Multitask Vision-Language Prompt Tuning

Sheng Shen* Shijia Yang* Tianjun Zhang* Bohan Zhai
Joseph E. Gonzalez Kurt Keutzer Trevor Darrell
University of California, Berkeley
{sheng.s,shijiayang,tianjunz,zhaibohan,jegonzal,keutzer,trevordarrell}@berkeley.edu

Abstract

Prompt Tuning, conditioning on task-specific learned prompt vectors, has emerged as a data-efficient and parameter-efficient method for adapting large pretrained vision-language models to multiple downstream tasks. However, existing approaches usually consider learning prompt vectors for each task independently from scratch, thereby failing to exploit the rich shareable knowledge across different vision-language tasks. In this paper, we propose multitask vision-language prompt tuning (MVLPT), which incorporates cross-task knowledge into prompt tuning for vision-language models. Specifically, (i) we demonstrate the effectiveness of learning a single transferable prompt from multiple source tasks to initialize the prompt for each target task; (ii) we show many target tasks can benefit each other from sharing prompt vectors and thus can be jointly learned via multitask prompt tuning. We benchmark the proposed MVLPT using three representative prompt tuning methods, namely text prompt tuning, visual prompt tuning, and the unified vision-language prompt tuning. Results in 20 vision tasks demonstrate that the proposed approach outperforms all single-task baseline prompt tuning methods, setting the new state-of-the-art on the few-shot ELEVATER *benchmarks and cross-task generalization benchmarks. To understand where the cross-task knowledge is most effective, we also conduct a large-scale study on task transferability with 20 vision tasks in 400 combinations for each prompt tuning method. It shows that the most performant MVLPT for each prompt tuning method prefers different task combinations and many tasks can benefit each other, depending on their visual similarity and label similarity.*

1. Introduction

Recent large-scale vision-language models, pretrained on a wide variety of images with natural language supervision (*i.e.*, CLIP [67], ALIGN [38] and Florence [96]), have demonstrated strong open-set recognition abilities for image classification in-the-wild [50, 67] and open-vocabulary detection [29]. Despite the impressive zero-shot transfer capabilities, adapting these large-scale vision-language models to downstream tasks presents its own challenges. It is usually prohibitive to fine-tune the entire model due to both huge parameter sizes and well-known overfitting issues for few-shot learning.

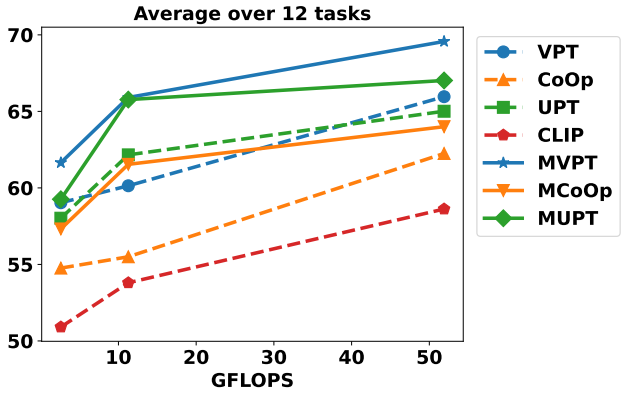


Figure 1. Our MVLPT approach (MCoOp, MVPT, MUPT) —which transfers a prompt learned from a mixture of source tasks (here, 11 Image Classification tasks) onto non-overlapped target tasks— outperforms vanilla CoOp [105], VPT [39] on 12 ELEVATER tasks by a large margin, across all CLIP model sizes (ViT-B/32, ViT-B/16 and ViT-L/14).

Such a trend emerges the essential need to study different adaptation methods [36, 37, 55], where Prompt Tuning [48, 105] has shown to be one of the most effective strategies. Typically, Prompt Tuning tunes only a small number of parameters for each task in a model's input spaces (prompt vectors) while keeping the pretrained model frozen. It was first introduced in NLP community [48, 55, 61] and has recently demonstrated superior few-shot adaptation performance [39, 105, 106] for vision-language models. CoOp [105] and VPT [39] are two representative vision-language prompt tuning methods, in which the former uses a textual prompt and the latter leverages the visual prompt.

