GalLoP: Learning Global and Local Prompts for Vision-Language Models

Marc Lafon, Elias Ramzi, Clément Rambour, Nicolas Audebert, Nicolas Thome

GalLoP: Learning Global and Local Prompts for Vision-Language Models

Marc Lafon*[1], Elias Ramzi*[1], Clément Rambour[1], Nicolas Audebert[1,2], and Nicolas Thome[3]

[1] Conservatoire national des arts et métiers, CEDRIC, F-75141 Paris, France
[2] Univ. Gustave Eiffel, ENSG, IGN, LASTIG, F-94160 Saint-Mandé, France
[3] Sorbonne Université, CNRS, ISIR, F-75005 Paris, France
{marc.lafon, elias.ramzi}@cnam.fr

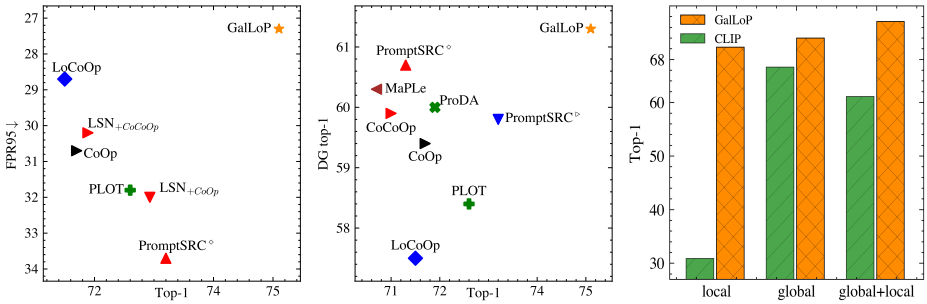
Abstract. Prompt learning has been widely adopted to efficiently adapt vision-language models (VLMs), *e.g.* CLIP, for few-shot image classification. Despite their success, most prompt learning methods trade-off between classification accuracy and robustness, *e.g.* in domain generalization or out-of-distribution (OOD) detection. In this work, we introduce **G**lob**al**-**Lo**cal **P**rompts (GalLoP), a new prompt learning method that learns multiple diverse prompts leveraging both global and local visual features. The training of the local prompts relies on local features with an enhanced vision-text alignment. To focus only on pertinent features, this local alignment is coupled with a sparsity strategy in the selection of the local features. We enforce diversity on the set of prompts using a new "prompt dropout" technique and a multiscale strategy on the local prompts. GalLoP outperforms previous prompt learning methods on accuracy on eleven datasets in different few shots settings and with various backbones. Furthermore, GalLoP shows strong robustness performances in both domain generalization and OOD detection, even outperforming dedicated OOD detection methods. Code and instructions to reproduce our results will be open-sourced.

Keywords: Vision-language models · Few shot classification · Prompt learning · Local and global prompts · Robustness · OOD detection

1 Introduction

Vision-Language Models (VLMs), *e.g.* CLIP [35] or ALIGN [20], have shown impressive performances for zero-shot image classification. Prompt learning [3, 21, 22, 26, 33, 51, 52] has been among the leading approaches to efficiently adapt VLMs to a specific downstream dataset. These methods train a learnable context in the form of *soft prompts* to optimize the text/image alignment. Prompt learning

* Equal contribution.



(a) OOD detection (b) Domain Generalization (c) Global-local

Fig. 1: Our GalLoP method demonstrates excellent performances in accuracy plus robustness, *i.e.* out-of-distribution detection (a) and domain generalization (b), while state-of-the-art prompt learning methods compromise between these aspects. Additionally, unlike recent methods utilizing ineffective local zero-shot CLIP features, GalLoP learns discriminative local prompts precisely aligned with sparse image regions at various scales, facilitating the discriminability between classes. GalLoP integrates both global and local prompts, with their diversity explicitly enforced during few-shot learning, which significantly enhances the performance of their combination (c).

methods benefit from the strong generalization capability of VLMs' textual encoder and are effective even when only a few labeled examples are available.

Despite their success, we observe that these methods trade off between classification accuracy and robustness. This is illustrated on Fig. 1(a), where methods exhibiting the best accuracy sacrifice out-of-distribution (OOD) detection performances, *e.g.* PromptSRC [22], while those excelling in OOD detection often have poor accuracy results, *e.g.* LoCoOp [29]. A similar observation is done in domain generalization, see Fig. 1(b): PromptSRC [22] presents two different versions, one optimized for accuracy (PromptSRC$^{\triangleright}$) and the other for domain generalization (PromptSRC$^{\diamond}$), highlighting the intrinsic conflict between both criteria.

To boost classification accuracy, prompt learning can involve learning multiple prompts [1] to emulate "prompt ensembling", *e.g.* prompts specialized for specific classes [33,45] or Transformer's layers [21,22], or casting multiple prompts learning within a probabilistic framework [26]. The key challenge in prompt ensembling lies in learning diverse prompts to optimize the combination. However, since these approaches only operate on global visual representations, they cannot utilize diverse prompts aligned with specific image regions to maximize their diversity.

Recently, attempts have been made to use local image representations in prompt learning, *e.g.* LoCoOp [29] or PLOT [3]. Although these approaches are promising, their performances in accuracy/robustness are suboptimal compared to state-of-the-art results, see Fig. 1(a),(b). Their limited performances stem from two main factors: i) they use "dense" (*i.e.* all) local features from CLIP, which includes irrelevant or noisy regions for a given concept, and ii) these local features are not as well aligned with the text due to CLIP's pre-training with the global representation. In consequence, the performance of prompts trained with those

local features is much lower than their global counterpart, and this degradation affects performances when combined with global, as illustrated in Fig. 1(c).

In this paper, we introduce **G**lob**al**-**Lo**cal **P**rompts (GalLoP), a new method to learn a diverse set of prompts by leveraging both global and local visual representations. GalLoP learns sparse discriminative local features, *i.e.* text prompts are aligned to a sparse subset of regions at multiple scales. This enables fine-grained and accurate text-to-image matching, making GalLoP local prompts highly competitive. Moreover, we train GalLoP with diverse global and local prompts, unlocking the complementarity between both sets and significantly improving their combination, as shown in Fig. 1(c).

To achieve this, GalLoP relies on two main methodological contributions:

- **Effective local prompts learning.** In GalLoP, we propose to align local prompts with sparse subsets of k image regions, enabling text-to-image matching that captures fine-grained semantics. To adapt visual representations to the downstream dataset, we refine the textual alignment of visual local features by employing a simple linear projection amenable to few-shots learning.
- **Enforcing ensemble diversity.** We learn both global prompts aligned with the whole image and local spatially-localized prompts, and enforce diversity between them to improve their combination. We induce diversity through randomization using a new "prompt dropout" strategy, which enhances generalization when learning multiple prompts. Additionally, we employ a multiscale strategy to align local prompts with image regions of varying sizes, capturing different visual aspects of a concept's semantics.

We conduct an extensive experimental validation of GalLoP on 11 few-shot image classification datasets and 8 datasets evaluating robustness. We show that GalLoP outperforms state-of-the-art prompt learning methods on classification accuracy, OOD detection, and domain generalization, therefore improving the observed tradeoff in these 3 criteria. We validate that our two main contributions, *i.e.* learning strong local prompts and diverse representations, are essential for reaching excellent performances.

2 Related work

Prompt learning. Prompt learning has emerged as an efficient way to adapt VLMs to downstream datasets. These methods, *e.g.* CoOp [52], learn *soft prompts* to adapt CLIP textual features to specific labels without the need for a cumbersome step of "prompt engineering" as performed in [35]. Following these seminal works, many variants have been proposed. [51] uses a meta-network to bias the learnable prompt using the global visual representation of the input image. To boost prompt learning performances, recent works have focused on learning multiple prompts [3,21,22,26]. MaPLe [21] introduces prompts in several layers of both textual and visual encoders. PromptSRC [22] builds upon this work by introducing several regularization losses, boosting both accuracy and robustness performances. We note that PromptSRC uses a set of hand-crafted prompts to regularize the learning of the textual prompts,

which is not fully aligned with the initial motivation behind prompt learning. Furthermore, both MaPLe and PromptSRC are limited to the use of vision transformer architectures. ProDA [26] models the distribution over the textual representation of classes using a multivariate Gaussian distribution, and indirectly learns the distribution over prompts using a surrogate loss. PromptStyler [4] learns several prompts that represent different "styles" to perform source-free domain generalization. These two approaches achieve prompt diversity by enforcing orthogonality among the prompts. In GalLoP, we induce diversity with a "prompt dropout" technique, which randomly drops subsets of prompts during training, thus avoiding the introduction of an additional loss, while limiting prompt over-fitting observed in [22] with a method inspired by a standard deep learning approach, *i.e.* Dropout [40]. To further improve diversity, we specialize the local prompts on different image scales, thus aligning them with different sets of attributes for each class.

Prompt learning using visual local features. There has been a growing interest in leveraging CLIP's local features in prompt learning methods [3, 29, 41]. PLOT [3] learns a set of prompts by using the optimal transport (OT) [44] distance between them and the set of local features, which is prohibitive to compute. Furthermore, the OT distance enforces the prompts to use information from *all* local visual features during training, including possibly detrimental ones. Also, PLOT adds the global visual features to the local features to achieve strong results on the ImageNet dataset. In GalLoP, we use a sparse mechanism to learn localized prompts. This removes the negative influence of background features while being computationally efficient. Finally, GalLoP learns prompts from the local features without any access to CLIP's original global visual feature. LoCoOp [29] introduced an entropy loss leveraging "irrelevant" local visual features in an outlier exposure fashion [18] to improve out-of-distribution detection but at the expense of accuracy. [41] introduces a method specifically designed for multi-label classification, which learns prompts using local visual features. While these methods obtained promising results, we show in this work that their performance is intrinsically limited by the lower discriminative power of CLIP's zero-shot local visual features.

