FATE: Feature-Adapted Parameter Tuning for Vision-Language Models

Zhengqin Xu, Zelin Peng[*], Xiaokang Yang[†], Wei Shen

MoE Key Lab of Artifcial Intelligence, AI Institute, Shanghai Jiao Tong University
{fate311, zelin.peng, xkyang, wei.shen}@sjtu.edu.cn

Abstract

Following the recent popularity of vision language models, several attempts, e.g., parameter-efficient fine-tuning (PEFT), have been made to extend them to different downstream tasks. Previous PEFT works motivate their methods from the view of introducing new parameters for adaptation but still need to learn this part of weight from scratch, i.e., random initialization. In this paper, we present a novel strategy that incorporates the potential of prompts, e.g., vision features, to facilitate the initial parameter space adapting to new scenarios. We introduce a **F**eature-**A**dapted parame**T**er **E**fficient tuning paradigm for vision-language models, dubbed as **FATE**, which injects informative features from the vision encoder into language encoder's parameters space. Specifically, we extract vision features from the last layer of CLIP's vision encoder and, after projection, treat them as parameters for fine-tuning each layer of CLIP's language encoder. By adjusting these feature-adapted parameters, we can directly enable communication between the vision and language branches, facilitating CLIP's adaptation to different scenarios. Experimental results show that FATE exhibits superior generalization performance on 11 datasets with a very small amount of extra parameters and computation.

Introduction

In recent years, we have witnessed impressive developments in vision-language modelling. Many foundation models (Dou et al. 2022b; Radford et al. 2021; Li et al. 2022a; Kim, Son, and Kim 2021; Li et al. 2021; Kamath et al. 2021; Dou et al. 2022a; Li et al. 2022b), e.g., CLIP (Radford et al. 2021), have facilitated many vision tasks: instead of training a model from scratch for each individual task, one can simply apply these pre-trained vision-language models (VLMs) and then (partially) fine-tuning them on various downstream tasks to achieve promising performance.

Existing algorithms often perform fine-tuning in an efficient style, which mainly encompasses two techniques: (i) Prompt learning (Rao et al. 2022; Tsimpoukelli et al. 2021; Bulat and Tzimiropoulos 2023; Yao, Zhang, and Xu 2023a; Zhou et al. 2022b,a). They often attempt to incorporate tunable or task-orientated prompts, e.g., vision features (Zhou et al. 2022a), with handcrafted input prompts of a model, targeted adjust the initial prompt space to satisfy downstream scenarios (see Fig. 1 (a)). (ii) Parameter-efficient fine-tuning (PEFT) (Chen et al. 2022a,b; Gao et al. 2024; Hu et al. 2022; Mou et al. 2024). As shown in Fig. 1 (b), their common idea is to learn a small set of parameters from scratch to help the initial parameter space adapt to new scenarios. Considering features can be treated as a type of pre-trained parameter incorporated into a prompt space tailored for new scenarios, we raise a question by the light of nature: Is there any way to take advantage of these informative features for the initial parameter space to facilitate its adaptation?

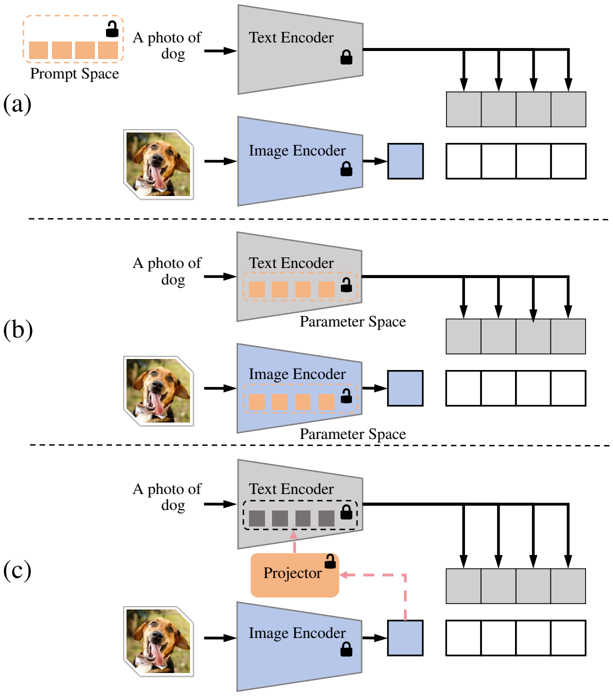

Figure 1: **Comparative Overview: Previous Methods vs. FATE**. Previous method using (a) Prompt Learning, (b) Parameter-Efficient Fine-Tuning. (c) our proposed FATE.

In this paper, to offer a solution to this question, we

[*]Project Leader

[†]Corresponding Author: Xiaokang Yang

propose a simple yet effective paradigm for CLIP, called **F**eature-**A**dapted parame**T**er-**E**fficient tuning (**FATE**). In FATE, we exploit visual features from the image encoder of CLIP and then incorporate them into the parameter space of CLIP's language encoder, as illustrated at the bottom of Fig. 1. Specifically, we extract vision features from the last layer of the vision encoder and generate new parameters for each layer of the language encoder via a projection head. By fine-tuning these feature-adapted parameters, we can effectively integrate visual information into the parameter space of the language model. Consequently, our proposed FATE efficiently and effectively adapts CLIP to different few-shot generalization scenarios with a very small amount of extra parameters and computation.

Comprehensive experiments on 11 datasets covering a diverse set of visual recognition tasks demonstrate that FATE shows leading performance. Additionally, FATE demonstrates remarkable acceleration compared with the current prompt engineering and PEFT methods.