However, on the one hand, most of these vision-language

*Equal contribution

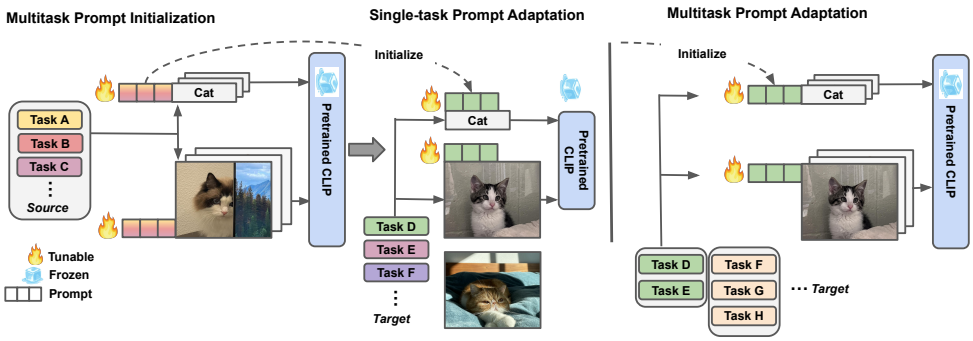


Figure 2. An illustration of our *multitask prompt initialization* (left) and *multitask prompt adaptation* (right) approaches in MVLPT. **Left**: We learn single generic source prompt vector on various *source tasks*, which is then used to initialize the prompt for each single *target task*. **Right**: After use the source prompt vector for initialization. We group relevant *target tasks* together and perform multitask prompt tuning within each group. Noted that grouping one task means single-task adaptation. (see Section 3 for details).

prompt tuning methods (*i.e.*, CoOp, VPT) focuses on learning a prompt for each downstream task independently, failing to incorporate cross-task knowledge when adapting to various downstream tasks. On the other hand, multitask learning has a rich literature [8, 80, 84, 102] for vision. Applying multitask prompt tuning to language models has also presented impressive few-shot [4,58] or zero-shot generalization capability [13, 71]. This motivates us to investigate the question: *Can vision-language model also benefit from multitask knowledge sharing via prompt tuning during adaptation?*

To this end, we propose multitask vision-language prompt tuning (MVLPT), to the best of our knowledge, the first method incorporating the cross-task knowledge into vision-language prompt tuning. MVLPT is a simple yet effective way to enable information sharing between multiple tasks. MVLPT consists of two stages: *multitask source prompt initialization* and *multitask target prompt adaptation*. Specifically, multitask prompt initialization first learns shared prompt vectors from various source tasks. Then, the shared prompt can initialize the prompt for target tasks. To adapt to target tasks, multitask prompt adaptation will group relevant tasks together then perform multitask prompt tuning within the selected groups. We remark that we could also perform single-task adaption with setting group size as one. This simple scheme enables passing cross-task knowledge from *source tasks* to *target tasks* through multitask prompt initialization, and exploiting shareable knowledge within *target tasks* via multitask prompt adaptation further.

We conduct extensive evaluations of MVLPT on 20 vision tasks in few-shot ELEVATER [50] in Section 4.2. Comparing to CoOp [105], VPT [39] and UPT (Section 3.1), MVLPT improves the baselines by 0.72%, 1.73% and 0.99% respectively and sets the new state-of-the-art on 20-shot ELEVATER benchmark. We also show the strong generalizability of MVLPT where MVLPT improves CoOp, VPT and UPT by 1.73%, 4.75% and 4.53%, respectively on cross-task generalization benchmark in Section 4.1 and study task transferability with the 20 vision tasks and in 400 combinations for each prompt method in Section 4.3.

In summary, we make the following contributions:

- We propose the multitask vision-language prompt tuning (MVLPT) framework, including multitask prompt initialization and multitask prompt adaptation, and demonstrate the efficacy for each component.
- We rigorously study the task transferability across 20 vision tasks with 400 combinations for each prompt tuning method to understand when MVLPT is most effective.
- We systematically evaluate the proposed MVLPT on the few-shot ELEVATER and cross-task generalization benchmarks, which sets the new state-of-the-art on 20-shot ELEVATER benchmark.