Prompt learning for OOD detection. As VLMs are becoming increasingly prevalent in few-shot classification applications, their zero-shot OOD detection capabilities have received increasing attention. In the seminal work [28], the authors proposed the maximum concept matching (MCM) score to detect OOD examples. Recently, [30] improved upon the MCM score by combining zero-shots global and local visual information to construct the GL-MCM score, which achieves strong zero-shots OOD detection results. In addition to the previously mentioned LoCoOp [29], other works tackle the few-shots OOD detection problem using prompt learning. To avoid degraded accuracy performances, [31] introduces the concept of negative prompts to perform OOD detection by "Learning to Say No" (LSN). OOD samples are then detected by computing the difference in MCM scores between positive and negative prompts. GalLoP already achieves strong OOD detection performances without introducing an extra loss or additional negative prompts.

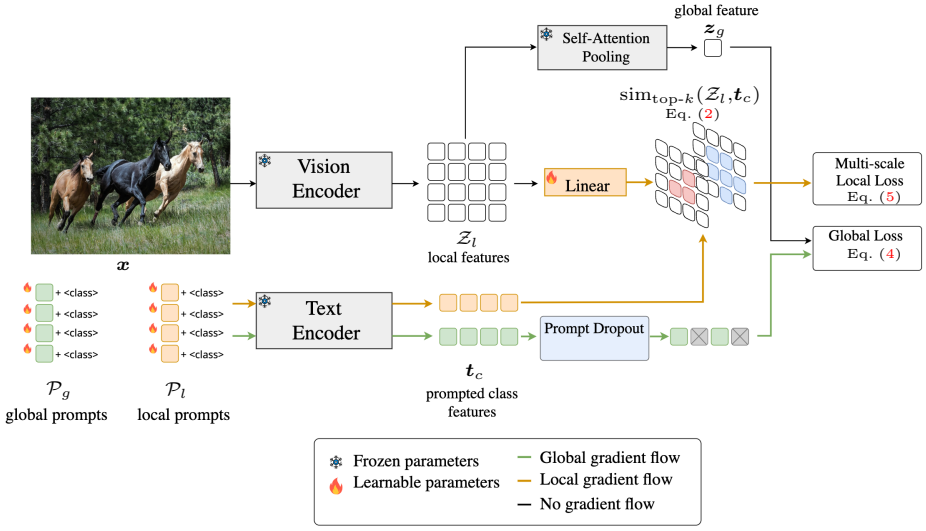


Fig. 2: Illustration of GalLoP. GalLoP learns a diverse set of global prompts and local prompts. Pertinent local prompts are learned using only the most relevant regions of the image for each class. We further improve the limited text-vision alignment of CLIP's local features using a simple linear layer. The diversity is encouraged using a new "prompt dropout" technique for global prompts, and a multiscale loss for local prompts.

Indeed, GalLoP has strong classification accuracy, compared to zero-shot CLIP used in [28] or LoCoOp [29], which helps OOD detection performances. Similarly, our discriminative local features further increase the gain of using local features for OOD detection observed in GL-MCM [30].

3 Combining global and local prompts with GalLoP

In this section, we describe our proposed method, GalLoP, which seeks to learn an ensemble of diverse prompts from both global and local CLIP's visual representations. As illustrated in Fig. 2, GalLoP learns two specialized sets of prompts: the "global prompts" receiving a signal from the global visual representation, and the "local prompts" trained using local features only.

Formally, let us consider a set of n learnable local prompts $\mathcal{P}_l = (\boldsymbol{p}_1^l, \cdots, \boldsymbol{p}_n^l)$ and a set of m learnable global prompts $\mathcal{P}_g = (\boldsymbol{p}_1^g, \cdots, \boldsymbol{p}_m^g)$. Each of these prompt $\boldsymbol{p}$ is composed of V learnable embeddings, *i.e.* $\boldsymbol{p} := [p^1, ..., p^V] \in \mathbb{R}^{V \times d'}$, and are prepended to the class name embeddings $\boldsymbol{c}$ to perform classification. Let $\mathcal{D} = \{(\boldsymbol{x}, y)\}$ denote the downstream dataset, where $\boldsymbol{x}$ is an image and y its class, and let $\mathcal{T}$ and $\mathcal{V}$ denote CLIP's text and vision encoder, respectively. The textual encoder produces a normalized textual representation $\boldsymbol{t}_c = \mathcal{T}([\boldsymbol{p}, \boldsymbol{c}]) \in \mathbb{R}^d$ of the c^{th} class. Given the input image $\boldsymbol{x}$, the visual encoder produces a visual representation $\boldsymbol{z}$. $\boldsymbol{z}$ can be a global vector for learning global prompts, *i.e.* the global visual feature on which

CLIP has been pre-trained. For local prompts, $\boldsymbol{z}$ will be a set of localized features outputted by the encoder. From its visual representation $\boldsymbol{z}$, the probability for the image $\boldsymbol{x}$ to be classified into the class y_c can be expressed as:

$$p(y=y_c|\boldsymbol{x};\boldsymbol{p}) = \frac{\exp(\text{sim}(\boldsymbol{z}, \boldsymbol{t}_c) \;/\; \tau)}{\sum_{c'} \exp(\text{sim}(\boldsymbol{z}, \boldsymbol{t}_{c'}) \;/\; \tau)}, \tag{1}$$

where $\text{sim}(\cdot,\cdot)$ is a measure of similarity, and τ is fixed a temperature scaling parameter. With this general definition of the probability in Eq. (1), we can train a prompt $\boldsymbol{p}$ using the standard cross-entropy loss $\mathcal{L}_{\text{CE}}(p(y=y_c|\boldsymbol{x};\boldsymbol{p}))$.

To train a global prompt, $\boldsymbol{p}_i^g \in \mathcal{P}_g$, we use the global visual representation for the image $\boldsymbol{x}$, *i.e.* $\boldsymbol{z} = \boldsymbol{z}_g \in \mathbb{R}^d$. The similarity between the global vector $\boldsymbol{z}_g$ and the prompt simply reduces to cosine similarity, *i.e.* $\text{sim}(\boldsymbol{z}_g, \boldsymbol{t}_c) = \langle \boldsymbol{z}_g, \boldsymbol{t}_c \rangle$, and the global prompt $\boldsymbol{p}^i$ can be trained by minimizing $\mathcal{L}_{\text{CE}}(p(y=y_c|\boldsymbol{x};\boldsymbol{p}_i^g))$.

In Sec. 3.1, we introduce a relevant similarity measure $\text{sim}(\boldsymbol{z}, \boldsymbol{t}_c)$ for implementing Eq. (1) on local prompts. We rely on a sparsification strategy that only considers a small subset of class-relevant regions of the image. Furthermore, we use a linear projection to improve the vision-text alignment of local features, thus enhancing the quality of the learned prompts. In Sec. 3.2 we describe how we learn a diverse set of global and local prompts, whose combination can improve predictions' performance. We introduce "prompt dropout" to increase the diversity of global prompts by randomly selecting a subset of prompts for each image. Finally, we introduce a multiscale loss by dedicating each local prompt to select different sub-region sizes of the input image.

3.1 Learning prompts from local visual representations

In this section, we temporarily consider a single local prompt $\boldsymbol{p}_j^l \in \mathcal{P}_l$ without loss of generality. In this case, the visual representation $\boldsymbol{z}$ that we consider is the set of visual local features, *i.e.* $\boldsymbol{z} = \mathcal{Z}_l \in \mathbb{R}^{L \times d}$, obtained following [7] (see details in supplementary A.1). Here, we can not directly compute the probability of Eq. (1) as we need to define the similarity between the set of vectors $\mathcal{Z}_l = (\boldsymbol{z}_1^l, \cdots, \boldsymbol{z}_L^l)$ and the textual representation of the c^{th} class, $\boldsymbol{t}_c = \mathcal{T}([\boldsymbol{p}_j^l, \boldsymbol{c}])$.

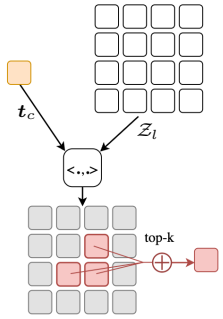


Fig. 3: GalLoP sparse local similarity $\text{sim}(\mathcal{Z}_l, \boldsymbol{t}_c)$ between class prompt $\boldsymbol{t}_c$ and visual features $\mathcal{Z}_l$ is the average of the top-k highest similarities (here, k=3).

Sparse local similarity. A naive way to obtain a single similarity for all regions is to average the similarities of each spatial location with the textual representation of the class. However, a substantial portion of the local features are irrelevant to the class, *e.g.* features from background areas, which may introduce noise and perturb the learning process. To solve this problem, we adopt a sparse approach, where only local features semantically

related to the class are kept to perform classification. As illustrated in Fig. 3, we select the top-k local features with the highest similarities with the prompted class textual representation, and average their similarities to measure $\text{sim}(\mathcal{Z}_l, \boldsymbol{t}_c)$.

Formally, we define the similarity between a prompt $\boldsymbol{t}_c$ and the set of visual features $\mathcal{Z}_l$ as the average similarity for the k most similar regions:

$$\text{sim}_{\text{top-}k}(\mathcal{Z}_l, \boldsymbol{t}_c) \coloneqq \frac{1}{k} \sum_{i=1}^{L} \mathbb{1}_{\text{top-}k}(i) \cdot \langle \boldsymbol{z}_i^l, \boldsymbol{t}_c \rangle \tag{2}$$

$$\text{where} \qquad \mathbb{1}_{\text{top-}k}(i) = \begin{cases} 1 & \text{if} \quad \text{rank}_i(\langle \boldsymbol{z}_i^l, \boldsymbol{t}_c \rangle) \leq k, \\ 0 & \text{otherwise.} \end{cases}$$

which we plug into Eq. (1) to compute the probability for class c. We show in Sec. 4.3 that relying on sparsity is mandatory for local prompt learning, boosting performances by almost 20pt in top-1 accuracy.

