Average** 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-ViT-B/16. Detailed analyses are provided in Sec. 4.2.

| **Method** | **Flower102** [35] | **DTD** [8] | **OxfordPets** [37] | **UCF** [42] | **Caltech101** [11] | **Aircraft** [34] |
|---|---|---|---|---|---|---|
| CLIP-ViT-B/16 | 67.94($bs.$) | 44.10($bs.$) | 85.71($bs.$) | 63.37($bs.$) | 90.29($bs.$) | 24.70($bs.$) |
| Ensemble | 67.65 | 44.87 | 86.20 | 64.36 | 90.89 | 24.40 |
| CoOp$_{2021}$ [53] | 66.08 | 42.17 | **89.00** | 66.04 | 91.69 | 18.00 |
| CoCoOp$_{2022}$ [52] | 70.88 | 44.78 | 88.71 | 68.42 | 92.49 | 24.20 |
| PromptAlign$_{2023}$ [17] | 51.60(16.34) ↓ | 27.60(16.50) ↓ | 75.82(9.89) ↓ | 57.31(6.06) ↓ | 87.18(3.11) ↓ | 6.96(17.74) ↓ |
| PromptAlign$^{\dagger}_{2023}$ [17] | 70.56 | 45.57 | 88.96 | 69.10 | 92.86 | 23.70 |
| TPT$_{2022}$ [41] | 69.31(1.37) ↑ | 46.23(2.13) ↑ | 86.49(0.78) ↑ | 66.44(3.07) ↑ | 92.49(2.20) ↑ | **24.90**(0.20) ↑ |
| CALIP$_{2023}$ [16] | 67.64(0.30) ↓ | 44.44(0.34) ↑ | 87.82(2.11) ↑ | 64.05(0.68) ↑ | 93.27(2.98) ↑ | 24.12(0.58) ↓ |
| **TTL (Ours)** | **70.48**(2.54) ↑ | **46.69**(2.59) ↑ | 88.72(3.01) ↑ | **69.20**(5.83) ↑ | **93.63**(3.34) ↑ | 23.82(1.78) ↓ |

| **Method** | **EuroSAT** [19] | **StanfordCars** [28] | **Food101** [3] | **SUN397** [48] | **Average** |
|---|---|---|---|---|---|
| CLIP-ViT-B/16 | 40.64($bs.$) | 66.58($bs.$) | 85.05($bs.$) | 61.88($bs.$) | 63.03($bs.$) |
| Ensemble | 47.01 | 67.60 | 85.35 | 64.65 | 64.30 |
| CoOp$_{2021}$ [53] | 35.36 | 63.44 | 85.15 | 61.54 | 61.85 |
| CoCoOp$_{2022}$ [52] | 39.23 | 65.22 | 86.53 | 64.65 | 64.51 |
| PromptAlign$_{2023}$ [17] | 35.57(5.07) ↓ | 58.70(7.88) ↓ | 82.23(2.82) ↓ | 57.84(4.04) ↓ | 54.08(8.95) ↓ |
| PromptAlign$^{\dagger}_{2023}$ [17] | 34.91 | 67.43 | 86.85 | 67.73 | 64.76 |
| TPT$_{2022}$ [41] | 37.15(3.49) ↓ | 66.50(0.08) ↓ | **86.93**(1.88) ↑ | 63.48(1.60) ↑ | 63.99(0.96) ↑ |
| CALIP$_{2023}$ [16] | **42.27**(1.63) ↑ | 65.80(0.78) ↓ | 82.76(2.29) ↓ | 62.52(0.64) ↑ | 63.47(0.44) ↑ |
| **TTL (Ours)** | 42.02(1.38) ↑ | **67.96**(1.38) ↑ | 85.05(0.00) ≈ | **66.32**(4.44) ↑ | **65.39**(2.36) ↑ |

