TextRefiner: Internal Visual Feature as Efficient Refiner for Vision-Language Models Prompt Tuning

Jingjing Xie[1], **Yuxin Zhang**[1], **Jun Peng**[1], **Zhaohong Huang**[1], **Liujuan Cao**[1 *]

[1] Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, Xiamen, China

Abstract

Despite the efficiency of prompt learning in transferring vision-language models (VLMs) to downstream tasks, existing methods mainly learn the prompts in a coarse-grained manner where the learned prompt vectors are shared across all categories. Consequently, the tailored prompts often fail to discern class-specific visual concepts, thereby hindering the transferred performance for classes that share similar or complex visual attributes. Recent advances mitigate this challenge by leveraging external knowledge from Large Language Models (LLMs) to furnish class descriptions, yet incurring notable inference costs. In this paper, we introduce TextRefiner, a plug-and-play method to refine the text prompts of existing methods by leveraging the internal knowledge of VLMs. Particularly, TextRefiner builds a novel local cache module to encapsulate fine-grained visual concepts derived from local tokens within the image branch. By aggregating and aligning the cached visual descriptions with the original output of the text branch, TextRefiner can efficiently refine and enrich the learned prompts from existing methods without relying on any external expertise. For example, it improves the performance of CoOp from 71.66% to 76.96% on 11 benchmarks, surpassing CoCoOp which introduced instance-wise feature for text prompts. Equipped with TextRefiner, PromptKD achieve state-of-the-art performance while keep inference efficient.

Introduction

Vision-Language Models (VLMs), such as CLIP (Radford et al. 2021) and ALIGN (Jia et al. 2021), have manifested exceptional generalization abilities, significantly benefiting a vast spectrum of applications like open-vocabulary image recognition (Zhai et al. 2023), object detection (Gu et al. 2022), and image segmentation (Li et al. 2022). Generally, VLMs optimize an image encoder and a text encoder to align text and image features in a unified embedding space via contrastive loss. By pre-training on large-scale web data, VLMs can acquire transferable representations in both visual and textual domains, facilitating their application to a broad range of downstream visual tasks. As such, numerous research efforts have been directed towards devising approaches to adapt VLMs in a manner mindful of both parameters and data (Khattak et al. 2023a; Shen et al. 2024; Gao et al. 2024b; Yu et al. 2023; Li et al. 2024a).

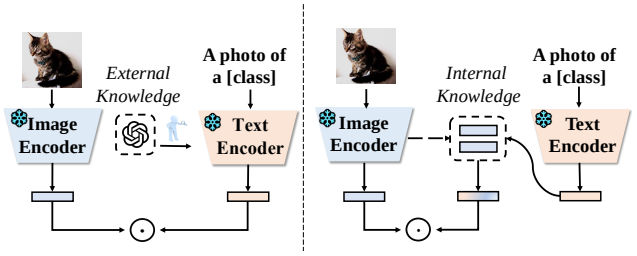


Figure 1: Comparison of different paradigms for enriching text prompts. Previous methods (left) introduced ***external*** knowledge experts to furnish fine-grained descriptions of each class, necessitating an extra filtering process to maintain alignment with downstream datasets. In contrast, our proposed TextRefiner (right) leverages the ***internal*** knowledge of the image branch to supply fine-grained, localized region information, thereby drastically reducing the inference overhead while maintaining the performance.

To date, the predominant focus of research has predominantly fallen on employing prompt learning (Zhou et al. 2022b,a; Kunananthaseelan, Zhang, and Harandi 2024; Khattak et al. 2023b), a widely acknowledged strategy for adapting large pre-trained models. As an initial foray, CoOp (Zhou et al. 2022b) leveraged prompt learning to adapt text prompts of CLIP, resulting in marked performance improvements in downstream tasks. Subsequent research efforts have sought to extend prompt learning into the visual (Wu et al. 2023; Li et al. 2024b; Shen et al. 2024) and multimodal domains (Khattak et al. 2023a; Xing et al. 2023; Gao et al. 2024a; Xin et al. 2024), aiming to thoroughly fine-tune the representations across modalities to enhance their alignment.

Despite the ongoing advancements, a prevalent limitation of these works is their tendency to enhance high-level semantics in a coarse-grained, global manner, resulting in holistic alignment across modalities. Consequently, the tailored prompts may fail to instruct the model to discern diverse visual concepts from local regions, thereby hindering its capability to generalize across classes that share similar visual attributes. To overcome this limitation, recent works

*Corresponding author

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

such as ArGue (Tian et al. 2024) and LLaMP (Zheng et al. 2024) utilize large language models (LLMs) to enrich the text prompts with fine-grained class descriptions that enhance transfer performance, as shown in Figure 1 (left). Unfortunately, these approaches necessitate additional processes to sift through the LLMs outputs to ensure their relevance to downstream datasets, thereby introducing significant inference overhead.

This work aims to enrich VLM prompt tuning without the reliance on external knowledge experts, as previously described. Our core impetus stems from a widely accepted consensus in the traditional computer field: *fine-grained visual concepts at local regions are prominently present in the intermediate layer outputs of the visual networks* (Zeiler and Fergus 2014; Kim, Nam, and Ko 2022a; Selvaraju et al. 2020). As such, one can intuitively exploit the internal knowledge within the VLM, specifically the intermediate outputs of the image encoder, to deliver precise class descriptions and achieve local alignment without the need for external LLMs assistance.

