

# Mixture of Prompt Learning for Vision Language Models

Yu Du<sup>†</sup>, Tong Niu<sup>†</sup>, Rong Zhao<sup>\*</sup>

Center for Brain-Inspired Computing Research

Department of Precision Instrument, Tsinghua University

{duyu20, nt20}@mails.tsinghua.edu.cn, r\_zhao@tsinghua.edu.cn

<sup>†</sup> Equal contribution, <sup>\*</sup> Corresponding author

## Abstract

As powerful pre-trained vision-language models (VLMs) like CLIP gain prominence, numerous studies have attempted to combine VLMs for downstream tasks. Among these, prompt learning has been validated as an effective method for adapting to new tasks, which only requiring a small number of parameters. However, current prompt learning methods face two challenges: first, a single soft prompt struggles to capture the diverse styles and patterns within a dataset; second, fine-tuning soft prompts is prone to overfitting. To address these challenges, we propose a mixture of soft prompt learning method incorporating a routing module. This module is able to capture a dataset’s varied styles and dynamically selects the most suitable prompts for each instance. Additionally, we introduce a novel gating mechanism to ensure the router selects prompts based on their similarity to hard prompt templates, which both retaining knowledge from hard prompts and improving selection accuracy. We also implement semantically grouped text-level supervision, initializing each soft prompt with the token embeddings of manually designed templates from its group and applied a contrastive loss between the resulted text feature and hard prompt encoded text feature. This supervision ensures that the text features derived from soft prompts remain close to those from their corresponding hard prompts, preserving initial knowledge and mitigating overfitting. Our method has been validated on 11 datasets, demonstrating evident improvements in few-shot learning, domain generalization, and base-to-new generalization scenarios compared to existing baselines. The code will be available at <https://anonymous.4open.science/r/mocoop-6387>

## 1 Introduction

Recently, pre-trained vision-language models like CLIP become increasingly prominent, numerous studies have explored their application in vari-

ous downstream tasks such as image classification (Zhou et al., 2022b), visual question answering (VQA) (Eslami et al., 2021), and cross-modal generation (Crowson et al., 2022). Prompt learning has emerged as an effective method by optimizing the prompts fed into the model, significantly improving performance on new downstream tasks without requiring large-scale fine-tuning of the entire model.

For example, take the downstream task of image classification, the prompt essentially serves as a template that can be positioned before, after, or surrounding the class name. Traditionally, manually designed text templates were used during the training of CLIP, guide the model in associating textual descriptions with visual content. These manually designed prompts are called hard prompts. Prompt learning takes this a step further by replacing these fixed text templates with learnable continuous vectors. By fine-tuning these vectors with a small number of samples, the performance on downstream tasks can be significantly improved. These vector-based prompts are called soft prompts to distinguish them from hard prompts.

We focus on two challenges of soft prompt learning in this work. 1) Dataset style variations. As seen in Figure. 1 For one dataset, a single soft prompt may not be sufficient to capture the diverse styles present in the data. Difference instances in the same dataset may be compatible with different prompts. Therefore, it is more natural to use multiple prompts to represent these variations adequately. 2) Overfitting Issue. Improper finetuning of the soft prompts may result in performance that even lags behind the zero shot capabilities of the original VLMs (Radford et al., 2021; Zhou et al., 2022b). This is related to over-training on base classes and the catastrophic forgetting of domain-general knowledge (Zhu et al., 2023).

To address these challenges, we propose a mixture of soft prompt learning method. This method incorporates a routing module that selects the most

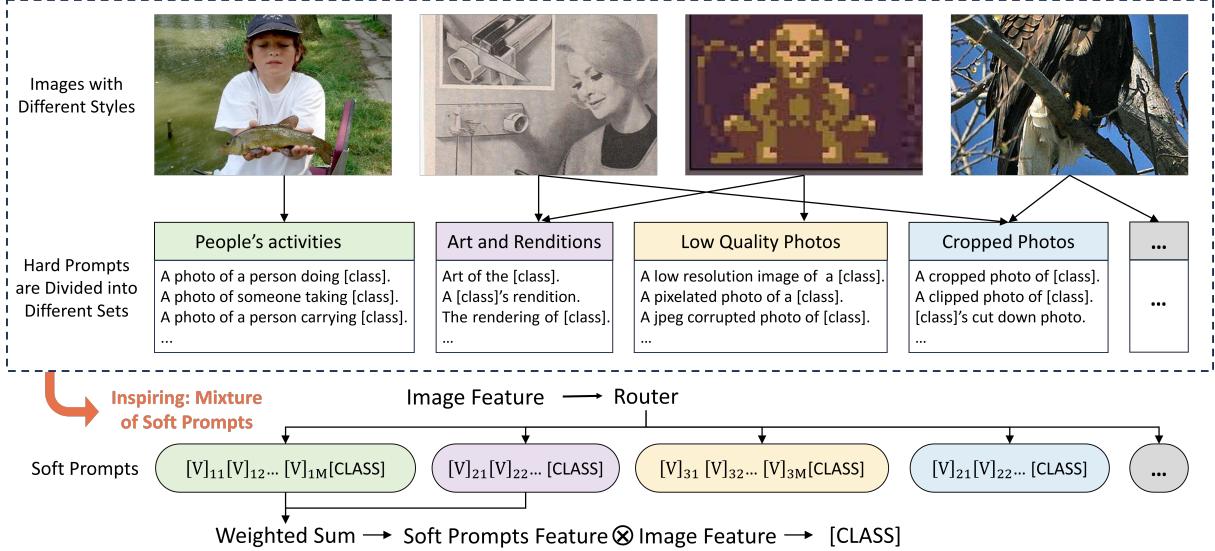


Figure 1: For a dataset, the existing hard templates can be divided into different sets based on the different styles and patterns they describe in the images (such as different contents within the different colored blocks). Furthermore, one image can simultaneously possess multiple different styles. Traditionally, only one soft prompt is used to fit all images, but we use multiple soft prompts. Each soft prompt represents a style, and a router selects the best matches. This approach better bridges the gap between visual and text features by taking different styles into consideration.