baselines on the 10 $\mathcal{C}_2$ datasets, we analyze the results within the `strict zero-shot` settings, as shown in Table 2. We can see that: **Our method outperforms all seven baselines on average across all $\mathcal{C}_2$ datasets**, individually surpassing six out of ten $\mathcal{C}_2$ datasets. Among the $\mathcal{M}_1$ methods, TTL exhibits consistent improvements, outperforming CLIP, TPT, and CALIP, with average improvements of +2.36, +1.40, and +1.91, reaching up to **65.39** average accuracy. Additionally, compared to $\mathcal{M}_2$ methods, TTL achieves average improvements of +1.09, +3.53, +0.88, and +11.31 when compared to Ensemble, CoOp, CoCoOp, and PromptAlign respectively. These results affirm that our proposed TTL, which combines the effect of weighted entropy minimization and low-rank adaptation, achieves superior distribution alignment compared to all baselines, including CLIP. This indicates that our method is a promising approach for zero-shot adaptation and demonstrates robustness to cross-data distributional variations.

**Low-Rank Adaptation *vs.* Prompt Tuning**: We assess TTL's performance against various prompt tuning approaches in `strict zero-shot` scenarios, categorizing these approaches into Text Prompt $\mathcal{T}_p$, Visual Prompt $\mathcal{V}_p$, and Multi-Modal Prompt $\mathcal{N}_p$. As illustrated in Figure 4, text prompt-tuning approaches $\mathcal{T}_p$ like TPT [41] is effective for test-time zero-shot adaptation as it learns a text prompt instead of a standard template. However, approaches utilizing visual $\mathcal{V}_p$ and multi-modal prompts $\mathcal{N}_p$ at test-time without pre-training not only fail to improve but also show reduced generalization compared to base CLIP across diverse domain shifts.

Table 3. **Effect of Weighted Entropy** under `strict zero-shot settings`, where **Average** indicates average accuracies of the `Cross-Datasets Generalization`. 'TTL *w/o* Wt. Ent.' indicate TTL without weighted entropy approach

| Method | **Flower102** [35] | **DTD** [8] | **OxfordPets** [37] | **UCF** [42] | **Caltech101** [11] | **Aircraft** [34] |
|---|---|---|---|---|---|---|
| CLIP-ViT-B/16 | 67.94(*bs.*) | 44.10(*bs.*) | 85.71(*bs.*) | 63.37(*bs.*) | 90.29(*bs.*) | 24.70(*bs.*) |
| TPT$_{2022}$ [41] | 69.31(1.37) ↑ | 46.23(2.13) ↑ | 86.49(0.78) ↑ | 66.44(3.07) ↑ | 92.49(2.20) ↑ | **24.90**(0.20) ↑ |
| TPT *w* Wt. Ent. | 69.56(1.62) ↑ | 46.69(2.59) ↑ | 88.58(2.87) ↑ | 69.18(5.81) ↑ | 93.55(3.26) ↑ | 23.14(1.56) ↓ |
| TTL *w/o* Wt. Ent. | 68.78(0.84) ↑ | 45.57(1.47) ↑ | **88.91**(3.20) ↑ | 68.09(4.72) ↑ | **94.04**(3.75) ↑ | 24.72(0.88) ↓ |
| **TTL (Ours)** | **70.48**(2.54) ↑ | **46.69**(2.59) ↑ | 88.72(3.01) ↑ | **69.20**(5.83) ↑ | 93.63(3.34) ↑ | 23.82(1.78) ↓ |

| Method | **EuroSAT** [19] | **StanfordCars** [28] | **Food101** [3] | **SUN397** [48] | **Average** |
|---|---|---|---|---|---|
| CLIP-ViT-B/16 | 40.64(*bs.*) | 66.58(*bs.*) | 85.05(*bs.*) | 61.88(*bs.*) | 63.03(*bs.*) |
| TPT$_{2022}$ [41] | 37.15(3.49) ↓ | 66.50(0.08) ↓ | **86.93**(1.88) ↑ | 63.48(1.60) ↑ | 63.99(0.96) ↑ |
| TPT *w* Wt. Ent. | 41.96(1.32) ↑ | 66.37(0.21) ↓ | 84.92(0.13) ↓ | 64.96(3.08) ↑ | 64.89(1.86) ↑ |
| TTL *w/o* Wt. Ent. | **42.07**(1.43) ↑ | 66.75(0.17) ↑ | 83.65(1.40) ↓ | 62.59(0.71) ↑ | 64.52(1.40) ↑ |
| **TTL (Ours)** | 42.02(1.38) ↑ | **67.96**(1.38) ↑ | 85.05(0.00) ≈ | **66.32**(4.44) ↑ | **65.39**(2.36) ↑ |

