International Journal of Computer Vision (2024) 132:1108–1125
https://doi.org/10.1007/s11263-023-01904-9

Language-Aware Soft Prompting: Text-to-Text Optimization for Few- and Zero-Shot Adaptation of V&L Models

Adrian Bulat[1] · **Georgios Tzimiropoulos**[1,2]

Received: 2 April 2023 / Accepted: 1 September 2023 / Published online: 25 October 2023

Abstract

Soft prompt learning has emerged as a promising direction for adapting V&L models to a downstream task using a few training examples. However, current methods significantly overfit the training data suffering from large accuracy degradation when tested on unseen classes from the same domain. In addition, all prior methods operate exclusively under the assumption that both vision and language data is present. To this end, we make the following 5 contributions: (1) To alleviate base class overfitting, we propose a novel Language-Aware Soft Prompting (LASP) learning method by means of a text-to-text cross-entropy loss that maximizes the probability of the learned prompts to be correctly classified with respect to pre-defined hand-crafted textual prompts. (2) To increase the representation capacity of the prompts, we also propose *grouped* LASP where each group of prompts is optimized with respect to a separate subset of textual prompts. (3) Moreover, we identify a visual-language misalignment introduced by prompt learning and LASP, and more importantly, propose a re-calibration mechanism to address it. (4) Importantly, we show that LASP is inherently amenable to including, during training, *virtual classes*, i.e. class names for which no visual samples are available, further increasing the robustness of the learned prompts. Expanding for the first time the setting to language-only adaptation, (5) we present a novel zero-shot variant of LASP where no visual samples at all are available for the downstream task. Through evaluations on 11 datasets, we show that our approach (a) significantly outperforms all prior works on soft prompting, and (b) matches and surpasses, for the first time, the accuracy on novel classes obtained by hand-crafted prompts and CLIP for 8 out of 11 test datasets. Finally, (c) we show that our zero-shot variant improves upon CLIP without requiring any extra data. Code will be made available.

Keywords Multimodal learning · Few-shot recognition · Zero-shot recognition · Prompting · Domain generalization

1 Introduction

Large-scale pre-training of neural networks has recently resulted in the construction of a multitude of foundation models for Language (Devlin et al., 2018; Radford et al., 2019) and Vision & Language (V&L) understanding (Radford et al., 2021; Jia et al., 2021; Yu et al., 2022; Alayrac et al., 2022). Unlike the previous generation of neural networks, such models can better capture the distribution of the world from which new favorable properties and characteristics emerge. Of particular interest to this work are V&L models trained with contrastive learning (i.e. CLIP-like models (Radford et al., 2021; Jia et al., 2021; Li et al., 2021; Yao et al., 2021; Yu et al., 2022)), which have enabled seamless few-shot and even zero-shot adaptation to new downstream tasks and datasets. Specifically, this paper proposes a simple yet highly effective way to drastically improve soft prompt learning for the few-shot adaptation of the V&L model to a given downstream task.

Similarly to their NLP counterparts (Radford et al., 2021; Lester et al., 2021; Li & Liang, 2021), prompt engineering and learning has emerged as one of the most powerful techniques for adapting a V&L to new tasks. Initially, in Radford et al. (2021), a set of manually-defined hand-engineered templates (or prompts) like `a photo of a {cls_name}`, or `a black and white photo of a {cls_name}` were passed through the text encoder of the V&L model to

Communicated by Kaiyang Zhou.

✉ Georgios Tzimiropoulos
g.tzimiropoulos@qmul.ac.uk

Adrian Bulat
adrian@adrianbulat.com

[1] Samsung AI Cambridge, Cambridge, UK

[2] Queen Mary University of London, London, UK

create class-specific weights for category `cls_name` that can be used for zero-shot recognition. Following research in NLP (Lester et al., 2021; Li & Liang, 2021), subsequent work (Zhou et al., 2022, a) has proposed replacing the manually picked templates with a sequence of learnable vectors, also coined *soft prompts*, which are fed as input to the text encoder along with the class name `cls_name`. The soft prompts are learned from a few training examples, with the parameters of the entire V&L model kept frozen. The whole process can be seen as parameter efficient fine-tuning of the V&L model on a small training dataset.

However, a clearly identifiable problem with prompt learning is base class overfitting: while the accuracy on the classes used for training (base classes) significantly increases, the accuracy on unseen, during training, (novel) classes significantly drops. This is to some extent expected, as soft prompts are learned from few examples belonging to the base classes. Notably, on novel classes, direct, zero-shot recognition using hand-engineered prompts outperforms all existing soft prompt learning methods.

In addition to this, for adaptation, all prior works assume the existence of paired vision and language data. Herein, we seek to relax this setting and advance the idea of vision-language adaptation *without* images, i.e. using solely language data, namely the class names of interest.

Key ideas: Firstly, to alleviate base class overfitting, in this work, we propose a solution motivated by the following observation: since prompt learning improves the accuracy on base classes, but prompt engineering is significantly better on novel classes, we propose to learn the soft prompts by adding a cross entropy text-to-text loss that enforces the learned prompts to be close, in embedding space, to the textual ones, thus exploiting the intrinsic information captured by the text encoder. The proposed text-to-text loss enables language-only optimization for vision-language adaption for the first time. This is in contrast with prior soft-prompt learning methods that only capture vision-language interactions.

Secondly, as CLIP learns a joint shared representation for the two domains, i.e. vision and language, one can approximate, to some extent, the vision domain with language (limited by the induced contrastive domain gap). Hence, by exploiting this, we devise a prompt learning framework for vision language adaptation that can learn solely based on the class names.

Key contributions: Based on the above, we propose a novel framework for soft prompt learning which we call Language-Aware Soft Prompting (LASP) trained either with labeled vision-language data or solely in the language domain (LASP-Z). Our main contributions within the LASP framework are as follows:

- We propose, for the first time, language-only optimization for vision-language adaption. Specifically, we propose a novel text-to-text cross-entropy loss that maximizes the probability of the learned prompts to be correctly classified with respect to the hand-engineered ones and show its effectiveness in terms of alleviating base-class overfitting.
- To increase the representation capacity of the prompts, and inspired by grouped convolution and multi-head attention, we propose a grouped language-aware prompt representation where *each group* of prompts specializes to a different subset of the pre-defined manual templates.
- We identify a visual-language misalignment introduced by prompt learning and LASP which impacts the generalization. More importantly, we propose a re-calibration mechanism based on (a) Layer Normalization fine-tuning and (b) learning a class-agnostic bias to address it.
- Thanks to our language-only learning framework, we propose training LASP with virtual classes by including, during training, class names for which no visual samples are available. Importantly, we show that this further increases the robustness of the learned prompts.
- Finally, by capitalizing on our language-only optimization framework, we present a zero-shot variant of LASP where no visual samples at all are available for the downstream adaptation task and show its superiority upon CLIP with prompt engineering. Effectively, this accomplishes vision-language adaptation *without* vision data.

Main results: Our methods set a new state-of-the-art for few-shot and zero-shot image classification on 11 datasets, significantly outperforming all soft prompting prior works. Importantly, we present, for the first time, a prompt learning method that outperforms, for the majority of the test datasets (8 out of 11), the very strong baseline based on hand-crafted prompts and CLIP for the recognition of novel classes (i.e. zero-shot setting). Moreover, our zero-shot V&L adaptation approach, LASP-Z, improves upon zero-shot CLIP without requiring any images at train time.

2 Related Work

Contrastive Vision-Language Models: Recently, large scale V&L pre-training with contrastive learning has been used to train foundation models resulting in robust representations, transferable to new tasks both under few-shot and zero-shot settings (Radford et al., 2021; Jia et al., 2021; Li et al., 2021; Yao et al., 2021; Yu et al., 2022). Such networks consist of a vision encoder (typically a ViT (Dosovitskiy et al., 2020)) and a Transformer-based text encoder (Vaswani et al., 2017). Highly parameterized instantiations of such architectures are trained on large corpora of image-caption pairs (e.g. Radford et al. (2021) uses 400 M and Jia et al. (2021)

1B pairs) using contrastive learning. We used CLIP (Radford et al., 2021) as the foundation model for our method.

Domain generalization aims to learn models that generalize to out-of-domain data. Current approaches attempt to perform data alignment (Hu et al., 2020; Mahajan et al., 2021; Shao et al., 2019), augmentation (Shi et al., 2020; Zhou et al., 2023), meta-learning (Balaji et al., 2018; Dou et al., 2019), self-supervised learning (Albuquerque et al., 2020) or reinforcement learning (Laskin et al., 2020; Yarats et al., 2021). As our approach can generalize outside of the source data, either via few-shot adaptation (LASP) or more extremely, using solely the class names (LASP-Z), it can be also considered as a domain-generalization method.

Zero/few-shot learning is concerned with the construction of models that can be adapted to downstream tasks using few or even no labeled samples. Both scenarios are currently dominated by large-scale constrastively pretrained vision-language models (Radford et al., 2021; Yao et al., 2021), a line of work which our approach builds upon too. While a full review goes beyond the scope of this work, we note that this is a vast research field (Nichol et al., 2018; Rajeswaran et al., 2019; Li et al., 2017), referring the reader to (Song et al., 2023).

Prompt Learningis about adapting pre-trained foundational models on (downstream) tasks, typically in a zero-shot or few-shot setting. Firstly proposed in the context of Language Models (LM), prompting was initially about prepending hand-crafted instructions/examples to the task input so that the LM generates the appropriate output conditioned to the input (Radford et al., 2019; Brown et al., 2020). In (Schick & Schutze, 2020a, b), the main idea is to reformulate the downstream task as a *cloze* task using hand-crafted patterns (or templates), thus avoiding the need to train a task-specific classifier. As finding the optimal patterns is laborious, recent works have attempted to address this by learning a set of soft (continuous) prompts (Lester et al., 2021; Li & Liang, 2021).