suitable prompts for each instance. The selected prompts are then encoded by a text encoder to obtain several sets of class text features. These features are weighted and averaged to produce the final set of class text features, which are then compared with image features to calculate similarities. Conceptually, this process can be deemed as selecting the most compatible style prompts for each instance, thereby enhancing the system’s adaptability and performance.

For the router, we also propose a hard prompt guided gating loss to ensure it selects the soft prompts initialized from the hard prompt templates whose text features are the most similar to the image feature. This mechanism distills the knowledge of hard prompt templates into the router and encourages it to make more accurate and relevant selections.

Additionally, to mitigate the overfitting issue, we introduce semantically grouped text-level supervision. Each soft prompt corresponds to a set of manually designed templates (hard prompts), where the semantics within each set are relatively close. We use the token embeddings of one of the templates from each set as the initialization for each soft prompt. During training, the text features obtained by the text encoder for each soft prompt are constrained to stay close to the text features obtained from their corresponding hard prompts.

This ensures that the initial knowledge from the manual text templates is preserved and integrated into the soft prompts.

We validated our method on 11 datasets, under the few-shot learning, domain generalization and base-to-new generalization from three main aspects. Our methods achieve improvements compared to existing baselines. We also designed ablation experiments to verify the contribution of different modules in our method to the performance improvement.

In summary, our contributions are as follows:

- We propose a mixture of soft prompt learning method that incorporates a routing module to select the most suitable prompts for each instance.
- We introduce a hard prompt guided gating loss to ensure the router selects prompts based on their similarity to hard prompt templates, thus improving selection accuracy.
- We implement semantically grouped text-level supervision to maintain the initial knowledge from manual text templates and mitigate overfitting.
- We validate our method on 11 datasets, demonstrating improvements in few-shot learning, domain generalization, and base-to-new generalization scenarios compared to existing baselines.

## 2 Related Works

**Prompt Learning.** In the realm of vision-language models, prompt learning aims to bridge the gap between visual and textual representations more effectively. A pioneering work in this area is the CoOp (Context Optimization) model (Zhou et al., 2022b), which optimizes the context of prompts to enhance the performance of models like CLIP (Radford et al., 2021) in few-shot learning scenarios.

Researchers have also introduced the concept of a vision prompt (Zang et al., 2022; Khattak et al., 2023), which involves appending learnable vectors to the inputs of a vision encoder, similar to text prompts. This approach can significantly enhance performance, although it also increases computational demands. In this paper, we focus exclusively on text-based prompts. In the future, our methodology could potentially be extended to include vision prompts.

Despite their success, most prompt learning methods trade-off between classification accuracy and robustness, e.g. in domain generalization or out-of-distribution (OOD) detection. A variety of methods have been developed to constrain the update of soft prompts using features from the original manual templates. These methods either directly restrict the gradient update direction or employ knowledge distillation. Among them, Pro-Grad (Zhu et al., 2023) prevents prompt tuning from forgetting general knowledge in VLMs by updating prompts only when their gradients align with the "general direction" represented by the KL loss gradient of a predefined prompt. LASP (Bulat and Tzimiropoulos, 2022) use grouped manual templates encoded feature as supervision to regularize the learning of the prompt. KgCoOp (Yao et al., 2023) reduces the difference between the textual embeddings generated by learned prompts and those from hand-crafted prompts. We also incorporate this technique by distilling the knowledge from original text features into each expert soft prompt. Additionally, we apply gating regularization to distill prior knowledge from discrete text into the router.

PLOT (Chen et al.) first explored to learn multiple comprehensive prompts to describe diverse characteristics of categories, using optimal transport to align visual and textual features. This method improves few-shot recognition tasks by applying a two-stage optimization strategy, demonstrating superior performance across various

datasets compared to conventional prompt learning approaches. We in another way, use multiple prompts to capture the diverse styles in the dataset and learning to prompt in a sparse mixture of experts way.

**Mixture of Experts.** The mixture of experts (MoE) framework (Zhou et al., 2022c; Masoudnia and Ebrahimpour, 2014), initially introduced decades ago, has brought significant advancements for AI, especially with the advent of sparsely-gated MoE in transformer-based large language models (Sukhbaatar et al., 2024; Liu et al., 2024). This framework allows different parts of a model, known as experts, to specialize in various tasks, engaging only relevant experts for a given input to maintain computational efficiency while leveraging specialized knowledge. A major issue of MoE is effectively balancing the load among different expert models, as poor load distribution can result in inefficiencies and unstable model performance (Masoudnia and Ebrahimpour, 2014).

