Black-Box Test-Time Prompt Tuning for Vision-Language Models

Fan'an Meng[1], Chaoran Cui[1*], Hongjun Dai[2*], Shuai Gong[1]

[1]Shandong University of Finance and Economics
[2]Shandong University
fameng@mail.sdu.edu.cn, crcui@sdufe.edu.cn, dahogn@sdu.edu.cn, gsh8210@163.com

Abstract

Test-time prompt tuning (TPT) aims to adjust the vision-language models (e.g., CLIP) with learnable prompts during the inference phase. However, previous works overlooked that pre-trained models as a service (MaaS) have become a noticeable trend due to their commercial usage and potential risk of misuse. In the context of MaaS, users can only design prompts in inputs and query the black-box vision-language models through inference APIs, rendering the previous paradigm of utilizing gradient for prompt tuning is infeasible. In this paper, we propose black-box test-time prompt tuning (B^2TPT), a novel framework that addresses the challenge of optimizing prompts without gradients in an unsupervised manner. Specifically, B^2TPT designs a consistent or confident (CoC) pseudo-labeling strategy to generate high-quality pseudo-labels from the outputs. Subsequently, we propose to optimize low-dimensional intrinsic prompts using a derivative-free evolution algorithm and to project them onto the original text and vision prompts. This strategy addresses the gradient-free challenge while reducing complexity. Extensive experiments across 15 datasets demonstrate the superiority of B^2TPT. The results show that B^2TPT not only outperforms CLIP's zero-shot inference at test time, but also surpasses other gradient-based TPT methods.

Introduction

Recent large Vision-Language Models (VLMs), such as CLIP (Radford et al. 2021), have attracted widespread attention for their robust zero-shot capabilities, significantly improving performance on downstream tasks. Adapting large models to downstream tasks poses considerable challenges. Fine-tuning all parameters of a large model is not only costly but also impractical, particularly for modern transformer-based architectures. Consequently, instead of modifying or fine-tuning the pre-trained model, the inputs are adjusted. Drawing inspiration from natural language processing (NLP) techniques, prompt tuning provides a straightforward and efficient method to adapt large models to downstream tasks. However, on the one hand, the diverse range of downstream tasks makes it challenging to design prompts for each specific domain. On the other hand, training data with annotations are costly and difficult to obtain in practice. To address the above challenge, test-time prompt tuning (TPT) (Shu et al. 2022) has been proposed to learn domain-specific prompts from test data in the absence of labeled training data.

Typically, existing TPT methods leverage tunable prompts to adapt PTMs to unseen domains and many studies (Zhang, Zhou, and Li 2024; Ma et al. 2024; Tsai, Mao, and Yang 2024; Ben-David, Oved, and Reichart 2022) have investigated tuning prompts via back-propagation. For example, TPT (Shu et al. 2022) and PromptSync (Khandelwal 2024) optimize instance-specific prompts under the guidance of entropy minimization (Grandvalet and Bengio 2004) or contrastive learning objectives (Cui et al. 2023) based on inputs. PromptAlign (Abdul Samadh et al. 2024) extends TPT by incorporating a token alignment strategy, which addresses distribution shifts in test samples. The means and variances of the test image token embeddings are aligned with the image token embeddings of an offline proxy source dataset.

Nevertheless, existing TPT methods overlook that VLMs are often released as services due to commercial use and the potential for misuse. In this scenario, referred to as Model as a Service (MaaS), the model can only be accessed through black-box APIs. It offers several APIs that allow users to design custom prompts for querying the VLMs without access to gradients. However, all previous methods rely on back-propagation to tune the prompts, and they become ineffective when gradients are unavailable for prompt updates.

In the absence of gradients, one can apply derivative-free optimization algorithms (Kennedy and Eberhart 1995; Bertsimas and Tsitsiklis 1993; Hansen and Ostermeier 2001). Despite relying solely on the function values without gradient back-propagation, such algorithms converge slowly when the search space is large. Fortunately, several studies (Aghajanyan, Gupta, and Zettlemoyer 2021; Qin et al. 2021) have demonstrated that VLMs, despite having recent parameters, possess low intrinsic dimensionality. Sun et al. (2022) successfully applied derivative-free optimization algorithms to the large language model RoBERTa (Liu et al. 2019), but their approach focused on optimizing text prompts for NLP, which makes it inapplicable to TPT in VLMs. Yu et al. (2023) demonstrated the feasibility of optimizing the vision-language model CLIP (Radford et al. 2021) using similar

*Corresponding Author

algorithms, but their work was conducted under supervised training, which renders it unsuitable for unlabeled TPT during testing.

Based on the above insights, this paper proposes a Black-Box Test-time Prompt Tuning (B^2TPT) method to optimize prompts without gradients when VLMs are accessed as interface APIs. Following previous research (Shu et al. 2022; Yu et al. 2023), we use CLIP as the backbone to optimize text and vision prompts jointly in input. B^2TPT addresses the two challenges of the lack of labels and gradients. For the absence of labels, B^2TPT introduces a consistent or confident (CoC) strategy to obtain high-quality pseudo-labels. Initially, the zero-shot inference capability of CLIP is used by default to generate pseudo-labels. In the intermediate phase, since the model may still be insufficiently trained and the generated pseudo-labels could contain significant errors, we continue to rely on CLIP to obtain pseudo-labels. Finally, pseudo-labels are determined by comparing the outputs of CLIP with those of the prompt-based adjusted model. If the results are consistent, the pseudo-label is confirmed; otherwise, the pseudo-label with higher confidence is selected. If the results are consistent, a definite pseudo-label is obtained; otherwise, the pseudo-label with higher confidence is selected. For the absence of gradients, the prompts are updated using covariance matrix adaptation evolution strategy (CMA-ES) (Hansen and Ostermeier 2001). To address the challenges of derivative-free algorithms in high-dimensional spaces, we optimize intrinsic prompts in a low-dimensional subspace rather than in the original parameter space.