2. Related Work

Vision-Language Models [11, 101] align images and texts into a joint embedding space using image and text encoding, and loss functions for alignment. Traditionally, models are designed and learned independently for images and texts, connected only by a loss module. Images are encoded using hand-crafted descriptors [19, 77] or neural networks [23, 47], while texts can be encoded with pre-trained word vectors [23, 77] or frequency-based features [19, 47]. To align these modalities, metric learning [23], multi-label classification [28, 41], and n-gram language learning [49] are used.

With the rise of large-scale pretraining, vision-language models [24, 44, 51–54, 56, 74, 75, 81, 83, 88, 90, 93, 98] now learn two encoders jointly and use larger neural networks (up to 80B parameters as in [2]) and datasets. As discussed in Zhe et al. [25], recent successes in vision-language models can mainly attribute to developments in Transformers [85],

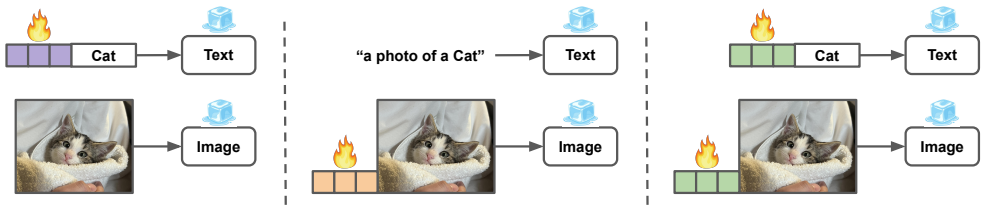


Figure 3. The architecture of (**a**) CoOp (textual prompt tuning), (**b**) VPT (visual prompt tuning), and (**c**) UPT (unified prompt tuning).

contrastive representation learning [10, 31, 35, 95], and web-scale training datasets [38, 67, 96]. CLIP [67] is a representative approach that trains two neural network-based encoders using a contrastive loss to match image-text pairs. After consuming 400 million data pairs, CLIP demonstrates remarkable zero-shot image recognition capability.

Prompt Tuning originated in the NLP community [48, 59] to improve the practical applicability of large-scale pre-trained language models [7, 18, 68, 72, 100]. The target NLP task is reformulated as a "fill-in-the-blank" cloze test,, which queries the language model to predict the masked token in "`I enjoyed the movie. It was [MASK].`" as either "positive" or "negative" for sentiment classification. The vital component lies in both designing the "verbalizer" (the label for the mask token) and the underlined part, known as prompt (template), in such a format familiar to the model. Efforts have focused on developing prompt-based learning approaches, such as handcrafted prompts [73], prompt mining and paraphrasing [40], gradient-based search [76], and automatic prompt generation [26]. Since discrete prompt was found to be sub-optimal and sensitive to the choice of the prompt [61, 103], more recent work has shifted toward learning continuous prompt learning methods [48, 55, 60, 87, 104], where the main idea is to turn a prompt into a set of continuous vectors that can be end-to-end optimized with respect to an objective function, which is also is most related to our research. In computer vision, prompt learning is a nascent research direction [5, 9, 17, 39, 42, 62, 70, 94, 99, 105, 106]. Our research focuses on incorporating cross-task knowledge in the prompt tuning process, which is distinct from previous vision-language prompt studies.

Multitask Prompt Tuning has recently been explored extensively in the NLP community. One line of the research [13, 64, 71, 89, 91] finetunes the pretrained model on massive, human-crafted, (thousands of) prompt-formatted downstream tasks and find the resulting model expresses strong generalization ability to unseen NLP tasks. Another line of the research explores multitask continuous prompt tuning [4]. For example, ATTEMPT [3] and SPoT [87], transferring source prompts to the target task, only incorporate multitask learning for prompt initialization, while our method explores both multitask initialization and adaptation. Additionally, we propose a different task grouping method and compare with related works in Section 4.3.

3. Methodology

We first revisit the CLIP [67], in company with text, visual, and unified prompt tuning approaches for visual recognition in Section. 3.1. We then present technical details of our proposed MVLPT learning in Section. 3.2.