## 3 Method

### 3.1 Overview

As illustrated in Figure 2, during inference, an image is first processed by the CLIP image encoder to obtain an image feature. This feature is then routed to select the  $k$  soft prompts with the highest probabilities. These selected prompts are concatenated with the available classes and fed into the CLIP text encoder, resulting in  $k$  sets of class text features. These  $k$  sets are then averaged weighted by the router's gating distribution (after the softmax layer) to produce a single set of class text features. The final feature set is compared with the image feature to produce the classification logits. In this way, only  $k$  soft prompts are activated at a time, keeping the inference cost comparable to using a single prompt. During training, there are three parts of gradient flow. First, we apply a cross entropy loss to the final classification probabilities with the ground truth label. Second, for the router, we calculate the similarity between the image feature and the text features from each hard prompt template set (using the average feature of all classes and all templates in the set). These similarities serve as a reference distribution. Then, a KL divergence objective function is used to align the router's gating distribution with this reference distribution. Finally, for the soft prompts, we use another cross entropy loss to ensure that each class's text feature from

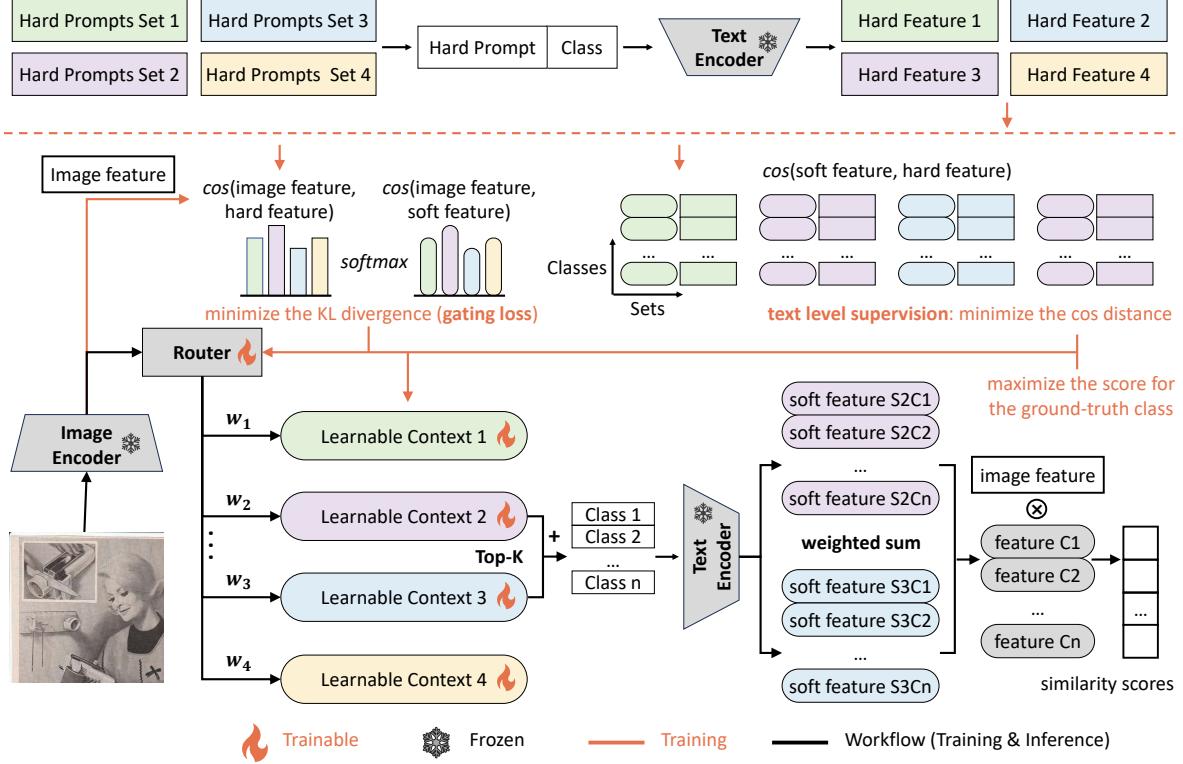


Figure 2: Overview of MoCoOp. The orange lines signify the extra flow for training while the black lines are shared by training and inference. During inference, two soft prompts with the highest probabilities are selected and combined with the available classes for text encoding. The resulting text features are averaged and used for classification. During training, the hard prompt guided routing and semantically grouped text level supervision are introduced to supervise the router and soft prompts respectively. In our experiments, we set k to 2.

each soft prompt closely matches the corresponding class’s feature from the associated hard prompt.

### 3.2 Preliminary of CoOp

Here we give a brief introduction of CoOp (Zhou et al., 2022b), the pioneering work in prompt learning of VLMs.

#### Notation:

First, here are some notations used in prompt learning of VLMs.

- $\mathbf{x}$ : Input image
- $\mathbf{p}$ : Text prompt
- $f_{\text{img}}$ : CLIP image encoder
- $f_{\text{txt}}$ : CLIP text encoder
- $\mathbf{h}_x = f_{\text{img}}(\mathbf{x})$ : Encoded image feature
- $\mathbf{h}_p = f_{\text{txt}}(\mathbf{p})$ : Encoded text feature
- $\mathbf{C}$ : Context vectors (learnable parameters)

**Prompt Representation.** The text prompt  $\mathbf{p}$  is represented as a sequence of tokens, including learnable context tokens and a class token.

$$\mathbf{p} = [\mathbf{C}, \text{CLASS}]$$

The context tokens can also be placed after or around the class token.

#### Context:

- The context is learnable vectors  $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_M]$ , where  $\mathbf{c}_i \in \mathbb{R}^d$  and  $M$  is the number of context tokens.
- All classes share the same context  $\mathbf{C}$  or each class  $c$  has its own context  $\mathbf{C}_c$ .