Improving local text-vision alignment. While previous works [29, 41, 50] have exploited the text-vision alignment of CLIP's local features, we empirically verified in Sec. 4.3 that using these features leads to poor zero-shots classification results on ImageNet. This is expected, as CLIP is pre-trained to align the global visual features with its textual representation. Local features are thus suboptimal to learn effective prompts for image classification. Motivated by this observation, we propose to improve the discriminative power of CLIP's local visual features by realigning them with the textual representations of the class labels of the downstream dataset. To do so, we propose to use a simple linear projection $h_{\boldsymbol{\theta}}$. To ease the learning process, we initialize the linear layer h_θ to identity, so that the initial features are close to CLIP's representations. Henceforth, we use the set of linearly transformed local visual features $h_{\boldsymbol{\theta}}(\mathcal{Z}_l)$ to compute the probability of Eq. (1), which becomes:

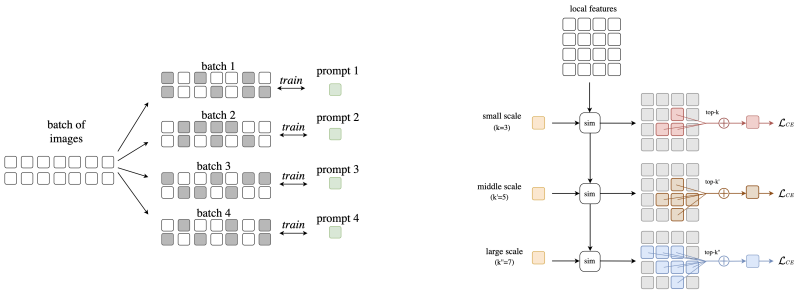
$$p(y = y_c | \boldsymbol{x}; \boldsymbol{p}_j^l, k, \boldsymbol{\theta}) = \frac{\exp(\text{sim}_{\text{top-}k}(h_{\boldsymbol{\theta}}(\mathcal{Z}_l), \boldsymbol{t}_c) \,/\, \tau)}{\sum_{c'} \exp(\text{sim}_{\text{top-}k}(h_{\boldsymbol{\theta}}(\mathcal{Z}_l), \boldsymbol{t}_{c'}) \,/\, \tau)}. \tag{3}$$

Thus, a local prompt can be optimized by maximizing this probability with the cross-entropy loss. These design choices in GalLoP allow us to train a powerful classifier for local features: the sparsity helps to focus on the most relevant regions of an image and to remove potential background noise, while the linear projection enhances the text-vision alignment and boosts the fine-grained discriminating power of the local features. We study these design choices in Sec. 4.3.

3.2 Learning multiple diverse prompts

In this section, we describe how we induce diversity among the learned prompts. Besides exploiting different sources of information – the global and visual ones –, we introduce two mechanisms to increase diversity: "prompt dropout" and multiscale training.

Prompt dropout. Motivated by the success of the "dropout" [10, 40] technique classically used in deep learning, we introduce "prompt dropout" into the prompt



(a) Prompt dropout (b) Multiscale loss

Fig. 4: (a) Prompt dropout induces diversity by randomly selecting different subsets of prompts for each image of the batch. In (a), each image will be used by half the prompts. (b) To learn diverse local prompts, we specialize each one of them using a different number of regions, and therefore a different level of sparsity.

learning framework. In "prompt dropout", we randomly mask a subset of prompts for each image of the batch. Alternatively, from the perspective of each prompt, we select a different subset of the batch of images, thus inducing diversity in the learning process of the prompts through input randomization.

The training of our set of global prompts is performed with the following loss:

$$\mathcal{L}_{\text{global}}(\mathcal{P}_g) = \sum_{i=1}^{m} \mathcal{L}_{\text{CE}}(\boldsymbol{p}_i^g). \tag{4}$$

Multiscale training. To specifically improve the diversity of the local prompts, we specialize each local prompt to select a different number of class-specific visual patches (scales). In this way, prompts dedicated to small scales will get more signals from classes corresponding to small visual concepts, *e.g.* "daisy flower" or "tailed frog", while prompts learned with larger scales will receive more signals from images with wider concepts, *e.g.* "castle" or "valley". More formally, let $(k_1, k_1+\Delta_k, \cdots, k_1+(n-1)\cdot\Delta_k)$ denote a set of increasing scales with k_1 the first scale and Δ_k the expansion factor. Each local prompt $\boldsymbol{p}_j^l$ will be learned with its associated scale $k_j = k_1+(j-1)\cdot\Delta_k$.

The training of our n local prompts is then performed by optimizing the probability defined in Eq. (3) for each prompt with a different scale, *i.e.* value of k:

$$\mathcal{L}_{\text{x-scale}}(\mathcal{P}_l, \boldsymbol{\theta}) = \sum_{j=1}^{n} \mathcal{L}_{\text{CE}}(\boldsymbol{p}_j^l, \boldsymbol{\theta}, k_j). \tag{5}$$

The overall loss to train our set of prompts $\mathcal{P} = \mathcal{P}_l \cup \mathcal{P}_g$ is the sum of the local multiscale and global losses:

$$\mathcal{L}_{\text{total}}(\mathcal{P}, \boldsymbol{\theta}) = \mathcal{L}_{\text{global}}(\mathcal{P}_g) + \mathcal{L}_{\text{x-scale}}(\mathcal{P}_l, \boldsymbol{\theta}) \tag{6}$$

4 Experimental results

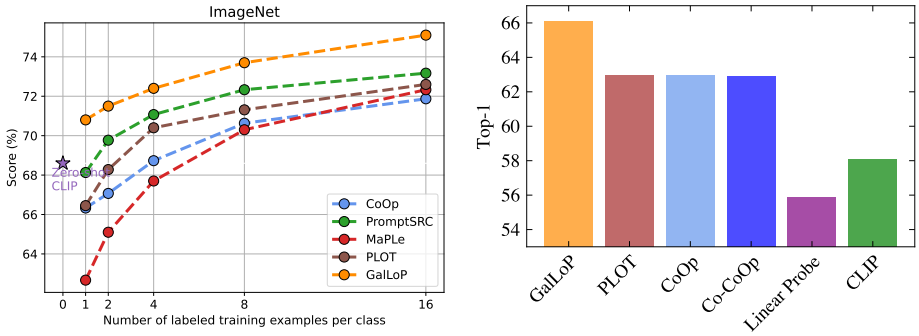
In this section, we present the experimental validation of GalLoP. We first show that GalLoP outperforms previous methods on top-1 accuracy on a collection of 11 datasets in Sec. 4.1 with ViT-B/16 [8]. We also show that GalLoP performs well for different few shot settings on ImageNet and with a ResNet-50 [14]. In Sec. 4.2, we compare robustness performances of GalLoP and other prompts learning methods in domain generalization and OOD detection, and show that GalLoP has better trade-off with top-1 accuracy contrary to previous methods. In Sec. 4.3, we conduct ablation studies of the different components of GalLoP.

Implementation details. We experiment with both ResNet-50 and ViT-B/16 CLIP models. When not specified, we use ViT-B/16. We train for 50 epochs on ImageNet and 200 epochs for other datasets with SGD, a learning rate of 0.002 decayed using cosine annealing and a weight decay of 0.01, following the setting of [52]. Unless specified otherwise, we train the models using 16 shots. Our base parameters for GalLoP are as follows: $m=4$ global prompts with a dropout of 75% (in practice we keep a single prompt for each image), $n=4$ local prompts with scales $k_1=10$ and $\Delta_k=10$ for ViT-B/16 and $k_1=5$ and $\Delta_k=5$ for ResNet-50 as there are fewer local patches. We keep τ fixed from CLIP.

Baselines. We compare GalLoP to recent prompt learning methods. Including, single prompt learning CoOp and Co-CoOp. Multi-prompt learning MaPLe, ProDA, PLOT, PromptSRC. We denote by PromptSRC$^{\triangleright}$ the version designed for accuracy and PromptSRC$^{\diamond}$ the version designed for domain generalization. We also include OOD detection specific methods such as LoCoOp and LSN.

Table 1: Top-1 accuracy with ViT-B/16 backbone. Comparison of GalLoP to other prompt learning methods on several standard benchmarks. †results based on our own re-implementation.

Dataset	*ImageNet* [6]	*Caltech101* [9]	*OxfordPets* [34]	*Cars* [23]	*Flowers102* [32]	*Food101* [2]	*Aircraft* [27]	*SUN397* [47]	*DTD* [5]	*EuroSAT* [15]	*UCF101* [39]	*Average*
CLIP [35]	66.7	92.2	88.4	65.5	70.7	84.8	24.8	62.3	44.1	48.3	64.7	75.7
Linear Probe	67.3	95.4	85.3	80.4	97.4	82.9	45.4	73.3	70.0	87.2	82.1	78.8
CoOp [52]	71.7	95.6	91.9	83.1	97.1	84.2	43.4	74.7	69.9	84.9	82.2	79.9
Co-CoOp [51]	71.0	95.2	93.3	71.6	87.8	**87.2**	31.2	72.2	63.0	73.3	78.1	74.9
MaPLe [21]	72.3	96.0	92.8	83.6	97.0	85.3	48.4	75.5	71.3	**92.3**	85.0	81.8
PLOT [3]	72.6	96.0	93.6	84.6	97.6	**87.1**	46.7	76.0	71.4	92.0	85.3	82.1
PromptSRC$^{\triangleright}$ [22]	73.2	96.1	93.7	85.8	97.6	86.5	50.8	**77.2**	72.7	**92.4**	86.5	82.9
LoCoOp† [29]	71.5	94.9	92.4	79.8	96.3	84.7	40.7	74.2	69.5	86.1	81.6	79.2
ProDA† [26]	71.9	95.5	93.5	79.8	96.8	86.8	40.2	75.7	70.9	85.1	83.3	80.0
GalLoP	**75.1**	**96.7**	**94.1**	**89.2**	**98.8**	86.5	**58.3**	**77.2**	**75.5**	90.1	**86.9**	**84.4**



(a) Few shot results.