3.1. Preliminaries

CLIP [67] is a model that trains both an image encoder and a text encoder to create similar embeddings for image-text pairs. It accomplishes this through minimizing a symmetric contrastive loss during pretraining, which predicts a positive sample in a batch of image-text combinations:

$$l_i^{u \to v} = \frac{\exp\left(\cos\left(\boldsymbol{u}_i, \boldsymbol{v}_i\right)/\tau\right)}{\sum_{j=1}^{N} \exp\left(\cos\left(\boldsymbol{u}_i, \boldsymbol{v}_j\right)/\tau\right)} \tag{1}$$

where $\boldsymbol{u} = \psi(\boldsymbol{x}) \in \mathbb{R}^d$ indicating the projection of image $\boldsymbol{x}$ to the final hidden space of dimension d; $\boldsymbol{v} = \phi(\boldsymbol{y}) \in \mathbb{R}^d$ indicating the projection of text $\boldsymbol{y}$; $\cos(\cdot, \cdot)$ denotes the cosine similarity; τ is a learnable temperature value. In zero-shot prediction, CLIP takes an image and a set of target classes, constructs a fixed prompt "`a photo of a [CLASS]`" for each class, and predicts the class with the highest cosine similarity between the encoded image and the set of prompts.

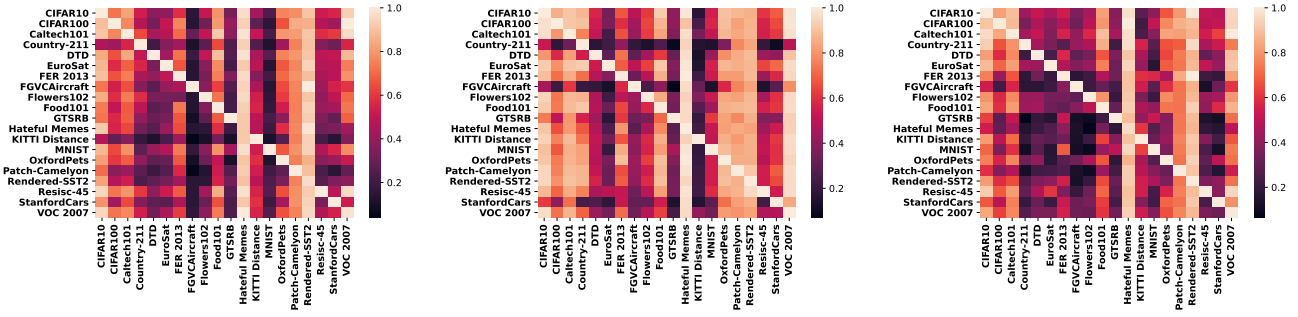
While the definition of a "task" is unclear, we borrow the definition from CLIP. For clarity, we formally distinct different tasks: K-way classification on dataset D is a different task than M-way classification on different dataset D', where K and M are different.

Text Prompt Tuning is a method used for adapting CLIP-like vision-language models to downstream tasks. It is a more efficient approach than finetuning the entire CLIP model. CoOp [105] proposed this method by replacing a prompt's context words with a learnable vector $\boldsymbol{P} \in \mathbb{R}^{d \times n}$ of adjustable length n. The text input is modified to:

$$\boldsymbol{P} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_n, \texttt{CLASS}]. \tag{2}$$

This modification allows for freezing the image and text encoders while optimizing only $\boldsymbol{P}$ with task-specific objective functions.

CoCoOp [106] is a newer method that adds a network to obtain an input-conditional token and achieves better performance than CoOp. However, its limitation in training



(a) CoOp Transferability. (b) VPT Transferability. (c) UPT Transferability.

Figure 4. A heatmap of our task transferability results. Each cell shows the relative performance on the target task of the transferred prompt from the associated source task (row) to the associated target task (column).

efficiency (training speed and GPU memory) makes it difficult to compare as a multitask baseline, so we only include this in Section 4.1.