To this end, we propose a plug-and-play method, dubbed TextRefiner, to refine the text prompt in VLM tuning without the need for external LLMs' assistance. As depicted in Fig 1, TextRefiner incorporates a novel local cache module to encapsulate fine-grained visual concepts derived from local tokens within the image branch. More particularly, amidst VLM tuning, the image branch continuously writes fine-grained semantic information into the cache in a class-wise manner. The fine-grained information retrieved for each category is concatenated with the corresponding class embedding, followed by a feature aggregation module to fuse global and local information to enhance the representation capability for the text branch. To further align the intermediate embeddings between modalities, we furnish TextRefiner with a feature alignment module to transform local tokens into the text embedding space. Owing to such design, the fine-grained local features from the image branch can be seamlessly utilized to deliver class descriptions, thereby effectively refining the vanilla text prompting performance.

It is noteworthy that TextRefiner is both scalable and orthogonal to augment the efficacy of prevailing VLM tuning methods, *i.e*, the local cache module operates independently with existing prompt tuning paradigms. For example, TextRefiner enhances CoOp by 5.30% on 11 benchmarks, surpassing CoCoOp which introduced instance-specific features for text prompts to improve generalization. Equipped with TextRefiner, PromptKD achieves state-of-the-art performance, surpassing LLaMP, which relies on external experts, by 1.06%. Moreover, TextRefiner demonstrates high inference efficiency. CoCoOp achieves 20.45 FPS, LLaMP reaches 1473.46 FPS, while PromptKD *w/*TextRefiner achieves 12793.26 FPS. This efficiency is due to its reliance on simple matrix multiplication at the text output, unlike CoCoOp and LLaMP, which require additional modules to extract and combine information with the text input.

The contributions of this paper are delineated as follows:

- We introduce TextRefiner, which, for the first time, exploits the internal knowledge of the image branch to refine the text prompts of VLMs.
- TextRefiner introduces three principal innovations, encompassing local cache, feature aggregation, and feature alignment, which collaboratively enhance the incorporation of fine-grained visual attributes into the text prompts.
- Extensive experiments demonstrate the effectiveness of TextRefiner in boosting VLM prompt tuning, even without additional inference costs associated with prior methods that utilize external knowledge from LLMs.

Related Works

Vision-Language Model

Traditional visual representation learning relied exclusively on a predefined set of labels limiting its expansion to unseen classes (Deng et al. 2009; He et al. 2016; Dosovitskiy et al. 2021; Liu et al. 2022). Vision-language models redress this deficiency by incorporating natural language supervision, thereby aiding vision systems in mastering more visual concepts. These models are pre-trained on copious amounts of raw text paired with images from the Internet in a self-supervised manner. Marked advances such as ALIGN (Jia et al. 2021) and CLIP (Radford et al. 2021) utilize contrastive loss during pre-training to align both modalities into a unifying embedding space. Benefiting from the large-scale dataset and contrastive training objective, the learned visual representations connect with language and are highly generalizable, which has been widely applied to various downstream vision tasks, such as object detection (Gu et al. 2022), semantic segmentation (Li et al. 2022) and depth estimation (Zhang et al. 2022). Despite these formidable capabilities, the transfer of VLMs to downstream tasks at few-shot scenarios poses a formidable challenge, where the quantity of data in such tasks falls substantially short relative to pre-training, leading to a significant falloff in generalizability (Zhu et al. 2023; Khattak et al. 2023b; Roy and Etemad 2024). Our proposed TextRefiner in this paper is crafted to achieve efficient transfer learning in low-data scenarios while preserving generalization.

Prompt Learning in VLMs

Prompt learning is progressively favored in adapting Vision-and-Language Models (VLMs) to downstream tasks without necessitating a full re-training of the original model. For VLMs, prompt learning entails introducing learnable text or visual prompts that transcend manually-defined prompts like `a photo of a {class}`. CoOp (Zhou et al. 2022b) incorporates learnable tokens into predefined text prompts within the text branch. These tokens, universally shared across classes, glean task-related knowledge to align with the image features derived from the downstream dataset. Zhou *et al.* (Zhou et al. 2022a) proposed conditional prompt learning (CoCoOp), constraining image features in an instance-specific manner to curtail overfitting in few-shot scenarios and bolster generalization towards unseen classes. PromptSRC (Khattak et al. 2023b) further modulates learned prompts utilizing pre-trained features to enhance generalization. MaPLe (Khattak et al. 2023a) expands prompt learning into multimodal branches, enabling the prompt to not only capture characteristics within the

text branch but also enhance inter-modal alignment. Despite their effectiveness, these methods generally tune the prompt on a rather coarse granularity scale, which tends to overlook the fine-grained image feature, potentially impairing the generalization across classes that share similar visual attributes. To mitigate this, ArGue (Tian et al. 2024) and LLaMP (Zheng et al. 2024) utilize large language models (LLMs) to provide detailed attribute information of classes to the text branch, achieving fine-grained alignment in downstream tasks. However, such LLM-based methods necessitate well-curated filtering of knowledge generated by the LLMs to ensure its relevance to downstream tasks, consequently incurring additional computational expenses. Our method capitalizes on the semantic information embedded in image local tokens to realize fine-grained alignment without the need for complex filtering or huge inference costs.