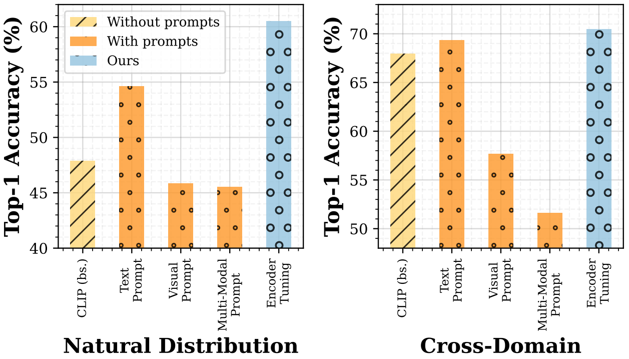


Figure 4. **Test-time performance of zero-shot methods.** CLIP *vs.* Textual Prompt Tuning (TPT) *vs.* Visual Prompt Tuning *vs.* Multi-modal Prompt Tuning *vs.* **TTL (Ours)** (See Figure 13).

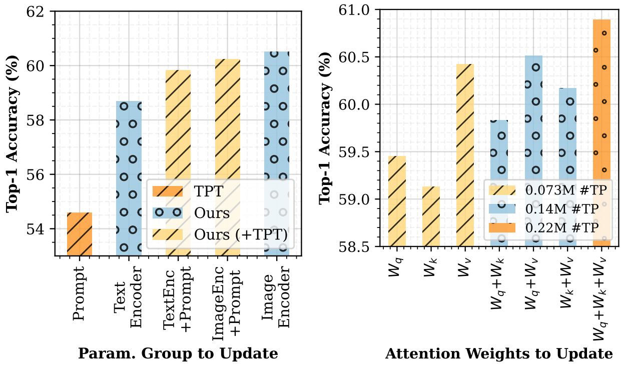


Figure 5. **Test-time Low-Rank Adaption** across (a) (left) different combinations of trainable model components (b) (right) different combinations of query, key, and value of image encoder.

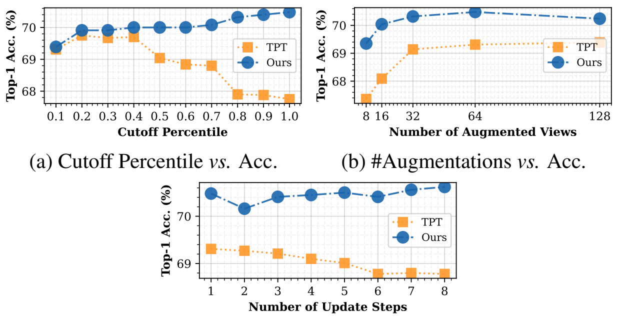
Supporting our observation, the $\mathcal{N}_p$ method PromptAlign [17], which appends prompts to both text and image encoders, shows that explicitly aligning visual feature distributions with an alignment loss enhances CLIP's generalization. However, this effectiveness relies on few-shot pretrained multi-modal prompts for test-time adaptation. Without pre-training, PromptAlign underperforms base CLIP. When pre-trained, PromptAlign surpass zero-shot methods, but at the expense of eliminating the essence of zero-shot generalization, essential for real-world scenarios (See Table 4). This reliance on pre-trained prompts indicates that without embedded prior knowledge, PromptAlign struggles with distribution shifts and meaningful representation learning. In contrast, TTL outperforms all $\mathcal{T}_p$, $\mathcal{V}_p$, and $\mathcal{N}_p$ approaches, achieving higher performance with respective gains of +2.11, +8.98, and +10.00 in $\mathcal{C}_1$ and $\mathcal{C}_2$ cases without any pre-training (Figure 7).