**Training Objective.** Given a dataset with images  $\{\mathbf{x}_i\}$  and corresponding labels  $\{y_i\}$ , the goal is to find the optimal context vectors  $\mathbf{C}$  (or  $\mathbf{C}_c$  for class-specific context) by minimizing the cross-entropy loss:

$$\mathcal{L} = - \sum_i \log \frac{\exp(\text{sim}(\mathbf{h}_x^i, \mathbf{h}_p^{y_i})/\tau)}{\sum_c \exp(\text{sim}(\mathbf{h}_x^i, \mathbf{h}_p^c)/\tau)}$$

where

- $\mathbf{h}_x^i = f_{\text{img}}(\mathbf{x}_i)$  is the image feature for image  $i$ .
- $\mathbf{h}_p^c = f_{\text{txt}}([\mathbf{C}, \text{CLASS}_c])$  is the text feature for class  $c$ .
- $\text{sim}(\cdot, \cdot)$  denotes a similarity function, such as cosine similarity.
- $\tau$  is the temperature.

**Optimization.** The context vectors  $\mathbf{C}$  are updated through backpropagation to minimize the loss  $\mathcal{L}$ , while keeping the pre-trained parameters of  $f_{\text{img}}$  and  $f_{\text{txt}}$  fixed.

In summary, CoOp involves learning optimal context vectors  $\mathbf{C}$  for text prompts, which are used to synthesize classification weights for downstream tasks. This process automates prompt engineering and enhances the adaptability and performance of vision-language models like CLIP on various image recognition tasks.

### 3.3 Mixture of Prompt Learning

The essential idea of this work is to learn to prompt like mixture of experts. In LLMs, the router selects the top K experts for each input token. Similarly, we use a router to select the top K contexts. Then the selected contexts are concatenated with the class names and encoded by the text encoder to obtain several sets of class features:

$$\mathbf{h}_{p_i} = f_{\text{txt}}([\mathbf{C}_i, \text{CLASS}]) \quad (1)$$

for  $i = 1, 2, \dots, K$ , where  $\mathbf{C}_i$  are the context vectors for the  $i$ -th selected prompt.

The features are then weighted and averaged to produce the final set of class features:

$$\mathbf{h}_p = \sum_{i=1}^K w_{\text{router}}^i \mathbf{h}_{p_i} \quad (2)$$

where  $w_i$  are the weights assigned to each prompt feature. A cross entropy loss is utilized to optimize these prompts:

$$\mathcal{L}_{\text{cls}} = - \sum_i \log \frac{\exp(\cos(\mathbf{h}_x^i, \mathbf{h}_p^{y_i})/\tau)}{\sum_{c \in \mathcal{C}} \exp(\cos(\mathbf{h}_x^i, \mathbf{h}_p^c)/\tau)} \quad (3)$$

### 3.4 Hard Prompt Guided Routing

Given  $G$  sets of hard prompts ( $I_1, I_2, \dots, I_G$ ), each concatenated with every class and encoded through the CLIP text encoder, we obtain  $G$  sets of hard text features for all classes. Specifically, for a hard

prompt concatenated with a specific  $\text{CLASS}_c$ , the corresponding hard text features can be similarly obtained using the CLIP text encoder, resulting in:

$$\mathbf{h}_c = f_{\text{txt}}([\text{hard\_prompt}, \text{CLASS}_c]) \quad (4)$$

where  $c$  denotes the specific class.

These hard text features are then averaged to generate  $G$  group text features, each representing one of the  $G$  groups. Specifically, the group text feature  $\mathbf{h}_g$  for the  $g$ -th group is computed by averaging the hard text features for all classes and all templates within that group as:

$$\mathbf{h}_g = \frac{1}{|I_g|} \sum_{i \in I_g} \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \mathbf{h}_{i,c} \quad (5)$$

where  $\mathcal{C}$  represents the set of all classes, and  $\mathbf{h}_{i,c}$  represents the  $i$ -th hard text feature for class  $c$  in the  $g$ -th group.

The cosine similarity between the image feature  $\mathbf{v}$  and each group's text feature, is calculated. The hard prompt guided gating distribution  $W_{\text{hard}}$  is then derived by applying the softmax function to these similarity scores, expressed as:

$$W_{\text{hard}} = \text{Softmax} \begin{pmatrix} \cos(\mathbf{h}_1, \mathbf{v}) \\ \cos(\mathbf{h}_2, \mathbf{v}) \\ \vdots \\ \cos(\mathbf{h}_G, \mathbf{v}) \end{pmatrix} \quad (6)$$

The router's output gating distribution is denoted by  $W_{\text{router}}$ . To ensure coherence between the two distributions, KL divergence is employed as a constraint, with the loss function defined as:

$$\mathcal{L}_{\text{router}} = D_{\text{KL}}(W_{\text{router}} \parallel W_{\text{hard}}) \quad (7)$$

### 3.5 Semantically Grouped Text Level Supervision

To mitigating the overfitting issue, we introduce semantically grouped text level supervision to alleviating the overfitting issue.