Method

Background

Preliminaries. By aligning image and text in a joint embedding space, Vision-Language Models (VLMs) like CLIP (Radford et al. 2021) and ALIGN (Jia et al. 2021) mark significant advancements in vision applications. Following previous works (Zhou et al. 2022a; Khattak et al. 2023b; Zheng et al. 2024), this paper utilizes CLIP as the foundation model, with relevant preliminary details provided herein. CLIP consists of an image encoder, labeled as f_I, and a text encoder referred to as f_T. Both encoders incorporate a feature extractor and a projection layer the transfer the multimodal inputs into a joint embedding space. Without loss of generality, we utilize the CLIP model equipped with a vision transformer (ViT) for illustration.

During the training phase, a contrastive loss (Chen et al. 2020) is employed to maximize the cosine similarity between embeddings of corresponding image-text pairs $\{I, T\}$, where I denotes the image and T the associated caption. For testing, CLIP integrates each class name within a hard prompt formatted as `a photo of a {class}`, creating textual class descriptions $\mathbf{P} \in \mathbb{R}^{C\times l}$, where C is the number of classes and l is the harmonized length of text sequences. The text encoder f_T then generates textual embeddings $\mathbf{E} \in \mathbb{R}^{C\times d}$ for $\mathbf{P}$, which act as class templates. Then, with the image feature $v \in \mathbb{R}^d$ outputted by the image encoder f_I, the prediction probabilities are determined by the corresponding similarity between textual embeddings and the image feature, expressed as:

$$P(c) = \frac{\exp(\cos(v, \mathbf{E}_c)/\tau)}{\sum_{c=1}^{C} \exp(\cos(v, \mathbf{E}_c)/\tau)}, \tag{1}$$

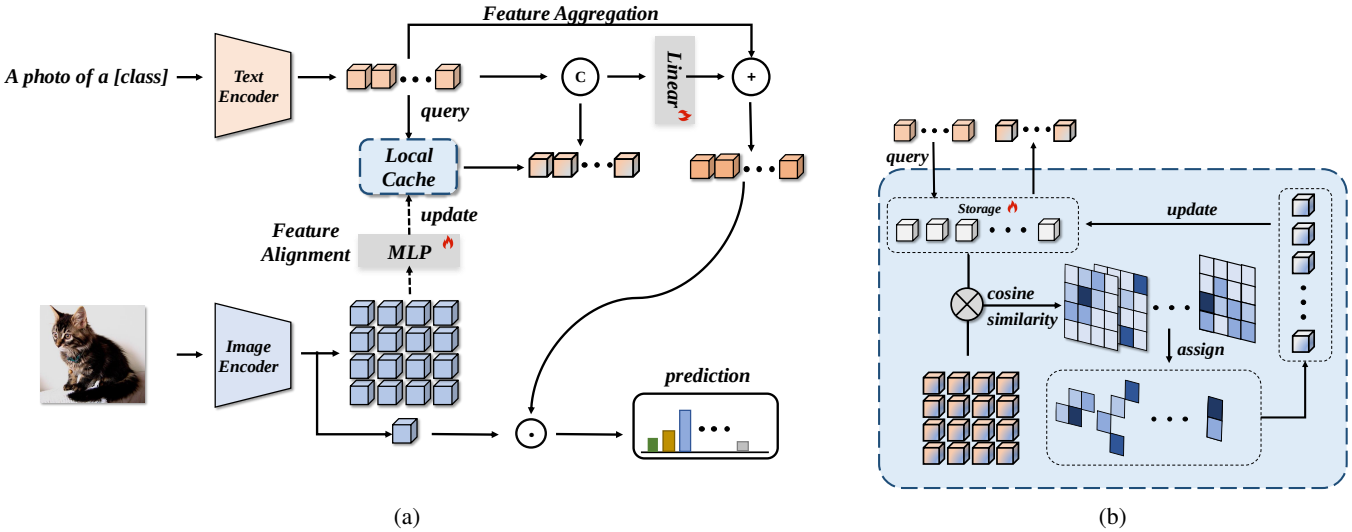
where c represents a specific class, $\cos(\cdot)$ calculates the cosine distance between vectors, and τ is a tuning parameter for scaling the softmax function.

VLMs Prompt Tuning. Instead of manually crafted hard prompts, VLM prompt tuning methods, pioneered by CoOp (Zhou et al. 2022b), learn soft prompts to further improve the transferring performance on downstream tasks. Specifically, CoOp incorporates multiple learnable word embeddings p_i shared within all classes to construct textual descriptions as $\mathbf{P} = [p_1, p_2, ..., p_L]\{\texttt{class}\}$, where L specifies the number of soft prompts. With these learnable prompts, CLIP can enhance task-relevant textual descriptions across all classes. Subsequent research efforts introduce learnable context tokens into the image branch or both branches to better learn downstream representations (Wu et al. 2023; Shen et al. 2024; Khattak et al. 2023a; Xing et al. 2023). Despite their efficacy, the prevailing paradigms largely enhance embeddings in a coarse-grained manner, with prompts that are universal across all classes, achieving only global alignment and often failing to reflect the diverse visual attributes of localized regions. Although recent works have revealed the potential in utilizing external knowledge from LLMs (Touvron et al. 2023a,b) to refine the learned prompts with fine-grained class-wise information, it comes at additional inference costs associated with filtering the outputs of LLMs to ensure their pertinence to specific downstream datasets.

TextRefiner

To mitigate the aforementioned challenges, this paper presents a plug-and-play method to refine the text prompt in VLM prompt tuning without depending on external assistance, termed TextRefiner. The core contribution of TextRefiner lies in enhancing the textual embedding with linguistic visual attributes using local features within the image branch, which have been demonstrated to encompass rich fine-grained visual concepts at local regions (Ghiasi et al. 2022; Ma et al. 2023; Kim, Nam, and Ko 2022b). TextRefiner achieves this objective through three complementary innovations, including local cache, feature aggregation and feature alignment, as shown in Fig. **??**. We detailedly introduce them as below.