## 5. Analysis and Ablation

We conduct a range of empirical analyses and ablation studies to assess the impact of different design choices in our method. Unless specified otherwise, we present the analyses using the ImageNet-A dataset with ViT-B/16 backbone, opting for the smallest domain generalization variant for simplicity.

**Optimizing Different Parameter Groups**: We investigate the effectiveness of optimizing different components within TTL framework for test-time adaptation. We compare four different parameter groups for optimization at test-time: `Text Encoder + TTL`, `Text Encoder + Prompt`, `Image Encoder + Prompt`, and `Image Encoder + TTL`. From Figure 5a, we notice that simply optimizing TTL in the `Text Encoder` achieves higher performance than prompt tuning methods. Additionally, utilizing TTL inside `Image Encoder` achieves the maximum performance gain compared to other groups.

**Optimizing Different Attention Groups**: We investigate the impact of optimizing different combinations of attention weight groups ($W_Q$, $W_K$, and $W_V$) within the self-attention module. The results, as depicted in Figure 5b, indicate that optimizing LoRA parameters in $W_Q$+$W_K$+$W_V$ produces maximum performance gains. However, there is a trade-off as the inclusion of more weight groups results in linear increment of trainable parameters. Therefore, to ensure com-



(a) Cutoff Percentile *vs.* Acc.

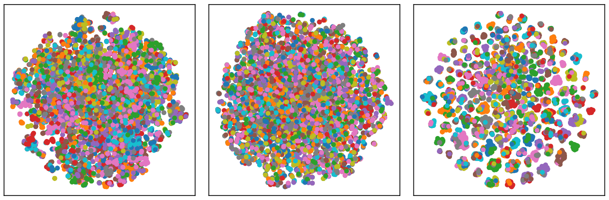
(b) #Augmentations *vs.* Acc.

(c) #Update steps *vs.* Accuracy

Figure 6. Analysis of compute resource constraints on performance across Cross-domain data on average.

putation efficiency while maintaining decent performance, we optimize LoRA parameters within $W_Q$ and $W_V$ projection layers.

**Analysing the Effect of Weighted Entropy**: We observe that instead of discarding the low entropy augmented views and assigning variable weights to the self-entropy of the predictions from each augmented view is advantageous for test-time optimization. Specifically, it is important to recognize that *a model predicting a test sample with high confidence* i.e. *low entropy, does not necessarily indicate a correct prediction* (Figure 1a). Table 3 illustrates a notable +0.87 average performance gain achieved by our method when using weighted entropy, surpassing the performance without weighted entropy. This integration effectively considers the contribution of both low and high entropy samples during optimization, enhancing test-time adaptation and improving generalization across both $\mathcal{C}_1$ and $\mathcal{C}_2$ datasets as shown in Table 1 and 2.



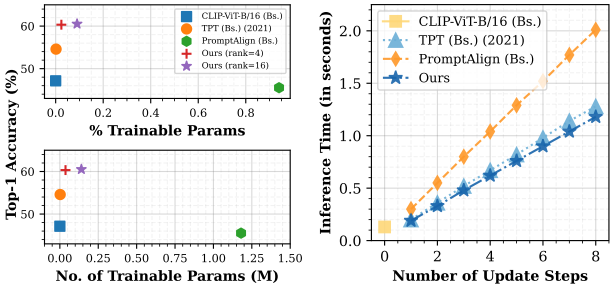
(a) TPT [41]

(b) PromptAlign [17]

(c) **TTL** (Ours)

Figure 7. **t-SNE visualizations** of the final class embedding from the test sets of $\mathcal{C}_1$ dataset: ImageNet-A, following Table 1. TTL could produce linearly separable features for zero-shot generalization baselines TPT, and PromptAlign.

**Trade-off between Inference Efficiency and Performance**: We analyze three factors influencing TTL's efficiency and performance: the cutoff percentile $\rho$ for confidence selection, the number of augmented views $N$ at test-time, and the number of update steps $S$. As shown in Figure 6a, TTL achieves maximum performance when $\rho = 1$ *i.e.* incorporating the entropy contribution from all the augmented views through our weighted entropy approach. Figure 6b shows a performance gain with increasing $N$, plateauing around $N = 64$. Figure 6c shows that with additional update steps, TTL consistently adapts better to the test sample, in contrast to TPT, whose performance trajectory is lower than TTL and exhibits optimal performance with $S$=1, and further updates do not enhance the performance. This suggests that a few optimization steps suffice for optimal generalization.