In V&L foundation models, like CLIP, the class names are used to create hand-crafted prompts (Radford et al., 2021) that are fed as input to the text encoder, enabling zero-shot visual recognition. CoOp (Zhou et al., 2022) extends work on soft prompt optimization to the V&L domain by learning a set of M prompts which are used as input to the text encoder alongside the class name. The prompts are learned by minimizing the classification error on a training set consisted of the given base classes. One major limitation of CoOp is weak generalization: the learned prompts overfit the base classes and do not work well when tested on novel classes. To alleviate this, CoCoOp (Zhou et al., 2022a) proposes a dynamic version of Zhou et al. (2022) where a small network is trained to produce a visual feature from the input image that is added to the learned prompts, hence making them input specific (i.e. dynamic). ProDA (Lu et al., 2022) adopts a probabilistic approach by modelling the distribution of the prompts at the output of the text encoder as a multivariate Gaussian distribution. The estimated mean is used during inference. UPL (Huang et al., 2022) uses CLIP to generate pseudo-labels on the target dataset and then self-training to learn the soft prompts. Finally, ProGrad (Zhu et al., 2022) aims to adapt the V&L model to each target domain by encouraging it “not to forget” CLIP’s zero-shot predictions using a KL visual-text loss between the CLIP’s logits and their model’s logits (i.e. they use visual features). The weights are then updated in the direction perpendicular to CLIP gradients. In contrast, our loss is a pure text-to-text loss, further allowing for the incorporation of virtual classes. Unlike (Zhu et al., 2022), we outperform CLIP on novel classes.

The proposed LASP framework alleviates base class overfitting and significantly improves upon the previously reported best results without resorting to a dynamic approach as in CoCoOp (Zhou et al., 2022a). In its basic version, LASP deploys a text-to-text loss that enforces the learned prompts to be “close” to a set of manually defined textual prompts in the text encoder space. Importantly, the basic LASP can be extended in three important ways: (1) by allowing the incorporation of virtual classes i.e. novel class name information for which no (visual) training data is available (LASP-V). This is shown to significantly improve the robustness of the learned prompts at no extra cost during inference; (2) by allowing the use of a grouped prompt representation within the proposed language-aware training which is shown to increase the representation capacity of the learned prompts; (3) by performing further optimization of the visual encoder so that the visual and text embeddings are realigned resulting in significant accuracy gains. Finally, we present a zero-shot variant of LASP where no training images at all are available for the downstream adaptation task. Notably, our approach is very efficient (as efficient as Zhou et al. (2022)) as opposed to Zhou et al. (2022a) which requires recomputing all the class-related text embeddings every time a new image is to be classified.

3 Method

3.1 Background

Prompt engineering enables zero-shot visual recognition using V&L models trained with contrastive learning (CLIP in this work) as follows: Given a set $\mathcal{V}$ of C class names, $\texttt{class\_name}_c$, $c \in \{1, \ldots, C\}$, a prompt, i.e. a manually designed template concatenated with the class name like $h_c = \texttt{a photo of a}\,\{\texttt{class\_name}_c\}$, is passed through the V&L’s text encoder $g_T(.)$ to compute the class specific text feature (weight) $\mathbf{t}_c^h = g_T(h_c)$. Moreover, an image $\mathbf{x}$

to be classified is passed through the V&L's image encoder $g_I(.)$ to compute image specific feature $\mathbf{f} = g_I(\mathbf{x})$. A probability distribution over the class labels is given by:

$$P_h(y|\mathbf{x}) = \frac{\exp\Big(\cos(\mathbf{t}_y^h, \mathbf{f})/\tau\Big)}{\sum_{c=1}^{C} \exp\Big(\cos(\mathbf{t}_c^h, \mathbf{f})/\tau\Big)}, \tag{1}$$

where τ is a temperature factor and $\cos$ the cosine similarity. Finally, the class for $\mathbf{x}$ is given by $\tilde{y} = \arg_{max} P_h(y|\mathbf{x})$. Note that, to compute $\mathbf{t}_c^h$, no training with class specific image data is required, thus enabling zero-shot recognition for any given class name.

Soft prompt learning (Lester et al., 2021; Li & Liang, 2021; Zhou et al., 2022) is concerned with parameter efficient fine-tuning of a pre-trained V&L model by learning a sequence of M learnable vectors $\mathbf{p}_m \in \mathbb{R}^d$, $m = \{1, \ldots, M\}$ using a few labeled samples. Specifically, the manually picked prompt h_c is replaced by a new learnable one $\mathbf{r}_c$ formed by concatenating the sequence of $\mathbf{p_m}$ with the word embedding $\mathbf{w}_c$ of $\texttt{class\_name}_c$, that is: $\mathbf{r}_c = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_M, \mathbf{w}_c\}$, and, finally, a class specific text feature $\mathbf{t}_c^r = g_T(\mathbf{r}_c)$ is obtained. A probability distribution over the class labels is:

$$P_r(y|\mathbf{x}) = \frac{\exp\Big(\cos(\mathbf{t}_y^r, \mathbf{f})/\tau\Big)}{\sum_{c=1}^{C} \exp\Big(\cos(\mathbf{t}_c^r, \mathbf{f})/\tau\Big)}. \tag{2}$$

The prompts can be learned by minimizing the cross-entropy loss:

$$\mathcal{L}_{VL} = -\sum_{c=1}^{C} \log P_r(c|\mathbf{x}) y_c. \tag{3}$$

Note that the V&L model remains entirely frozen during training. Moreover, as the soft prompts are typically shared across all classes, they can be directly used for zero-shot evaluation on additional novel classes.

3.2 Language-Aware Soft Prompting (LASP)

Despite its strong performance on base classes, vanilla soft prompt learning (see Sect. 3.1) under-performs on novel classes (i.e. zero-shot setting). While CoCoOp (Zhou et al., 2022) partially alleviates this by conditioning on the image feature, its accuracy for the zero-shot setting is still trailing that of CLIP with hand-crafted prompts. Moreover, it requires passing the prompts for all classes through the text encoder every time a new image is to be classified.

In this work, we propose, for the first time, language-only optimization for vision-language downstream adaption. This is in contrast with prior soft-prompt learning methods that only capture vision-language interactions. Specifically, since the hand-engineered textual prompts outperform the learnable soft prompts for the zero-shot setting, then, in order to avoid base-class overfitting and strengthen generalizability, we propose that the learnable ones should be trained so that they can be correctly classified in language space where the class weights are given by the textual prompts. In other words, the model is forced to correctly classify the learnable prompts into the corresponding hand-crafted ones.

To this end, a second cross entropy loss is used to minimize the distance between the encoded learned soft prompts and the encoded textual ones. Specifically, recall that $\mathbf{t}_c^h = g_T(h_c)$ is the class weight for class c obtained by encoding the corresponding textual prompt. Assuming that L manually defined textual prompts are available,[1] we have $\mathbf{t}_c^{h,l}$, $l = 1, \ldots, L$. Moreover, $\mathbf{t}^r$ is an encoded learnable prompt to be classified in one of the C classes. Finally, the probability of prompt $\mathbf{t}^r$ being classified as class y is:

$$P_{rh}(y|\mathbf{t^r}) = \frac{1}{L}\sum_{l=1}^{L} \frac{\exp\Big(\cos(\mathbf{t}_y^{h,l}, \mathbf{t}^r)/\tau\Big)}{\sum_{c=1}^{C} \exp\Big(\cos(\mathbf{t}_c^{h,l}, \mathbf{t}^r)/\tau\Big)}. \tag{4}$$

The language-aware training loss is computed similarly to the vision-language loss:

$$\mathcal{L}_{TT} = -\sum_{c=1}^{C} \log P_{rh}(c|\mathbf{t}^r) y_c, \tag{5}$$

with the overall training objective defined as:

$$\mathcal{L} = \alpha_{VL}\mathcal{L}_{VL} + \alpha_{TT}\mathcal{L}_{TT}, \tag{6}$$

where α_{VL} and α_{TT} are user-defined scaling coefficients controlling the magnitude of the $\mathcal{L}_{VL}$ and $\mathcal{L}_{TT}$ losses, respectively. Overall, we call the proposed learning formulation Language-Aware Soft Prompting (LASP). See also Fig. 1.

Interpretations: LASP can be interpreted in a number of ways:

LASP as a regularizer: Although the learned prompts constitute a small number of parameters, especially in the few-shot setting, the resulting models (prompts) are prone to overfitting to base classes (Zhou et al., 2022). As the proposed language-aware loss encourages the learned prompts to be close in embedding space to the textual ones, LASP can be naturally viewed as a regularizer that prevents the learned prompt-conditioned features from diverging too much from the hand-crafted ones.

LASP as language-based augmentation: Current soft prompt learning methods restrict augmentation to the vision

[1] The original CLIP prompts serve as textual prompts without any tweaking or change. Note, that our method can even work with random sentences (see Sect. 5.3).