Local Cache. The visual encoder ViT splits input image into non-overlapping patches and feeds them into stacked transformer blocks to obtain a series of local tokens $\mathbf{V} = \{v_1, v_2, ..., v_N\}$, where N represents the number of patches. These local tokens have been widely demonstrated to be capable of capturing specialized visual features such as edges, textures, and represent discernible conceptual categories (Ghiasi et al. 2022). Therefore, we propose to leverage these internal local tokens within VLMs as visual attribute descriptions for classes, forsaking the practice of using external experts as previous methods do. To achieve this, we propose a novel local cache module to store the fine-grained visual concepts from local tokens. Instead of coarsely storing all local tokens in bulk, we establish a fixed number of entries within the local cache and cluster similar local tokens into a single entry to selectively capture corresponding visual attributes. For instance, when processing a zebra sample, our objective is to amalgamate its pronounced black and white stripe features into a cache entry that epitomizes textures. To this end, we construct an storage $\mathbf{A} \in \mathbb{R}^{M\times d}$, where M entries in $\mathbf{A}$ are used to collect information from local tokens across all instances based on similarity and d denotes the feature dimension. Given local tokens, we calculate the cosine similarity between the attribute



(a) (b)

Figure 2: (a) shows the framework of TextRefiner, which is composed of local cache, feature aggregation, and feature alignment. (b) demonstrate the mechanism of local cache. Each item in the local cache can be considered as an attribute prior, updated by local tokens from the image branch. Textual class embedding can obtain corresponding linguistic visual attributes by querying this cache.

storage and local tokens, followed by a softmax function to obtain probability, as follows:

$$\mathbf{D}_{i,j} = \frac{\exp(\cos(v_i, \mathbf{A}_j))}{\sum_{j=1}^{M} \exp(\cos(v_i, \mathbf{A}_j))}. \tag{2}$$

Then, the local tokens are assigned to the entry in $\mathbf{A}$ with the highest probability. For example, the allocated position of the j-th entry in the local cache is formulated as follows:

$$G_j = \{\, i \,|\, \arg\max_k \mathbf{D}_{i,k} = j \}. \tag{3}$$

Subsequently, the j-th entry gathers information from local tokens as:

$$\mathbf{A}_j = \gamma \cdot \mathbf{A}_j + (1-\gamma) \sum_{i \in G_j} \mathbf{D}_{i,j} \cdot v_i, \tag{4}$$

where γ denotes the momentum coefficient. Through the above updates, we achieve continuously memorizing the fine-grained information inherent in the local tokens and this information subsequently can be utilized to enhance the descriptive capabilities of text representation.

Feature Aggregation. To harness the potential of the fine-grained information in the local cache, we introduce feature aggregation operation. As shown in Figure **??**, we enhance text embedding of classes by integrating local context from the local cache into the global context from CLIP via residual connection (He et al. 2016; Yu et al. 2023). Specifically, we begin by matching each text embedding $\mathbf{E}_i$ from CLIP with the corresponding visual attributes stored in entries of the local cache as:

$$\mathbf{W}_{i,j} = \frac{\exp(\cos(\mathbf{E}_i, \mathbf{A}_j))}{\sum_{j=1}^{M} \exp(\cos(\mathbf{E}_i, \mathbf{A}_j))}, \tag{5}$$

Then, the detailed descriptive embedding associated with the corresponding i-th class is calculated as:

$$\overline{\mathbf{E}}_i = \sum_{j=1}^{M} \mathbf{W}_{i,j} \cdot \mathbf{A}_j. \tag{6}$$

Then, we concatenate the detailed descriptive embedding with the original text embedding and subsequently employ a linear layer with residual connection to aggregate the two types of features:

$$\hat{\mathbf{E}}_i = \alpha \cdot \text{Linear}([\mathbf{E}_i, \overline{\mathbf{E}}_i]) + \mathbf{E}_i, \tag{7}$$

where α is the coefficient hyperparameter.

Feature Alignment. CLIP aligns the global feature from images and texts to learn a joint embedding space. However, local features in images lack explicit alignment, leading to the modality gap between the local feature $\mathbf{V}$ and the textual embedding $\mathbf{E}$. To address potential feature shift, we further incorporate a feature alignment module to transform local features into the text embedding space, mitigating the modality gap. The feature alignment module simply consists of a 2-layer MLP where $\mathbf{V}$ will be transformed as:

$$\hat{\mathbf{V}} = \mathbf{W}_2 \sigma(\text{norm}(\mathbf{W}_1 \mathbf{V})). \tag{8}$$

Training

During the training, we adhere to the original CLIP framework, utilizing contrastive loss as the primary classification supervision, formulated as follows:

$$\mathcal{L}_{cls} = -\log \frac{\exp(\cos(v, \hat{\mathbf{E}}_c)/\tau)}{\sum_{j=1}^{C} \exp(\cos(v, \hat{\mathbf{E}}_j)/\tau)}. \tag{9}$$