Extensive experiments were conducted on two representative benchmarks: domain generalization and cross-dataset generalization. Our B^2TPT method demonstrates significant improvements over numerous state-of-the-art TPT methods, including gradient-based ones.

In summary, our main contributions are as follows[1]:

- We design a framework named B^2TPT that jointly optimizes text and vision prompts to address the challenge of test-time prompt tuning in the MaaS scenario. To the best of our knowledge, this is the first work to investigate TPT in the MaaS scenario.
- We propose an intrinsic prompt generation strategy that successfully extends CMA-ES to black-box prompt tuning without gradients. In particular, we design a consistent or confident (CoC) pseudo-labeling strategy for obtaining high-quality pseudo-labels.
- We evaluate the effectiveness of our method from different perspectives and achieve state-of-the-art results on two benchmark datasets.

Related Work

Vision-Language Models

With the success of pre-trained models in CV and NLP, many efforts have been made to pre-train large-scale models on both vision and language modalities, known as Vision-Language Models (VLMs). CLIP (Radford et al. 2021) is a notable example, training text and vision encoders with contrastive loss to compute similarity scores. After pre-training on 400 million data pairs collected from the Internet, CLIP demonstrates impressive zero-shot capabilities across various downstream tasks. Based on CLIP, VLM fine-tuning methods can be categorized into two groups: prompt-based methods (Zhou et al. 2022b,a) and adapter-based methods. Prompt-based methods aim to learn continuous prompts for downstream tasks using back-propagation on few-shot datasets. For example, CoOp (Zhou et al. 2022b) transformed hand-crafted prompts into learnable continuous prompts and fine-tuned them to adapt to downstream tasks. CoCoOp (Zhou et al. 2022a) extended CoOp by employing a meta-network and image features to generate specific prompts for each image. Adapter-based methods do not modify the prompts but instead add additional classification layers to the backbone models. CLIP-Adapter (Gao et al. 2024) added an additional bottleneck layer to learn new features and employed residual-style feature blending with the original pre-trained features. Tip-Adapter (Zhang et al. 2022) used non-parametric key-value cache models to fine-tune the CLIP model. Graph-Adapter (Li et al. 2024) implemented a textual adapter with a dual knowledge graph consisting of textual and visual knowledge sub-graph, yielding a more effective classifier for downstream tasks.

Despite impressive progress, these methods heavily rely on labeled data, making them unsuitable for test-time settings. Therefore, numerous studies focus on test-time prompt tuning (TPT) (Shu et al. 2022; Karmanov et al. 2024), exploring how to fine-tune VLMs during test-time without training data.

Test-time Prompt Tuning

With the increasing public concerns regarding expensive data annotations and their unavailability during testing, TPT has received much attention in the literature.

For instance, Shu et al. (2022) proposed test-time prompt tuning (TPT), utilizing an entropy minimization method to optimize tunable prompt for each testing sample. Feng et al. (2023) enhanced TPT by utilizing pre-trained diffusion models (DiffTPT) to generate diverse augmented images with richer visual variations. TPS (Sui, Wang, and Yeung-Levy 2024) was developed by precomputing and caching prototypes generated using a pre-trained text encoder to mitigate shifts in test environments.

However, existing TPT methods ignore that VLMs, which tend to be released as services, can only be accessed only through black-box APIs. Therefore, it becomes essential to explore methods for fine-tuning black-box VLMs at test time.

Black-box Optimization

Black-box optimization, also known as derivative-free optimization, is a discipline in mathematical optimization that does not use derivative information in the classical sense to find optimal solutions. In recent years, fine-tuning of VLMs in black-box scenarios has garnered increasing attention. It can be divided into two categories: gradient estimation and evolutionary algorithms. The former optimizes