Related Work

Vision-Language Models

Recent advancements in Vision-Language Models (VLMs), e.g., CLIP (Radford et al. 2021), ALIGN (Jia et al. 2021), LiT (Zhai et al. 2022), and Kosmos (Huang et al. 2024), have engaged considerable interest from industrial and research community. The core idea of these models is to leverage abundantly available data for learning joint image-language representations. For example, CLIP (Radford et al. 2021) and ALIGN (Jia et al. 2021) train a multi-modal network by using approximately 400M and 1B image-text pairs, respectively. Despite the pre-trained VLMs can learn generalized representations, directly adapting them to satisfy various downstream tasks is a significant challenge. Numerous works have been proposed to adapt VLMs for different tasks, e.g., few-shot image classification (Gao et al. 2024; Zhou et al. 2022c), object detection (Zang et al. 2022), and segmentation (Peng et al. 2024a; He et al. 2023). In this work, we propose a novel paradigm for CLIP to effectively facilitate its adaptation to different downstream tasks.

Prompt Learning

It is acknowledged that the concept of prompt learning is proposed by (Lester, Al-Rfou, and Constant 2021) for natural language process. Lately, (Wang et al. 2022) introduces prompt learning for image recognition. As a pioneer, Wang et al (Petroni et al. 2019) propose that the pre-trained language models can be treated as knowledge databases. In light of this, the following works (Shin et al. 2020; Zhou et al. 2022b) argue that a prompt can be manually designed for a downstream task or automatically learned during fine-tuning. For example, LPAQA (Jiang et al. 2020) chooses the optimal prompt from a group of candidate prompts that generated by using text mining and paraphrasing. Autoprompt (Shin et al. 2020) selects the best token that causes the great gradient changes from a vocabulary. CoOp (Zhou et al. 2022b) uses learned continuous vectors as a prompt. CoCoOp (Zhou et al. 2022a), as a variation of CoOp, generates an input-conditional token as a prompt for each image to enhance generalization. Maple (Khattak et al. 2023) intensifies the collaboration by mapping text prompts to image prompts. These approaches, however, predominantly focus on a prompt space. Differently, our FATE directly incorporates the visual features into the parameter space of CLIP's text encoder, offering a simple yet effective solution to facilitate its adaptation.

Parameter Efficient Fine-Tuning

Parameter Efficient Fine-Tuning (PEFT) methods (Hu et al. 2022; Chen et al. 2022a) can efficiently adapt pre-trained large-scale models to various downstream tasks, which has already been widely studied in the field of computer vision (Peng et al. 2024c,b) and multi-modal (Peng et al. 2024a). These PEFT methods mainly include the reparameterization-based methods (e.g. LoRA (Hu et al. 2022)) and addition-based methods (e.g. Adapterformer (Chen et al. 2022a)). For the reparameterization-based PEFT methods, they retrain a small fraction of pre-trained parameters (Hu et al. 2022) or reconstruct the whole parameter space (Peng et al. 2024c) of the pre-trained model. For the addition-based method, they add a learnable lightweight network in parallel with a pre-trained model, e.g., Clip-Adapter (Gao et al. 2024) and Tip-Adapter (Zhang et al. 2022), which add a trainable small network connected with CLIP's image encoder. Besides, cross-modal adapter (Jiang et al. 2022) inserts a single adapter after every self-attention and feed-forward MLP modules for text-video retrieval, respectively. Nevertheless, these existing PEFT methods learn new parameters from scratch for diverse downstream tasks, ignoring the model's inherent feature information which can be treated as factual knowledge. In this paper, our FATE utilizes a novel paradigm for CLIP, that incorporates visual features as a correlation knowledge into text encoder parameter space with minimal additional parameters and computation.

Preliminaries

Vision Language Model

Since we build upon our method on CLIP (Radford et al. 2021), one of the representative vision language models, we briefly introduce it. CLIP consists of a text encoder $\mathcal{T}$ and an image encoder $\mathcal{V}$. During pre-training on large-scale image-text pairs, CLIP is required to encode images and text descriptions simultaneously to learn an aligned embedding space for visual and language modalities by contrastive learning. Consequently, pre-trained CLIP shows remarkable performance on zero-shot transferability for various downstream tasks.

More specifically, denote $\mathbf{F}_V$ is the image feature generated from an image $\mathbf{I}$ by CLIP's image encoder $\mathcal{V}$, which includes K transformer layers $\mathcal{V}_{i=1}^{K}$. Formally, $\mathbf{F}_V$ can be