Visual Prompt Tuning [39] is similar to text prompt tuning but for vision models. It adds a tunable vector $\mathbf{V} \in \mathbb{R}^{d \times n}$ to the input of each i-th transformer layer in the model. The modified input includes a classification token $c^i = \texttt{[CLS]}$, patchified image tokens $\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_m$, and $\mathbf{V}$:

$$\hat{\boldsymbol{Q}}^i = [c^i, \boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n, \boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_m]. \quad (3)$$

During finetuning, the image encoder is frozen, and only the visual prompt is optimized.

Unified Prompt Tuning (UPT) [97] is an approach for adapting VL models. [1] Specifically, instead of introducing two sets of isolated modality-specific prompts (*i.e.*, $\boldsymbol{P}$ in Eq. (2) and $\mathbf{V}$ in Eq. (3)) for the text and visual encoders, UPT considers learning a set of vision-language modality-agnostic prompts for tuning VL models. UPT defines a set of learnable prompts $\boldsymbol{U} = [\boldsymbol{U}_T, \boldsymbol{U}_V] \in \mathbb{R}^{d \times n}$ with length n, where $\boldsymbol{U}_T \in \mathbb{R}^{d \times n_T}$, $\boldsymbol{U}_V \in \mathbb{R}^{d \times n_V}$ is later employed as textual prompt and visual prompts, respectively. A lightweight Transformer layer θ is used to transform and interact with vision-language prompts $\boldsymbol{U}$ before appending the vision-language prompts into the text and visual encoders:

$$\begin{aligned} \boldsymbol{U}' &= \mathrm{SA}\left(\boldsymbol{U}\right) + \mathrm{LN}\left(\boldsymbol{U}\right), \\ \hat{\boldsymbol{U}} &= \mathrm{FFN}\left(\mathrm{LN}\left(\boldsymbol{U}'\right)\right) + \mathrm{LN}\left(\boldsymbol{U}'\right), \end{aligned} \quad (4)$$

where the self-attention operator SA, feed-forward network FFN and layer normalization LN are applied to obtain the transformed prompts $\hat{\boldsymbol{U}}$. The self-attention module in the lightweight Transformer layer allows beneficial interaction between two modalities so as to maximize the complementary effects. During downstream training, UPT froze both the text and visual encoder (ϕ and ψ) and only optimizes the vision-language prompts $\boldsymbol{U}$ and the lightweight Transformer layer θ. In this way, both the dynamic classifiers $\mathbf{W}$ and visual features $\boldsymbol{z}$ in Eq. (1) are effectively tuned for reliable prediction in the downstream task.

[1] Due to the recency, [97] does not release their model details or code. We therefore implement our own variant that simply concatenates the CoOp prompt vectors $\boldsymbol{U}_T$ and VPT-deep prompt vector $\boldsymbol{U}_V$ together as $\boldsymbol{U}$, we set the context length of $\boldsymbol{U}_T$ and $\boldsymbol{U}_V$ the same as 4 unless specify. We use a one-layer one-head Transformer block θ whose hidden dimension is cut to be 128. Before and after feeding $\boldsymbol{U}$ to θ, a linear layer is employed to match the dimensionality. We ablate this design choice in Appendix.

3.2. Multitask Vision-Language Prompt Tuning

Our proposed framework MVLPT mainly consists of two stages as shown in Figure 2, *multitask source prompt initialization* and *multitask target prompt adaptation*.

Multitask Prompt Initialization. In this stage, the shareable prompts for all *source tasks* are pretrained jointly through multitask prompt tuning. Note that we only use few-shot training set from *source tasks* to perform this pretrain versus using the entire set in NLP community [4, 87].

Multitask Prompt Adaptation. In this stage, we transfer the shareable source prompt to target tasks. For single-task target prompt adaptation, we then directly use the learned source prompt to initialize the target prompt and optimize with the regular task loss on each task (*i.e.*, cross-entropy loss). For multitask prompt adaptation, we first group relevant tasks together, then perform multitask prompt tuning within the selected groups from the same multitask-initialized source prompt. The grouping strategies are further discussed in Section 4.3. A theoretical justification of task grouping is provided in Appendix.