(b) Results with ResNet-50.

Fig. 5: Results on ImageNet with different few shot settings Fig. 5a, and ResNet-50 Fig. 5b.

4.1 Main in-distribution results.

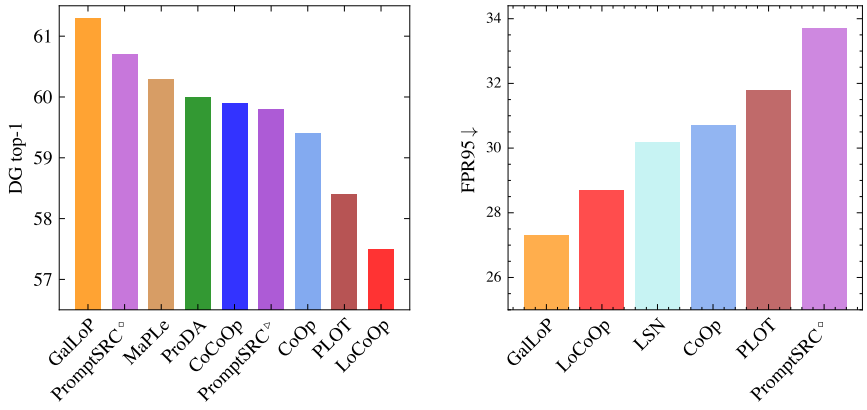
On Tab. 1, we compare GalLoP with a ViT-B/16 backbone on a suite of 11 datasets, a standard benchmark for prompt learning methods. On average, GalLoP outperforms previous methods by a large margin with +1.5pt compared to PromptSRC$^\triangleright$ the next best performing method. Furthermore, GalLoP performs well on most datasets, achieving state-of-the-art among prompt learning methods. For instance, on the large-scale ImageNet dataset, it outperforms PLOT by +2.5pt and PromptSRC$^\triangleright$ by +1.9pt. On some datasets, *e.g.* FGVC Aircraft, GalLoP outperforms the next best method by a large margin, with +7.5pt compared to PromptSRC$^\triangleright$.

We then compare GalLoP on Fig. 5a to prompt learning methods in different few-shot settings on ImageNet. GalLoP performs well in all configurations, outperforming for each setting the very competitive method, PromptSRC$^\triangleright$. Finally, in Fig. 5b we show that GalLoP works well with a ResNet-50, outperforming PLOT and CoOp by +3.1pt. Note that compared to other methods, *e.g.* MaPLe and PromptSRC, GalLoP is amenable to both convolutional and transformer vision backbones. Detailed results for ResNet-50 can be found in the supplementary material B.2.

4.2 Robustness results.

In this section, we compare the robustness performances of GalLoP *vs*. other prompt learning methods, see Fig. 6, on domain generalization and OOD detection. For both benchmarks, models are trained on ImageNet (16 shots).

Domain generalization results. We compare on Fig. 6a the domain generalization performances of GalLoP *vs*. other prompt learning methods. After being trained on ImageNet (16 shots), the models are evaluated on top-1 accuracy for different domains with the same classes as ImageNet, *i.e.* ImageNet-V2 [36], ImageNet-Sketch [46], ImageNet-A [19] and ImageNet-R [16]. GalLoP outperforms the domain-generalization specific method PromptSRC$^\diamond$ by +0.5pt on average,



(a) Domain generalization. **(b)** OOD detection.

Fig. 6: GalLoP robustness performances. GalLoP achieves strong performances on domain generalization Fig. 6a and on OOD detection Fig. 6b, while outperforming prompt learning methods on top-1 accuracy.

while outperforming it by +4.9pt on ImageNet. This illustrates the trade-off made by PromptSRC between top-1 accuracy and domain generalization. Indeed, GalLoP outperforms PromptSRC$^{\triangleright}$, designed for ImageNet accuracy, by +1.9pt on ImageNet and +1.5pt on average in domain generalization. GalLoP achieves the best trade-off between top-1 performances and domain generalization. The detailed results can be found in supplementary material B.4.

Results on OOD detection. In OOD detection the models must recognize between in-distribution examples (ImageNet test set) and different OOD datasets, namely iNaturalist [43], SUN [47], Places [49] and Textures [5], a standard benchmark in the OOD detection literature. We plot on Fig. 6b the average results on the ImageNet OOD benchmark of GalLoP and other prompt learning methods measured in FPR95 (lower is better, ↓). GalLoP outperforms traditional prompt learning methods, *e.g.* CoOp -3pt FPR95, as well as dedicated OOD detection methods, *e.g.*-1.4pt FPR95 *vs*. LoCoOp or -2.9pt FPR95 *vs*. LSN. Meanwhile, GalLoP also outperforms both LSN and LoCoOp by a large margin in top-1 accuracy, *i.e.* +3.2pt and +3.6pt respectively. The detailed results can be found in the supplementary material B.5.

4.3 Ablation studies.

In this section, we investigate the design choices for GalLoP. We first show how GalLoP leverages the complementarity of strong global and local prompts to boost performances Tab. 2. We then demonstrate the benefit of sparsity and local alignment in Fig. 7. Finally, we show the impact of our choice when learning multiple prompts for both global and local features Fig. 8.

Combining global and local features. On Tab. 2, we show that leveraging global and local features requires some important design choices. Indeed, we experiment with a baseline using CoOp on local features ("$\text{CoOp}_{\text{Local}}$"), learning a single prompt, without sparsity and no alignment. This baseline already outperforms using zero-shot local features, +28.7pt top-1. However, its combination with a standard $\text{CoOp}_{\text{Global}}$, *i.e.* "$\text{CoOp}_{\text{GL}}$", is detrimental to final top-1 performances, with -1.9pt top-1 or -3.6pt DG compared to $\text{CoOp}_{\text{Global}}$. On the other hand, GalLoP enjoys a boost in performances on all metrics when combining the learned global ($\text{GalLoP}_{\text{Global}}$) and local ($\text{GalLoP}_{\text{Local}}$) prompts. We can see that the top-1 performances of GalLoP increase by +3.1pt compared to ($\text{GalLoP}_{\text{Global}}$). Similarly, on OOD detection, GalLoP has a decrease of -8.9pt FPR95 compared to $\text{GalLoP}_{\text{Local}}$. Tab. 2 illustrates how the resulting performances of GalLoP, in both accuracy and robustness, come from the complementarity of both the local and global features.

Table 2: Ablation studies for the different components of our GalLoP.

	Top-1	DG	FPR95↓	AUC
$\text{CLIP}_{\text{Global}}$	66.6	57.2	42.8	90.8
$\text{CLIP}_{\text{Local}}$	12.5	9.49	73.3	73.7
CLIP_{GL}	61.1	49.3	35.5	90.8
$\text{CoOp}_{\text{Global}}$	71.4	59.2	39.1	91.1
$\text{CoOp}_{\text{Local}}$	41.2	30.1	65.2	78.3
CoOp_{GL}	69.5	55.6	33.7	90.5
$\text{GalLoP}_{\text{Global}}$	72.0	60.4	37.0	91.7
$\text{GalLoP}_{\text{Local}}$	70.9	54.1	36.0	90.1
GalLoP	**75.1**	**61.3**	**27.3**	**93.2**

The need for sparsity. In Fig. 7 we show how the sparsity when using local features allows achieving higher performances than attending to each local feature, for three regimes: zero-shot CLIP ("zero-shot"), while learning a local prompt ("w/o linear"), and when aligning a local prompt and our linear projection ("w. linear"). On the three regimes, the difference between looking at all local features and the best reported sparsity level is, respectively, +18.4pt, +17.6pt, and +8.5pt. Furthermore, we can see that when aligning a local prompt and the linear layer, our sparsity ratio works for a wide range of k, with performances above 69pt between $k=5$ and $k=50$. This shows the robustness to the choice of k. Finally, learning a local prompt allows to significantly boost the performances for the local features, *e.g.* +27.9pt for $k=10$, and aligning with a linear projection further boosts performances, with +10pt for $k=10$ compared to learning the prompt only. Fig. 7 shows the interest of both enforcing the sparsity when looking at local features and further aligning the local features with a local prompt.

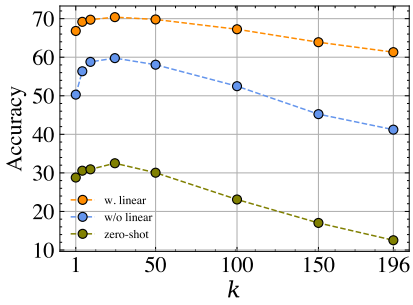
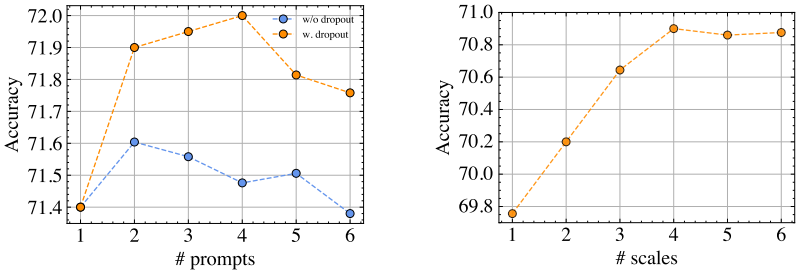


Fig. 7: Impact of our sparsity choice for three regimes, zero-shot CLIP, learning a local prompt ("w/o linear") and aligning our linear projection with a local prompt ("w. linear").