(a) #Params vs Accuracy

(b) Test-Time Update Steps

Figure 8. **Computational Efficiency.** (a) Trainable Parameters *vs.* Accuracy. Total number of parameters in base CLIP-ViT-B/16 is 124.32 M (b) Number of optimization steps per sample *vs.* the Inference time (in seconds).

**Trade-off Between Computational Cost and Performance**: TTL introduces trainable parameters (`TP`) for the attention matrices of the Image Encoder. In comparison, prompt tuning involves 2K `TP`, and multi-modal tuning employs 1.18M `TP`, while TTL utilizes only 36K `TP` for updates during inference. This signifies a subtle trade-off between the number of `TP` and efficiency, as depicted in Figure 8a. Notably, this additional `TP` in TTL, compared to TPT and CLIP, does not adversely impact or introduce extra latency during test-time optimization. In fact, as illustrated in Figure 8b, TTL achieves slightly lower inference time with an increasing number of update steps, showcasing superior performance with higher parameter efficiency.

## 6. Conclusion

We present Test-Time Low-rank adaptation (TTL), a novel parameter-efficient strategy for achieving zero-shot generalization in vision-language models (VLMs). TTL provides an efficient alternative to traditional test-time prompt tuning methods by updating the attention weights of CLIP's visual encoder using Low-Rank adapters, thereby adapting the model for downstream recognition tasks without any fine-tuning or pre-training. Additionally, TTL incorporates a confidence maximization mechanism through the utilization of weighted entropy loss derived from augmented sample predictions. Notably, TTL achieves superior performance without the requirement of a source dataset or pre-trained prompts, outperforming current state-of-the-art CLIP zero-shot generalization methods in both domain generalization and cross-dataset evaluation scenarios.

## References

[1] Beery, S., Van Horn, G., Perona, P.: Recognition in terra incognita. In: Proceedings of the European conference on computer vision (ECCV). pp. 456–473 (2018) 2

[2] Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. arXiv preprint arXiv:2311.15127 (2023) 2

[3] Bossard, L., Guillaumin, M., Van Gool, L.: 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. pp. 446–461. Springer (2014) 5, 6, 7, 16

[4] Cascante-Bonilla, P., Shehada, K., Smith, J.S., Doveh, S., Kim, D., Panda, R., Varol, G., Oliva, A., Ordonez, V., Feris, R., Karlinsky, L.: Going beyond nouns with vision & language models using synthetic data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 20155–20165 (October 2023) 2

[5] Chen, D., Wang, D., Darrell, T., Ebrahimi, S.: Contrastive test-time adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 295–305 (2022) 1

[6] Chen, G., Liu, F., Meng, Z., Liang, S.: Revisiting parameter-efficient tuning: Are we really there yet? arXiv preprint arXiv:2202.07962 (2022) 2

[7] Chen, J., Zhu, D., Shen, X., Li, X., Liu, Z., Zhang, P., Krishnamoorthi, R., Chandra, V., Xiong, Y., Elhoseiny, M.: Minigpt-v2: large language model as a unified interface for vision-language multi-task learning. arXiv preprint arXiv:2310.09478 (2023) 2

[8] Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3606–3613 (2014) 5, 6, 7, 16

[9] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255. IEEE (2009) 5

[10] Fan, Y., Watkins, O., Du, Y., Liu, H., Ryu, M., Boutilier, C., Abbeel, P., Ghavamzadeh, M., Lee, K., Lee, K.: Dpok: Reinforcement learning for fine-tuning text-to-image diffusion models. In: Oh, A., Neumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems. vol. 36, pp. 79858–79885. Curran Associates, Inc. (2023), https://proceedings.neurips.cc/paper_files/paper/2023/file/fc65fab891d83433bd3c8d966edde311-Paper-Conference.pdf 2