The hard prompts are semantically grouped into  $G$  sets  $I_1, I_2, \dots, I_G$ . (See [A](#) for details). For each learnable soft prompt  $\mathbf{t}_g^s$  and its corresponding hard prompt group  $I_g$ , the probability of a class  $y$  filled in this prompt being classified as its proper class  $y$  is given by:

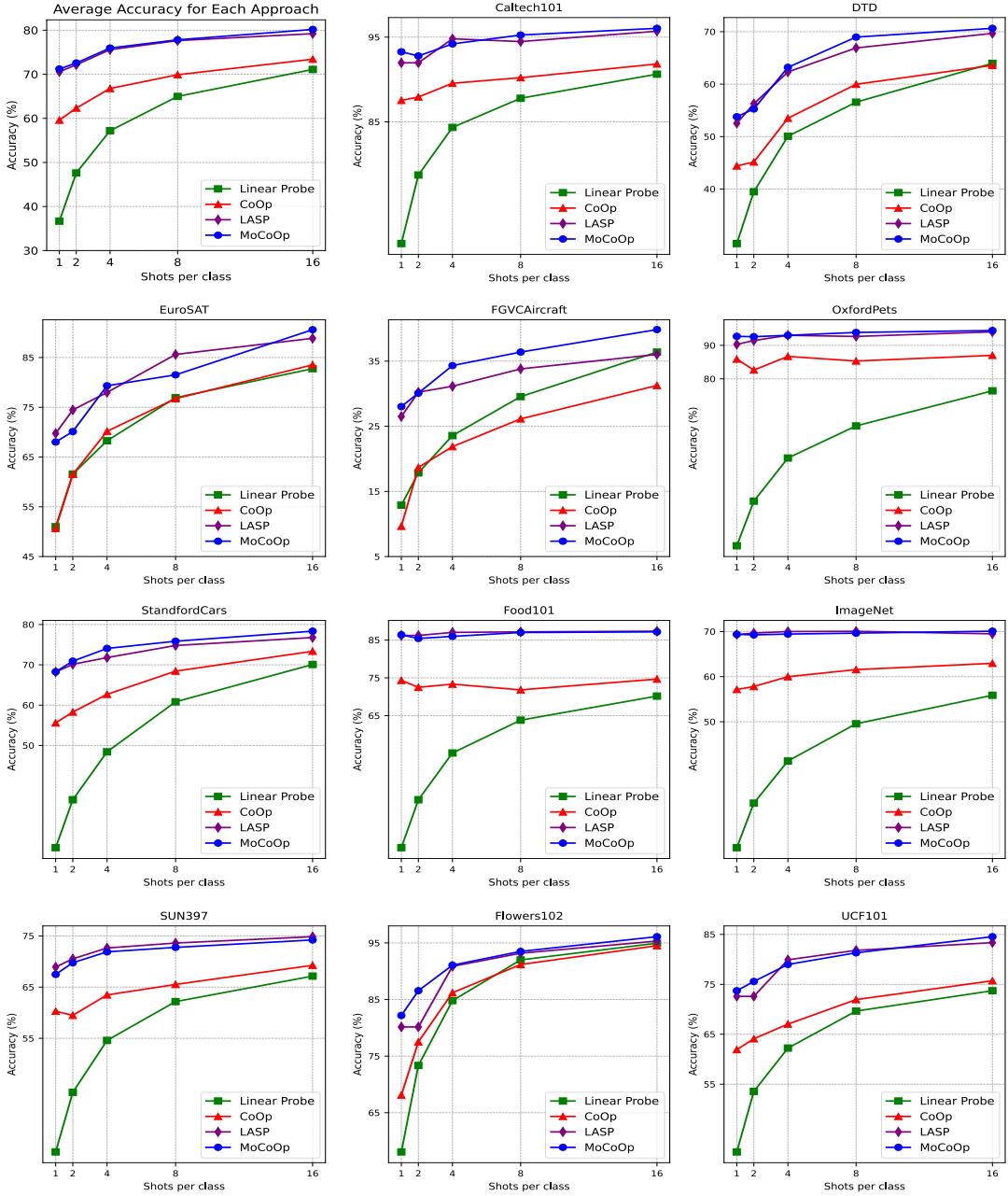


Figure 3: The few-shot learning results on 11 datasets. We plot the results across 1,2,4,8,16 shots. It can be seen that our MoCoOp consistently and significantly surpasses CoOp, LASP, and the Linear Probe approach across most datasets. This is evident in the average accuracy displayed in the top left corner. For LASP (Bulat and Tzimiropoulos, 2022), we use our reproduced results.

$$P(y|\mathbf{t}_g^s) = \frac{1}{|I_g|} \sum_{i \in I_g} P_i(y|\mathbf{t}_g^s)$$

$$P_i(y|\mathbf{t}_g^s) = \frac{\exp(\cos(\mathbf{h}_{i,y}, f_{\text{txt}}([\mathbf{t}_g^s, y])) / \tau)}{\sum_{c \in \mathcal{C}} \exp(\cos(\mathbf{h}_{i,c}, f_{\text{txt}}([\mathbf{t}_g^s, c])) / \tau)} \quad (8)$$

where  $P_i(y|\mathbf{t}_g^s)$  is the possibility of  $\mathbf{t}_g^s$  applied to class  $y$  be classified as the  $i$ -th hard template in

$I_g$  applied to class  $y$ ,  $\cos(\cdot, \cdot)$  denotes the cosine similarity, and  $\tau$  is a temperature parameter,  $\mathcal{C}$  is the class set.

Next, we use the cross-entropy loss to minimize the distance between the encoded learnable soft prompts and the manually defined text prompts in the encoded space. The loss function can be expressed as:

$$\mathcal{L}_{\text{text}} = -\frac{1}{G} \sum_{g=1}^G \sum_{c \in \mathcal{C}} \frac{1}{|\mathcal{C}|} \log P(c | \mathbf{t}_g^s) \quad (9)$$

The overall training objective is

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda_1 \mathcal{L}_{\text{router}} + \lambda_2 \mathcal{L}_{\text{text}} \quad (10)$$

Where  $\lambda_1$  and  $\lambda_2$  are weights that balance the importance of each loss term.