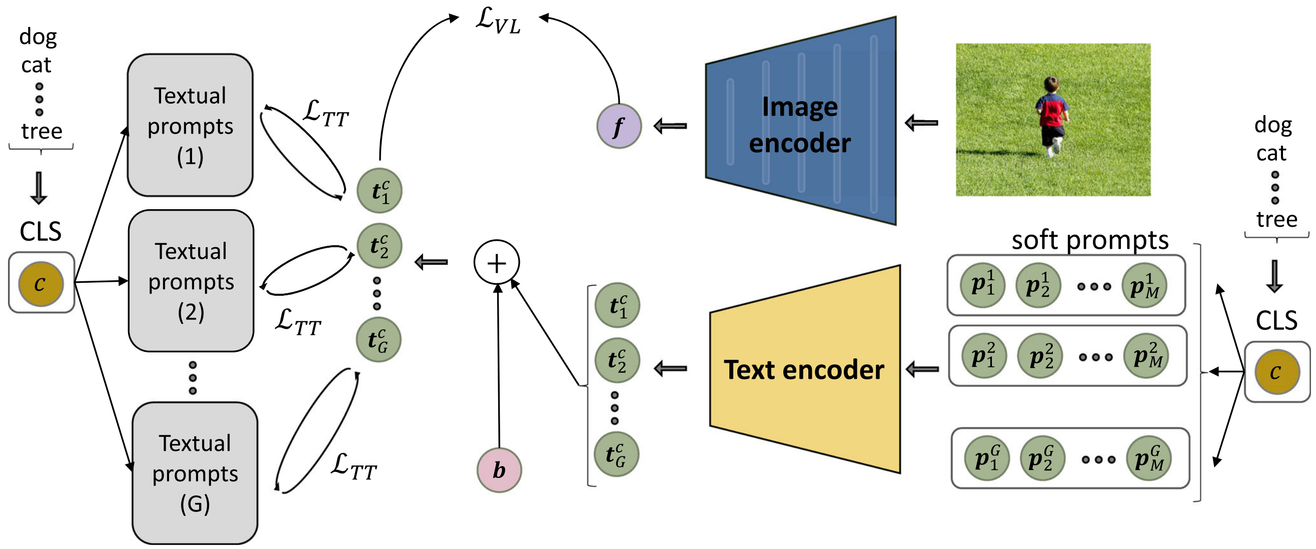


Fig. 1 Overall idea. While standard prompt learning is based on image-text interactions (L_{VL} loss; Eq. 3), LASP additionally models text-text interactions using the proposed Text-to-Text loss L_{TT} (Eq. 5). There are G groups of learned prompts $\mathbf{p}_i^j$ passed through the text encoder to form G text embeddings $\mathbf{t}_j$ summarizing the input. The L_{TT} loss is then applied over the different groups of the text embeddings and the textual prompts. Moreover, to alleviate data distribution shift and visual-language misalignment, the LN layers of the visual encoder are fine-tuned and the embeddings are "corrected" at the output space by the learnable vector **b**, shared for all classes. The text encoder remains entirely frozen. Notably, LASP can be trained with virtual classes by including, during training, class names for which no visual samples are available

domain, where random transformations, such as rotation, colour jittering or scaling, increase the robustness of the system, especially for cases with limited number of training samples. However, no augmentations are performed in the language domain. Ideally, we want the prompt-conditioned text embedding to be robust too, capturing the full space of each class. In practice, we can achieve this by targeted prompting, where we can specify certain characteristics and/or apply text-based transformations to the class name, e.g.: "A sketch of *dog*" or "A rotated photo of a *dog*".

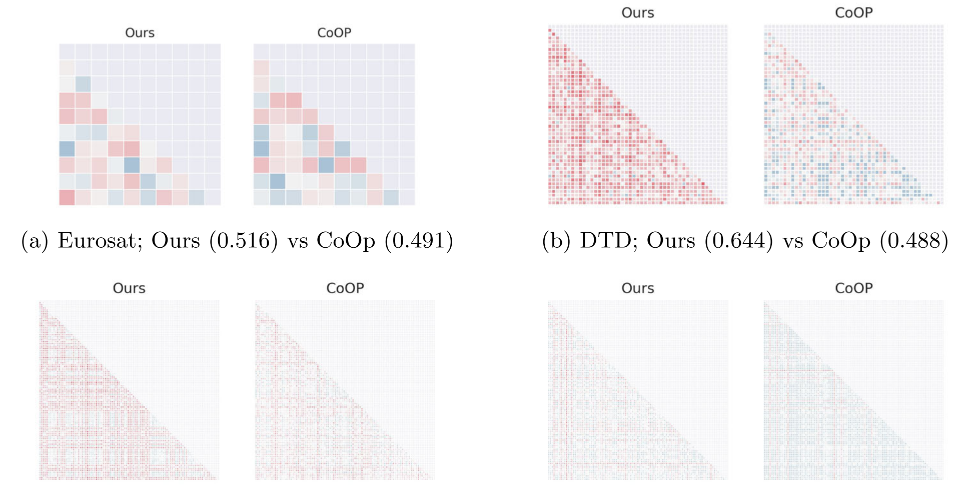
At train time, as reflected by Eq. 4, we compute the class label distribution per l-th template and then average over all templates. Hence, we opt not to mix across templates during training as we want the model to focus on class information solely. For example, the model could distinguish easier between a "a sketch of a *dog*" and "a photo of a wolf" compared to "a sketch of a *dog*" and "a sketch of a wolf", as in the former case, the style could be used as an additional queue. We validated this in preliminary experiments (intermixing the templates was found to hurt performance).

LASP for discriminative class centroids: By optimizing w.r.t both image and text, our method produces class centroids that are more discriminative and have a higher separation margin. This can be visualized in Fig. 2 where we plot the cosine distance between the embeddings of each class. Our approach learns class centroids that have a higher cosine distance than those of our baseline, CoOp.

LASP as data-free distillation: Typically, knowledge distillation requires a training set of images where a teacher network provides a training signal for the student (Hinton, 2015). LASP's text-to-text loss can be also interpreted as a data-free distillation (i.e. does not use any image data) where the learnable prompts define the "samples". As CLIP learns a joint vision-language space, similar concepts are close together across both domains. Hence, optimizing against a concept or object in the language domain, using the proposed loss, should also help make a step in the visual domain, improving the classification of the images.

3.3 Grouped LASP

Grouped convolutions (Krizhevsky et al., 2017) and multi-head attention (Vaswani et al., 2017) have been shown to learn strong representations. The groups or the number of heads, respectively, can be also interpreted as a set of experts that are then combined to produce a strong feature. Drawing inspiration from this, we propose a grouped prompt representation, where each group is optimized with respect to a separate subset of textual prompts. Effectively, the prompts from each group will learn a transformation specialized to its corresponding subset (analogous to the aforementioned techniques that also specialize to a part of the signal). In particular, we split the set of L templates into G equally sized sub-sets. Moreover, each sub-set is associated with a sequence of M prompts $\mathbf{r}_c^g = \{\mathbf{p}_1^g, \ldots, \mathbf{p}_M^g, \mathbf{w}_c\}, g = 1, \ldots, G$ each producing a class specific text feature $\mathbf{t}_c^{r,g} = g_T(\mathbf{r}_c^g)$. Finally, our text-to-text loss in Eq. 5 becomes:



(a) Eurosat; Ours (0.516) vs CoOp (0.491)

(b) DTD; Ours (0.644) vs CoOp (0.488)

(c) Flowers102; Ours (0.628) vs CoOP (0.554)

(d) Food101; Ours (0.487) vs CoOP (0.397)

Fig. 2 Cosine distance between the class embeddings produced by the CLIP text encoder on Eurosat, DTD, Flowers102 and Food101 for LASP and CoOp. Class centroids situated further apart are more separable, as the underlying image features are identical across both methods. Brighter colors indicate bigger distances. The numbers shown are the average cosine distance between the classes

$$\mathcal{L}_{TT-G} = -\sum_{g=1}^{G}\sum_{c=1}^{C} \log P_{rh}^{g}(c|\mathbf{t}^{g})y_c, \tag{7}$$

with P_{rh}^{g} computed for each group similarly to Eq. 4. At test time, the final result is computed by taking the average of the cosine similarity scores between each group and the visual feature $\mathbf{f}$.

3.4 Re-aligning LASP

Combating data distribution shift: for some downstream tasks, it is possible that there is a data distribution shift between the downstream image dataset and the one used by CLIP during training. Hence, we would like this aspect to be captured by the downstream adaptation method. To this end, some optimization of the visual encoder can be performed; nevertheless this can very easily result in base class overfitting if, after the training, the V&L embeddings are pushed away from the joint space learned by CLIP. For example, preliminary results with visual adapters have shown that they hurt zero-shot accuracy. On the contrary, we found that Layer Normalization (LN) (Ba et al., 2016) fine-tuning is a much more robust way to adapt the visual encoder. Overall, we propose fine-tuning the LN of the CLIP encoder as a way to combat distributional shift.

Combating V&L misalignment: Because after LN fine-tuning the V&L are not guaranteed to continue to be aligned, we also propose to learn a "correction" at the output of the text encoder in the form of a learnable offset (bias) that aims to re-align the two modalities. Let $\mathbf{W}$ be the set of weights of the linear classifier obtained by passing the learned prompts from the text encoder. We propose to learn a vector $\mathbf{b} \in \mathbb{R}^{d}$ that is simply added to $\mathbf{W}$, that is $\mathbf{W} = \mathbf{W} + \mathbf{b}$. Importantly, the learned offset is shared among all classes, and in this way it can be readily applied for the case of novel classes too.

3.5 LASP with Virtual Classes (LASP-V)

A direct observation that can be drawn from Eq. 4 is that, in practice, we do not have to use only the class names for which we have labelled image data, as the value of L_{TT} is independent of the input image. To this end, we propose to learn the prompts using both annotated image-text pairs and *class names* outside of the base set (for which we have no images available). We call this setting as training *LASP with virtual classes*. Our setting combines the best of both words: the guidance from the few annotated image samples and the zero-shot generalizability of language-based training. As our results show, LASP with virtual classes can significantly improve the robustness of the prompts learned. We refer to this variant of our method as **LASP-V**.

Note that training with virtual classes does not violate the zero-shot setting (Xian et al., 2017).[2] Moreover, from a practical perspective, if the novel class names are not known during initial training, the model can be simply retrained in a zero-shot manner when they become available.