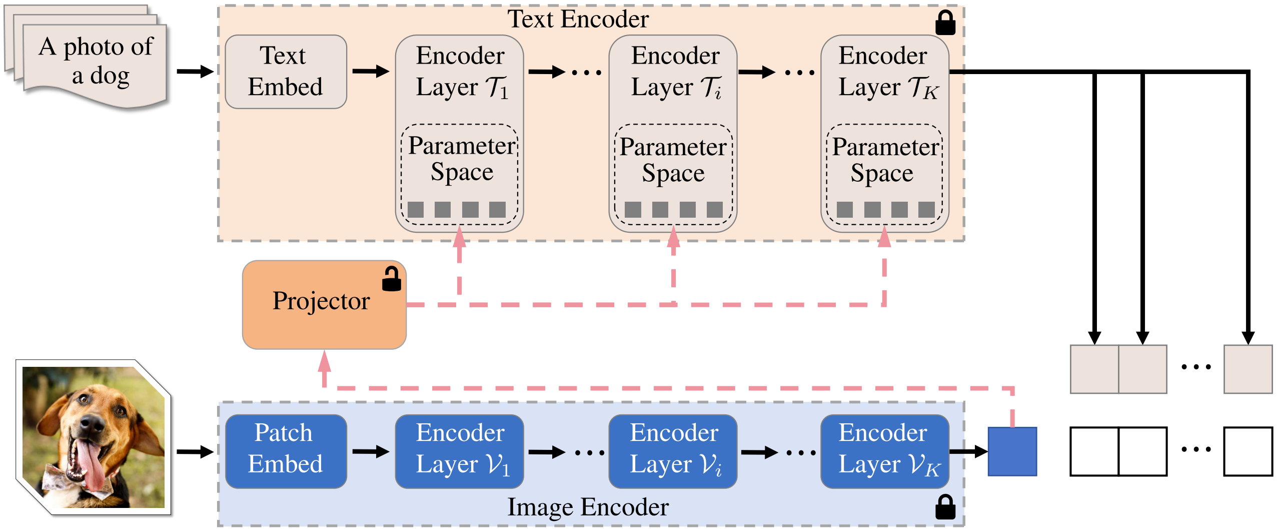
A photo of a dog
Text Encoder
Text Embed
Encoder Layer $\mathcal{T}_1$
Encoder Layer $\mathcal{T}_i$
Encoder Layer $\mathcal{T}_K$
Parameter Space
Parameter Space
Parameter Space
Projector
Patch Embed
Encoder Layer $\mathcal{V}_1$
Encoder Layer $\mathcal{V}_i$
Encoder Layer $\mathcal{V}_K$
Image Encoder

Figure 2: Overview of the proposed FATE framework for fine-tuning in CLIP. Our FATE tunes both image and text encoders where only the projector and position embedding are learned, while the whole pre-trained CLIP models are frozen. FATE integrates the visual information into language modal parameter space to facilitate the model's adaptation with a very small amount of extra parameters and computation.

obtained as follow:

$$\begin{aligned} \mathbf{F}_{V,0} &= \textit{PatchEmbed}(\mathbf{I}) && (1)\\ [\mathbf{c}_i, \mathbf{F}_{V,i}] &= \mathcal{V}_i([\mathbf{c}_{i-1}, \mathbf{F}_{V,i-1}]), i = 1, 2, \cdots, K && (2)\\ \mathbf{F}_V &= \textit{PatchProj}(\mathbf{c}_K), && (3) \end{aligned}$$

where $\textit{PatchEmbed}(\cdot)$ splits the image $\mathbf{I}$ into fixed-size patches and projects these patches into feature embeddings. These features are concatenated with a learnable class token embeddings $\mathbf{c}_1$ are input to K transformer layers $\mathcal{V}_{i=1}^K$. The class token $\mathbf{c}_K$ is protected to obtain the final image feature $\mathbf{F}_V$ via $\textit{PatchProj}(\cdot)$. Similarly, the corresponding text feature $\mathbf{F}_T$ is first generated from a text description $\mathbf{T}$ and then feeding into CLIP's text encoder $\mathcal{T}$ with K transformer layers $\mathcal{T}_{i=1}^K$ as follow:

$$\begin{aligned} [\mathbf{F}_{T,0}^j]_{j-1}^N &= \textit{TextEmbed}(\mathbf{T}) && (4)\\ [\mathbf{F}_{T,i}^j]_{j-1}^N &= \mathcal{T}_i(\mathbf{F}_{T,i-1}), i = 1, 2, \cdots, K && (5)\\ \mathbf{F}_T &= \textit{TextProj}(\mathbf{F}_{T,K}^N), && (6) \end{aligned}$$

where $\textit{TextEmbed}(\cdot)$ tokenize and project the text description $\mathbf{T}$ into N word embeddings, which are then input to K transformer layers $\mathcal{T}_{i=1}^K$. $\textit{TextProj}(\cdot)$ projects the token $\mathbf{F}_{T,K}^N$ to obtain the final text feature $\mathbf{F}_T$. The text feature $\mathbf{F}_T$ and text feature $\mathbf{F}_T$ lie in the common vision-language latent space. During training, CLIP maximizes the cosine similarity scores $\text{sim}(\mathbf{F}_V, \mathbf{F}_T)$ between the image with matched text description, and minimizes the cosine similarity between the image with unmatched text description. CLIP can perform task-specific predictions in different scenarios via computing the cosine similarity scores. For zero-shot classification, the class labels $y = 1, 2, \cdots, C$ are designed as text prompt, and the prediction $\hat{y}$ can be computed as:

$$p(\hat{y} \| \mathbf{F}_V) = \frac{\exp(\text{sim}(\mathbf{F}_V, \mathbf{F}_T^y)/\tau)}{\sum_{i=1}^C \exp(\text{sim}(\mathbf{F}_V, \mathbf{F}_T^i)/\tau)}, \quad (7)$$

where τ is a temperature parameter.

Methodology

In this section, we present the whole framework of our method in detail. After that, we provide a detailed analysis of its complexity.

Feature-Adapted Parameter-Efficient Tuning

This section gives a new perspective on fine-tuning CLIP for various downstream tasks, i.e., feature-adapted parameter tuning (FATE). Considering that the parameter space of pre-trained VLMs stores abundant knowledge for creating distinguish features for new scenarios (Han et al. 2021), rather than training new parameters from scratch, i.e., random initialization, we leverage the distinguish features and project them into parameter space for adapting CLIP, which is simple yet effective. The core of the proposed FATE lies in fine-tuning a vision encoder projector, aiming to project existing visual information into CLIP's language encoder's parameter spaces tailored to new scenarios. Fig. 2 provides a visual overview of our proposed FATE.