(a) Impact of prompt dropout when learning multiple global prompts.

(b) Impact of multiple scales when learning local prompts, with $k_1 = 10$, $\Delta_k = 10$.

Fig. 8: Impact of our design choices on learning global Fig. 8a and local prompts Fig. 8b .

Global prompt learning with prompt dropout. We display on Fig. 8a how prompt dropout allows learning efficiently multiple prompts for the global features. We display the top-1 accuracy when using more and more prompts, with ("w.") or without ("w/o") prompt dropout. We can observe that adding more prompts does not result in better performances without prompt dropout. For example, performances with 6 prompts decrease compared to using a single prompt . This is due to limited diversity among the learned prompts. In comparison, adding more prompts is always beneficial when using prompt dropout.

Local prompt learning at multiple scales. On Fig. 8b, we show the interest of our multiscale approach. We experiment with various number of scales, *i.e.* from 1 to 6 scales with $k_1 = 10$ and $\Delta_k = 10$ and report the top-1 accuracy. We can observe a steady increase from 1 scale to 4 scales (+1pt). Performances stabilize afterward for 5 and 6 scales. Fig. 8b shows that learning at different scales is beneficial, but also that GalLoP is not too sensitive to the choice of number of prompts. Furthermore, learning at different scale also reduce the need to select an optimal k, although we show in Fig. 7 that performances are stable with respect to k.

4.4 Qualitative study.

We conduct in this section a qualitative study of GalLoP, by comparing it to CLIP on Fig. 9, and visualizing its different scales on Fig. 10. We show other qualitative results in supplementary material B.6.

Comparison to CLIP. On Fig. 9, we compare GalLoP and CLIP local features. We can observe that CLIP's local features are not discriminative and do not allow to classify images correctly, which was observed in Sec. 4.3. On the other hand, GalLoP classifies correctly the images, even with a single scale. We can also observe GalLoP accurately segments the object of interest when using all its scales.

Visualize multiple scales. Finally, we show the different regions each of the local prompts attend to. We can see that scale # 1 focuses on the most discriminative features, *i.e.* the head and tail of the "Ring tailed lemur". Each scale progressively attends to different parts of the body, leading to an accurate prediction.

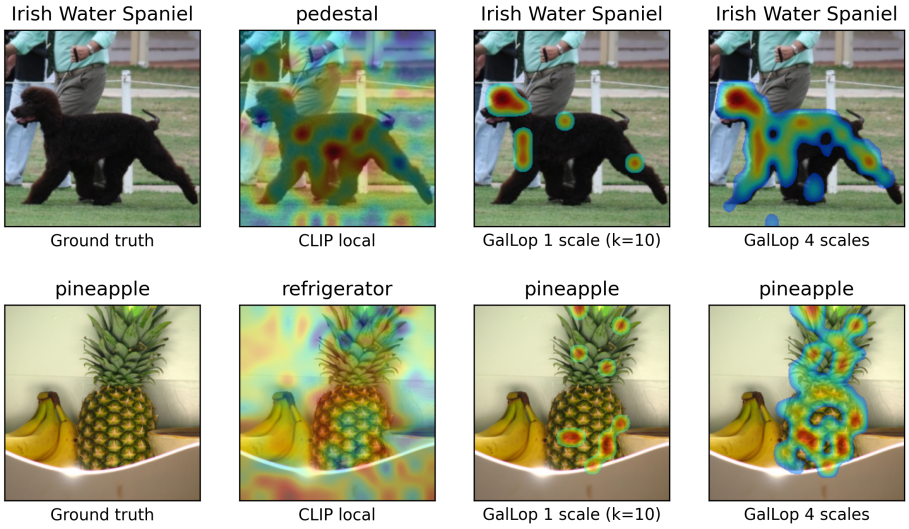


Fig. 9: Qualitative comparison of CLIP and GalLoP. From left to right, the original image with its ground truth, CLIP local wrong prediction, one scale (k=10) of GalLoP with correct prediction and GalLoP multiscale, resulting in correct prediction and segmentation.

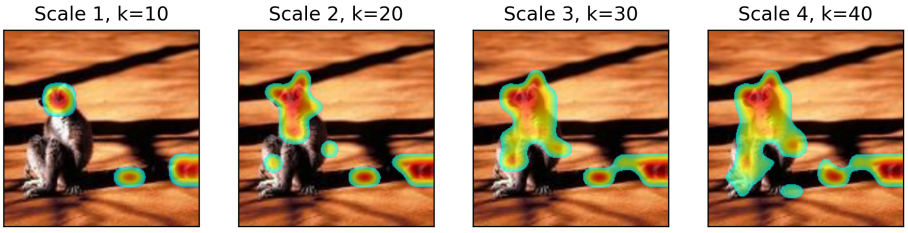


Fig. 10: GalLoP multiscale visualization. Regions observed by the different prompts of GalLoP for a "Ring tailed lemur".

5 Conclusion

This paper introduces GalLoP, a new prompt learning method that leverage both global and local visual representations. The key features of GalLoP are the strong discriminability of its local representations and its capacity to produce diverse predictions from both local and global prompts. Extensive experiments show that GalLoP outperforms previous prompt learning methods on top-1 accuracy on average for 11 datasets; that it works in different few shot settings; and for both convolutional and transformer vision-backbones. We show in ablation studies the interest of the design choices that make GalLoP work, *i.e.* complementarity between local and global prompts; sparsity and enhanced alignment; encouraging diversity. Finally, we conduct a qualitative study to show what local prompts focus on when classifying an image. Future works include learning the local feature alignment on a large vision-language dataset.

Acknowledgements

This work was done under grants from the DIAMELEX ANR program (ANR-20-CE45-0026) and the AHEAD ANR program (ANR-20-THIA-0002). It was granted access to the HPC resources of IDRIS under the allocation AD011012645R1 and AD011013370R1 made by GENCI.

References

1. Agnolucci, L., Baldrati, A., Todino, F., Becattini, F., Bertini, M., Del Bimbo, A.: Eco: Ensembling context optimization for vision-language models. In: ICCV (2023) 2
2. Bossard, L., Guillaumin, M., Van Gool, L.: Food-101 – mining discriminative components with random forests. In: ECCV (2014) 9, 24
3. Chen, G., Yao, W., Song, X., Li, X., Rao, Y., Zhang, K.: Plot: Prompt learning with optimal transport for vision-language models. In: The Eleventh International Conference on Learning Representations (2023) 1, 2, 3, 4, 9
4. Cho, J., Nam, G., Kim, S., Yang, H., Kwak, S.: Promptstyler: Prompt-driven style generation for source-free domain generalization. In: IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023. pp. 15656–15666. IEEE (2023) 4, 20
5. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., , Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2014) 9, 11, 24, 25, 26
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009) 9, 24
7. Dong, X., Bao, J., Zheng, Y., Zhang, T., Chen, D., Yang, H., Zeng, M., Zhang, W., Yuan, L., Chen, D., et al.: Maskclip: Masked self-distillation advances contrastive language-image pretraining. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10995–11005 (2023) 6
8. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 9
9. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer Vision and Pattern Recognition Workshop (2004) 9, 24
10. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: International Conference on Machine Learning. pp. 1050–1059. PMLR (2016) 7
11. Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., Qiao, Y.: Clip-adapter: Better vision-language models with feature adapters. International Journal of Computer Vision **132**(2), 581–595 (2024) 23, 24
12. Gondal, M.W., Gast, J., Ruiz, I.A., Droste, R., Macri, T., Kumar, S., Staudigl, L.: Domain aligned clip for few-shot classification. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 5721–5730 (2024) 24
13. Goyal, S., Kumar, A., Garg, S., Kolter, Z., Raghunathan, A.: Finetune like you pretrain: Improved finetuning of zero-shot vision models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023. pp. 19338–19347. IEEE (2023) 23, 24

14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arxiv e-prints. arXiv preprint arXiv:1512.03385 **10** (2015) 9
15. Helber, P., Bischke, B., Dengel, A., Borth, D.: Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification (2017) 9, 24
16. Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., Song, D., Steinhardt, J., Gilmer, J.: The many faces of robustness: A critical analysis of out-of-distribution generalization. ICCV (2021) 10, 25
17. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136 (2016) 20
18. Hendrycks, D., Mazeika, M., Dietterich, T.: Deep anomaly detection with outlier exposure. arXiv preprint arXiv:1812.04606 (2018) 4
19. Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., Song, D.: Natural adversarial examples. CVPR (2021) 10, 25
20. Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: International conference on machine learning. pp. 4904–4916. PMLR (2021) 1
21. Khattak, M.U., Rasheed, H., Maaz, M., Khan, S., Khan, F.S.: Maple: Multi-modal prompt learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19113–19122 (2023) 1, 2, 3, 9
22. Khattak, M.U., Wasim, S.T., Naseer, M., Khan, S., Yang, M.H., Khan, F.S.: Self-regulating prompts: Foundational model adaptation without forgetting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15190–15200 (2023) 1, 2, 3, 4, 9, 22
23. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13). Sydney, Australia (2013) 9, 24
24. Lafon, M., Ramzi, E., Rambour, C., Thome, N.: Hybrid energy based model in the feature space for out-of-distribution detection. In: International Conference on Machine Learning. pp. 18250–18268. PMLR (2023) 20
25. Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks. Advances in neural information processing systems **31** (2018) 20
26. Lu, Y., Liu, J., Zhang, Y., Liu, Y., Tian, X.: Prompt distribution learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5206–5215 (2022) 1, 2, 3, 4, 9, 20
27. Maji, S., Kannala, J., Rahtu, E., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. Tech. rep. (2013) 9, 24
28. Ming, Y., Cai, Z., Gu, J., Sun, Y., Li, W., Li, Y.: Delving into out-of-distribution detection with vision-language representations. Advances in Neural Information Processing Systems **35**, 35087–35102 (2022) 4, 5, 20, 25
29. Miyai, A., Yu, Q., Irie, G., Aizawa, K.: Locoop: Few-shot out-of-distribution detection via prompt learning. NeurIPS **36** (2023) 2, 4, 5, 7, 9, 26
30. Miyai, A., Yu, Q., Irie, G., Aizawa, K.: Zero-shot in-distribution detection in multi-object settings using vision-language foundation models. CoRR (2023) 4, 5, 19, 20
31. Nie, J., Zhang, Y., Fang, Z., Liu, T., Han, B., Tian, X.: Out-of-distribution detection with negative prompts. In: The Twelfth International Conference on Learning Representations (2024) 4, 26

32. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing (Dec 2008) 9, 24
33. Parisot, S., Yang, Y., McDonagh, S.: Learning to name classes for vision and language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 23477–23486 (2023) 1, 2
34. Parkhi, O.M., Vedaldi, A., Zisserman, A., Jawahar, C.V.: Cats and dogs. In: IEEE Conference on Computer Vision and Pattern Recognition (2012) 9, 24
35. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021) 1, 3, 9
36. Recht, B., Roelofs, R., Schmidt, L., Shankar, V.: Do imagenet classifiers generalize to imagenet? In: International conference on machine learning. pp. 5389–5400. PMLR (2019) 10, 25
37. Sehwag, V., Chiang, M., Mittal, P.: SSD: A unified framework for self-supervised outlier detection. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021 (2021) 20
38. Shu, Y., Guo, X., Wu, J., Wang, X., Wang, J., Long, M.: Clipood: Generalizing CLIP to out-of-distributions. In: ICML (2023) 23, 24
39. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012) 9, 24
40. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research **15**(1), 1929–1958 (2014) 4, 7
41. Sun, X., Hu, P., Saenko, K.: Dualcoop: Fast adaptation to multi-label recognition with limited annotations. Advances in Neural Information Processing Systems **35**, 30569–30582 (2022) 4, 7, 19
42. Sun, Y., Ming, Y., Zhu, X., Li, Y.: Out-of-distribution detection with deep nearest neighbors. In: International Conference on Machine Learning. pp. 20827–20840. PMLR (2022) 20
43. Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., Belongie, S.: The inaturalist species classification and detection dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8769–8778 (2018) 11, 25, 26
44. Villani, C., et al.: Optimal transport: old and new, vol. 338. Springer (2009) 4
45. Wang, F., Li, M., Lin, X., Lv, H., Schwing, A., Ji, H.: Learning to decompose visual features with latent textual prompts. In: The Eleventh International Conference on Learning Representations (2022) 2
46. Wang, H., Ge, S., Lipton, Z., Xing, E.P.: Learning robust global representations by penalizing local predictive power. In: Advances in Neural Information Processing Systems. pp. 10506–10518 (2019) 10, 25
47. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: CVPR (2010) 9, 11, 24, 25, 26
48. Zhang, R., Zhang, W., Fang, R., Gao, P., Li, K., Dai, J., Qiao, Y., Li, H.: Tip-adapter: Training-free adaption of CLIP for few-shot classification. In: ECCV. pp. 493–510 (2022) 24
49. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017) 11, 25, 26

50. Zhou, C., Loy, C.C., Dai, B.: Extract free dense labels from CLIP. In: Avidan, S., Brostow, G.J., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel. Lecture Notes in Computer Science, vol. 13688, pp. 696–712. Springer (2022) 7, 19
51. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Conditional prompt learning for vision-language models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16816–16825 (2022) 1, 3, 9
52. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Learning to prompt for vision-language models. International Journal of Computer Vision **130**(9), 2337–2348 (2022) 1, 3, 9, 21, 22

A Additional details on method.

In this section, we give additional details about GalLoP. In Sec. A.1, we describe how the local features are extracted from CLIP's vision encoder, for both ResNet and ViT architectures. In Sec. A.2, we describe the inference procedure in GalLoP as well as the GL-MCM score [30] which we use for OOD detection. Finally, we discuss in Sec. A.3 the use of an additional explicit diversity loss to train GalLoP.

A.1 CLIP's local visual features.

To obtain the visual local features from CLIP we follow previous works [30, 41, 50], which we describe in the following.

ViT backbone. When the vision encoder is a ViT, the output of the vision encoder is composed of the class token embedding, $\boldsymbol{z}_{\text{cls}}$, and a set of L local features $\mathcal{Z}_l = (\boldsymbol{z}_1^l, ..., \boldsymbol{z}_L^l)$. The global visual representation used in CLIP is the class token embedding, *i.e.* $\boldsymbol{z}_g = \boldsymbol{z}_{\text{cls}}$, however the local features after the last transformer block are of low quality as only the class token receives a supervision signal during training. Hence, prior studies [30, 41, 50] have recommended utilizing visual local features from the penultimate transformer block and forward them through the last transformer block without using the self attention mechanism.

Specifically, we have $\forall i \in \{1, ..., L\}$:

$$\boldsymbol{z}_i^l = \boldsymbol{z}_i^l + v(\boldsymbol{z}_i^l) + f(\boldsymbol{z}_i^l + v(\boldsymbol{z}_i^l)),$$

where $v(\cdot)$ denotes the linear projection used to compute the values in the self-attention module and $f(\cdot)$ is the feed-forward network of the last transformer block.

ResNet backbone. When the vision encoder is a ResNet the vision encoder outputs a feature map containing L local patches $\mathcal{Z}_l = (\boldsymbol{z}_1^l, ..., \boldsymbol{z}_L^l)$. Then, the global visual feature, $\boldsymbol{z}_g$, is obtained using a self-attention pooling module:

$$\boldsymbol{z}_g = \sum_i \text{softmax}\left(\frac{q(\overline{\boldsymbol{z}^l})\ k(\boldsymbol{z}_i^l)^T}{\sqrt{d}}\right) \cdot v(\boldsymbol{z}_i^l),$$

where d is the feature dimension, $\overline{\boldsymbol{z}^l} = \frac{1}{L}\sum_{i=1}^{L} \boldsymbol{z}_i^l$ is the average-pooled feature used as unique query, and $q(\cdot)$, $k(\cdot)$, $v(\cdot)$ denote the query, key and value projections, respectively. To obtain useful visual local features, it is then sufficient to use the values of the local features without the attention mechanism, *i.e.* $\boldsymbol{z}_i^l = v(\boldsymbol{z}_i^l)$.

A.2 Details on GalLoP's inference.

In this section, we give more details on our inference procedure. As described in Sec. 3.2, GalLoP is trained by summing the global and multiscale losses, associated to global and local prompts. Therefore, we naturally adopt an "ensembling-style" inference strategy by averaging the similarities obtained with each prompt to obtain a final similarity, $\text{sim}(\boldsymbol{z}, \boldsymbol{t}_c)$, for each class y_c.

Specifically, writing $\boldsymbol{z} = [\boldsymbol{z}_g, \mathcal{Z}_l]$, we compute:

$$\text{sim}(\boldsymbol{z}, \boldsymbol{t}_c) = \frac{1}{n}\sum_{i=1}^{n}\langle \boldsymbol{z}_g, \boldsymbol{t}_c(\boldsymbol{p}_i^{\boldsymbol{g}})\rangle + \frac{1}{m}\sum_{j=1}^{m}\text{sim}_{\text{top-}k}(\mathcal{Z}_l, \boldsymbol{t}_c(\boldsymbol{p}_j^{l})),$$

where $\text{sim}_{\text{top-}k}(\mathcal{Z}_l,\ \boldsymbol{t}_c(\boldsymbol{p}_j^{l}))$ is defined in Eq. (2) of the main paper. Then with this final similarity computed, we use Eq. (1) of the main paper to compute the probability for class y_c.

To perform out-of-distribution detection with GalLoP we use the GL-MCM score [30] which rely on both global and local information. The idea behind the MCM score [28] and the GL-MCM score [30] is to perform a maximum concept matching, which is a natural extension of the maximum class probability (MCP) score [17] which is widely used baseline within the OOD community [24, 25, 37, 42].

Formally, the GL-MCM score is expressed as:

$$\text{S}_{\text{GL-MCM}} = \text{S}_{\text{G-MCM}} + \text{S}_{\text{L-MCM}}$$

where

$$S_{\text{G-MCM}} = \max_{c} \frac{\exp(\frac{1}{n}\sum_{i=1}^{n}\langle \boldsymbol{z}_g, \boldsymbol{t}_c(\boldsymbol{p}_i^{\boldsymbol{g}})\rangle \ / \ \tau)}{\sum_{c'}\exp(\frac{1}{n}\sum_{i=1}^{n}\langle \boldsymbol{z}_g, \boldsymbol{t}_{c'}(\boldsymbol{p}_i^{\boldsymbol{g}})\rangle \ / \ \tau)},$$

$$S_{\text{L-MCM}} = \max_{c,\ i} \frac{\exp(\frac{1}{m}\sum_{j=i}^{m}\langle \boldsymbol{z}_i^{l}, \boldsymbol{t}_c(\boldsymbol{p}_j^{l})\rangle \ / \ \tau)}{\sum_{c'}\exp(\frac{1}{m}\sum_{j=1}^{m}\langle \boldsymbol{z}_i^{l}, \boldsymbol{t}_{c'}(\boldsymbol{p}_j^{l})\rangle \ / \ \tau)}.$$

A.3 Diversity loss.

Previous works on prompt ensembling have explored the use of an explicit loss term encouraging the semantic orthogonality between prompts to increase their diversity [4, 26]. This loss is expressed as:

$$\mathcal{L}_{\text{div.}}(\mathcal{P}) = \frac{1}{N\cdot(N-1)}\sum_{i=1}^{N}\sum_{j=i+1}^{N}|\langle \boldsymbol{t}_i, \boldsymbol{t}_j\rangle|,$$

where $\mathcal{P}$ is a set of N prompts and $\forall i \in \{1,\cdots,N\}$, $\boldsymbol{t}_i$ are the textual representations of the prompts without incorporating class names. The strength of the diversity loss is controlled with a hyper-parameter $\lambda_{\text{div.}}$.