4. Experiments

Our approach is mainly evaluated in the following three problem settings: 1) cross-task generalization (Section 4.1) that measures the efficacy of multitask prompt initialization; 2) few-shot ELEVATER (Section 4.2) that shows the effectiveness of multitask prompt adaption; and, 3) zero-shot task transferability (Section 4.3) that is based on the 20 vision

Table 1. **Comparison of CoOp, CoCoOp, VPT, UPT, and our MCoCoOp, MCoOp, MVPT, and MUPT in the cross-task generalization setting**. The results strongly justify the **strong generalizability** of multitask prompt initialization. Specifically, each multitask variant learns shared prompt vectors from 11 *source tasks* before single task adaptation to 12 *target tasks*. The shots number (1, 5, 20) denotes both the number of shots we use for multitask prompt initialization and single task adaptation. For instance, 1 shot means we use 1 shot from each *source task* for multitask prompt initialization and adapt that for 1 shot learning to each *target task*. **Boldface** text denotes the best performance in that setting. Noted that we include the CIFAR-10 in the averaged task table and the CIFAR-10 performance is in Appendix.

(a) **Average over 12 tasks.**

# shots	1	5	20
CoOp	$50.51_{\pm 1.8}$	$55.50_{\pm 2.1}$	$65.87_{\pm 0.5}$
CoCoOp	$53.23_{\pm 1.6}$	$57.37_{\pm 1.7}$	$66.34_{\pm 0.6}$
VPT	$57.06_{\pm 1.3}$	$60.14_{\pm 1.0}$	$66.98_{\pm 0.7}$
UPT	$56.76_{\pm 0.7}$	$62.16_{\pm 0.8}$	$67.62_{\pm 0.6}$
MCoOp	$55.85_{\pm 1.1}$	$61.54_{\pm 1.6}$	$67.60_{\pm 0.5}$
MCoCoOp	$57.61_{\pm 0.6}$	$63.49_{\pm 0.5}$	$70.54_{\pm 0.4}$
MVPT	$60.98_{\pm 0.4}$	$\mathbf{65.91}_{\pm 0.4}$	$71.73_{\pm 0.3}$
MUPT	$\mathbf{61.66}_{\pm 0.2}$	$65.77_{\pm 0.4}$	$\mathbf{72.15}_{\pm 0.4}$

(b) CIFAR-100.

# shots	1	5	20
CoOp	64.65	70.48	72.90
CoCoOp	65.73	71.21	73.09
VPT	70.29	73.01	77.02
UPT	69.12	72.50	75.98
MCoOp	63.50	71.81	73.20
MCoCoOp	63.03	71.14	72.14
MVPT	70.67	72.71	77.22
MUPT	**71.17**	**73.66**	**77.45**

(c) Hateful Memes.

# shots	1	5	20
CoOp	48.40	52.60	52.40
CoCoOp	49.44	53.30	52.58
VPT	55.40	53.20	57.20
UPT	51.80	54.93	56.60
MCoOp	54.00	53.80	59.40
MCoCoOp	54.63	54.53	60.56
MVPT	**56.20**	**55.27**	**57.60**
MUPT	**56.20**	55.20	56.60

(d) MNIST.

# shots	1	5	20
CoOp	49.98	78.31	91.79
CoCoOp	51.61	79.41	92.07
VPT	71.61	74.00	88.62
UPT	60.44	81.64	89.88
MCoOp	65.06	78.30	94.14
MCoCoOp	66.17	79.52	95.08
MVPT	**82.36**	**89.57**	**95.31**
MUPT	81.29	88.48	94.54

(e) Resisc-45.

# shots	1	5	20
CoOp	68.65	78.23	84.25
CoCoOp	72.20	80.63	84.86
VPT	69.08	68.47	83.94
UPT	63.68	77.25	84.05
MCoOp	67.39	79.70	85.12
MCoCoOp	69.80	**82.04**	**88.84**
MVPT	**70.58**	77.79	84.63
MUPT	70.23	77.94	85.06

(f) Country-211.

# shots	1	5	20
CoOp	12.16	21.63	22.76
CoCoOp	**16.01**	24.23	23.42
VPT	13.76	18.26	20.71
UPT	13.62	21.62	21.11
MCoOp	11.75	22.04	23.56
MCoCoOp	14.37	**24.61**	**27.64**
MVPT	11.85	17.40	19.81
MUPT	11.37	21.33	23.53

(g) VOC 2007 Classification.