[1]https://github.com/MFAaaaaaa/B2TPT

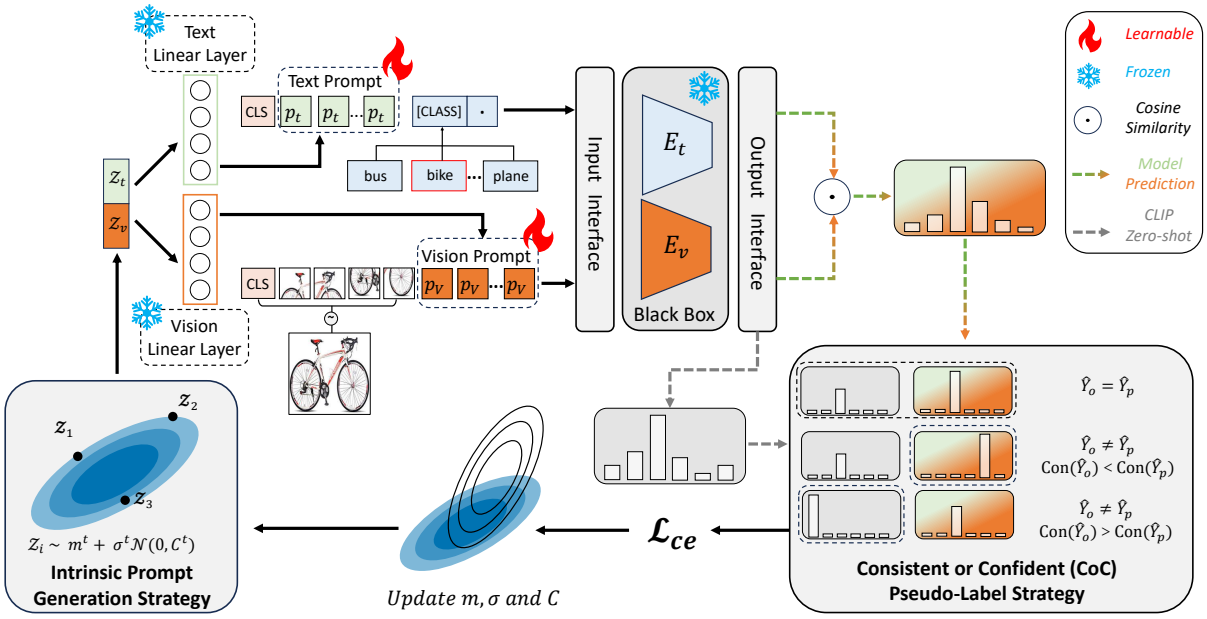


Figure 1: The overview framework of our B^{2}TPT method. Intrinsic prompt generation utilizes the CMA-ES algorithm to learn low-dimensional intrinsic prompts of $\boldsymbol{z}_t$ and $\boldsymbol{z}_v$, which are then mapped to the prompt space through linear layers. The text linear layer and vision linear layer are kept fixed with W_t and W_v, respectively. Next, $\boldsymbol{p}_t$ is prefixed, and $\boldsymbol{p}_v$ is suffixed to the input query. The black-box output interface is subsequently queried, and the CoC strategy is employed to obtain high-quality pseudo-labels. Finally, cross-entropy loss is computed to update the parameters of CMA-ES, initiating the next iteration.

prompts by estimating gradients or finding proxy gradients, while the latter uses evolutionary algorithms to generate prompt candidates from low-dimensional vectors. For example, CBBT (Guo et al. 2023) approximated the gradients of text prompts by analyzing predictions from perturbed prompt inputs. BlackVIP (Oh et al. 2023) designed an asymmetric autoencoder-style coordinator to generate input-dependent image-shaped visual prompts and optimize the coordinator by simultaneous perturbation stochastic approximation. In contrast, BPT-VLM (Yu et al. 2023) extended traditional evolution strategies to black-box prompt tuning on VLMs as a prompt generation module. It incorporates cross-modal interaction by sharing the intrinsic parameter subspaces between the vision and language modalities. Inspired by BPT-VLM, CraFT (Wang et al. 2024) added a prediction refinement module, which is designed to learn text prompts and refine output predictions. Additionally, it introduced a collaborative training algorithm to train these modules together.

Unfortunately, no methods currently address the fine-tuning of black-box VLMs during test-time, which presents a significant challenge for real-world applications. To address this, we propose B^{2}TPT, a method that uses evolutionary algorithms to solve the gradient-free optimization problem. Specifically, B^{2}TPT optimizes text and vision prompts separately, generating them from a low-dimensional space instead of optimizing the high-dimensional space of the entire model's parameters. By utilizing evolutionary strategies, we iteratively evolve prompt candidates that are best suited to the test-time scenario, all without the need for gradient-based optimization.

Preliminaries

CLIP Revisiting

CLIP consists of two encoders, namely: an image encoder $E_v(\cdot)$ that maps the image input into a feature vector and a text encoder $E_t(\cdot)$ that does the same for text inputs. It employs a contrastive loss function (Radford et al. 2021) to maximize the similarity between the two vectors, ensuring that the text and image representations are consistent in the feature space. For zero-shot classification, the prediction probability for a test sample X can be obtained by

$$p\left(y_i \mid X\right)=\frac{\exp\left(\operatorname{sim}\left(\boldsymbol{t}_i \cdot \boldsymbol{v}\right)/\tau\right)}{\sum_{i=1}^{K}\exp\left(\operatorname{sim}\left(\boldsymbol{t}_i \cdot \boldsymbol{v}\right)/\tau\right)} \tag{1}$$

where $\operatorname{sim}(\cdot,\cdot)$ denotes cosine similarity, $\boldsymbol{v}$ represents the image features extracted by E_v, and $\boldsymbol{t_i}$ represents the text features for class i extracted by E_t. Here, τ is the temperature parameter.

Test-time Prompt Tuning

Despite CLIP containing rich knowledge from diverse datasets, effectively extracting this knowledge at test time remains challenging. Fine-tuning all the parameters of a large model is both expensive and infeasible, especially for modern transformer-based architectures. Therefore, another line of research focuses on optimizing prompts based on a single test sample. We denote the forward pass of CLIP as f. The above $\{\boldsymbol{t}_i\}_{i=1}^{K}$ represents the text prompts for K

classes, and $\boldsymbol{v}$ refers to the image input to E_v. Specifically, $\boldsymbol{t}_i = E_t\left(\underline{\boldsymbol{p}_t^i}, \boldsymbol{e}_t^i\right)$, $i = 1, 2, \ldots, K$ for K categories, and $\boldsymbol{v} = E_v\left(\underline{\boldsymbol{e}_v}\right)$. Here, $\boldsymbol{e}_t^i$ denotes the word embedding of the text input and $\boldsymbol{e}_v$ denotes the patch embedding of image input. For conciseness, we omit the superscript i in the following discussion. The objective of traditional TPT methods (Shu et al. 2022) can be formulated as follows:

$$\boldsymbol{p}_t^* = \min_{\boldsymbol{p}_t} \mathcal{L}(f(E_t(\boldsymbol{p}_t, \boldsymbol{e}_t), E_v(\boldsymbol{e}_v)), \widehat{Y}) \tag{2}$$

where $\boldsymbol{p}_t$ represents the learnable prompts, and $\mathcal{L}$ denotes the cross-entropy loss function. Since no labels are available during testing, $\widehat{Y}$ represents the pseudo-label. The details of which will be discussed in the next section.

In fact, $\boldsymbol{p}_t \in \mathbb{R}^{L \times D}$ is the actual text prompt optimized by TPT methods, where L denotes the length of the prompt and D represents their dimension.