4 Zero-Shot LASP (LASP-Z)

The LASP framework presented so far combines vision-language and language-language optimization for both few-shot and in-domain zero-shot accuracy. However, as $\mathcal{L}_{VT}$ and $\mathcal{L}_{TT}$ can be applied independently, one can fully transition from the few-shot setting to the zero-shot one, where

[2] according to Xian et al. (2017) "Zero-shot learning aims to recognize objects whose **instances** may not have been seen during training."

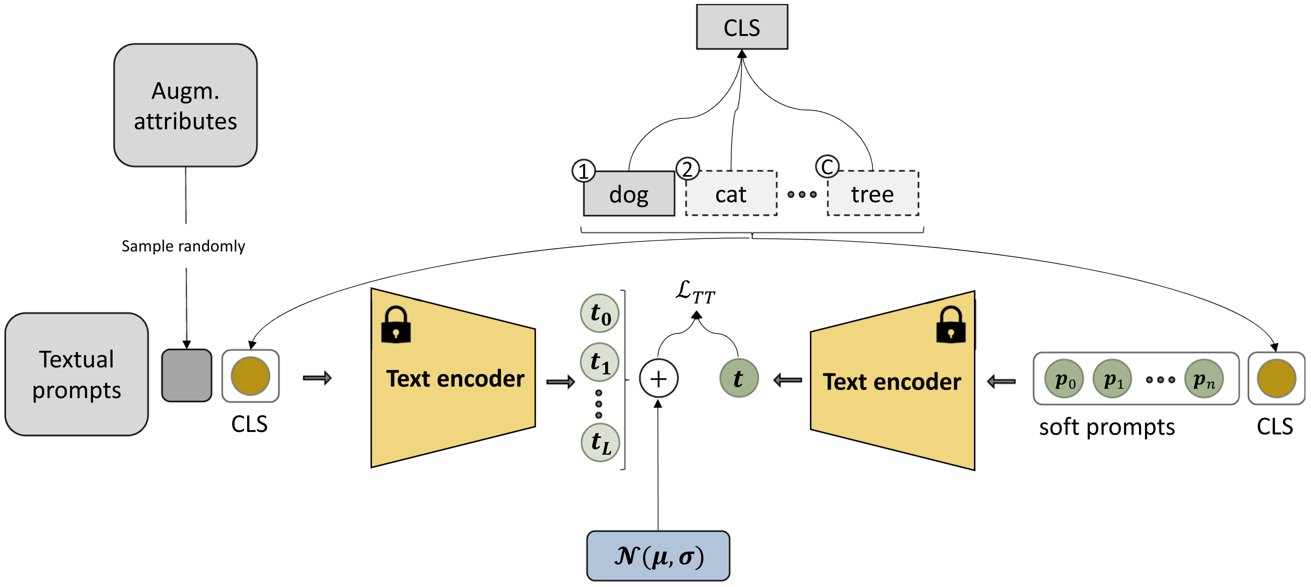


Fig. 3 LASP-Z is a pure zero-shot variant of LASP which does not use visual samples at all for downstream adaptation. LASP-Z exclusively operates within the language domain by optimizing the prompts using the proposed Text-to-Text loss L_{TT} (Eqs. 5, 8). During training, in order to explicitly alleviate the domain gap and explore the vicinity of the class embeddings, we propose **a** to augment the input space using a series of randomly sampled adjectives/attributes and **b** to augment the output space by injecting additive Gaussian noise to the class embedding. *Note* in the figure above, the text encoder remains entirely frozen, and its weights are shared

no visual examples at all are available, just the class names. Herein, we attempt to study this zero-shot setting, introducing LASP-Z, a visual data-free approach capable of zero-shot in-domain specialization.

Not entirely surprising, a naive, direct application of Eq. 5 is heavily sensitive to overfitting: Firstly, there is an implicit domain gap between the vision and language modalities within the CLIP embedding space (Liang et al., 2022), hence overly specializing to the textual data amplifies and further expose this dissimilarity. Secondly, for image data, it is common practice to apply random train-time augmentations with the goal of alleviating overfitting. In fact, image augmentation has been shown to be a key component in many state-of-the-art self-supervised representation learning methods (Chen et al., 2020; Caron et al., 2020). Hence, to optimize Eq. 5 without overfitting, one should aim at applying inexpensive, yet effective augmentations in the language domain.

Specifically, we define two “augmentation” inducing functions, applied pre- and post-encoding: $f_{pre}(.)$ and $f_{post}(.)$, respectively. The goal of f_{pre} inserts a set of adjectives or attributes before the class name (e.g. `large, small, rotates, pixelated, colorful` etc.). This explores, in essence, class-generic appearance variations directly in the text domain, analogous to the image ones. Moreover, $f_{post}(\mathbf{t}) = \mathbf{t} + \mathbf{x}, \mathbf{x} \sim \mathcal{N}(\mu, \sigma^2)$ adds to the text feature descriptor t a noise vector sampled from a normal distribution. Depending on its magnitude, this allows the model to explore the immediate vicinity of the prompt in the CLIP embedding space, increasing the chance of matching points located in the proximity of true visual samples, mitigating to some extent the domain gap.

Recall that $\mathbf{t}_c^{h,l} = g_T(h_c^l)$ is a feature descriptor obtained by encoding the $c-$th class name with the l-th predefined textual template ($l = 1, \ldots, L$), and $\mathbf{t}^r$ is an encoded learnable prompt to be classified in one of the C classes. Then, the probability of prompt $\mathbf{t}^r$ being classified as class y is:

$$P_{rh}(y|\mathbf{t^r}) = \frac{1}{L}\sum_{l=1}^{L}\frac{\exp\Big(\cos(\hat{\mathbf{t}}_y^{h,l},\mathbf{t}^r)/\tau\Big)}{\sum_{c=1}^{C}\exp\Big(\cos(\hat{\mathbf{t}}_c^{h,l},\mathbf{t}^r)/\tau\Big)}, \tag{8}$$

where $\hat{\mathbf{t}}_y^{h,l} = f_{post}\Big(g_T(f_{pre}(t_y^l))\Big)$. We call this variant of LASP Zero-shot LASP (LASP-Z) as no visual samples at are used for the downstream adaptation. See Fig. 3 for an overview of LASP-Z.

5 Experiments

Following (Radford et al., 2019; Zhou et al., 2022a), we mainly evaluated the accuracy of our approach on generalization to novel classes (i.e. zero-shot recognition) for 11 datasets in total. Each dataset is split into two equal partitions with disjoint classes, named *base* and *new*. We trained our model using text-image pairs from the base classes and test on both base and new classes. To further analyze the performance of our approach, we also report results for the cross-dataset transfer and domain generalization settings.

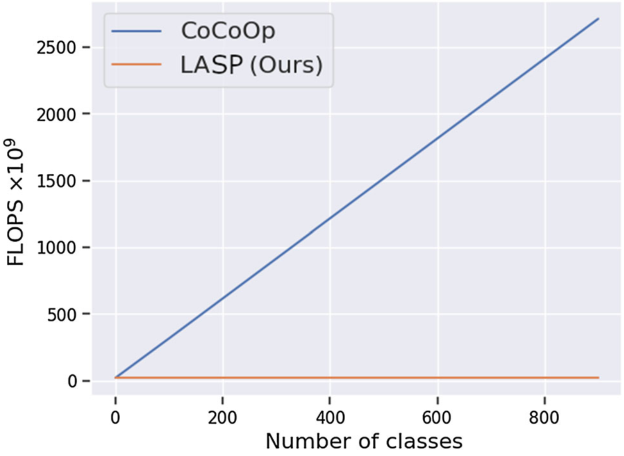


Fig. 4 Comparison between LASP and CoCoOp in terms of number of FLOPs. While the inference cost of LASP remains largely constant with respect to the number of classes, CoCoOp's cost increases linearly (from around ≈ 20 GFLOPs for 1 class to over 2500 GFLOPs for 1000)

Datasets: We used 11 in total, namely: 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 & Zisserman, 2008), Food101 (Bossard et al., 2014), FGVC Aircraft (Maji et al., 2013), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019) and UCF-101 (Soomro et al., 2012).

Models: For all experiments, unless otherwise specified, we used a pretrained CLIP model with a ViT-B/16 image encoder, $M = 4$ learnable prompts and 16 samples per class. The number of groups G (when used) is set to 3. In all experiments, we report the average across 3 runs.

Training: For LASP and LASP-V, largely, we followed the training procedure described in CoOp (Zhou et al., 2022) and CoCoOp (Zhou et al., 2022a) (i.e. same image augmentation, SGD with initial learning rate of 0.002 and a cosine annealing scheduler with 1 epoch of warm-up). In Eq. 6, α_{VL} was set to 1 and α_T to 20. The number of textual templates L was set to 34. The templates were taken from CoOp and CLIP. For LASP-Z, as no images are used during training, we increase the scheduler length to 50 epochs-equivalent and re-adjust the learning rate to 0.08. All training and testing was done on a single NVIDIA V100 GPU (except for ImageNet where 4 GPUs were used). The code was implemented using PyTorch (Paszke et al., 2017).

Methods compared: We report the performance of LASP and its improved version trained with virtual classes (LASP-V). For LASP-V, the *class names only* of the novel classes are used during training as virtual classes. We also study the impact of adding other types of virtual classes. The direct baseline that our method is compared with is CoOp (Zhou et al., 2022), as we add the proposed components on top of it. Note that both methods have *exactly* the same inference (as our method adds in addition a text-to-text loss during training). We also compare with ProDA (Lu et al., 2022) and CoCoOp (Zhou et al., 2022a) which conditions the prompts on image features and hence induces significant additional computation during inference. See also Fig. 4 for a comparison.