Method	*Average*			ImageNet			Caltech101			OxfordPets		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP	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	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	80.47	71.69	75.83	75.98	70.43	73.10	97.96	93.81	95.84	95.20	97.69	96.43
PromptSRC	84.26	76.10	79.97	77.60	70.73	74.01	98.10	94.03	96.02	95.33	97.30	96.30
MaPLe	82.28	75.14	78.55	76.66	70.54	73.47	97.74	94.36	96.02	95.43	97.76	96.58
PromptKD	84.11	78.28	81.09	77.63	70.96	74.15	98.31	96.29	97.29	93.42	97.44	95.39
LLaMP	85.16	77.71	81.27	**77.99**	71.27	**74.48**	98.45	95.85	97.13	**96.31**	97.74	**97.02**
CoOp *w*/TextRefiner	79.74	74.32	76.94	76.84	70.54	73.56	98.13	94.43	96.24	95.27	97.65	96.45
PromptKD *w*/TextRefiner	**85.22**	**79.64**	**82.33**	77.51	**71.43**	74.35	**98.52**	**96.52**	**97.51**	95.60	**97.90**	96.74

Method	StanfordCars			Flowers102			Food101			FGVCAircraft		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP	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	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	70.49	73.59	72.01	94.87	71.75	81.71	90.70	91.29	90.99	33.41	23.71	27.74
PromptSRC	78.27	74.97	76.58	98.07	76.50	85.95	90.67	91.53	91.10	42.73	37.87	40.15
MaPLe	72.94	74.00	73.47	95.92	72.46	82.56	90.71	92.05	91.38	37.44	35.61	36.50
PromptKD	80.48	81.78	81.12	98.69	81.91	89.52	89.43	91.27	90.34	43.61	39.68	41.55
LLaMP	**81.56**	74.54	77.89	97.82	77.40	86.42	91.05	91.93	91.49	**47.30**	37.61	41.90
CoOp *w*/TextRefiner	71.40	70.90	71.15	95.92	74.33	83.76	90.88	91.43	91.15	35.35	35.87	35.61
PromptKD *w*/TextRefiner	80.91	**81.83**	**81.37**	**99.30**	**82.91**	**90.37**	**91.42**	**92.71**	**92.06**	45.01	**40.12**	**42.42**

Method	SUN397			DTD			EuroSAT			UCF101		
	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM	Base	Novel	HM
CLIP	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	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	79.74	76.86	78.27	77.01	56.00	64.85	87.49	60.04	71.21	82.33	73.45	77.64
PromptSRC	82.67	78.47	80.52	83.37	62.97	71.75	92.90	73.90	82.32	87.10	78.80	82.74
MaPLe	80.82	78.70	79.75	80.36	59.18	68.16	**94.07**	73.23	82.35	83.00	78.66	80.77
PromptKD	82.53	**80.88**	81.70	82.86	69.15	75.39	92.04	71.59	80.54	86.23	80.11	83.06
LLaMP	**83.41**	79.90	81.62	83.49	64.49	72.77	91.93	**83.66**	**87.60**	87.13	80.66	83.77
CoOp *w*/TextRefiner	80.96	76.49	78.66	75.35	58.09	65.60	74.57	72.82	73.68	82.52	75.01	78.59
PromptKD *w*/TextRefiner	83.02	80.50	**81.74**	**83.91**	**71.01**	**76.92**	92.99	79.22	85.55	**89.20**	**81.90**	**85.39**

Table 1: Comparison between our method and other existing methods on base-to-novel generalization.

In addition to the vanilla contrastive loss, we augment TextRefiner with a semantic loss and a regularization loss component. Respectively, we select the top-k transformed local features with attention scores and calculate the average of all tokens to obtain $\mathbf{S} \in \mathbb{R}^{(k+1)\times d}$. The semantic loss ensures that $\mathbf{S}$ align with their corresponding text embeddings, expressed as:

$$\mathcal{L}_{sem} = \frac{1}{k+1}\sum_{i=1}^{k+1}\log\frac{\exp(\cos(\mathbf{S}_i, \hat{\mathbf{E}}_c)/\tau)}{\sum_{j=1}^{C}\exp(\cos(\mathbf{S}_i, \hat{\mathbf{E}}_j)/\tau)}. \quad (10)$$

On the other hand, we implement a regularization loss to curb overfitting to the limited amount of training images in VLMs fine-tuning scenarios, which can be denoted as:

$$\mathcal{L}_{reg} = |\mathbf{E} - \hat{\mathbf{E}}|. \quad (11)$$

The overall loss can be formulated as follows:

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda_1 * \mathcal{L}_{sem} + \lambda_2 * \mathcal{L}_{reg}, \quad (12)$$

where λ_1 and λ_2 are hyperparameters utilized to balance the loss components.

Inference

During the inference phase for the input image denoted by x, we compute the cosine similarity between $v = f_I(x)$ and the fused textual embedding $\hat{\mathbf{E}}_i$ for all classes. The predicted label y is ascertained based on the maximum likelihood, expressed as:

$$y = \arg\max_i \frac{\exp(\cos(v, \hat{\mathbf{E}}_i)/\tau)}{\sum_{j=1}^{C}\exp(\cos(v, \hat{\mathbf{E}}_j)/\tau)} \quad (13)$$

Unlike other methods that insert hard or carefully designed learnable prompts representing visual attributes into the input embedding of the text branch, our approach only requires feature aggregation between the textual output embeddings and the local cache, providing an efficiency advantage.