Method

Approach Overview

Both TPT (Shu et al. 2022) and DiffTPT (Feng et al. 2023) employ multi-view augmentation for a single test sample and filter confident predictions based on low entropy, subsequently optimizing only the text prompts via back-propagation. On the one hand, CLIP includes both text and vision encoders, while traditional TPT methods lack supervised signals from vision prompts. On the other hand, these methods fail when VLMs are encapsulated as black-box models that do not provide gradients.

In contrast, B^{2}TPT seeks to optimize text and vision prompts separately for a single batch of test samples without relying on gradients. Given the lack of gradients in MaaS, we can only adjust models such as CLIP in the input and output spaces. In the input space, we propose an intrinsic prompt generation strategy that employs CMA-ES (Hansen and Ostermeier 2001) to learn text and vision prompts separately. In the output space, we introduce a Consistent or Confident (CoC) pseudo-labeling strategy to select high-quality pseudo-labels from the black-box outputs. We adopt CLIP as the backbone due to its open-source nature and extensive research foundation, and B^{2}TPT is also compatible with other vision-language models.

Consistent or Confident Pseudo-labeling

Since the test-time prompt tuning is unlabeled, we choose to leverage pseudo-labels for the cross-entropy loss computation. On the one hand, the pseudo-label $\widehat{Y}$ can be derived as $\widehat{Y}_o$, obtained through CLIP's zero-shot inference capability. On the other hand, the output of the black-box model can be transformed into the pseudo-label $\widehat{Y}_p$ via a softmax operation. Throughout the entire testing process, if the predictions of $\widehat{Y}_o$ and $\widehat{Y}_p$ are consistent, we obtain a definitive pseudo-label $\widehat{Y}$. If $\widehat{Y}_o$ and $\widehat{Y}_p$ are inconsistent, we propose a Consistent or Confident (CoC) strategy to identify $\widehat{Y}$: At the start of the test, i.e., before the final iteration, if $\widehat{Y}_o$ and $\widehat{Y}_p$ differ, we prefer $\widehat{Y}_o$ as a pseudo-label for evaluating and updating the generation solutions of CMA-ES. This is because, in the initial iterations, $\widehat{Y}_p$ is uncertain and, thus, does not provide high-quality pseudo-labels. However, in the final iteration, if $\widehat{Y}_o$ and $\widehat{Y}_p$ differ, we select the prediction with higher confidence as the pseudo-label, $\widehat{Y}$. The reason is that, after multiple iterations of tuning, the black-box output obtained based on the prompts $\boldsymbol{p}_t$ and $\boldsymbol{p}_v$ has gained confidence.

Many previous studies (Shu et al. 2022; Zhang, Shen, and Foo 2023; Wang et al. 2021) on acquiring pseudo-labels have employed a threshold selection approach to filter test samples. For example, Tent (Wang et al. 2021) manually sets a fixed entropy threshold to filter and obtain pseudo-labels based on prediction uncertainty. However, applying this method to small batches of test samples increases the risk of overfitting. Therefore, it is important to note that CoC does not exclude any test samples. Since the number of samples tested in each batch within B^{2}TPT is quite small, filtering based on confidence would significantly reduce the sample size. In addition, CoC does not rely on fixed threshold-based filtering, thereby ensuring greater robustness across different datasets.

Intrinsic Prompt Generation

Learning prompts with a single test sample in TPT is time-consuming due to multi-view transformations. Therefore, we explore learning prompts with a batch of test samples. However, it is challenging to achieve accurate predictions for a batch of samples solely by relying on text prompts. Inspired by Maple (Khattak et al. 2023), we design a vision prompt $\boldsymbol{p}_v$ that is optimized alongside $\boldsymbol{p}_t$:

$$\boldsymbol{P}^* = \min_{\boldsymbol{p}_t, \boldsymbol{p}_v} \mathcal{L}(f(E_t\left(\underline{\boldsymbol{p}_t}, \boldsymbol{e}_t\right), E_v(\boldsymbol{e}_v, \underline{\boldsymbol{p}_v})), \widehat{Y}) \tag{3}$$

where P^* denotes the unified formulation of $(\boldsymbol{p}_t, \boldsymbol{p}_v)$. As is widely used and evaluated in previous works (Li and Liang 2021), prefix positioning is adopted for the text encoder, so we place the text prompt $\boldsymbol{p}_t$ before $\boldsymbol{e}_t$. Unlike the text encoder, we use suffix positioning for the image encoder (ViT) to retain the pre-trained positional embedding information.

We can directly utilize CMA-ES to generate $\boldsymbol{p}_t$ and $\boldsymbol{p}_v$, corresponding to the embedding dimension of 512 and 756 for E_t and E_v, respectively. The total parameters of our prompts $\boldsymbol{P}^* \in \mathbb{R}^{D_1 + D_2}$ can be in the tens of thousands, which poses a challenge for black-box optimization. Fortunately, some studies (Aghajanyan, Gupta, and Zettlemoyer 2021; Qin et al. 2021) have shown that VLMs actually have a very low intrinsic dimension, indicating that we can achieve the same effect as the full parameter space by optimizing only a low-dimensional subspace.

Specifically, we use a matrix W that projects intrinsic prompts $\boldsymbol{z}$ in the low dimension d onto the VLMs tokens $\boldsymbol{p}$, i.e., $\boldsymbol{p} = W\boldsymbol{z}$. We define $\boldsymbol{p}_t = W_t\boldsymbol{z}_t$, $\boldsymbol{p}_v = W_v\boldsymbol{z}_v$, and concatenate $\boldsymbol{z}_t$ and $\boldsymbol{z}_v$ denoted as $\boldsymbol{Z} \in \mathbb{R}^{d_1 + d_2}$, which is simple and efficient to generate via CMA-ES ($d_1 + d_2 \ll D_1 + D_2$). Our ultimate optimization objective is as follows:

$$\boldsymbol{Z}^* = \min_{\boldsymbol{z}_t, \boldsymbol{z}_v} \mathcal{L}(f(E_t(\underline{W_t\boldsymbol{z}_t}, \boldsymbol{e}_t), E_v(\boldsymbol{e}_v, \underline{W_v\boldsymbol{z}_v})), \widehat{Y}) \tag{4}$$

Algorithm 1: The training process of B^{2}TPT.