We have experimented optimizing GalLoP with the following loss: $\mathcal{L}_{\text{total}}(\mathcal{P}, \boldsymbol{\theta}) + \lambda_{\text{div.}} \cdot \mathcal{L}_{\text{div.}}(\mathcal{P})$. In Fig. 11, we show that training GalLoP with $\mathcal{L}_{\text{div}}$ does not improve top-1 accuracy, even when increasing $\lambda_{\text{div.}}$. As a result, we did not include $\mathcal{L}_{\text{div.}}$ in GalLoP, as it did not lead to significant improvement in either accuracy or robustness, while introducing an extra hyperparameter, $\lambda_{\text{div.}}$.

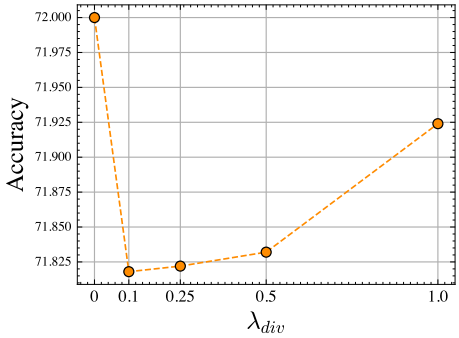


Fig. 11: Impact of $\lambda_{\text{div.}}$.

B Additional experimental results.

In this section, we give additional experimental results of GalLoP. In Sec. B.1 we conduct more results for few-shots settings experiments on the suite 11 of datasets. In Sec. B.2 we give results of GalLoP when using a ResNet-50 CLIP backbone. In Sec. B.4 and Sec. B.5 we give the detailed results for the ImageNet-1k domain generalization and out-of-distribution detection benchmarks, respectively. In Sec. B.3 we compare GalLoP to other few-shots learning methods. Finally, we show additional qualitative results in Sec. B.6.

Additional implementation details.

In this section, we give more implementation details of GalLoP. We show on Tab. 3 the hyperparameters used to train GalLoP on ImageNet for the 16-shots setting. We use the same data augmentation than CoOp [52].

Table 3: Hyperparameters to train GalLoP on ImageNet (16 shots) with ViT-B/16 backbone.

Hyperparameters	Value
batch size	128
learning rate	0.002
lr-scheduler	CosineAnnealingLR
epochs	50
optimizer	SGD
weight decay	0.01
momentum	0.9
local prompts	4
global prompts	4
tokens per prompt	4
prompt init	“A photo of a”

B.1 Full few shot results.

In this section, we give the detailed results for different few-shots settings. We report the top-1 accuracy of GalLoP on each dataset of the few-shot learning benchmark introduced in [52]. We can see in Tab. 4 that GalLoP outperforms other prompt learning baselines for all shots on average on the suite of 11 datasets. Specifically, GalLoP consistently outperforms the second-best method PromptSRC by +0.5pt with 1-shot, +1.1pt with 2-shots, +0.8pt with 4-shots, +1.5pt with 8-shots and +1.6pt with 16-shots. All results for each of the 11 datasets are ploted on Fig. 12.

Table 4: Averaged few-shots results on the suite 11 datasets with ViT-B/16 backbone.

Method	0-shot	1-shot	2-shots	4-shots	8-shots	16-shots
CLIP	64.9	-	-	-	-	-
CoOp	-	67.6	70.6	74.0	77.0	79.9
MaPLe	-	69.3	72.6	75.8	78.9	81.8
PLOT	-	70.7	74.0	76.9	79.6	82.1
PromptSRC	-	72.3	75.3	78.3	80.7	82.9
GalLoP	-	**72.8**	**76.4**	**79.1**	**82.2**	**84.5**

B.2 Detailed results for ResNet-50.

In this section we give detailed results for GalLoP when trained using a ResNet-50 backbone. We can see in Tab. 5 that GalLoP outperforms other ResNet-50 compatible prompt learning methods on all the datasets except Food101. Specifically, GalLoP achieves 77.3% accuracy on average on the suite 11 of datasets, outperforming PLOT by +3.4pt and CoOp by +3.9pt. Note that the second-best method on ViT-B/16, PromptSRC [22], is not compatible with convolutional backbones such as the ResNet-50.

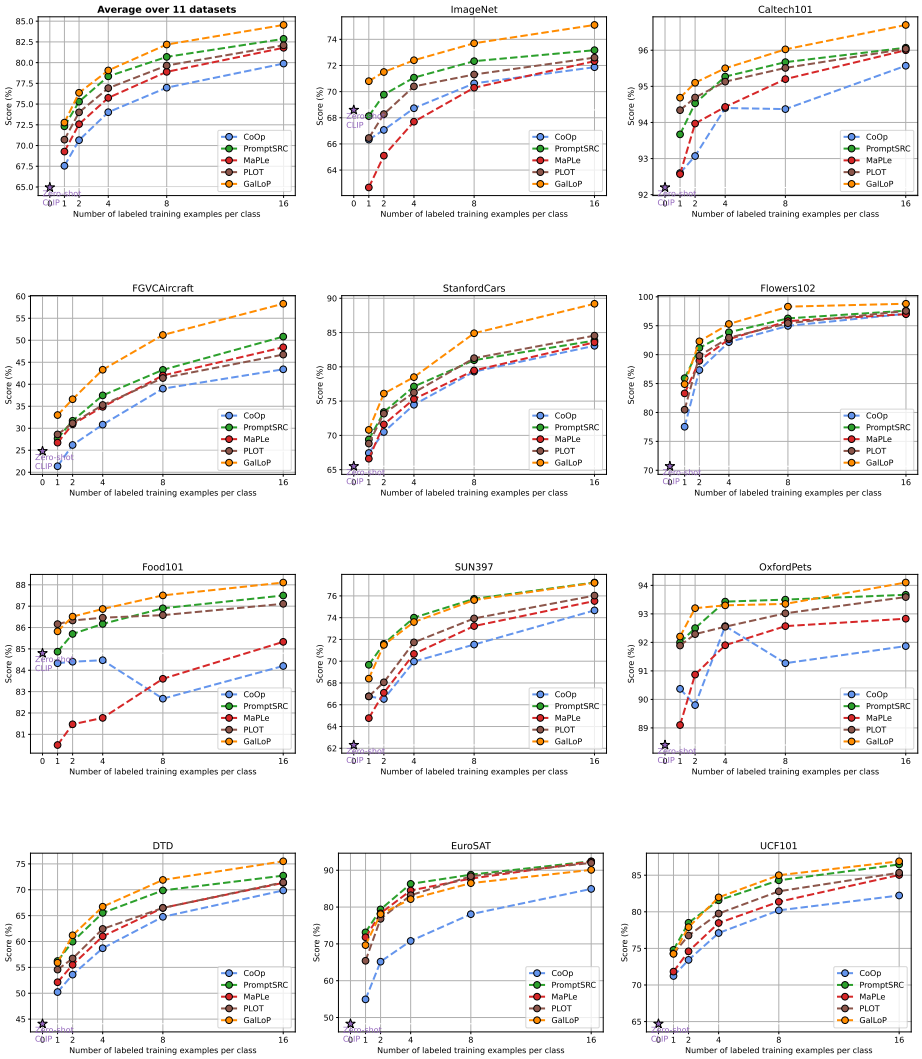


Fig. 12: Few-shot learning results of GalLoP on the 11 datasets with the ViT-B/16 backbone.

B.3 GalLoP *vs*. other few-shots learning methods.

In this section, we compare GalLoP to other type of few-shots learning methods. We compare GalLoP against the standard fine-tuning of all the parameters of CLIP's vision and text encoders as well as FLYP [13], which is a more recent version using the same contrastive objective as CLIP to fine-tune on downstream datasets. We also consider CLIP_{OOD} [38], which only trains the visual encoder. Furthermore, we also include adapters, *e.g.* the recent CLIP-Adapter [11], which uses residual adapters on both the visual and textual representations. Finally, we

Table 5: Top-1 accuracy with a Resnet-50 backbone in the 16-shots setting. Comparison of GalLoP to other prompt learning methods on the suite of 11 datasets.

Dataset	*ImageNet* [6]	*Caltech101* [9]	*OxfordPets* [34]	*Cars* [23]	*Flowers102* [32]	*Food101* [2]	*Aircraft* [27]	*SUN397* [47]	*DTD* [5]	*EuroSAT* [15]	*UCF101* [39]	*Average*
CLIP	58.1	84.1	82.7	55.8	66.0	75.0	17.0	57.1	42.9	36.3	57.9	57.5
Linear Probe	55.9	90.6	76.4	70.1	95.0	70.2	36.4	67.2	64.0	82.8	73.7	71.1
CoOp	63.0	91.8	87.0	73.4	94.5	74.7	31.3	69.3	63.6	83.5	75.7	73.4
Co-CoOp	62.9	90.2	88.3	61.6	78.3	**80.0**	21.3	67.3	56.2	70.1	71.1	67.9
PLOT	63.0	92.2	87.2	72.8	94.8	77.1	31.5	70.0	65.6	82.2	77.3	73.9
GalLoP	**66.1**	**92.8**	**89.3**	**79.3**	**96.7**	76.5	**41.6**	**72.2**	**67.6**	**87.6**	**80.4**	**77.3**

compare against cached-based methods like Tip-Adapter / Tip-Adapter-F [48] and DAC-V / DAC-VT [12].