[11] Fei-Fei, L., Fergus, R., Perona, P.: 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. pp. 178–178. IEEE (2004) 5, 6, 7, 16

[12] Feng, C.M., Yu, K., Liu, Y., Khan, S., Zuo, W.: Diverse data augmentation with diffusions for effective test-time prompt tuning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2704–2714 (2023) 2, 5, 13

[13] Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. Nature Machine Intelligence **2**(11), 665–673 (2020) 2

[14] Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp. 249–256. JMLR Workshop and Conference Proceedings (2010) 5, 12

[15] Guo, Y., Yang, C., Rao, A., Wang, Y., Qiao, Y., Lin, D., Dai, B.: Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. arXiv preprint arXiv:2307.04725 (2023) 2

[16] Guo, Z., Zhang, R., Qiu, L., Ma, X., Miao, X., He, X., Cui, B.: Calip: Zero-shot enhancement of clip with parameter-free attention. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 746–754 (2023) 2, 5, 6

[17] Hassan, J., Gani, H., Hussein, N., Khattak, M.U., Naseer, M., Khan, F.S., Khan, S.: Align your prompts: Test-time prompting with distribution alignment for zero-shot generalization (2024) 1, 2, 3, 4, 5, 6, 7, 8, 12, 13, 14

[18] He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the

IEEE International Conference on Computer Vision. pp. 1026–1034 (2015) 12

[19] Helber, P., Bischke, B., Dengel, A., Borth, D.: 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) 5, 6, 7, 16

[20] Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., 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. pp. 8340–8349 (2021) 5

[21] Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D.: Natural adversarial examples. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15262–15271 (2021) 5

[22] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021) 2, 3

[23] Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: ICML. pp. 4904–4916. PMLR (2021) 2

[24] Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N.: Visual prompt tuning. In: European Conference on Computer Vision. pp. 709–727. Springer (2022) 1

[25] Khattak, M.U., Naeem, M.F., Naseer, M., Van Gool, L., Tombari, F.: Learning to prompt with text only supervision for vision-language models. arXiv preprint arXiv:2401.02418 (2024) 1

[26] Khattak, M.U., Rasheed, H., Maaz, M., Khan, S., Khan, F.S.: Maple: Multi-modal prompt learning. In: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023) 1, 3

[27] Khattak, M.U., Wasim, S.T., Naseer, M., Khan, S., Yang, M.H., Khan, F.S.: Self-regulating prompts: Foundational model adaptation without forgetting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15190–15200 (2023) 1

[28] Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 554–561 (2013) 5, 6, 7, 16

[29] Kumar, A., Raghunathan, A., Jones, R., Ma, T., Liang, P.: Fine-tuning can distort pretrained features and underperform out-of-distribution. In: International Conference on Learning Representations (2022) 1

[30] Lai, X., Tian, Z., Chen, Y., Li, Y., Yuan, Y., Liu, S., Jia, J.: Lisa: Reasoning segmentation via large language model. arXiv preprint arXiv:2308.00692 (2023) 2

[31] Lee, J., Jung, D., Lee, S., Park, J., Shin, J., Hwang, U., Yoon, S.: Entropy is not enough for test-time adaptation: From the perspective of disentangled factors. In: The Twelfth International Conference on Learning Representations (2023) 3, 4

[32] Liu, F., Lin, K., Li, L., Wang, J., Yacoob, Y., Wang, L.: Aligning large multi-modal model with robust instruction tuning. arXiv preprint arXiv:2306.14565 (2023) 2

[33] Liu, Y., Kothari, P., van Delft, B., Bellot-Gurlet, B., Mordan, T., Alahi, A.: Ttt++: When does self-supervised test-time training fail or thrive? In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 21808–21820. Curran Associates, Inc. (2021), https://proceedings.neurips.cc/paper_files/paper/2021/file/b618c3210e934362ac261db280128c22-Paper.pdf 2

[34] Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151 (2013) 5, 6, 7, 16

[35] Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing. pp. 722–729. IEEE (2008) 5, 6, 7, 16