Specifically, for a text encoder's pre-trained weight $\mathbf{W}_0 \in \mathbb{R}^{d \times k}$, we constrain its update by representing the latter with visual features $\mathbf{F_V} \in \mathbb{R}^{n' \times d'}$: $\mathbf{W}_0 + \mathbf{W}' = \mathbf{W}_0 + \alpha \mathcal{F}(\mathbf{F_V})$, where $\mathcal{F}(\cdot)$ is a projector contained two fully-connected layers, $\mathcal{F}(\mathbf{F_V}) \in \mathbb{R}^{d \times k}$, α is a scaling factor. For $\mathbf{h} = \mathbf{W}_0 \mathbf{x}$, where $\mathbf{x} \in \mathbb{R}^{n \times d}$ is the textual feature, the forward pass of our proposed FATE can be written as:

$$\mathbf{h} = \mathbf{W}_0 \mathbf{x} + \mathbf{W}' \mathbf{x} = \mathbf{W}_0 \mathbf{x} + \alpha \mathcal{F}(\mathbf{F_V}) \mathbf{x}. \quad (8)$$

As for the projector $\mathcal{F}(\cdot)$, it should realize one key function that aligns the dimension between visual feature $\mathbf{F_V} \in \mathbb{R}^{n' \times d'}$ and the text encoder's pre-trained weight $\mathbf{W}_0 \in \mathbb{R}^{d \times k}$. We use a non-linear activation (RELU$(\cdot)$) and two FC layers f_1 and f_2 to project the visual feature to the corresponding dimension of the text encoder's pre-trained weight as a visual prompt, respectively:

$$\mathcal{F}(\mathbf{F_V}) = f_1(\mathrm{RELU}(f_2(\mathbf{F_V}))). \tag{9}$$

In principle, we can apply the visual prompt to any subset of weight matrices in the CLIP's text encoder to inject the visual information into the language modal parameter space. In the text encoder, there are four weight matrices in the self-attention module ($\mathbf{W}_q$, $\mathbf{W}_k$, $\mathbf{W}_v$, $\mathbf{W}_o$) and two in the MLP module. We limit our study to only injecting the visual information into some of the textual parameter matrices for downstream tasks and freeze other parameter matrices without injecting both for simplicity and efficiency. Besides, we further study the effect of injecting the visual information into different types of weight matrices in the CLIP's text encoder via ablation study.

Complexity Analysis

This section studies the complexity of our method in terms of floating point operations (FLOPs). For simplicity, we assume that a layer of CLIP's text encoder only contains a multi-head self-attention (MHSA) and an FFN block without taking other modules into account. For a token sequence that the length is L and the dimension of each token is d, the FLOPs of a single text encoder layer are calculated as:

$$\begin{aligned} \mathrm{FLOPs}_{\textit{text}} =& \mathrm{FLOPs}_{\textit{MHSA}} + \mathrm{FLOPs}_{\textit{FFN}} \\ =& (8Ld^2 + 4L^2d) + (16Ld^2) \\ =& 4Ld(6d + L). \end{aligned} \tag{10}$$

For the prompt learning method which uses a vision prompt whose length is n' and dimension is d', the length of the input tokens is $L + n'$ and the FLOPs of a single text encoder layer become:

$$\mathrm{FLOPs}_{\textit{Prompt}} = 4d(L + n')(6d + L + n'), \tag{11}$$

and the addition FLOPs of K layers text encoder are:

$$\mathrm{FLOPs}_{\textit{Prompt}} - \mathrm{FLOPs}_{\textit{text}} = Kn'd(6d + n' + 2L). \tag{12}$$

For the PEFT method, the adapter is composed of two FC layers in which the dimension of latent space is r. The additional FLOPs of the whole text encoder are:

$$\mathrm{FLOPs}_{\textit{add}} - \mathrm{FLOPs}_{\textit{text}} = 4Ldr. \tag{13}$$

Whereas for our FATE, assume that the size of parameter space injected visual information is $d \times k$, the additional FLOPs are $n'd(k + d')$. Compared with the prompt learning method, the additional FLOPs of the PEFT method and our FATE are negligible. Therefore, FATE is a computation-efficient paradigm to adapt CLIP to different scenarios.

Experiments

In all experiments, we follow prior works (Khattak et al. 2023; Zhou et al. 2022a) to evaluate the performance of FATE across three different scenarios, including Base-to-Novel Generalization, Cross-dataset Evaluation, and Domain Generalization. We begin by introducing the benchmark setting and implementation details. Following this, we benchmark FATE against various prevailing approaches. Finally, we delve into an ablation study of our FATE.

Benchmark Setting

Base-to-Novel Generalization. We evaluate the generalizability of our proposed FATE on 11 image classification datasets, including 2 general object recognition datasets: ImageNet (Deng et al. 2009) and Caltech101 (Fei-Fei, Fergus, and Perona 2004); 5 fine-grained image recognition datasets: OxfordPets (Parkhi et al. 2012), StandfordCars (Krause et al. 2013), Flowers102 (Nilsback and Zisserman 2008), Food101 (Bossard, Guillaumin, and Van Gool 2014), and FGVCAircraft (Maji et al. 2013); a scene understanding dataset: SUN397 (Xiao et al. 2010); a texture dataset: DTD (Cimpoi et al. 2014); a satellite-image recognition dataset: EuroSAT (Helber et al. 2019)and an action classification dataset: UCF101 (Soomro, Zamir, and Shah 2012). The model is trained using only the base classes in a few-shot setting while evaluation is conducted on base and novel categories to test generalizability.