5.1 Comparison with State-of-the-Art

Standard setting of Zhou et al. (2022a): Table 1 compares our approach with the current state-of-the-art. We conclude:

- **Conclusion 1: In terms of harmonic mean, LASP outperforms all methods by large margin.** It outperforms, on average, the second best (ProDA) by $>$ 2%. The improvement on specific datasets is even bigger (e.g. $>$ 3% on Flowers102, $>$ 11% on EuroSAT, $>$ 3% on UCF101).
- **Conclusion 2: On the novel classes, LASP outperforms all methods by large margin.** It is the first reported method outperforming CLIP by 0.68% (but notice that CLIP performs very poorly on the bases classes). It also outperforms ProDA (third best) by $>$ 2.5%. Again, compared to ProDA, the improvement on specific datasets is even bigger (e.g. $>$ 5% on Flowers102, $>$ 3% on Food101, $>$ 11% on EuroSAT, $>$ 6% on UCF101).
- **Conclusion 3: On new classes, LASP with virtual classes has significant impact for specific datasets**. These include datasets with informative class names like EuroSAT and DTD where the improvement over LASP is $\sim$ 5.5% and $\sim$ 4.0%, respectively.

Generalized zero-shot setting: The current evaluation protocol used in Zhou et al. (2022), Zhou et al. (2022a) computes accuracy, considering the base and new classes in isolation. That is, the two disjoint sets, consisting of C_{base} and C_{novel} classes (i.e., $C = C_{base} + C_{novel}$), are each evaluated using a C_{base}-way and a C_{novel}-way classifier, respectively. A more realistic evaluation protocol should consider the classes across both subsets, base and novel, jointly as in practice one would expect to run the same classifier across the combined sets. In this instance, a C-way classifier, that includes the class prototypes from both the base and new subsets would be used when evaluating either of them. Beyond increasing the difficulty, this setting better expose cases where overfitting to base classes occurs.

We report results using this setting in Table 3. To ground the results, as no pretrained models were available, we retrain CoCoOp using the official code released by the authors. As it can be observed, the same conclusions, previously made using the protocol proposed in Zhou et al. (2022) hold true.

Table 1 Comparison with the state-of-the-art on 11 datasets

Dataset	Set	CLIP	CoOp	CoCoOp	ProDA	LASP (Ours)	LASP-V (Ours)	Δ
Average	Base	69.34	82.69	80.47	81.56	82.70	**83.18**	+0.49
	New	74.22	63.22	71.69	72.30	74.90	**76.11**	+1.89
	H	71.70	71.66	75.83	76.65	78.61	**79.48**	+2.83
ImageNet	Base	72.43	**76.47**	75.98	75.40	76.20	76.25	−0.22
	New	68.14	67.88	70.43	70.23	70.95	**71.17**	+0.74
	H	70.22	71.92	73.10	72.72	73.48	**73.62**	+0.52
Caltech101	Base	96.84	98.00	97.96	**98.27**	98.10	98.17	−0.10
	New	94.00	89.91	93.81	93.23	94.24	**94.33**	+0.33
	H	95.40	93.73	95.84	95.86	96.16	**96.21**	+0.35
OxfordPets	Base	91.17	93.67	95.20	95.43	**95.90**	95.73	+0.30
	New	97.26	95.29	97.69	97.83	**97.93**	97.87	+0.04
	H	94.12	94.47	96.43	96.62	**96.90**	96.79	+0.16
Stanford Cars	Base	63.37	**78.12**	70.49	74.70	75.17	75.23	−2.89
	New	**74.89**	60.40	73.59	71.20	71.60	71.77	−3.12
	H	68.85	68.13	72.01	72.91	73.34	**73.46**	+0.55
Flowers102	Base	72.08	97.60	94.87	**97.70**	97.00	97.17	−0.53
	New	**77.80**	59.67	71.75	68.68	74.00	73.53	−4.27
	H	74.83	74.06	81.71	80.66	83.95	**83.71**	+2.00
Food101	Base	90.10	88.33	90.70	90.30	91.20	**91.20**	+0.50
	New	91.22	82.26	91.29	88.57	91.70	**91.90**	+0.61
	H	90.66	85.19	90.99	89.43	91.44	**91.54**	+0.55
FGVC Aircraft	Base	27.19	**40.44**	33.41	36.90	34.53	38.05	−2.39
	New	**36.29**	22.30	23.71	34.13	30.57	33.20	−3.09
	H	31.09	28.75	27.74	**35.46**	32.43	**35.46**	0.00
SUN397	Base	69.36	80.60	79.74	78.67	80.70	**80.70**	+0.10
	New	75.35	65.89	76.86	76.93	78.60	**79.30**	+2.37
	H	72.23	72.51	78.27	77.79	79.63	**80.00**	+1.73
DTD	Base	53.24	79.44	77.01	80.67	81.40	**81.10**	+1.53
	New	59.90	41.18	56.00	56.48	58.60	**62.57**	+3.10
	H	56.37	54.24	64.85	66.44	68.14	**70.64**	+4.20
EuroSAT	Base	56.48	92.19	87.49	83.90	94.60	**95.00**	+2.81
	New	64.05	54.74	60.04	66.00	77.78	**83.37**	+17.37
	H	60.03	68.9	71.21	73.88	85.36	**88.86**	+14.98
UCF101	Base	70.53	84.69	82.33	85.23	84.77	**85.53**	+0.30
	New	77.50	56.05	73.45	71.97	78.03	**78.20**	+0.70
	H	73.85	67.46	77.64	78.04	81.26	**81.70**	+3.66

We provide the results of LASP and LASP trained with virtual classes (LASP-V). Δ denotes the absolute improvement of our best variant, LASP-V, over the previous best result

Cross-Dataset Transfer setting: Following (Zhou et al., 2022a), we measure how well the soft prompts learned on ImageNet perform when evaluated on different datasets. In this setting, the training is performed on images from all 1,000 classes, using 16 images for each class. As the results from Table 5 show, our approach surpasses CoOp by 2.5% while outperforming the more computationally demanding CoCoOp (0.8% better on average).

Domain generalization setting: Following the encouraging results reported in Zhou et al. (2022), Zhou et al. (2022a) on domain generalization, herein, we attempt to evaluate whether our approach can improve the quality of the leaned prompts under domain shift too. To this end, we trained LASP on all classes from ImageNet (16-shot setting) and evaluate the learned prompts on 5 datasets with class names compatible with those of ImageNet, but different data distribution. Following (Zhou et al., 2022), we used ImageNet (Deng et al., 2009) as the source dataset, and ImageNetV2 (Recht et al., 2019), ImageNet-Sketech (Wang et al., 2019), ImageNet-A (Hendrycks et al., 2021) and ImageNet-R (Hendrycks et al., 2021) as the test datasets.

Table 2 Performance analysis for LASP-Z

Dataset	Set	CLIP Zhou et al. (2022b)	LASP-Z w/o $f_{pre/post}$	LASP-Z (Ours)	Δ
Average	Base	69.34	65.21	**70.33**	**+0.99**
	New	74.22	71.58	**75.54**	**+1.32**
	H	71.70	68.24	**72.84**	**+1.14**
ImageNet	Base	72.43	67.90	**73.61**	**+1.18**
	New	68.15	65.98	**69.72**	**+1.57**
	H	70.22	66.92	**71.61**	**+1.39**
Caltech101	Base	**96.84**	92.85	**96.81**	−0.03
	New	94.00	93.77	**95.23**	**+1.23**
	H	95.40	93.31	**96.01**	**+0.61**
OxfordPets	Base	91.17	91.00	91.46	**+0.29**
	New	97.26	95.97	97.40	**+0.14**
	H	94.12	93.41	94.33	**+0.21**
Stanford Cars	Base	63.37	57.38	**64.57**	**+1.20**
	New	74.89	65.33	**75.53**	**+0.64**
	H	68.85	61.09	**69.62**	**+0.77**
Flowers102	Base	72.08	58.11	**72.50**	**+0.42**
	New	**77.80**	74.63	77.13	**−0.67**
	H	**74.83**	65.34	74.74	−0.09
Food101	Base	**90.10**	89.05	89.28	**−0.82**
	New	91.22	90.85	**91.47**	**+0.25**
	H	**90.66**	89.94	90.36	**−0.30**
FGVC Aircraft	Base	27.19	13.14	**27.23**	+0.04
	New	**36.29**	26.83	34.19	**−2.10**
	H	**31.09**	17.64	30.31	**−0.78**
SUN397	Base	69.36	67.03	**73.05**	**+3.69**
	New	75.35	74.05	**77.81**	**+2.46**
	H	72.23	70.36	**75.35**	**+3.12**
DTD	Base	53.24	56.94	**57.33**	**+4.09**
	New	59.90	59.00	**60.53**	**+0.63**
	H	56.37	57.95	**58.88**	**+2.51**
EuroSAT	Base	56.48	55.96	**56.62**	**+0.14**
	New	64.05	69.60	**73.80**	**+9.75**
	H	60.02	62.03	**64.07**	**+4.05**
UCF101	Base	70.53	68.01	**71.19**	**+0.66**
	New	77.50	71.19	**78.20**	**+0.70**
	H	73.85	69.56	**74.53**	**+0.68**

As the results from Table 6 show, with the exception of ImageNet-V2, our approach outperforms all prior work, showing strong domain generalization capabilities.