## 4 Experiment

**Settings:** We conduct experiments under three settings: base to new generalization, few-shot learning, and domain generalization. For base to new generalization, we train on the base class and test on both the base class and new class. For few-shot learning, we train and test on all classes. And domain generalization refers to training on ImageNet and testing on other datasets. The few-shot capability reflects the method's fitting ability, while base-to-new generalization and domain generalization can measure the model's robustness.

**Implementation Details:** We build our framework based on LASP (Bulat and Tzimiropoulos, 2022). For each expert, we use different context positions depending on the handcrafted template object used to initialize it. We used 4 to 20 experts. The number of experts and corresponding templates varies for datasets. For example, for FGVC\_Aircraft, we use the template "a photo of a {}, a type of aircraft." For the Oxford\_Flowers dataset, we use "a photo of a {}, a type of flower." Generally, a unique template for the dataset is combined with some general templates like "a photo of a ". Since ImageNet covers a wide range of categories, we use 20 groups of templates. Specific templates can be found in the appendix A. Based on existing studies, we use ViT-B/16 as the backbone. Specifically, we use the publicly available CLIP-ViT-B/16 models (<https://github.com/openai/CLIP>). The resolution of CLIP's feature map is  $14 \times 14$  for CLIP-ViT-B/16. The  $\lambda_1$  and  $\lambda_2$  is set as 1. and 5. respectively. The  $\tau$  in Eq.3 and Eq.8 is set to 0.07. For base-to-new generalization, we use virtual classes during training following LASP (Bulat and Tzimiropoulos, 2022) by incorporating new classes as text-level supervision. This approach helps mitigate overfitting to some extent.

**Evaluation Metrics:** For few-shot experiments, we use top-1 accuracy. For base to new generalization, we evaluate by base class accuracy, new class

accuracy, and the harmonic mean of base and new classes.

**Training:** Our training schedule is consistent with LASP (Bulat and Tzimiropoulos, 2022), and both training and testing are conducted on four NVIDIA GeForce RTX 3090 GPUs.

**Baselines:** In the few-shot experiment, we compared with Linear Probe, CoOp (Zhou et al., 2022b), PLOT (Chen et al.), and LASP (Bulat and Tzimiropoulos, 2022). In the base-to-new generalization experiment, we compare with CoOp (Zhou et al., 2022b), CoCoOp (Zhou et al., 2022a), Kg-CoOp (Yao et al., 2023) and LASP (Bulat and Tzimiropoulos, 2022). Note that CoOp (Zhou et al., 2022b), KgCoOp (Yao et al., 2023), LASP (Bulat and Tzimiropoulos, 2022), PLOT (Chen et al.) are textual only methods while CoCoOp (Zhou et al., 2022a) is instance-conditioned. Textual-only methods typically have poorer generalization to unseen classes within the same task, even lagging behind the original CLIP on some datasets. Instance-conditioned methods improves the generalization by generating different contexts based on various image visual features, and then obtain different text features through the CLIP text encoder. Therefore, they require significant computational resources. Our method, MoCoOp, also partially relies on visual information but does not generate new contexts. Instead, it combines different text features for different images, thus eliminating the heavy computational cost of the text encoder during inference.

**Dataset:** Following previous studies (Zhou et al., 2022b,a; Chen et al.; Yao et al., 2023; Bulat and Tzimiropoulos, 2022), we primarily evaluate the accuracy of our approach across a total of 11 datasets. The datasets used include: ImageNet (Deng et al., 2009), Caltech101 (Fei-Fei et al., 2004), Oxford-Pets (Parkhi et al., 2012), Stanford Cars (Krause et al., 2013), Flowers102 (Nilsback and Zisserman, 2008), Food101 (Bossard et al., 2014), FGVC Aircraft (Maji et al., 2013), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014), EuroSAT (Heller et al., 2019), and UCF-101 (Soomro et al., 2012).

### 4.1 Main Results

Here we show the results of few-shot experiments and base-to-new generalization. The domain generalization results can be found in Appendix. B