Input:
The pre-trained black-box model f, unlabeled test samples $\{x_i\}_{i=1}^{N}$, the population size λ, batch-size B, iteration size I and the hyperparameters L_t, L_v, d_1, d_2.

Initalization:
Initialize parameters m^0, σ^0, C^0 and matrices W_t, W_v.

for $n = 1$ to N/B **do**
  for $i = 1$ to I **do**
    Sample λ intrinsic prompts solutions based on m^t, σ^t, C^t as shown in Eq.(5);
    Project the intrinsic prompts into the corresponding text and vision prompt using W_t and W_v;
    Obtain the pseudo-label $\widehat{Y}$ using **CoC**;
    Compute the fitness as shown in Eq. (4);
    Update m^t, σ^t, C^t using CMA-ES.
  end for
end for

where W_t and W_v are kept fixed during generation. We initialize W_t with the observed mean and deviation of the word embedding layer in E_t and initialize W_v with the counterparts of the entry convolutional layer in E_v, which can speed up the convergence process compared to standard uniform distribution.

Because the model gradient is not accessible, we address the problem using the derivative-free optimization algorithm CMA-ES (Hansen and Ostermeier 2001) to generate intrinsic prompts $\boldsymbol{z}_t$ and $\boldsymbol{z}_v$. CMA-ES is a parametric distribution search strategy that generates solutions from a multivariate normal distribution. The solutions are generated at each iteration as follows:

$$\boldsymbol{z}_i \sim m^t + \sigma^t \mathcal{N}\left(0, C^t\right), \quad i = 1, 2, \ldots, \lambda \tag{5}$$

where i represents the index of the sample solution, λ represents the population size, m^t represents the mean of the distribution in the t-th iteration, σ^t represents the step-size, and C^t represents the covariance matrix of the distribution. In Algorithm 1, we summarize the entire process of our B^{2}TPT framework.

Experiments

Datasets

We conducted experiments on two benchmarks for test-time prompt tuning: the out-of-distribution (OOD) benchmark and the cross-dataset benchmark.

In the OOD setting, we evaluate the model's robustness to natural distributional shifts on the following four ImageNet variants considered as out-of-distribution (OOD) data, based on ImageNet (Deng et al. 2009): **ImageNet-A** (Hendrycks et al. 2021b), **ImageNet-V2** (Recht et al. 2019), **ImageNet-R** (Hendrycks et al. 2021a), **ImageNet-Sketch** (Wang et al. 2019). For the cross-dataset setting, on the other hand, we evaluate the model's performance across 10 diverse image classification datasets: Flowers102 (Nilsback and Zisserman 2008), texture classification with DTD (Cimpoi et al. 2014), fine-grained image recognition with OxfordPets (Parkhi et al. 2012), StanfordCars (Krause et al. 2013), action classification with UCF101 (Soomro, Zamir, and Shah 2012), general objects classification with Caltech101 (Fei-Fei, Fergus, and Perona 2004), Food101 (Bossard, Guillaumin, and Van Gool 2014), scene recognition with SUN397 (Xiao et al. 2010), Aircraft (Maji et al. 2013), and satellite image classification with EuroSAT (Helber et al. 2019).

Hyperparameters	Default Value
Batch-size B	32
Iteration Size I	4
Population Size λ	30
Vision Prompt Length L_v	5
Text Prompt Length L_t	8
Intrinsic Dimension $d_1 + d_2$	200

Table 1: Hyperparameters in B^{2}TPT.

Baselines

We compare the classification accuracy of our B^{2}TPT method with **CLIP** (Radford et al. 2021) which performs zero-shot inference directly using the pre-trained model of ViT-B/16 on the test data , as well as 5 state-of-the-art TPT methods. These methods update tunable prompts based on gradient back-propagation. **CoOp** (Zhou et al. 2022b) and **CoCoOp** (Zhou et al. 2022a) are both train-time methods trained on ImageNet's training set with 16 shots per category then tested on other datasets. **CoOp** tunes a learnable context prompt for each downstream dataset, while **CoCoOp** extends **CoOp** by incorporating input-conditional tokens for image features. **TPT** (Shu et al. 2022) tunes adaptive prompts on the fly with a single test sample, without requiring additional training data or annotations. **DiffTPT** (Feng et al. 2023) first utilizes the pre-trained stable diffusion to generate data with richer visual appearance variations. **C-TPT** (Yoon et al. 2024) observes that prompting the dispersion of text features can serve as a guiding regularizer for test-time optimization.

Implementation Details

We built the model using the open-source CLIP, employing ViT-B/16 as the backbone for the visual and textual encoder. These encoders are initialized with CLIP's pretrained weights and kept frozen during testing, ensuring they remain unseen. We used the CMA-ES algorithm to optimize the text and vision prompts, setting the batch-size to 32. For clarity, Table 1 provides the default configuration of hyperparameters used in our experiments. We implemented all TPT baseline methods using their publicly available code[2].

[2]TPT: https://azshue.github.io/TPT/;
DiffTPT: https://github.com/chunmeifeng/DiffTPT;
C-TPT: https://github.com/hee-suk-yoon/C-TPT.