Cross-dataset Evaluation. As suggested in CoCoOp (Zhou et al. 2022a), we also use the 11 datasets mentioned above for cross-dataset evaluation, in which all models are trained on ImageNet with 1000 categories (each category having 16 training samples) and directly transfer the model to evaluate on other datasets.

Domain Generalization. Following the suggestion (Zhou et al. 2022a), we employ our proposed FATE pre-trained on the ImageNet (Deng et al. 2009) dataset and subject them to direct testing on four other specific datasets (ImageNetV2 (Recht et al. 2019), ImageNet-Sketch (Wang et al. 2019), ImageNet-A (Hendrycks et al. 2021b), and ImageNet-R (Hendrycks et al. 2021a)) with different data distributions to evaluate the robustness of models on out-of-distribution tasks.

Implementation Details. In line with previous works (Zhou et al. 2022b,a; Khattak et al. 2023), we use a few-shot setting that randomly samples 16 shots for each class in all experiments. The pre-trained ViT-B/16 CLIP model is used throughout the experiments. We train FATE for 10 epochs with a batch size of 10 and an initial learning rate of 0.002 via an SGD solver. All models are trained with a cosine learning rate schedule on a single NVIDIA 3090 GPU. To maintain robust results, we report the results of Base and Novel class accuracy, and their harmonic mean (HM) averaged over three times with different seeds.

Main Results

Base-to-Novel Generalization. Table. 1 presents the performance of our FATE and the result compared to several prevailing approaches in the base-to-novel generalization set-

Methods	Average			ImageNet			Caltech101			OxfordPets		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP (ICML'21)	69.34	74.22	71.70	72.43	68.14	70.22	96.84	94.00	95.40	91.17	97.26	94.12
CoOp (IJCV'22)	82.69	63.22	71.66	76.47	67.88	71.92	98.00	89.81	93.73	93.67	95.29	94.47
CoCoOp (CVPR'22)	80.47	71.69	75.83	75.98	70.43	73.10	97.96	93.81	95.84	95.20	97.69	96.43
KgCoOp (CVPR'23)	80.73	73.60	77.00	75.83	69.96	72.78	97.72	94.39	96.03	94.65	97.76	96.18
MaPLe (CVPR'23)	82.28	75.14	78.55	**76.66**	70.54	73.47	97.74	94.36	96.02	95.43	97.76	96.58
LASP (CVPR'23)	**82.70**	74.90	78.61	76.20	70.95	73.48	98.10	94.24	96.16	**95.90**	97.93	**96.90**
RPO (ICCV'23)	81.13	75.00	77.78	76.60	71.57	74.00	97.97	94.37	96.03	94.63	97.50	96.05
FATE	82.11	**76.29**	**79.09**	76.40	**71.81**	**74.03**	98.17	**95.13**	**96.63**	95.12	**98.04**	96.56

Methods	StanfordCars			Flowers102			Food101			FGVCAircraft		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP (ICML'21)	63.37	74.89	68.65	72.08	**77.80**	74.83	90.10	91.22	90.66	27.19	36.29	31.09
CoOp (IJCV'22)	**78.12**	60.40	68.13	97.60	59.67	74.06	88.33	82.26	85.19	**40.44**	22.30	28.75
CoCoOp (CVPR'22)	70.49	73.59	72.01	94.87	71.75	81.71	90.70	91.29	90.99	33.41	23.71	27.74
KgCoOp (CVPR'23)	71.76	75.04	73.36	95.00	74.73	83.65	90.50	91.70	91.09	36.21	33.55	34.83
MaPLe (CVPR'23)	72.94	74.00	73.47	95.92	72.46	82.56	90.71	**92.05**	91.38	37.44	35.61	36.50
LASP (CVPR'23)	75.17	71.60	73.34	97.00	74.00	83.95	**91.20**	91.70	**91.44**	34.53	30.57	32.43
RPO (ICCV'23)	73.87	75.53	**74.69**	94.13	76.67	84.50	90.33	90.83	90.58	37.33	34.20	35.70
FATE	71.98	**75.82**	73.85	95.98	75.73	**84.66**	90.91	91.54	91.22	38.69	**35.72**	**37.15**

Methods	SUN397			DTD			EuroSAT			UCF101		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP (ICML'21)	69.36	75.35	72.23	53.24	59.90	56.37	56.48	64.05	60.03	70.53	77.50	73.85
CoOp (IJCV'22)	80.60	65.89	72.51	79.44	41.18	54.24	92.19	54.74	68.69	84.69	56.05	67.46
CoCoOp (CVPR'22)	79.74	76.86	78.27	77.01	56.00	64.85	87.49	60.04	71.21	82.33	73.45	77.64
KgCoOp (CVPR'23)	80.29	76.53	78.36	77.55	54.99	64.35	85.64	64.34	73.48	82.89	76.67	79.65
MaPLe (CVPR'23)	80.82	**78.70**	**79.75**	80.36	59.18	68.16	94.07	73.23	82.35	83.00	78.66	80.77
LASP (CVPR'23)	80.70	78.60	79.63	**81.40**	58.60	68.14	**94.60**	**77.78**	**85.36**	**84.77**	78.03	**81.26**
RPO (ICCV'23)	80.60	77.80	79.18	76.70	**62.13**	68.61	86.63	68.97	76.79	83.67	75.43	79.34
FATE	**80.92**	77.87	79.37	79.24	61.21	**69.07**	92.04	77.67	84.25	83.81	**78.67**	81.16

Table 1: Comparison with existing state-of-the-art methods in the Base-to-Novel Generalization setting. "Base" and "Novel" are the recognition accuracies on base and novel classes respectively. "HM" is the harmonic mean of base and new accuracy, providing the trade-off between adaption and generalization. The proposed FATE demonstrates strong generalization results over existing methods on 11 recognition datasets.