[36] Niu, S., Wu, J., Zhang, Y., Chen, Y., Zheng, S., Zhao, P., Tan, M.: Efficient test-time model adaptation without forgetting. In: International conference on machine learning. pp. 16888–16905. PMLR (2022) 3

[37] Parkhi, O.M., Vedaldi, A., Zisserman, A., Jawahar, C.: Cats and dogs. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 3498–3505. IEEE (2012) 5, 6, 7, 16

[38] Qiu, Z., Liu, W., Feng, H., Xue, Y., Feng, Y., Liu, Z., Zhang, D., Weller, A., Schölkopf, B.: Controlling text-to-image diffusion by orthogonal finetuning. In: Oh, A., Neumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems. vol. 36, pp. 79320–79362. Curran Associates, Inc. (2023), https://proceedings.neurips.cc/paper_files/paper/2023/file/faacb7a4827b4d51e201666b93ab5fa7-Paper-Conference.pdf 2

[39] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763. PMLR (2021) 1, 2, 5, 13, 14, 15

[40] Recht, B., Roelofs, R., Schmidt, L., Shankar, V.: Do imagenet classifiers generalize to imagenet? In: International Conference on Machine Learning. pp. 5389–5400. PMLR (2019) 5

[41] Shu, M., Nie, W., Huang, D.A., Yu, Z., Goldstein, T., Anandkumar, A., Xiao, C.: Test-time prompt tuning for zero-shot generalization in vision-language models. Advances in Neural Information Processing Systems **35**, 14274–14289 (2022) 1, 2, 3, 4, 5, 6, 7, 8, 13, 14

[42] Soomro, K., Zamir, A.R., Shah, M.: A dataset of 101 human action classes from videos in the wild. Center for Research in Computer Vision **2**(11) (2012) 5, 6, 7, 16

[43] Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., Hardt, M.: Test-time training with self-supervision for generalization under distribution shifts. In: International Conference on Machine Learning. pp. 9229–9248. PMLR (2020) 2

[44] Wang, D., Shelhamer, E., Liu, S., Olshausen, B., Darrell, T.: Tent: Fully test-time adaptation by entropy minimization. arXiv preprint arXiv:2006.10726 (2020) 2, 3

[45] Wang, H., Ge, S., Lipton, Z., Xing, E.P.: Learning robust global representations by penalizing local predictive power. Advances in Neural Information Processing Systems **32** (2019) 5

[46] Wang, W., Chen, Z., Chen, X., Wu, J., Zhu, X., Zeng, G., Luo, P., Lu, T., Zhou, J., Qiao, Y., Dai, J.: Visionllm: Large language model is also an open-ended decoder for vision-centric tasks. In: Oh, A., Neumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems. vol. 36, pp. 61501–61513. Curran Associates, Inc. (2023), https://proceedings.neurips.cc/paper_files/paper/2023/file/c1f7b1ed763e9c75e4db74b49b76db5f-Paper-Conference.pdf 2

[47] Wortsman, M., Ilharco, G., Kim, J.W., Li, M., Kornblith, S., Roelofs, R., Lopes, R.G., Hajishirzi, H., Farhadi, A., Namkoong, H., et al.: Robust fine-tuning of zero-shot models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7959–7971 (2022) 1

[48] Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 3485–3492. IEEE (2010) 5, 6, 7, 16

[49] Xu, Z., Chen, Z., Zhang, Y., Song, Y., Wan, X., Li, G.: Bridging vision and language encoders: Parameter-efficient tuning for referring image segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 17503–17512 (October 2023) 2

[50] Zhang, M., Levine, S., Finn, C.: Memo: Test time robustness via adaptation and augmentation. Advances in Neural Information Processing Systems **35**, 38629–38642 (2022) 3

[51] Zhou, C., Ma, X., Michel, P., Neubig, G.: Examining and combating spurious features under distribution shift. In: International Conference on Machine Learning. pp. 12857–12867. PMLR (2021) 2

[52] Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Conditional prompt learning for vision-language models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022) 1, 3, 5, 6

[53] Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. International Journal of Computer Vision (IJCV) (2022) 1, 2, 5, 6