Method	ImageNet	ImageNet-A	ImageNet-V2	ImageNet-R	ImageNet-Sketch	Average	OOD Average
CLIP-ViT-B/16	66.73	47.87	60.86	73.98	46.09	59.11	57.20
CoOp	**71.51**	49.71	64.20	75.21	47.99	61.72	59.28
CoCoOp	71.02	50.63	64.07	76.18	48.75	62.13	59.91
TPT	68.98	54.77	63.45	77.06	47.94	62.44	60.81
DiffTPT	70.30	**55.68**	65.10	75.00	46.80	62.28	60.52
C-TPT	69.30	52.90	63.40	78.00	48.50	62.42	60.70
B^2TPT	69.57	55.26	**65.40**	**78.64**	**49.53**	**63.68**	**62.21**

Table 2: Accuracy (%) on OOD benchmark. CoOp and CoCoOp are fine-tuned on ImageNet using 16-shot training data for each category. TPT, DiffTPT and C-TPT are test-time prompt tuning methods. All the compared methods are built upon CLIP-ViT-B/16 baselines. The two evaluation metrics, "Average" and "OOD Average", are calculated by taking the mean accuracy across all five datasets and the four OOD datasets , excluding ImageNet. **Bold** represents the best result and underlined represents the second highest result.

Method	Flow	DTD	Pets	Cars	UCF	Cal	Food	SUN	Air	Euro	Avg.
CLIP-ViT-B/16	67.44	44.27	88.25	65.48	65.13	93.35	83.65	62.59	23.67	42.01	63.58
CoOp	68.71	41.92	89.14	64.51	66.55	93.70	85.30	64.14	18.47	46.39	63.88
CoCoOp	70.85	45.45	90.46	64.90	68.44	93.79	83.97	66.89	22.29	39.23	64.63
TPT	68.98	47.75	87.79	66.87	68.04	94.16	84.67	65.50	24.78	42.44	65.10
DiffiTPT	70.10	47.00	88.22	67.10	68.22	92.49	87.23	65.74	25.60	43.13	65.47
C-TPT	69.90	46.80	87.40	66.70	66.70	94.10	84.50	66.00	23.90	**48.70**	65.47
B^2TPT	**74.83**	**51.18**	**92.69**	**68.99**	**73.21**	**96.96**	**91.32**	**70.59**	**32.13**	46.80	**69.87**

Table 3: Accuracy (%) on Cross-dataset benchmark.CoOp and CoCoOp are fine-tuned on ImageNet using 16-shot training data for each category. TPT, DiffTPT and C-TPT are test-time prompt tuning methods. All the compared methods are built upon CLIP-ViT-B/16 baselines. The evaluation metric "Average" is determined by calculating the mean accuracy across all ten datasets. **Bold** represents the best result and underlined represents the second highest result.

For train-time baseline methods CoOp and CoCoOp, we directly adopted the results reported in their original papers. All experiments were conducted in PyTorch using NVIDIA GeForce RTX 4090 GPUs.

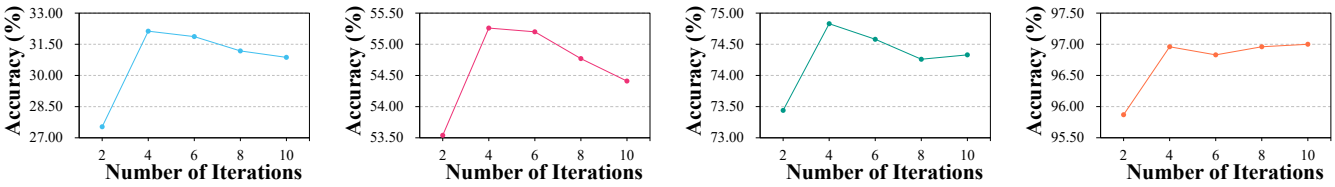
Performance Comparison

Natural Distribution Shifts. Table 2 presents the accuracy comparison of different methods on ImageNet and its four variants. In both overall and OOD-specific evaluations, we achieved the highest performance. B^2TPT maintains an advantage of over 4.5% compared to CLIP. Besides, B^2TPT outperforms other state-of-the-art TPT methods. The method's robustness to distribution shifts is better demonstrated on the OOD average, where B^2TPT achieves an improvement of more than 1.2% over the other TPT methods. For instance, ImageNet-A is particularly challenging as it consists of images misclassified by ResNet. Compared to DiffTPT, which utilizes a diffusion model to generate diverse augmented views, B^2TPT lags behind by only 0.4%. Similarly, for the variant with the most samples, i.e., ImageNet-Sketch, B^2TPT outperforms DiffTPT by approximately 3.0%.

Cross-Datasets Generalization. In Table 3, we compare B^2TPT with few-shot prompt tuning methods and typical TPT methods on their ability to generate from ImageNet to fine-grained datasets. On average, our B^2TPT achieves state-of-the-art performance. Specifically, we also achieve optimal performance on every dataset except EuroSAT. Among the numerous cross-domain datasets, Aircraft is particularly challenging due to the similar appearance of many plane types, which results in extremely low classification accuracy. Nevertheless, B^2TPT achieves an accuracy over 30% on Aircraft, which is not accomplished by any other TPT method. Furthermore, this demonstrates that our method can distinguish minor appearance differences and thereby reduce the classification error rate. Similarly, there are significant improvements such as DTD, UCF101, Food101 and SUN397, which have been elevated to a new level of accuracy.