	Source	**Target**									
	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UCF101	Average
CoOp (IJCV'22)	**71.51**	93.70	89.14	64.51	68.71	85.30	18.47	64.15	41.92	46.39	66.55	63.88
CoCoOp (CVPR'22)	71.02	**94.43**	90.14	65.32	71.88	86.06	22.94	**67.36**	45.73	45.37	68.21	65.74
MaPLe (CVPR'23)	70.72	93.53	90.49	65.57	72.23	**86.20**	24.74	67.01	46.49	**48.06**	68.69	66.30
PromptSRC (ICCV'23)	71.27	93.60	90.25	65.70	70.25	86.15	23.90	67.10	46.87	45.50	68.75	65.81
FATE	71.47	94.14	**90.52**	**66.02**	**72.33**	86.09	**25.12**	67.13	**46.99**	47.87	**68.77**	**66.95**

Table 2: Comparison of FATE with state-of-the-art methods in the Cross-Dataset Evaluation setting. Overall, our FATE obtains leading average performance over 10 datasets, demonstrating the good zero-shot transferable ability.

ting on 11 classification datasets. The comparison works include the zero-shot baseline - CLIP (Radford et al. 2021), text-based prompt learning methods - CoOp (Zhou et al. 2022b), CoCoOp (Zhou et al. 2022a), KgCoOp (Yao, Zhang, and Xu 2023b), LASP (Bulat and Tzimiropoulos 2023), and two recently parameter-efficient fine-tuning methods: RPO (Lee et al. 2023) and MaPLe (Khattak et al. 2023). In line with previous (Zhou et al. 2022a; Khattak et al. 2023), we utilize the classification accuracy of the base (Base) and novel (Novel) classes, as well as the trade-off between these two metrics – harmonic mean (HM), as the evaluation metrics.

Obviously, our proposed FATE surpasses the previous works and achieves the best average performance in terms of base and novel accuracy and their harmonic mean (HM) over 11 datasets. LASP (Bulat and Tzimiropoulos 2023), which uses handcrafted textual prompts, maximizes the correct classification probability of the prompts by using a text-to-text loss. In all prevailing approaches, LASP achieves the best trade-off in base-to-novel generalization. The proposed FATE outperforms LASP in that FATE performs better than LASP on the base class and novel class generalization. Meanwhile, compared with LASP used handcrafted textual prompts, FATE is more flexible to be deployed in various scenarios. Next, MaPle (Khattak et al. 2023) as a multimode prompt learning method adds learnable prompts of both text and image encoders and improves the alignment between text and image features via a couple function. FATE performs better than MaPLe in terms of base accuracy, novel accuracy, and their harmonic mean, respectively. The comparative results on all 11 datasets indicate that our FATE as a new paradigm injected visual information into the language modal provides the best trad-off in the Base-to-Novel generalization.

Cross-dataset Evaluation. In the cross-dataset evaluation setting, FATE is trained on all the 1,000 classes of ImageNet and then directly transferred for evaluation on the remaining 10 datasets. All experiments use the same learning rate as in the base-to-novel generalization. Table. 2 presents all comparative results of our FATE with other state-of-the-art methods. On the whole, our FATE outperforms the others and attains the best average accuracy of 66.95. It is noted that FATE surpasses MaPle, which is the second performer, in half of these 10 datasets. In addition, our FATE also obtains competitive performance against CoOP, CoCoOp, and MaPLe on ImageNet, which is the trained source dataset. These results demonstrate that our FATE achieves injecting visual information into the language model and has good zero-shot transferability.

Domain Generalization. In the domain generalization setting, Evaluating models on out-of-distribution tasks is vital, and we pre-train the models on ImageNet datasets and directly test them on specific datasets with diverse data distributions. As shown in Table. 3, we evaluate the model, which trained on the entire 1,000 classes of ImageNet, on 4 variant datasets derived from ImageNet. These results indicate that our FATE outperforms MaPLe and CoCoOp performance on out-of-distribution datasets and demonstrates the robustness of domain shifts in the domain generalization setting.

Methods	ImageNet	-V2	-S	-A	-R
CLIP (ICML'21)	66.73	60.83	46.15	47.77	73.96
CoOp (IJCV'22)	**71.51**	64.20	47.99	49.71	75.21
CoCoOp (CVPR'22)	71.02	64.07	48.75	50.63	76.18
MaPLe (CVPR'23)	70.72	64.07	49.15	50.90	76.98
FATE	71.47	**64.08**	**49.58**	**50.99**	**77.12**

Table 3: Comparison of FATE with state-of-the-art methods in the Domain Generalization setting. Overall, our FATE obtains the best performance in all out-of-distribution datasets, showing good robustness to domain shifts.

	SA. la.	FFN la.	Base	Novel	HM
Baseline †	-	-	53.24	59.90	56.37
Parameter Space	-	√	77.24	56.64	65.36
	√	-	78.98	58.72	67.36
	√	√	**79.24**	**61.21**	**69.07**

Table 4: Ablation on selecting which parameter space of text encoder for injection visual information. "SA. la": Self-Attention layers. "FFN la.": Feed-Forward Network layers. "Baseline †": zero-shot of CLIP.

Ablation Study

In this section, by conducting ablation studies, we determine which parameter sub-space of CLIP's text encoder, i.e., MHSA or FFN, can be injected with visual information to offer better performance. We also conduct an ablation study to demonstrate the effects of visual information injected layer depth on the CLIP's text encoder. Additionally, we conduct an ablation study to validate injecting the visual information into the parameter space or the prompt space of the CLIP's text encoder yields better performance. All results are reported on the DTD dataset.