5.2 Zero-Shot Adaptation Setting

Departing from the few-shot adaptation experiments of the previous section, herein, we evaluate the zero-shot V&L learning capabilities of the proposed image-free LASP-Z on the same set of 11 datasets used for few-shot evaluation. While the base/new partitions are no longer meaningful in this case, as no images are used, we report results preserving the data split structure to facilitate comparisons across different settings (i.e. few-shot and zero-shot adaptation). The results reported in Table 2 show that our zero-shot adaptation approach improves upon CLIP by +1.14% on average across 11 datasets, outperforming it on 8/11 datasets by up to 4% (on EuroSAT). Moreover, Table 2 shows the importance of the

Table 3 Comparison with the state-of-the-art for the generalized zero-shot setting

	Base	New	H
(a) Average over 11 datasets			
CoCoOp	72.46	64.77	68.39
LASP	76.59	67.55	71.78
LASP-V	**77.23**	**68.52**	**72.61**
(b) ImageNet			
	Base	New	H
CoCoOp	71.90	67.50	69.63
LASP	**72.00**	67.33	69.51
LASP-V	71.90	**68.00**	**69.78**
(c) Caltech101			
	Base	New	H
CoCoOp	95.20	90.67	92.87
LASP	94.87	92.20	93.51
LASP-V	95.54	**92.78**	**94.13**
(d) OxfordPets			
	Base	New	H
CoCoOp	91.01	93.10	92.04
LASP	91.53	92.87	92.19
LASP-V	**92.23**	**93.17**	**92.69**
(e) StanfordCars			
	Base	New	H
CoCoOp	67.26	**69.43**	68.33
LASP	**72.27**	68.73	**70.45**
LASP-V	71.00	68.50	69.27
(f) Flowers102			
	Base	New	H
CoCoOp	86.73	64.63	74.06
LASP	90.97	68.80	78.34
LASP-V	**92.20**	**69.93**	**79.53**
(g) Food101			
	Base	New	H
CoCoOp	85.73	85.50	85.61
LASP	87.53	**87.17**	87.34
LASP-V	**87.73**	**87.17**	**87.45**
(h) FGVCAircraft			
	Base	New	H
CoCoOp	24.50	25.93	25.19
LASP	24.33	27.03	25.61
LASP-V	**28.77**	**27.80**	**28.27**
(i) SUN397			
	Base	New	H
CoCoOp	71.13	67.76	69.40
LASP	**72.60**	67.21	69.80
LASP-V	72.55	**69.11**	**70.79**
(j) DTD			
	Base	New	H
CoCoOp	59.33	42.70	49.65
LASP	**67.53**	46.93	55.37
LASP-V	65.67	**49.90**	**56.71**
(k) EuroSAT			
	Base	New	H
CoCoOp	69.20	39.23	50.14
LASP	89.38	54.87	67.99
LASP-V	**90.80**	**56.80**	**69.88**
(l) UCF101			
	Base	New	H
CoCoOp	75.16	66.10	70.34
LASP	79.57	70.00	74.47
LASP-V	**81.20**	**70.60**	**75.52**

We have re-trained CoCoOp using the officially released code
Best method for each dataset and testing setting (base, new and H (harmonic mean))

proposed augmentations in LASP-Z. As it can be observed, without the augmentations, the accuracy of LASP-Z significantly deteriorates. Overall, we conclude:

- **Conclusion 4: Zero-shot LASP (LASP-Z) significantly outperforms CLIP for the zero-shot adaptation setting.** For this purpose, the proposed language-based augmentations are necessary.

5.3 Ablation Studies

Effect of different LASP components: LASP proposes a number of contributions which are evaluated incrementally. The start point is the proposed Text-to-Text loss of Eq. 5. On top of this, we incrementally apply the grouped prompt representation (Eq. 7), and then the re-alignment module (Sect. 3.4). This gives rise to LASP. Finally, we add virtual classes giving rise to LASP-V. Our baseline is CoOp. From the results of Table 4, we conclude:

- **Conclusion 5: Our idea in its plain form (Text-to-Text loss) outperforms its direct baseline (CoOp) by large margin**. Specifically, it improves upon CoOp by $\sim 4.5\%$ on average, demonstrating its effectiveness.
- **Conclusion 6: All components are needed to obtain high accuracy.**

Effect of size and content of the textual prompts: Herein, we study the effect of the size L and the content of the set

Table 4 Effect of different LASP components

Dataset	Set	Baseline Zhou et al. (2022)	Text-to-Text	+Grouped	+Align (LASP)	+ Virtual (LASP-V)
Average	Base	82.69	81.26	81.87	82.70	83.18
	New	63.22	71.54	73.48	74.90	**76.11**
	H	71.66	76.09	77.44	78.61	**79.48**
ImageNet	Base	**76.47**	75.97	76.20	76.20	76.25
	New	67.88	70.31	70.70	70.95	**71.17**
	H	71.92	73.03	73.34	73.48	**73.62**
Caltech101	Base	98.00	97.70	97.97	98.10	98.17
	New	89.91	94.08	94.27	94.24	**94.33**
	H	93.73	95.85	96.08	96.16	**96.21**
OxfordPets	Base	93.67	95.13	95.63	95.90	**95.73**
	New	95.29	96.23	97.87	97.93	**97.87**
	H	94.47	95.68	96.73	96.90	**96.79**
Stanford Cars	Base	**78.12**	72.46	73.50	75.17	75.23
	New	60.40	71.80	72.10	71.60	71.77
	H	68.13	72.19	72.93	73.34	**73.46**
Flowers102	Base	97.60	96.47	96.80	97.00	97.17
	New	59.67	70.70	74.00	74.00	73.53
	H	74.06	81.59	83.87	83.95	**83.71**
Food101	Base	88.33	90.30	91.00	91.20	**91.20**
	New	82.26	90.73	90.87	91.70	**91.90**
	H	85.19	90.51	90.93	91.44	**91.54**
FGVC Aircraft	Base	**40.44**	32.63	33.05	34.53	38.05
	New	22.30	30.46	31.80	30.57	33.20
	H	28.75	31.57	32.41	32.43	**35.46**
SUN397	Base	80.60	80.20	80.55	80.70	**80.70**
	New	65.89	75.56	77.11	78.60	**79.30**
	H	72.51	77.81	78.79	79.63	**80.00**
DTD	Base	79.44	79.13	80.50	81.40	**81.10**
	New	41.18	52.10	56.20	58.60	**62.57**
	H	54.24	62.82	66.19	68.14	**70.64**
EuroSAT	Base	92.19	91.23	91.90	94.60	**95.00**
	New	54.74	63.16	66.37	77.78	**83.37**
	H	68.90	74.64	77.07	85.36	**88.86**
UCF101	Base	84.69	82.70	83.47	84.77	**85.53**
	New	56.05	71.80	77.07	78.03	**78.20**
	H	67.46	76.86	80.14	81.26	**81.70**

Text-to-Text is Eq. 5, only. On top of this, we incrementally apply the grouped prompt of Eq. 7, and the re-alignment module of Sect. 3.4. Up to this point, this is equiv. to LASP. Finally, we add virtual classes (equiv. to LASP-V). Baseline is CoOp

of the textual prompts used by our method in Eq. 4. For simplicity, we report results using our Text-to-Text loss (Eq. 5), only. The hand-crafted templates are increased to 100 by including the rest of the prompts defined in CLIP (Radford et al., 2021), while their number is reduced to 1 by using the following template only: `a photo of {}`. Random templates are produced by sampling grammatically plausible random sentences that contain incoherent words, with length between 5 and 20 words. The class names are inserted at the end of these random templates. All variations use the same training scheduler and hyperparameters, except for the case of random templates, where $\alpha_{TT} = 5$.

Table 10 shows our results. We importantly note that the accuracy on the base classes remains similar across all settings (not shown in the table). Moreover, we conclude:

Table 5 Comparison with state-of-the-art for the cross-dataset transfer setting

	Source	Target										
	ImageNet	Caltech101	OxfordPets	StanfordCars	Flowers102	Food101	FGVCAircraft	SUN397	DTD	EuroSAT	UCF101	Average
CoOp	**71.51**	93.70	89.14	64.51	68.71	85.30	18.47	64.15	41.92	46.39	66.55	63.88
CoCoOp	71.02	94.43	**90.14**	65.32	**71.88**	86.06	22.94	**67.36**	**45.73**	45.37	68.21	65.74
LASP	71.30	**94.50**	89.36	**66.20**	71.74	**86.40**	**23.03**	67.0	45.54	**48.50**	**68.24**	**66.52**

- **Conclusion 7: The exact choice of the templates might not be so significant for the few-shot setting.**
- **Conclusion 8: For the case of novel classes, both the number and the content of the templates are important to obtain high accuracy.**

Effect of type of loss: In Table 7, we vary the choice of loss in LASP, i.e. we replace the Cross-Entropy (CE) with an L_2 and L_1 loss. Again, for simplicity, we report results using our Text-to-Text loss (Eq. 5), only.

- **Conclusion 9: The proposed CE loss based formulation outperforms other losses for LASP.**

Effect of out-domain distractors: Motivated by the recent work of Ren et al. (2022) suggesting that CLIP's performance drops as the number of classes used for testing increases, we introduce a new evaluation setting: Firstly, we select 4 test datasets with clear disjoint domains: EuroSAT (10 satellite terrain types), Food101 (101 food names), Flowers102 (102 flower names) and OxfordPets (37 dog and cat breed names). At test time, we define the classifier across the union of classes across all 4 datasets (250 classes in total). Note that LASP-V is the only method that benefits from knowledge of this expanded vocabulary during training. From Table 8, we can conclude:

- **Conclusion 10: The models are somewhat robust to out-of-domain distractors.** Specifically, the drop in accuracy is moderate (typically 1-2%). The exception is EuroSAT where the number of classes increases 25×. Importantly, LASP-V manages to largely recover the lost accuracy.