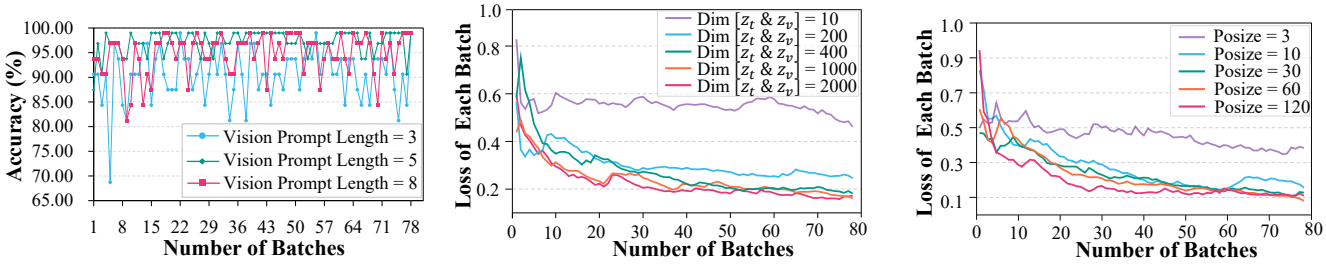
Ablation Study

To assess the contribution of key components in our B^2TPT method, we conducted ablation studies comparing the performance of CODA and its variants on **Aircraft**, **ImageNet-A**, **Flower102** and **Caltech101**. Table 4 presents the average accuracy achieved by different variants and summarizes the differences in their settings, where "Backbone" refers to directly utilizing zero-shot inference during testing, "CLIP" denotes a pseudo-labeling strategy based on zero-shot inference, and "CoC" represents our proposed pseudo-labeling strategy. Even without CoC, significant improve-



(a) Aircraft (b) ImageNet-A (c) Flower102 (d) Caltech101

Figure 2: Accuracy curves for different numbers of iterations on different datasets.



(a) Vision Prompt Length (b) Intrinsic Dimension (c) Population size

Figure 3: Results of ablation experiments for hyperparameters. (a) presents the accuracy curves corresponding to different vision prompt lengths, (b) illustrates the loss curves for various intrinsic dimensions, and (c) analyzes the loss curves based on different population sizes in the black-box optimization algorithm.

Backbone	CLIP	CoC	Air	Image-A	Flow	Cal	Avg.
✔	✘	✘	23.67	47.87	67.44	93.35	58.08
✔	✔	✘	30.89	54.48	73.77	96.27	63.85
✔	✘	✔	**32.13**	**55.26**	**74.83**	**96.96**	**64.80**

Table 4: Ablation study by comparing the performance of B^{2}TPT and its variants.

ments have been achieved by relying solely on CLIP for pseudo-labeling, demonstrating the feasibility of black-box optimization in B^{2}TPT. With the addition of CoC, it is evident that the accuracy of the model is further improved.

Hyperparameter Analysis

Each batch of data can be iteratively optimized to achieve the best possible results. As shown in Figure 2, for both cross-domain and generalization settings, we selected a total of four datasets to cover the accuracy range from 20% to 100%.

Inspired by previous studies (Sun et al. 2022; Yu et al. 2023; Zhou et al. 2022b) on text prompt tuning, the length of text prompts L_t was fixed at 8. Figure 3(a) illustrates the variation in the length of vision prompts, displaying the accuracy rates for each data batch. The highest accuracy was observed when the vision prompt length L_v was set to 5, both for individual batches and on average.

The parameter subspace of the intrinsic prompts represents the actual optimization space. As shown in Figure 3(b), higher dimensions lead to faster convergence but also involve more parameters. When the dimensionality d is set to 400, further increases only accelerate convergence without providing significant performance gains. Therefore, we chose an intrinsic dimension of 400 for our experiments.

In each generation of the evolutionary strategy, all individuals were sampled from a multivariate normal distribution. Figure 3(c) presents the loss curves for different population sizes λ. Smaller populations tend to converge faster, as fewer individuals are involved, but this often results in poorer performance due to limited diversity. In contrast, larger populations converge more slowly, allowing for better exploration of the solution space, which generally leads to better outcomes. A population size of 30 appears to strike a balance, providing more reliable performance.

Conclusion

In this paper, we address the challenge of gradient-free test-time prompt tuning and propose a method called Black-Box Test-time Prompt Tuning (B^{2}TPT). The primary distinction of B^{2}TPT from previous TPT methods is that it optimizes prompts without using gradients. Specifically, text and vision prompts are optimized jointly by generating low-dimensional intrinsic prompts within the parameter subspace and projecting them into the full parameter prompt space. During the tuning process, the CMA-ES algorithm is employed to generate the intrinsic prompts. Additionally, the proposed consistent and confident pseudo-labeling strategy enhances accuracy. Extensive experiments on two benchmark datasets demonstrate that B^{2}TPT achieves comparable or superior performance relative to state-of-the-art TPT methods.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 62077033, by the Shandong Provincial Natural Science Foundation under Grant ZR2020KF015, and by the Taishan Scholar Program of Shandong Province under Grant tsqn202211199.

References

Abdul Samadh, J.; Gani, M. H.; Hussein, N.; Khattak, M. U.; Naseer, M. M.; Shahbaz Khan, F.; and Khan, S. H. 2024. Align your prompts: Test-time prompting with distribution alignment for zero-shot generalization. *Advances in Neural Information Processing Systems*, 36.

Aghajanyan, A.; Gupta, S.; and Zettlemoyer, L. 2021. Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 7319–7328.

Ben-David, E.; Oved, N.; and Reichart, R. 2022. PADA: Example-based Prompt Learning for on-the-fly Adaptation to Unseen Domains. *Transactions of the Association for Computational Linguistics*, 10: 414–433.

Bertsimas, D.; and Tsitsiklis, J. 1993. Simulated annealing. *Statistical science*, 8(1): 10–15.

Bossard, L.; Guillaumin, M.; and Van Gool, L. 2014. 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*, 446–461. Springer.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; and Vedaldi, A. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3606–3613.

Cui, C.; Zhang, C.; Liu, Z.; Zhu, L.; Gong, S.; Lin, X.; et al. 2023. Adversarial Source Generation for Source-Free Domain Adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, 34(6): 4887–4898.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2004. 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*, 178–178. IEEE.

Feng, C.-M.; Yu, K.; Liu, Y.; Khan, S.; and Zuo, W. 2023. Diverse data augmentation with diffusions for effective test-time prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2704–2714.

Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; and Qiao, Y. 2024. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2): 581–595.

Grandvalet, Y.; and Bengio, Y. 2004. Semi-supervised learning by entropy minimization. *Advances in Neural Information Processing Systems*, 17: 529–536.

Guo, Z.; Wei, Y.; Liu, M.; Ji, Z.; Bai, J.; Guo, Y.; and Zuo, W. 2023. Black-box tuning of vision-language models with effective gradient approximation. *arXiv preprint arXiv:2312.15901*.

Hansen, N.; and Ostermeier, A. 2001. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2): 159–195.

Helber, P.; Bischke, B.; Dengel, A.; and Borth, D. 2019. 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.

Hendrycks, D.; Basart, S.; Mu, N.; Kadavath, S.; Wang, F.; Dorundo, E.; Desai, R.; Zhu, T.; Parajuli, S.; Guo, M.; et al. 2021a. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, 8340–8349.

Hendrycks, D.; Zhao, K.; Basart, S.; Steinhardt, J.; and Song, D. 2021b. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 15262–15271.

Karmanov, A.; Guan, D.; Lu, S.; El Saddik, A.; and Xing, E. 2024. Efficient Test-Time Adaptation of Vision-Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14162–14171.

Kennedy, J.; and Eberhart, R. 1995. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, 1942–1948. ieee.

Khandelwal, A. 2024. PromptSync: Bridging Domain Gaps in Vision-Language Models through Class-Aware Prototype Alignment and Discrimination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7819–7828.

Khattak, M. U.; Rasheed, H.; Maaz, M.; Khan, S.; and Khan, F. S. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19113–19122.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, 554–561.

Li, X.; Lian, D.; Lu, Z.; Bai, J.; Chen, Z.; and Wang, X. 2024. Graphadapter: Tuning vision-language models with dual knowledge graph. *Advances in Neural Information Processing Systems*, 36.

Li, X. L.; and Liang, P. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 4582–4597.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ma, X.; Zhang, J.; Guo, S.; and Xu, W. 2024. Swapprompt: Test-time prompt adaptation for vision-language models. *Advances in Neural Information Processing Systems*, 36.

Maji, S.; Rahtu, E.; Kannala, J.; Blaschko, M.; and Vedaldi, A. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.

Nilsback, M.-E.; and Zisserman, A. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, 722–729. IEEE.

Oh, C.; Hwang, H.; Lee, H.-y.; Lim, Y.; Jung, G.; Jung, J.; Choi, H.; and Song, K. 2023. Blackvip: Black-box visual prompting for robust transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24224–24235.

Parkhi, O. M.; Vedaldi, A.; Zisserman, A.; and Jawahar, C. 2012. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, 3498–3505. IEEE.

Qin, Y.; Wang, X.; Su, Y.; Lin, Y.; Ding, N.; Yi, J.; Chen, W.; Liu, Z.; Li, J.; Hou, L.; et al. 2021. Exploring universal intrinsic task subspace via prompt tuning. *arXiv preprint arXiv:2110.07867*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Recht, B.; Roelofs, R.; Schmidt, L.; and Shankar, V. 2019. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, 5389–5400. PMLR.

Shu, M.; Nie, W.; Huang, D.-A.; Yu, Z.; Goldstein, T.; Anandkumar, A.; and Xiao, C. 2022. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35: 14274–14289.

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Sui, E.; Wang, X.; and Yeung-Levy, S. 2024. Just Shift It: Test-Time Prototype Shifting for Zero-Shot Generalization with Vision-Language Models. *arXiv preprint arXiv:2403.12952*.

Sun, T.; Shao, Y.; Qian, H.; Huang, X.; and Qiu, X. 2022. Black-box tuning for language-model-as-a-service. In *International Conference on Machine Learning*, 20841–20855. PMLR.

Tsai, Y.-Y.; Mao, C.; and Yang, J. 2024. Convolutional visual prompt for robust visual perception. *Advances in Neural Information Processing Systems*, 36.

Wang, D.; Shelhamer, E.; Liu, S.; Olshausen, B.; and Darrell, T. 2021. Tent: Fully Test-Time Adaptation by Entropy Minimization. In *International Conference on Learning Representations*.

Wang, H.; Ge, S.; Lipton, Z.; and Xing, E. P. 2019. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32.

Wang, Z.; Liang, J.; He, R.; Wang, Z.; and Tan, T. 2024. Connecting the Dots: Collaborative Fine-tuning for Black-Box Vision-Language Models. *arXiv preprint arXiv:2402.04050*.

Xiao, J.; Hays, J.; Ehinger, K. A.; Oliva, A.; and Torralba, A. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, 3485–3492. IEEE.

Yoon, H. S.; Yoon, E.; Tee, J. T. J.; Hasegawa-Johnson, M.; Li, Y.; and Yoo, C. D. 2024. C-TPT: Calibrated Test-Time Prompt Tuning for Vision-Language Models via Text Feature Dispersion. *arXiv preprint arXiv:2403.14119*.

Yu, L.; Chen, Q.; Lin, J.; and He, L. 2023. Black-box Prompt Tuning for Vision-Language Model as a Service. In *IJCAI*, 1686–1694.

Zhang, D.-C.; Zhou, Z.; and Li, Y.-F. 2024. Robust Test-Time Adaptation for Zero-Shot Prompt Tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 16714–16722.

Zhang, R.; Zhang, W.; Fang, R.; Gao, P.; Li, K.; Dai, J.; Qiao, Y.; and Li, H. 2022. Tip-adapter: Training-free adaption of clip for few-shot classification. In *European conference on computer vision*, 493–510. Springer.

Zhang, W.; Shen, L.; and Foo, C.-S. 2023. Rethinking the role of pre-trained networks in source-free domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 18841–18851.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16816–16825.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9): 2337–2348.