Select which parameter space for visual information injection? Each layer of CLIP's text encoder consists of an MHSA layer and an FFN layer. Accordingly, we inject the visual information into its parameter space, i.e., the space formed by the parameters from 1) the MHSA layer, 2) the FFN layers, and 3) both of them. As shown in Table. 4, only injecting the visual information into the SA layer parameter space or the FFN layer parameter space leads to obvious performance drops. This indicates that the visual information needs to be injected into the whole parameter space of each layer of CLIP's text encoder to promote visual and textual information fusion. **Notably, we observe that the required trainable parameter size FATE is only 15K while leading to superior performance.**

Inject which depth of text encoder for visual information? We ablate the experiment to illustrate the effect of visual information injecting depth for FATE. As shown in Fig. 3, the general trend indicates that as the visual information injecting depth increases, the performance of the model tends to improve. Overall, our FATE achieves the best per-

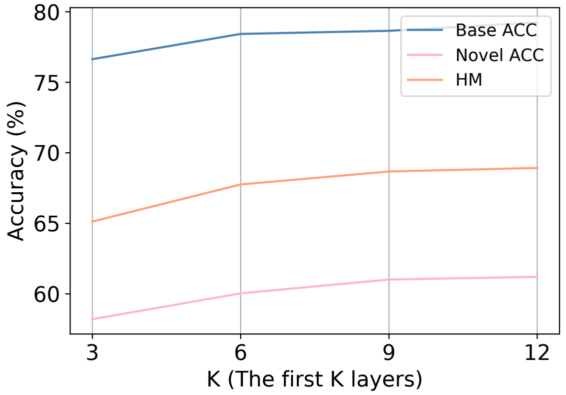

Figure 3: Ablation study on the first K layers for injecting visual information.

	Base	Novel	HM
Prompt space	77.01	56.00	64.85
Parameter space (ours)	**79.24**	**61.21**	**69.07**

Table 5: Ablation on selecting which space to inject visual prompts.

Numbers	Base	Novel	HM
1	79.01	60.12	68.28
2	79.24	**61.21**	**69.07**
3	**79.48**	60.77	68.88

Table 6: Ablation study of numbers of FC layers in the projector $\mathcal{F}(\cdot)$.

formance at a depth of 12. This further demonstrates that the visual information needs to be injected into the whole of CLIP's text encoder parameter space to achieve optimal performance.

Select which one of the parameter space and the prompt space in one signal layer is the injected space of visual information? Table. 5 shows the impact of selecting the parameter space or the prompt space. We observe that the parameter space is a more efficient space to inject visual information. Interestingly, the parameter space is significantly higher than the prompt space in the novel class accuracy. It means that injecting visual information into the parameter space capability contributes to improving the VLMs' generalization. In the language encoder's parameter space of VLMs, our method adeptly fuses the visual information and textual information and efficiently boosts better performance.

Numbers of MLP in the projector $\mathcal{F}(\cdot)$. In order to examine the impact of different numbers of MLP in the projector $\mathcal{F}(\cdot)$, we carry out experiments to study it. As presented in Table. 6, the result shows that the highest point of performance is reached at 2.

α	Base	Novel	HM
0.0001	75.46	57.89	65.52
0.0005	77.66	60.92	68.28
0.001	79.24	**61.21**	**69.07**
0.005	79.31	60.51	68.65
0.01	**79.42**	60.32	68.56

Table 7: Scaling factor α.

	Base	Novel	HM
Add (ours)	79.24	61.21	69.07
Concatenation	79.98	61.12	69.28
Non-linear	79.11	62.45	69.79

Table 8: Ablation study of fusion mechanism.

Scaling factor α. The scaling factor α balances the importance of the original textual information and the injected visual information. We assess the effect of the scaling factor α by carrying out a series of experiments. Table. 7 shows that with $\alpha = 0.001$, FATE achieves the optimal trade-off performance. Moderately increasing the injection of visual information is beneficial to facilitate our model's adaptation, but the excessive increase leads to a performance decline due to the visual information redundancy.

Fusion Mechanism. Table. 8 shows the impact of different fusion mechanisms. "Concatenation" improves the base performance but degrades the novel one. "Non-linear" show an opposite trend. Nevertheless, changing the fusion mechanism can further improve the performance of HM.

Conclusion

Existing attempts that transfer the pre-trained VLMs to different downstream tasks mainly focused on introducing new parameters for adaptation and needing to learn this part of weight from scratch. To facilitate the new parameter space adapting to new scenarios, we inject informative features from the vision encoder into language encoder's parameters space and propose a novel feature-adapted parameter efficient tuning paradigm, called FATE. In FATE, we leverage the relationship between the visual feature exploited from the image encoder of CLIP and the parameter space of the CLIP's text encoder to generate feature-adapted parameters within the text encoder parameter space. By adjusting these feature-adapted parameters, we can achieve communication between vision and language branches and efficiently adapt CLIP to different scenarios. Extensive experiments have produced compelling results, showing that our FATE has superior generalization performance with a few extra parameters and computation.

Acknowledgments

This work was supported by NSFC 62401361, 62322604, 62176159, U19B2035, and Shanghai Municipal Science and Technology Major Project 2021SHZDZX0102.

References

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.

Bulat, A.; and Tzimiropoulos, G. 2023. Lasp: Text-to-text optimization for language-aware soft prompting of vision & language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23232–23241.

Chen, S.; Ge, C.; Tong, Z.; Wang, J.; Song, Y.; Wang, J.; and Luo, P. 2022a. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35: 16664–16678.