Effect of in-domain distractors: Expanding on the idea from the previous section, herein, we propose to test the performance of the current soft prompting methods with in-domain distractors. Unlike the case of out-of-domain distractors, the in-domain distractors are selected such that they are closely related to the current dataset/classes, being part of the same super-category. We performed experiments on two datasets: Food101 and Flowers102. For Flowers102, we added 65 new class names while, for Food101, 53 new classes. Note again that, with the exception of LASP-V, the classes are only used at test time as distractors expanding the C-way classifier by 65 and 53, respectively. From the results of Table 9, we conclude:

- **Conclusion 11: In-domain distractors significantly increase the problem difficulty.** Specifically, the drop in accuracy is large (4-7%). LASP-V manages to recover part of the lost accuracy.

Table 6 Comparison with state-of-the-art for the domain generalization setting

	Learnable?	Source	Target			
		ImageNet	ImageNetV2	ImageNet-Sketch	ImageNet-A	ImageNet-R
CLIP		66.73	60.83	46.15	47.77	73.96
CoOp	✓	**71.51**	**64.20**	47.99	49.71	75.21
CoCoOp	✓	71.02	64.07	48.75	50.63	76.18
LASP	✓	71.10	63.96	**49.01**	**50.70**	**77.07**

Table 7 Effect of type of loss

Set	CE	L_1	L_2
Base	81.26	**81.50**	81.47
New	**71.54**	66.01	65.80
H	**76.09**	73.54	72.80

For simplicity, we report results using our Text-to-Text loss (Eq. 5), only

Table 8 Effect of out-domain distractors

Method	w/o distractors			with distractors		
(a) EuroSAT						
	Base	New	H	Base	New	H
LASP	86.25	64.63	73.89	86.00	55.80	67.68
LASP-V	**90.00**	**65.73**	**75.97**	**90.80**	**59.87**	**72.16**
(b) Food101						
Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	**87.17**	87.53	87.34	**87.01**	86.90	86.95
LASP-V	**87.17**	**87.63**	**87.39**	86.99	**87.10**	**87.04**
(c) Flowers102						
Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	90.97	67.8	77.69	90.0	67.10	76.68
LASP-V	**93.20**	**69.93**	**79.9**	**92.05**	**69.08**	**78.92**
(d) OxfordPets						
Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	**92.53**	**94.20**	**91.52**	91.53	92.60	92.06
LASP-V	92.25	93.97	93.10	**92.23**	**93.17**	**92.69**

w/o distractors are the results on the generalized zero-shot setting

Table 9 Effect of in-domain distractors

Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
(a) Food101.						
LASP	**87.17**	87.53	87.34	82.70	83.47	83.08
LASP-V	**87.17**	**87.63**	**87.39**	**83.11**	**83.95**	**83.52**
(b) Flowers102.						
Method	w/o distractors			with distractors		
	Base	New	H	Base	New	H
LASP	90.97	67.80	77.69	80.16	62.50	70.23
LASP-V	**93.20**	**69.93**	**79.90**	**83.95**	**65.31**	**73.47**

w/o distractors are the results on the generalized zero-shot setting evaluation

Table 10 Effect of dictionary size and content on new classes

#Templates	1	34	100
(a) DTD			
Text-to-Text (R)	49.02	51.63	52.64
Text-to-Text	**50.73**	**52.10**	**56.53**
(b) EuroSAT			
#Templates	1	34	100
Text-to-Text (R)	55.01	59.90	62.10
Text-to-Text	**56.97**	**63.16**	**65.13**
(c) UCF101			
#Templates	1	34	100
Text-to-Text (R)	67.50	68.60	70.03
Text-to-Text	**71.36**	**71.80**	**72.77**

Accuracy on the base classes remains similar across all settings, hence it is omitted. 34 templates were used for the paper's main results. For simplicity, we report results using our Text-to-Text loss (Eq. 5), only. Text-to-Text (R) denotes models trained using randomly constructed templates

Best method per dataset, within each setting defined by the number of textual prompts

Table 11 Impact of noise value τ on the overall performance of LASP-Z

s	0	0.05	0.15	0.3
LASP-Z	70.81	71.90	72.84	72.70

Results aggregated across 10 datasets

Effect of text-based augmentations: As detailed in Sect. 3.2, one way to view the proposed text-to-text component is as a direct extension of image-style augmentations to the language domain. To explore this, we construct a variation of the Oxford Pets dataset in which all test images are rotated by 90 or 180^o. We select Oxford Pets as the rotated pets images are far from the natural distribution of photos. During the training of the LASP model the images are kept under their original rotation (i.e. none) while the textual prompts are

Table 12 Impact of number of augmentations on the overall performance of LASP-Z

s	0	10	15	20
LASP-Z	69.70	71.73	72.40	72.45

Results aggregated across 10 datasets

augmented with extra keywords such as: “rotate”, “upside down”, “angled” etc. Based on the results from Table 13, we can conclude:

- **Conclusion 12: Text-based augmentations are a viable solution for increased robustness.**

Effect of noise level and transformation on LASP-Z. To alleviate the issue that the text features are not a perfect proxy for the vision domain, we explore the points located in their vicinity, hence, increasing the likelihood of overlapping with the data distribution from the vision domain. This is achieved, in practice, by adding Gaussian noise in the output space or by adjusting the prompts in the input space. In Table 11, we analyse the impact of the noise magnitude s (i.e, $\mathbf{x} \sim s\mathcal{N}(\mu, \sigma^2)$) on the performance of the model. While the model is overall resilient to the exact value of s, removing it completely leads to performance inferior to that of CLIP. We conclude that adding noise does not only help bridge the domain gap, but also alleviate overfitting.

Similar results can be observed for varying the number of augmentations. Here, we note that a higher number leads to better results as intuitively they allow for the exploration of more points around the class centroid.

Can LASP-Z be used as initialisation for LASP? LASP-Z tries to fully leverage the joint vision-language embedding space learned by CLIP, moving the optimization process fully in the text domain. While the text alone is a good proxy for representing visual data, due to the domain gap that naturally occurs as part of the contrastive training, it is not a full substitute for the visual data. Due to the above, when initialising LASP/LASP-V from LASP-Z weights, we observed no additional gains as the visual samples provided include the queues provided by the text training.

6 Conclusions

In this paper, we introduced LASP - a language aware soft prompting method for V&L adaptation that is shown to outperform prior work by large margin. Specifically, we made the following contributions: *Firstly*, we introduced a novel text-to-text loss that largely alleviates the problem of base-class overfitting. *Secondly*, we proposed a *grouped* language-aware prompting for learning more specialized and stronger prompt representations. *Thirdly*, we identified a visual-language misalignment within LASP and propose a re-calibration mechanism to address it. *Fourthly*, we showed that our approach, unlike prior work, is amenable to, including during training, *virtual classes*, i.e. class names for which no visual samples are available, significantly increasing the robustness of the learned prompts. *Fifthly*, we presented a zero-shot variant of LASP (LASP-Z) where no visual samples at all are available for the downstream task and showed its superiority over CLIP. We hope that LASP/LASP-V/LASP-Z will serve as a strong baseline for future works in the area of few-shot adaptation for V&L models.

Data Availability The datasets used and/or analyzed during the current study are available from their respective authors: 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 & Zisserman, 2008), Food101 (Bossard et al., 2014), FGVC Aircraft (Maji et al., 2013), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019) and UCF-101 (Soomro et al., 2012).

Table 13 Effect of text-based positional augmentations

Dataset	Text	clean			rotated		
	Augm	Base	New	H	Base	New	H
Oxford pets	✗	95.73	97.87	96.79	71.20	72.41	71.79
	✓	95.64	97.85	96.73	72.14	72.70	72.41

Appendix A: Training and Inference Speed Considerations

Once trained, LASP is as fast as CoOp. In terms of training time, the cost of adapting the text encoder for LASP, CoOp and CoCoOp is $G \cdot M \cdot C_T$, $M \cdot C_T$ and $B \cdot M \cdot C_T$, respectively, where B is the batch size, M is the number of classes and C_T is the text encoder's cost for 1 sample. In practice, for $B = 32$ and $G = 4$ LASP is, on average, 2.3× slower than CoOp and up to 10× faster than CoCoOp. Note that these numbers are subject to the implementation optimizations made for each method. For G=1, LASP's training cost is similar with that of CoOp's while losing only 0.5% on average, being slightly (1.05–1.2×) slower due to the additional gradients computed with respect to the weights of the layer norms inside the vision transformer.

Appendix B: Implementation Details

Hand-engineered prompts set ζ: Unless otherwise specified, we used the following set of hand-engineered templates (borrowed from CLIP and CoOp):

```
"a photo of a {}, a type of flower.",
"a photo of a person doing {}.",
"a centered satellite photo of {}.",
"a photo of a {}, a type of aircraft.",
"{} texture.",
"itap of a {}.",
"a bad photo of the {}.",
"a origami {}.",
"a photo of the large {}.",
"a {} in a video game.",
"art of the {}.",
"a photo of the small {}.",
"a photo of a {}.",
"a photo of many {}.",
"a photo of the hard to see {}.",
"a low resolution photo of the {}.",
"a rendering of a {}.",
"a bad photo of the {}.",
"a cropped photo of the {}.",
"a pixelated photo of the {}.",
"a bright photo of the {}.",
"a cropped photo of a {}.",
"a photo of the {}.",
"a good photo of the {}.",
"a rendering of the {}.",
"a close-up photo of the {}.",
"a low resolution photo of a {}.",
"a rendition of the {}.",
"a photo of the clean {}.",
"a photo of a large {}.",
"a blurry photo of a {}.",
"a pixelated photo of a {}.",
"itap of the {}.",
"a jpeg corrupted photo of the {}.",
"a good photo of a {}."
```

Note that {} represent the placeholder for the location of the class name w.