Method	Source	Target			
	ImageNet	-V2	-Sketch	-A	-R
CLIP	66.73	60.83	46.15	47.77	73.96
CoOpOp	71.02	64.07	48.75	50.63	76.18
PromptSRC	71.27	64.35	49.55	50.90	77.80
CoOp	71.51	64.20	47.99	49.71	75.21
+ TextRefiner	72.06	65.02	48.58	49.77	76.30
MaPLe	70.72	64.07	49.15	50.90	76.98
+ TextRefiner	71.13	64.54	49.08	51.49	77.71

Table 2: Comparison between our method and other existing methods on cross-domain generalization.

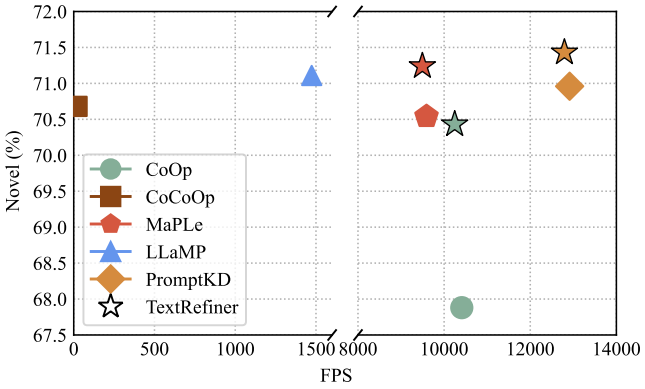


Figure 3: Comparison of inference efficiency among existing methods on the ImageNet dataset. Our TextRefiner is more efficient than LLaMP which relies on external knowledge.

Experiment

Settings

Datasets. We follow the common practice in the literature (Zhou et al. 2022a; Khattak et al. 2023b,a) to adopt two experimental settings: *base-to-novel* and *cross-domain*. For *base-to-novel* evaluation, we adopt a wide range of visual recognition benchmark datasets *i.e.*, ImageNet (Deng et al. 2009), Caltech101 (Fei-Fei, Fergus, and Perona 2004), OxfordPets (Parkhi et al. 2012), StanfordCars (Krause et al. 2013), Flowers102 (Nilsback and Zisserman 2008), Food101 (Bossard, Guillaumin, and Gool 2014), FGVCAircraft (Maji et al. 2013), SUN397 (Xiao et al. 2016), DTD (Cimpoi et al. 2014), EuroSAT (Helber et al. 2019)and UCF101 (Soomro, Zamir, and Shah 2012). In this setting, the datasets are split into two non-overlapping sets where the base classes and novel classes serve as training sets and test sets, respectively. For cross-domain evaluation, we train our model on ImageNet and test the generalization performance on its variants: ImageNetV2 (Recht et al. 2019), ImageNet-Sketch (Wang et al. 2019), ImageNet-A (Hendrycks et al. 2021b) and ImageNet-R (Hendrycks et al. 2021a). Both are trained using 16 shots per class from the training sets and are tested on the full test sets.

Implementation details. For a fair comparison, we adopt CLIP with ViT-B/16 vision encoder and verify the effectiveness of our method on CoOp (Zhou et al. 2022b), MaPLe and PromptKD (Li et al. 2024b). The setting of epochs, batch size, learning rate, optimizer and soft token length corresponds with the original papers, except for CoOp, where the number of epochs was reduced from 200 to 10. In particular, the momentum α and the fusion factor β are configured to be 0.8 and 0.2, respectively. For the hyper-parameters in Eq. 12, we set λ_1 to 0.02 and λ_2 to 20 based on empirical observation. For simplicity, we report ablation results on CoOp to verify its effectiveness.

Main Results

Base-to-novel. Firstly, we report the quantitative results of various methods for *base-to-novel* tasks on 11 benchmarks. Table 1 shows our method can consistently enhance the performance of existing methods. In particular, TextRefiner enhances CoOp's generalization on novel classes, improving accuracy from 63.22% to 74.32%. Compared to CoCoOp, which integrates instance-conditional context produced by meta-net to refine text prompt, TextRefiner effectively local tokens and offers additional improvements. For PromptKD, our method enhances their performance on fine-grained benchmarks. On *DTD*, TextRefiner increases novel accuracy from 69.15% to 71.01%, On *EuroSAT*, TextRefiner achieves improvements of 7.63%. Compared to LLaMP, which uses LLMs to provide detailed descriptions according to class names, our method achieves better generalization and provides a better balance between base and novel classes, where PromptKD *w/*TextRefiner achieves average gains of 1.97% in novel accuracy, and 1.06% in harmonic mean on average. This suggests that additional filtering modules run the risk of overfitting downstream tasks, requiring careful design. In contrast, our approach leverages the internal knowledge in visual networks to improve downstream performance while enhancing generalization. These improvements verify that our method can achieve better balances on seen and unseen data, thus showcasing stronger generalization compared to the original methods.

Cross-domain. Moving beyond fundamental *base-to-novel* benchmarks, we exploit the generalization capacity of TextRefiner across domains on four commonly-used datasets *i.e.*, ImageNet-V2, ImageNet-Sketch, ImageNet-A and ImageNet-R. The experimental results are shown in Table 2, where a significant improvement has been observed in all scenarios and the robustness to domain shift is verified.

Efficiency. We also provide a comparison of inference efficiency which is evaluated with one single A800 GPU based on the officially released code. As shown in Figure 3, TextRefiner only slightly reduces FPS and remains highly efficient. In contrast, LLaMP, which relies on external experts, significantly compromises inference speed.