Chen, Z.; Duan, Y.; Wang, W.; He, J.; Lu, T.; Dai, J.; and Qiao, Y. 2022b. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*.

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.

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.

Dou, Z.-Y.; Kamath, A.; Gan, Z.; Zhang, P.; Wang, J.; Li, L.; Liu, Z.; Liu, C.; LeCun, Y.; Peng, N.; et al. 2022a. Coarse-to-fine vision-language pre-training with fusion in the backbone. *Advances in neural information processing systems*, 35: 32942–32956.

Dou, Z.-Y.; Xu, Y.; Gan, Z.; Wang, J.; Wang, S.; Wang, L.; Zhu, C.; Zhang, P.; Yuan, L.; Peng, N.; et al. 2022b. An empirical study of training end-to-end vision-and-language transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18166–18176.

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.

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.

Han, X.; Zhang, Z.; Ding, N.; Gu, Y.; Liu, X.; Huo, Y.; Qiu, J.; Yao, Y.; Zhang, A.; Zhang, L.; et al. 2021. Pre-trained models: Past, present and future. *AI Open*, 2: 225–250.

He, W.; Jamonnak, S.; Gou, L.; and Ren, L. 2023. Clip-s4: Language-guided self-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11207–11216.

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.

Hu, E. J.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W.; et al. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Huang, S.; Dong, L.; Wang, W.; Hao, Y.; Singhal, S.; Ma, S.; Lv, T.; Cui, L.; Mohammed, O. K.; Patra, B.; et al. 2024. Language is not all you need: Aligning perception with language models. *Advances in Neural Information Processing Systems*, 36.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.-H.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, 4904–4916. PMLR.

Jiang, H.; Zhang, J.; Huang, R.; Ge, C.; Ni, Z.; Lu, J.; Zhou, J.; Song, S.; and Huang, G. 2022. Cross-modal adapter for text-video retrieval. *arXiv preprint arXiv:2211.09623*.

Jiang, Z.; Xu, F. F.; Araki, J.; and Neubig, G. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8: 423–438.

Kamath, A.; Singh, M.; LeCun, Y.; Synnaeve, G.; Misra, I.; and Carion, N. 2021. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1780–1790.

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.

Kim, W.; Son, B.; and Kim, I. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *International conference on machine learning*, 5583–5594. PMLR.

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.

Lee, D.; Song, S.; Suh, J.; Choi, J.; Lee, S.; and Kim, H. J. 2023. Read-only prompt optimization for vision-language few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1401–1411.

Lester, B.; Al-Rfou, R.; and Constant, N. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022a. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, J.; Selvaraju, R.; Gotmare, A.; Joty, S.; Xiong, C.; and Hoi, S. C. H. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34: 9694–9705.

Li, L. H.; Zhang, P.; Zhang, H.; Yang, J.; Li, C.; Zhong, Y.; Wang, L.; Yuan, L.; Zhang, L.; Hwang, J.-N.; et al. 2022b. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10965–10975.

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

Mou, C.; Wang, X.; Xie, L.; Wu, Y.; Zhang, J.; Qi, Z.; and Shan, Y. 2024. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 4296–4304.

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.

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.

Peng, Z.; Xu, Z.; Zeng, Z.; Wang, Y.; Xie, L.; Tian, Q.; and Shen, W. 2024a. Parameter-efficient Fine-tuning in Hyperspherical Space for Open-vocabulary Semantic Segmentation. *arXiv preprint arXiv:2405.18840*.

Peng, Z.; Xu, Z.; Zeng, Z.; Xie, L.; Tian, Q.; and Shen, W. 2024b. Parameter efficient fine-tuning via cross block orchestration for segment anything model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3743–3752.

Peng, Z.; Xu, Z.; Zeng, Z.; Yang, X.; and Shen, W. 2024c. Sam-parser: Fine-tuning sam efficiently by parameter space reconstruction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 4515–4523.

Petroni, F.; Rocktäschel, T.; Lewis, P.; Bakhtin, A.; Wu, Y.; Miller, A. H.; and Riedel, S. 2019. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*.

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.

Rao, Y.; Zhao, W.; Chen, G.; Tang, Y.; Zhu, Z.; Huang, G.; Zhou, J.; and Lu, J. 2022. Denseclip: Language-guided dense prediction with context-aware prompting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 18082–18091.

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.

Shin, T.; Razeghi, Y.; Logan IV, R. L.; Wallace, E.; and Singh, S. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

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*.

Tsimpoukelli, M.; Menick, J. L.; Cabi, S.; Eslami, S.; Vinyals, O.; and Hill, F. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34: 200–212.

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.; Zhang, Z.; Ebrahimi, S.; Sun, R.; Zhang, H.; Lee, C.-Y.; Ren, X.; Su, G.; Perot, V.; Dy, J.; et al. 2022. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European Conference on Computer Vision*, 631–648. Springer.

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.

Yao, H.; Zhang, R.; and Xu, C. 2023a. Visual-language prompt tuning with knowledge-guided context optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6757–6767.

Yao, H.; Zhang, R.; and Xu, C. 2023b. Visual-language prompt tuning with knowledge-guided context optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6757–6767.

Zang, Y.; Li, W.; Zhou, K.; Huang, C.; and Loy, C. C. 2022. Open-vocabulary detr with conditional matching. In *European Conference on Computer Vision*, 106–122. Springer.

Zhai, X.; Wang, X.; Mustafa, B.; Steiner, A.; Keysers, D.; Kolesnikov, A.; and Beyer, L. 2022. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 18123–18133.

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.

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.

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