We show in Tab. 6 the performance of GalLoP *vs*. the other few-shots learning methods in the 16-shots setting using a ViT-B/16 backbone. We can see that GalLoP obtains better top-1 accuracy than the recent fine-tuning method FLYP while it fine-tunes $\times 250$ more parameters than GalLoP. Furthermore, GalLoP outperforms the best cache-based method, DAC-VT, by $+0.5$pt, while having half the number of parameters. Also, when compared to DAC-V, which has the same number of parameters, GalLoP obtains $+2.1$pt in top-1 accuracy.

Table 6: Comparison of GalLoP *vs*. other few-shots learning methods on ViT-B/16 in the 16-shots setting.

	Top-1	# params ($\times 10^6$)
Zero-Shot CLIP	68.6	0
Tip-Adapter [48]	70.8	0
Full fine-tuning [13]	73.1	149.7
CLIP_{OOD} [38]	71.6	86.7
FLYP [13]	74.9	149.7
CLIP-Adapter [11]	71.1	0.2
Tip-Adapter-F [48]	73.7	16.4
DAC-V [12]	73.0	0.6
DAC-VT [12]	74.6	1.1
GalLoP	**75.1**	0.6

B.4 Detailed domain generalization results.

In this section, we give the detailed results for the ImageNet domain generalization benchmark. We compare the performances of GalLoP with several prompt learning

methods. Each method is trained on ImageNet with 16 shots per class and is evaluated on top-1 accuracy on four variants of ImageNet, *i.e.* ImageNet-V2 [36], ImageNet-Sketch [46], ImageNet-A [19] and ImageNet-R [16]. We can see in Tab. 7 that GalLoP outperforms previous prompt learning methods on average on the four ImageNet variants with +0.6pt top1-accuracy *vs*. PromptSRC$^{\diamond}$. More specifically, we obtain better results on ImageNet-V2, with +1.8pt with respect to the second-best method, and comparable results to PromptSRC$^{\diamond}$ on ImageNet-Sketch and ImageNet-R.

B.5 Detailed OOD detection results.

In this section, we give the detailed results of GalLoP for OOD detection. We use the OOD detection benchmark from [28] where ImageNet-1k is the in-distribution (ID) dataset, and iNaturalist [43], SUN [47], Places [49] and Textures [5] are used as OOD datasets. We report the results using the FPR95↓ and the AUC↑ metrics, two standard metrics used by the OOD detection community. The FPR95 is the false positive rate, using a threshold corresponding that classifies 95% of the ID images correctly. The AUC is the area under the receiver operating characteristic curve (ROC). We can see in Tab. 8 that GalLoP obtains better averaged FPR95 results than other prompt learning methods with -1.4pt *vs*. LoCoOp while achieving 93.2 averaged AUC, the second-best result after LoCoOp (93.5 averaged AUC).

Table 7: Domain generalization from ImageNet with ViT-B/16 backbone. Prompt learning methods are trained on ImageNet and evaluated on datasets with domain shifts. †results based on our re-implementation.

	Source	Target				
	ImageNet	-V2 [36]	-S [46]	-A [19]	-R [16]	Avg.
CLIP	66.7	60.8	46.2	47.8	74.0	57.2
CoOp	71.7	64.6	47.9	49.9	75.1	59.4
Co-CoOp	71.0	64.1	48.8	<u>50.6</u>	76.2	59.9
MaPLe	70.7	64.1	<u>49.2</u>	**50.9**	77.0	60.3
PLOT	72.6	64.9	46.8	48.0	73.9	58.4
PromptSRC$^{\diamond}$	71.3	64.4	**49.6**	**50.9**	**77.8**	<u>60.7</u>
PromptSRC$^{\triangleright}$	<u>73.2</u>	<u>65.7</u>	49.1	47.6	76.9	59.8
LoCoOp†	71.5	64.7	47.4	49.8	75.0	57.5
ProDA†	71.9	64.5	48.6	<u>50.7</u>	76.3	60.0
GalLoP	**75.1**	**67.5**	**49.5**	50.3	**77.8**	**61.3**

Table 8: OOD detection with ViT-B/16 as backbone. CoOp and LoCoOp results reported from [29]. CoCoOp and LSN results are reported from [31]. † denotes results based on our re-implementation. For PLOT, we use their released checkpoint and evaluate its OOD detection results ourselves.

	iNat [43]		SUN [47]		Places [49]		Textures [5]		**Average**		Top-1
	FPR95↓	AUC↑	FPR95↓	AUC↑	FPR95↓	AUC↑	FPR95↓	AUC↑	FPR95↓	AUC↑	
MCM	30.9	94.6	37.7	92.6	44.8	89.8	57.9	86.1	42.8	90.8	66.7
GL-MCM	15.2	96.7	30.4	93.1	38.9	89.9	57.9	83.6	35.5	90.8	66.7
PLOT	15.9	96.6	33.7	92.8	38.2	91.0	39.2	90.2	31.8	92.7	72.6
PromptSRC$^{\diamond}$	28.8	93.9	35.9	92.6	42.4	90.0	46.9	88.9	38.5	91.4	71.3
PromptSRC$^{\triangleright}$	20.6	95.7	30.1	93.7	38.0	91.1	46.0	89.0	33.7	92.4	<u>73.2</u>
ProDA†	32.4	93.2	35.7	92.4	42.6	90.0	46.2	89.3	39.2	91.2	71.9
CoOp$_{\text{MCM}}$	28.0	94.4	37.0	92.3	43.0	89.7	39.3	**91.2**	36.8	91.9	71.7
CoOp$_{\text{GL}}$	<u>14.6</u>	96.6	28.5	92.7	36.5	90.0	43.1	88.0	30.7	91.8	71.7
CoCoOp	30.7	94.7	31.2	93.2	38.8	90.6	53.8	87.9	38.6	91.6	71.0
LoCoOp$_{\text{MCM}}$	23.1	95.5	32.7	93.4	39.9	90.6	40.2	**91.3**	34.0	92.7	71.5
LoCoOp$_{\text{GL}}$	16.1	<u>96.9</u>	**23.4**	**95.1**	<u>32.9</u>	**92.0**	42.3	90.2	28.7	**93.5**	71.5
LSN$_{+\text{CoOp}}$	23.5	95.5	29.8	93.5	36.4	90.9	**38.2**	89.5	32.0	92.3	72.9
LSN$_{+\text{CoCoOp}}$	21.6	95.8	26.3	<u>94.4</u>	34.5	<u>91.3</u>	38.5	<u>90.4</u>	30.2	93.0	71.9
GalLoP	**13.7**	**97.1**	<u>24.9</u>	94.0	**32.5**	<u>91.3</u>	<u>38.4</u>	<u>90.4</u>	**27.3**	<u>93.2</u>	**75.1**

B.6 Additional qualitative results.

Finally, we display additional qualitative results of GalLoP$_{\text{Local}}$ on Fig. 13.

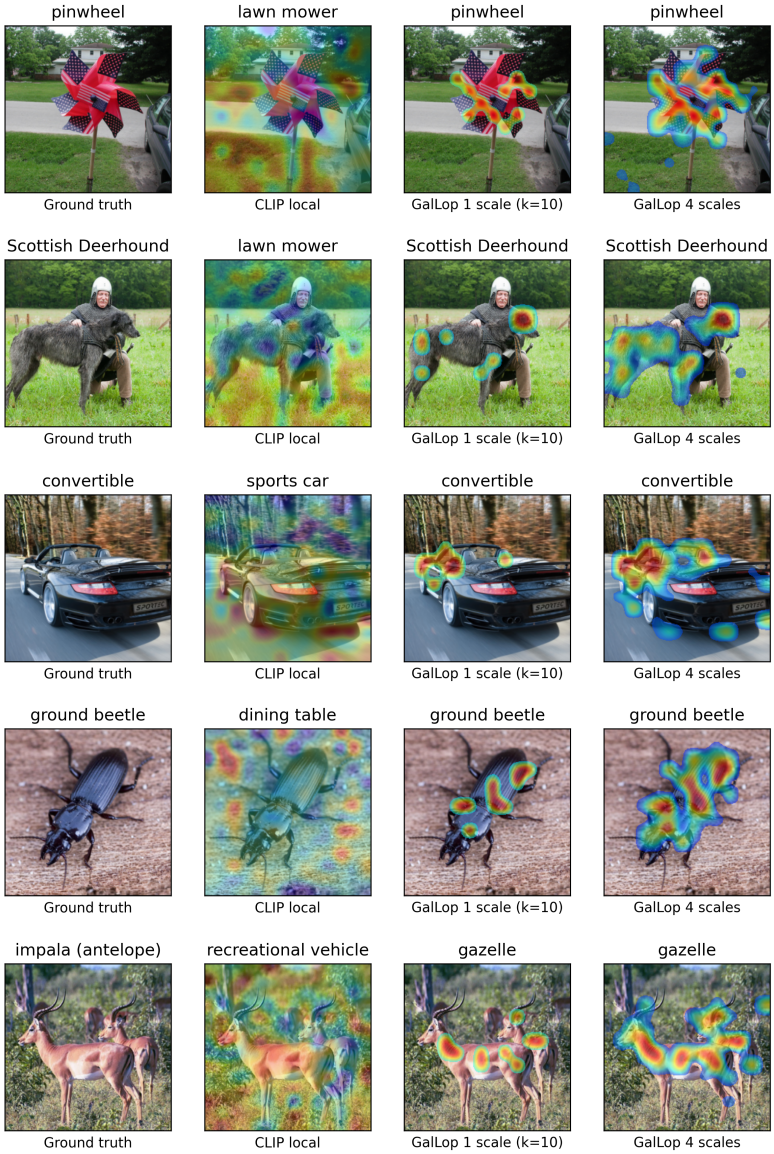


Fig. 13: Additional qualitative results for GalLoP. From left to right, the original image with its ground truth, CLIP local wrong prediction, one scale (k=10) of GalLoP$_{\text{Local}}$ with correct prediction and GalLoP$_{\text{Local}}$ multiscale.