Random prompts: For the experiments involving random prompts, we list bellow a few such examples:

```
"Ports, waterways, the subfield that {}.",
"In TCP, prepared mind, but some others, Milatiai, appear
to have {}.",
"Iron Age, The Eastern Shore of Virginia residents age 5
and {}.",
"Cat mostly all with {}.",
"Wind erosion. go unnoticed-it was {}.",
"River Delta, on six different {}.",
"12 hours. few times every million {}.",
etc.
```

Additional class names for in-domain ablation: Below, we list the manually defined in-domain class name distractors used to produce the results for with in-domain distractors. For Food-101, we added the following classes:

['aroma', 'bagel', 'batter', 'beans', 'biscuit', 'broth', 'burger', 'burrito', 'butter', 'candy', 'caramel', 'caviar', 'cheese', 'chili', 'chimichanga', 'cider', 'cocoa', 'coffee', 'cobbler', 'empanada', 'fish', 'flour', 'ketchup', 'margarine', 'mousse', 'muffin', 'mushrooms', 'noodle', 'nuts', 'oil', 'olives', 'pudding', 'raclette', 'rice', 'salad', 'salsa', 'sandwitch', 'soda', 'tea', 'stew', 'toast', 'waffles', 'yogurt', 'wine', 'sopapillas', 'chilli con carne', 'banana bread', 'yorkshire pudding', 'spaghetti carbonara', 'roast potatoes', 'sausage ragu', 'avocado panzanella', 'lamb biryani']

Respectively, for Flowers102 dataset:

['Agapanthus', 'Allium', 'Alstroemerias', 'Amaranthus', 'Astilbe', 'Begonia', 'brunia', 'California poppy', 'Calla lily', 'Campanula', 'Carnations', 'Celosia', 'Chrysanthemum', 'Cornflower', 'Delphinium', 'Dianthus', 'Dusty Miller', 'Eryngium', 'Freesia', 'Gardenias', 'Gerbera daisies', 'Gladiolus', 'Gypsophila', 'Hydrangea', 'Hypericum', 'Kale', 'Larkspur', 'Liatris', 'Lilies', 'Lisianthus', 'Orchids', 'Peony', 'Periwinkle', 'Ranunculus', 'Scabiosa', 'Sunflowers', 'Yarrow', 'Zinnia', 'Bellflower', 'Bleeding Heart', 'Browallia', 'Bugleweed', 'Butterfly Weed', 'Calendula', 'Cardinal Flower', 'Celosia', 'Clary Sage', 'Coreopsis', 'Forget-Me-Not', 'Freesias', 'Gaillardia', 'Glory of the Snow', 'Heather', 'Hollyhock', 'Hyssop', 'Impatiens', 'Jack-in-the-Pulpit', 'Lilac', 'Lilies', 'Lobelia', 'Periwinkle', 'Rue', 'Thunbergia', 'Verbena', 'Wisteria']

Attribute selection: For simplicity we selected (relatively) generic attributes, i.e. attributes that are class agnostic. The attributes were sampled automatically by prompting a LLM model for suggestions (i.e. To list a set of possible transformations that could be applied to a given image). Examples of attributes: blurred image, sharpened image, sepia- toned image, high contrast image, tilted image, vignette effect image, pencil sketch image, rotated image, duotone image, posterized image etc.

References

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., & Hasson, Y., et al. (2022). Flamingo: A visual language model for few-shot learning. arXiv:2204.14198

Albuquerque, I., Naik, N., Li, J., Keskar, N., & Socher, R. (2020). Improving out-ofdistribution generalization via multi-task self-supervised pretraining. arXiv:2003.13525

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. arXiv:1607.06450

Balaji, Y., Sankaranarayanan, S., & Chellappa, R. (2018). Metareg: Towards domain generalization using meta-regularization. In *Advances in neural information processing systems* (Vol. 31).

Bossard, L., Guillaumin, M., & Gool, L. V. (2014). Food-101-mining discriminative components with random forests. In *European conference on computer vision* (pp. 446–461).

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., & Dhariwal, P. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems, 33*, 1877–1901.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., & Joulin, A. (2020). Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems, 33*, 9912–9924.

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

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

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A largescale hierarchical image database. In *2009 ieee conference on computer vision and pattern recognition* (pp. 248–255).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., & Unterthiner, T., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv:2010.11929

Dou, Q., Coelho de Castro, D., Kamnitsas, K., & Glocker, B. (2019). Domain generalization via model-agnostic learning of semantic features. *Advances in Neural Information Processing Systems, 32*.

Fei-Fei, L., Fergus, R., & Perona, P. (2004). Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. In *2004 Conference on computer vision and pattern recognition workshop* (pp. 178–178).

Helber, P., Bischke, B., Dengel, A., & Borth, D. (2019). Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 12*(7), 2217–2226.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., & Dorundo, E., et al. (2021). The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 8340–8349).

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., & Song, D. (2021). Natural adversarial examples. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 15262–15271).

Hinton, G., Vinyals, O., & Dean, J., et al. (2015). Distilling the knowledge in a neural network. arXiv:1503.02531

Hu, S., Zhang, K., Chen, Z., & Chan, L. (2020). Domain generalization via multidomain discriminant analysis. In *Uncertainty in artificial intelligence* (pp. 292–302).

Huang, T., Chu, J., & Wei, F. (2022). Unsupervised prompt learning for vision-language models. arXiv:2204.03649

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., & Pham, H., et al. (2021). Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning* (pp. 4904–4916).

Krause, J., Stark, M., Deng, J., Fei-Fei, L. (2013). 3d object representations for finegrained categorization. In *Proceedings of the ieee international conference on computer vision workshops* (pp. 554–561).

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90.

Laskin, M., Lee, K., Stooke, A., Pinto, L., Abbeel, P., & Srinivas, A. (2020). Reinforcement learning with augmented data. *Advances in Neural Information Processing Systems, 33*, 19884–19895.

Lester, B., Al-Rfou, R., & Constant, N. (2021). The power of scale for parameter efficient prompt tuning. arXiv:2104.08691

Li, X. L., & Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. arXiv:2101.00190

Li, Y., Liang, F., Zhao, L., Cui, Y., Ouyang, W., & Shao, J., et al. (2021). Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. arXiv:2110.05208

Li, Z., Zhou, F., Chen, F., & Li, H. (2017). Metasgd: Learning to learn quickly for few-shot learning. arXiv:1707.09835

Liang, W., Zhang, Y., Kwon, Y., Yeung, S., & Zou, J. (2022). Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. arXiv:2203.02053

Lu, Y., Liu, J., Zhang, Y., Liu, Y., & Tian, X. (2022). Prompt distribution learning. In *Ieee conference on computer vision and pattern recognition*.

Mahajan, D., Tople, S., & Sharma, A. (2021). Domain generalization using causal matching. In *International conference on machine learning* (pp. 7313–7324).

Maji, S., Rahtu, E., Kannala, J., Blaschko, M., & Vedaldi, A. (2013). Fine-grained visual classification of aircraft. arXiv:1306.5151

Nichol, A., Achiam, J., & Schulman, J. (2018). On first-order meta-learning algorithms. arXiv:1803.02999

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

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

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., & DeVito, Z., et al. (2017). Automatic differentiation in pytorch.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., et al. (2021). Learning transferable visual models from natural language supervision. In *International conference on machine learning* (pp. 8748–8763).

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog, 1*(8), 9.

Rajeswaran, A., Finn, C., Kakade, S. M., & Levine, S. (2019). Meta-learning with implicit gradients. *Advances in Neural Information Processing Systems, 32*.

Recht, B., Roelofs, R., Schmidt, L., & Shankar, V. (2019). Do imagenet classifiers generalize to imagenet? In *International conference on machine learning* (pp. 5389–5400).

Ren, S., Li, L., Ren, X., Zhao, G., & Sun, X. (2022). Rethinking the openness of clip. arXiv:2206.01986

Schick, T., & Schütze, H. (2020a). Exploiting cloze questions for few shot text classification and natural language inference. arXiv:2001.07676

Schick, T., & Schütze, H. (2020b). It's not just size that matters: Small language models are also few-shot learners. arXiv:2009.07118

Shao, R., Lan, X., Li, J., & Yuen, P. C. (2019). Multiadversarial discriminative deep domain generalization for face presentation attack detection. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 10023–10031).

Shi, Y., Yu, X., Sohn, K., Chandraker, M., & Jain, A. K. (2020). Towards universal representation learning for deep face recognition. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 6817–6826).

Song, Y., Wang, T., Cai, P., Mondal, S. K., & Sahoo, J. P. (2023). A comprehensive survey of fewshot learning: Evolution, applications, challenges, and opportunities. *ACM Computing Surveys*.

Soomro, K., Zamir, A.R., & Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv:1212.0402

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*.

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

Xian, Y., Schiele, B., & Akata, Z. (2017). Zeroshot learning-the good, the bad and the ugly. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4582–4591).

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

Yao, L., Huang, R., Hou, L., Lu, G., Niu, M., Xu, H., et al. (2021). Filip: Finegrained interactive language-image pretraining. arXiv:2111.07783

Yarats, D., Zhang, A., Kostrikov, I., Amos, B., Pineau, J., & Fergus, R. (2021). Improving sample efficiency in model-free reinforcement learning from images. In *Proceedings of the aaai conference on artificial intelligence* (Vol. 35, pp. 10674–10681).

Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., & Wu, Y. (2022). Coca: Contrastive captioners are imagetext foundation models. arXiv:2205.01917.

Zhou, K., Loy, C.C., & Liu, Z. (2023). Semisupervised domain generalization with stochastic stylematch. *International Journal of Computer Vision*, 1–11.

Zhou, K., Yang, J., Loy, C.C., & Liu, Z. (2022a). Conditional prompt learning for visionlanguage models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 16816–16825).

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

Zhu, B., Niu, Y., Han, Y., Wu, Y., & Zhang, H. (2022). Prompt-aligned gradient for prompt tuning. arXiv:2205.14865