Dataset	CLIP			CoOp			CoCoOp			LASP			KgCoOp			MoCoOp (Ours)		
	Base	New	H	Base	New	H	Base	New	H	Base	New	H	Base	New	H	Base	New	H
Average	70.25	74.22	71.57	82.64	68.00	74.02	80.47	71.69	75.42	83.18	76.11	79.48	73.63	76.90	83.22	<b>83.32</b>	<b>77.34</b>	<b>80.17</b>
ImageNet	72.43	68.14	70.22	76.46	66.31	71.00	75.98	70.43	73.11	76.25	71.17	73.62	75.83	69.96	71.89	<b>76.52</b>	69.2	72.67
Caltech101	96.84	94.00	95.40	98.11	93.52	95.76	97.96	93.81	95.84	98.17	94.33	96.21	97.72	94.39	95.55	<b>98.43</b>	<b>94.87</b>	<b>96.61</b>
OxfordPets	91.17	<b>97.26</b>	94.11	94.24	96.66	<b>95.44</b>	95.20	97.69	<b>96.43</b>	95.73	<b>97.87</b>	<b>96.79</b>	94.65	<b>97.76</b>	96.18	95.59	96.64	96.11
StanfordCars	63.37	74.89	68.61	76.2	69.14	72.51	70.49	73.59	72.01	75.23	71.77	73.46	71.76	75.04	73.36	<b>76.34</b>	73.26	<b>74.77</b>
Flowers102	72.08	77.80	74.82	97.63	69.55	81.35	94.87	71.75	81.64	97.17	73.53	83.71	95.00	74.73	83.65	<b>97.18</b>	77.21	<b>86.05</b>
Food101	90.10	91.22	90.66	89.44	87.5	88.46	90.70	<b>91.29</b>	90.99	91.20	<b>91.90</b>	<b>91.54</b>	90.50	<b>91.70</b>	91.10	90.25	91.57	90.90
FGVCAircraft	27.19	36.29	31.07	39.24	30.49	34.23	33.41	23.71	27.73	38.05	33.20	35.46	36.21	33.55	34.83	<b>38.78</b>	<b>38.09</b>	<b>38.43</b>
SUN397	69.36	75.35	72.22	80.85	68.34	74.06	79.74	76.86	78.27	80.70	79.30	<b>80.00</b>	80.29	76.53	78.36	<b>81.43</b>	77.45	79.39
DTD	53.24	59.90	56.36	80.17	47.54	59.67	77.01	56.00	64.85	81.10	62.57	70.64	77.55	54.99	64.35	<b>81.94</b>	60.99	69.93
EuroSAT	56.48	64.05	60.03	91.54	54.44	68.15	87.49	60.04	71.11	<b>95.00</b>	83.37	88.86	<b>95.64</b>	64.34	76.93	94.79	85.18	<b>89.73</b>
UCF101	70.53	77.50	73.82	85.14	64.47	73.57	82.33	73.45	77.63	85.53	78.20	81.70	82.89	76.67	79.66	<b>85.28</b>	79.31	<b>82.17</b>

Table 1: The comparison with baselines on novel class prediction. H is the harmonic mean of the test accuracy on base and new class. The best results are marked in bold font.

	Caltech101			EuroSAT			UCF101			Flowers102		
	Base	New	H	Base	New	H	Base	New	H	Base	New	H
Baseline	95.40	98.11	93.52	91.54	54.44	68.15	85.14	64.47	73.57	97.63	69.55	81.35
+ MoE	98.38	92.03	95.10	94.90	58.79	72.60	85.78	69.50	76.79	97.63	70.64	81.97
+ $L_{\text{router}}$	98.39	92.47	95.34	95.17	57.05	71.34	86.97	73.88	79.89	97.34	72.77	83.28
+ $L_{\text{text}}$	98.43	94.87	96.61	94.79	85.18	89.73	85.28	79.31	82.17	97.18	77.21	86.05

Table 2: Component analysis. We sequentially add the components MoE,  $L_{\text{router}}$  and  $L_{\text{text}}$ . Our baseline is CoOp (Zhou et al., 2022b)

### 4.1.1 Results of Few-shot experiment

In the Figure 3, we plot the performance curves of our MoCoOp and the baselines across 11 datasets for various shots, along with the average accuracies of all datasets. It can be seen that our method achieves the best results in most cases. The performance on ImageNet is average, possibly because other methods utilized all 39 hand-crafted templates, whereas we need to control the number of groups and selected only a portion. Since ImageNet contains images with diverse styles, using only a subset of templates might not have been sufficient.

### 4.1.2 Results of Base-to-New Generalization

In the Table 1, we list the comparison results of MoCoOp and several baselines. It can be seen that our method surpasses the baselines in generalization ability on most datasets, especially compared to LASP (Bulat and Tzimiropoulos, 2022). The improvement can be attributed to the utilization of multiple prompts and the semantically grouped text supervision.

## 4.2 Ablations

### 4.2.1 Component Analysis.

Table.2 presents the performance as we progressively include components. Our baseline is CoOp (Zhou et al., 2022b). As can be seen in Table. 2,

adding MoE alone has already achieved significant improvement. Adding hard prompt guided routing provides a slight improvement, while incorporating semantically grouped text supervision brings a huge enhancement.

## 5 Conclusion

In this work, we introduce a novel mixture of prompt learning method for vision-language models, addressing key challenges such as dataset style variations and overfitting. Our approach employs a routing module to dynamically select the most suitable prompts for each instance, enhancing adaptability and performance. We also propose a hard prompt guided gating loss and semantically grouped text-level supervision, which help maintain initial knowledge and mitigate overfitting. Our method demonstrate significant improvements across multiple datasets in few-shot learning, domain generalization, and base-to-new generalization scenarios. Future work could explore extending this methodology to include vision prompts or instance-conditioned contexts for further enhancements. Another direction could be using ChatGPT for generating and grouping hard prompt templates.

## 6 Limitations

While our MoCoOp demonstrates improvements across various tasks, there are two limitations. First, despite the sparse gating of soft prompts, the training cost and memory usage remain high compared to single prompt methods. This can be a constraint in resource-limited environments, especially when dealing with large-scale datasets. Second, templates require manual grouping based on their semantics, potentially introducing human bias that could affect the model’s performance. To enhance efficiency and accuracy, developing automated grouping algorithms may be necessary in the future.