Ablations

The effects of each component. To verify the effectiveness of the proposed method, we examine different combinations of components. As shown in Table 3, TextRefiner can significantly enhance the performance of CoOp by pro-

TextRefiner	$\mathcal{L}_{sem}$	$\mathcal{L}_{reg}$	ImageNet		
			Base	Novel	HM
			76.47	67.88	71.92
✓			76.50	70.48	73.37
✓	✓		76.58	70.62	73.48
✓		✓	76.70	69.67	73.02
✓	✓	✓	76.84	70.54	73.56

Table 3: Ablation study on each component.

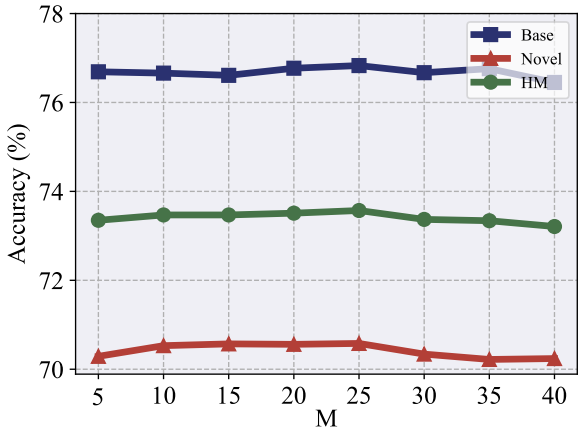


Figure 4: Ablation study on M in **A**.

$\frac{1}{\lambda_1}$	20	30	40	50	60
Base	76.33	76.45	76.52	76.84	76.60
Novel	70.14	70.04	70.28	70.54	70.27
HM	73.10	73.10	73.27	73.56	73.30
λ_2	5	10	15	20	25
Base	76.65	76.70	76.67	76.84	76.65
Novel	70.49	70.51	70.32	70.54	70.34
HM	73.44	73.47	73.36	73.56	73.36

Table 4: Ablation study on loss factors.

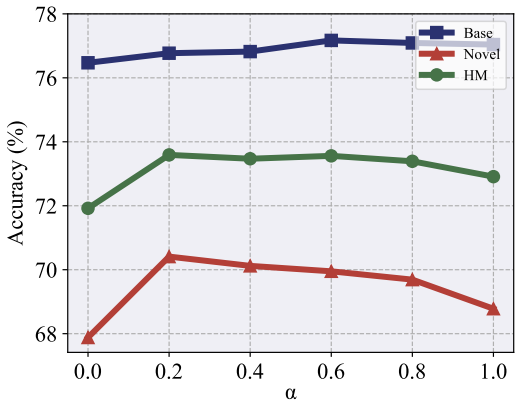


Figure 5: Ablation study on aggregation coefficient in Eq. 7.

viding fine-grained information in the local cache. The introduced semantic loss ensures that the semantic information stored in the local cache is distinguish and aligned with the text, further improving performance. Introducing regularization loss enhances performance on seen classes but restricts the utilization of fine-grained information in the local cache, which leads to decreased generalization on unseen classes. Utilizing all components together effectively balances the recognition of seen classes and the generalization of unseen classes.

The effects of M. We investigate the effect of M in **A**, controlling the condensation level of local tokens, where a larger M means more entries to summarize the visual attributes from local tokens. As shown in Figure 4, recognition performance improves as M increases, but once M reaches a certain value, performance starts to decline. This can be understood as follows: when M exceeds a certain threshold, complete visual attributes are split and stored in multiple entries compromising their integrity and text embedding struggles to recombine the correct attributes.

The effects of λ_1 and λ_2. To better understand the effects of these two loss factors, we fixed one and observed the performance changes as the other varied. As shown in Table 4, the performance of our method consistently increases first and then decreases when the loss balance factor λ_1 and λ_2 increase. This observation suggests that excessively aligning local tokens can cause the stored information to overfit the local features of seen samples, leading to a spurious relationship. In addition, applying a strong regularization results in insufficient learning for TextRefiner.

The effects of fusion coefficient α. We also studied the hyperparameter α, where a larger α indicates the final text embedding contains more local information from $\overline{\mathbf{E}}$. In Figure 5, as α increases, accuracy on the base class continuously improves, while accuracy on the novel class would increase first and then continuously decrease. This indicates that with a limited number of samples, over-reliance on fine-grained information from seen samples may lead to overfitting and damage generalization capability.

Conclusion

In this paper, we utilize visual concepts naturally embedded in local tokens of visual networks to enhance the text prompt with fine-grained information. Specifically, we introduce TextRefiner, a plug-and-play module, containing the local cache, feature aggregation, and feature alignment. The local cache is used to continuously store fine-grained information from local tokens. Meanwhile, feature aggregation provides a solution to fuse global and local information to enhance the representation capability for the text branch. Feature alignment module can alleviate the modality gap between the text embedding and local tokens. We also introduce two additional training losses, the semantic loss and regularization loss, to aid in the optimization process. We hope this study will inspire additional advancements in the field of multimodal learning and efficient transfer learning.

Acknowledgments

This work was supported by National Science and Technology Major Project (No. 2022ZD0118201), the National Science Fund for Distinguished Young Scholars (No.62025603), the National Natural Science Foundation of China (No. U21B2037, No. U22B2051, No. U23A20383, No. U21A20472, No. 62176222, No. 62176223, No. 62176226, No. 62072386, No. 62072387, No. 62072389, No. 62002305 and No. 62272401), and the Natural Science Foundation of Fujian Province of China (No. 2021J06003, No.2022J06001).