## References

- Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 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*, pages 446–461. Springer.
- Adrian Bulat and Georgios Tzimiropoulos. 2022. Lasp: Text-to-text optimization for language-aware soft prompting of vision & language models. *arXiv preprint arXiv:2210.01115*.
- Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. Plot: Prompt learning with optimal transport for vision-language models. In *The Eleventh International Conference on Learning Representations*.
- Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613.
- Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. 2022. Vqgan-clip: Open domain image generation and editing with natural language guidance. In *European Conference on Computer Vision*, pages 88–105. Springer.
- Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee.
- Sedigheh Eslami, Gerard de Melo, and Christoph Meinel. 2021. Does clip benefit visual question answering in the medical domain as much as it does in the general domain? *arXiv preprint arXiv:2112.13906*.
- Li Fei-Fei, Rob Fergus, and Pietro Perona. 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*, pages 178–178. IEEE.
- Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 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.
- Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, 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*, pages 8340–8349.
- Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. 2021b. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15262–15271.
- Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19113–19122.
- Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561.
- Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. 2024. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*.
- Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. 2013. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*.
- Saeed Masoudnia and Reza Ebrahimpour. 2014. Mixture of experts: a literature survey. *Artificial Intelligence Review*, 42:275–293.
- Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE.
- Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. 2012. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE.

- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.
- Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. 2019. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR.
- Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.
- Sainbayar Sukhbaatar, Olga Golovneva, Vasu Sharma, Hu Xu, Xi Victoria Lin, Baptiste Rozière, Jacob Kahn, Daniel Li, Wen-tau Yih, Jason Weston, et al. 2024. Branch-train-mix: Mixing expert llms into a mixture-of-experts llm. *arXiv preprint arXiv:2403.07816*.
- Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. 2019. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32.
- Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE.
- Hantao Yao, Rui Zhang, and Changsheng Xu. 2023. Visual-language prompt tuning with knowledge-guided context optimization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6757–6767.
- Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. 2022. Unified vision and language prompt learning. *arXiv preprint arXiv:2210.07225*.
- Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825.
- Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348.
- Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. 2022c. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems*, 35:7103–7114.
- Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. 2023. Prompt-aligned gradient for prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15659–15669.

## A Groups of Hard Prompt Templates

Here is the groups of hard prompt templates.

```
[# Photos of flowers
 "a photo of a {}, a type of flower.",
# Photos of people doing activities
 "a photo of a person doing {}.",
# Satellite photos
 "a centered satellite photo of {}.",
# Photos of aircraft
 "a photo of a {}, a type of aircraft.",
# "Itap" (I took a picture) photos
 "itap of a {}.",
 "itap of the {}.",
# Photos of large objects
 "a photo of the large {}.",
 "a photo of a large {}.",
# Art and renditions
 "art of the {}.",
 "a rendering of a {}.",
 "a rendering of the {}.",
 "a rendition of the {}.",
# Photos of small objects
 "a photo of the small {}.",
# General photo prompts
 "a photo of a {}.",
 "a photo of the {}.",
 "a photo of many {}.",
# Low resolution and pixelated photos
 "a low resolution photo of the {}.",
 "a low resolution photo of a {}.",
 "a pixelated photo of the {}.",
```

```

    "a pixelated photo of a
    {}.",
    "a jpeg corrupted photo
    of the {}.",
    "a blurry photo of a
    {}.",
    "a bad photo of the {}.",
# Cropped photos
    "a cropped photo of the
    {}.",
    "a cropped photo of a
    {}.",
# Bright photos
    "a bright photo of the
    {}.",
# Good quality photos
    "a good photo of the
    {}.",
    "a good photo of a {}.",
# Close-up photos
    "a close-up photo of the
    {}.",
# Jpeg corrupted photos
# Blurry photos
# Clean objects
    "a photo of the clean
    {}.",
# Video game screenshots
    "a {} in a video game.",
# Hard to see objects
    "a photo of the hard to
    see {}.",
# Bad quality photos
# Origami photos
    "a origami {}.",
# Texture photos
    "{} texture.",
]

```

## B Results of Domain Generalization

We evaluate our MoCoOp in the domain generalization setting. This evaluate the robustness to domain shift. We train on 16 shots of ImageNet and test on ImageNet-R (Hendrycks et al., 2021a), ImageNet-A (Hendrycks et al., 2021b), ImageNet-Sketch (Wang et al., 2019), ImageNet-V2 (Recht et al., 2019). As seen in Table 3, the results are comparable with LASP (Bulat and Tzimiropoulos, 2022).

Method	Source		Target			
	ImageNet		-R	-A	-Sketch	-V2
CLIP	66.73	73.96	47.77	46.15	60.83	
CoOp	71.51	75.21	49.71	47.99	64.20	
CoCoOp	71.02	76.18	50.63	48.75	64.07	
ProGrad	72.24	74.58	49.39	47.63	64.73	
KgCoOp	71.20	76.70	50.69	48.97	64.10	
LASP	69.49	75.54	47.08	47.59	62.52	
MoCoOp (Ours)	70.08	75.88	48.97	46.50	61.31	

Table 3: Comparisons on robustness to domain shift. All methods are trained on 16 shots per class of ImageNet and tested on ImageNet-R, ImageNet-A, ImageNet-Sketch and ImageNet-V2. For LASP (Bulat and Tzimiropoulos, 2022), we use our reproduced results.

	K=2			K=3			K=4		
	Base	New	H	Base	New	H	Base	New	H
Caltech101	98.43	94.87	96.61	98.39	94.43	96.37	98.00	94.87	96.41
EuroSAT	94.48	77.02	84.75	94.38	75.0	83.58	94.02	74.36	83.04
UCF101	85.28	79.31	82.17	84.23	68.63	75.63	86.40	79.23	82.66
Flowers102	97.18	77.21	86.05	98.10	71.42	82.66	97.34	76.67	85.78

Table 4: Ablations of the number of selected experts.

## C Additional Ablations

### C.0.1 The number of experts selected by the router

We also show the effect of the number of experts selected on the performance. As seen in Table 4, using top 2 prompts is the best in most cases.