References

Bossard, L.; Guillaumin, M.; and Gool, L. V. 2014. Food-101–mining discriminative components with random forests. In *ECCV*, 446–461. Springer.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Cimpoi, M.; Maji, S.; Kokkinos, I.; Mohamed, S.; and Vedaldi, A. 2014. Describing textures in the wild. In *CVPR*, 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 *CVPR*, 248–255. Ieee.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations, (ICLR)*.

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 *CVPR workshop*, 178–178. IEEE.

Gao, J.; Ruan, J.; Xiang, S.; Yu, Z.; Ji, K.; Xie, M.; Liu, T.; and Fu, Y. 2024a. LAMM: Label Alignment for Multi-Modal Prompt Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 1815–1823.

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

Ghiasi, A.; Kazemi, H.; Borgnia, E.; Reich, S.; Shu, M.; Goldblum, M.; Wilson, A. G.; and Goldstein, T. 2022. What do vision transformers learn? a visual exploration. *arXiv preprint arXiv:2212.06727*.

Gu, X.; Lin, T.-Y.; Kuo, W.; and Cui, Y. 2022. Open-vocabulary Object Detection via Vision and Language Knowledge Distillation. In *International Conference on Learning Representations*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

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.

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.

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

Khattak, M. U.; Wasim, S. T.; Naseer, M.; Khan, S.; Yang, M.-H.; and Khan, F. S. 2023b. Self-regulating prompts: Foundational model adaptation without forgetting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15190–15200.

Kim, S.; Nam, J.; and Ko, B. C. 2022a. ViT-NeT: Interpretable Vision Transformers with Neural Tree Decoder. In *International Conference on Machine Learning (ICLR)*, volume 162, 11162–11172.

Kim, S.; Nam, J.; and Ko, B. C. 2022b. Vit-net: Interpretable vision transformers with neural tree decoder. In *International conference on machine learning*, 11162–11172. PMLR.

Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3d object representations for fine-grained categorization. In *ICCV workshops*, 554–561.

Kunananthaseelan, N.; Zhang, J.; and Harandi, M. 2024. LaViP:Language-Grounded Visual Prompts.

Li, B.; Weinberger, K. Q.; Belongie, S.; Koltun, V.; and Ranftl, R. 2022. Language-driven Semantic Segmentation. In *International Conference on Learning Representations*.

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

Li, Z.; Li, X.; Fu, X.; Zhang, X.; Wang, W.; Chen, S.; and Yang, J. 2024b. Promptkd: Unsupervised prompt distillation for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26617–26626.

Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022. A convnet for the 2020s. In *Proceedings*

of the IEEE/CVF conference on computer vision and pattern recognition, 11976–11986.

Ma, J.; Bai, Y.; Zhong, B.; Zhang, W.; Yao, T.; and Mei, T. 2023. Visualizing and understanding patch interactions in vision transformer. *IEEE Transactions on Neural Networks and Learning Systems*.

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.

Parkhi, O. M.; Vedaldi, A.; Zisserman, A.; and Jawahar, C. 2012. Cats and dogs. In *CVPR*, 3498–3505. IEEE.

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 (ICML)*, 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.

Roy, S.; and Etemad, A. 2024. Consistency-guided Prompt Learning for Vision-Language Models. In *International Conference on Learning Representations, (ICLR)*.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2020. Grad-CAM: visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision (IJCV)*, 128: 336–359.

Shen, S.; Yang, S.; Zhang, T.; Zhai, B.; Gonzalez, J. E.; Keutzer, K.; and Darrell, T. 2024. Multitask vision-language prompt tuning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 5656–5667.

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*.

Tian, X.; Zou, S.; Yang, Z.; and Zhang, J. 2024. ArGue: Attribute-Guided Prompt Tuning for Vision-Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 28578–28587.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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.

Wu, X.; Zhu, F.; Zhao, R.; and Li, H. 2023. Cora: Adapting clip for open-vocabulary detection with region prompting and anchor pre-matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7031–7040.

Xiao, J.; Ehinger, K. A.; Hays, J.; Torralba, A.; and Oliva, A. 2016. Sun database: Exploring a large collection of scene categories. *IJCV*, 119(1): 3–22.

Xin, Y.; Du, J.; Wang, Q.; Yan, K.; and Ding, S. 2024. Mmap: Multi-modal alignment prompt for cross-domain multi-task learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 16076–16084.

Xing, Y.; Wu, Q.; Cheng, D.; Zhang, S.; Liang, G.; Wang, P.; and Zhang, Y. 2023. Dual modality prompt tuning for vision-language pre-trained model. *IEEE Transactions on Multimedia*.

Yu, T.; Lu, Z.; Jin, X.; Chen, Z.; and Wang, X. 2023. Task residual for tuning vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10899–10909.

Zeiler, M. D.; and Fergus, R. 2014. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*, 818–833. Springer.

Zhai, X.; Mustafa, B.; Kolesnikov, A.; and Beyer, L. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11975–11986.

Zhang, R.; Zeng, Z.; Guo, Z.; and Li, Y. 2022. Can language understand depth? In *Proceedings of the 30th ACM International Conference on Multimedia*, 6868–6874.

Zheng, Z.; Wei, J.; Hu, X.; Zhu, H.; and Nevatia, R. 2024. Large Language Models are Good Prompt Learners for Low-Shot Image Classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 28453–28462.

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 (CVPR)*, 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 (IJCV)*, 130(9): 2337–2348.

Zhu, B.; Niu, Y.; Han, Y.; Wu, Y.; and Zhang, H. 2023. Prompt-aligned gradient for prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15659–15669.