# shots	1	5	20
CoOp	55.78	63.70	77.43
CoCoOp	62.14	68.00	78.52
VPT	77.54	75.91	80.59
UPT	79.57	76.10	78.88
MCoOp	75.84	75.46	77.60
MCoCoOp	77.97	78.73	81.39
MVPT	78.39	79.19	**81.67**
MUPT	**80.18**	**80.51**	80.92

(h) Patch-Camelyon.

# shots	1	5	20
CoOp	59.71	51.93	59.65
CoCoOp	**62.30**	54.03	60.18
VPT	56.85	57.24	57.06
UPT	56.89	55.44	60.30
MCoOp	52.39	56.08	69.78
MCoCoOp	54.26	58.15	73.07
MVPT	59.06	**66.17**	**78.10**
MUPT	**62.30**	64.84	73.53

(i) Rendered-SST2.

# shots	1	5	20
CoOp	55.85	54.15	54.75
CoCoOp	58.01	56.35	55.31
VPT	57.28	54.13	57.55
UPT	52.72	55.83	57.66
MCoOp	56.40	56.67	57.77
MCoCoOp	58.53	58.88	60.28
MVPT	**59.09**	58.59	59.80
MUPT	56.07	**61.54**	**61.18**

(j) GTSRB.

# shots	1	5	20
CoOp	37.55	61.71	71.52
CoCoOp	41.55	64.41	72.21
VPT	52.58	72.42	86.17
UPT	**57.67**	70.72	85.34
MCoOp	37.89	59.31	72.09
MCoCoOp	40.54	62.08	76.49
MVPT	50.56	**75.03**	**89.75**
MUPT	51.79	69.22	85.30

(k) FER 2013.

# shots	1	5	20
CoOp	29.34	28.25	50.71
CoCoOp	34.08	31.45	51.52
VPT	49.76	47.48	56.39
UPT	49.76	47.85	56.77
MCoOp	52.49	47.76	50.24
MCoCoOp	55.50	50.90	55.23
MVPT	51.43	50.85	57.12
MUPT	**55.95**	**51.27**	**60.07**

(l) KITTI Distance.

# shots	1	5	20
CoOp	34.60	21.38	60.90
CoCoOp	32.08	19.68	60.47
VPT	23.77	40.79	47.68
UPT	37.41	42.57	53.54
MCoOp	45.01	46.69	57.38
MCoCoOp	43.36	45.04	54.76
MVPT	**52.60**	**58.46**	67.65
MUPT	50.77	53.73	**73.98**

tasks in ELEVATER.

Datasets For the domain generalization setting, we use the 11 image recognition tasks from [105] as *source tasks*. In Section 4.1, we use the non-overlapped 12 image recognition tasks in ELEVATER [50] as *target tasks*, covering a diverse set of recognition tasks. Specifically, the *source tasks* include ImageNet [15] and Caltech101 [21]

Table 2. **Comparison of prompt learning methods on the few-shot ELEVATER**. The number of shots is set to be 20 in each case, except for zero-shot CLIP. The results suggest the significant generalizability of multitask prompt initialization. † denotes the zero-shot CLIP results from ELEVATER [50] "Source" denotes the prompt initialization source, where "-" stands for random initialization, and "M" stands for using all 20 ELEVATER tasks for prompt initialization. "Adaptation" denotes the *target task* prompt adaptation method, where "S" stands for single *target task* prompt adaptation that each *target task* will be adapted independently, and 'M'" stands for multitask prompt adaptation that certain tasks (selected based on results in Section 4.3) will be learned together. Clearly, MVLPT demonstrates better **transferability** than single *target task* prompt adaptation counterparts. Δ denotes the best M-variant's gain over the respective baseline methods.

			Target																				
	Source	Adaptation	Caltech101	CIFAR10	CIFAR100	Country-211	DTD	EuroSat	FER 2013	FGVCAircraft	Flowers102	Food101	GTSRB	Hateful Memes	KITTI Distance	MNIST	OxfordPets	Patch-Camelyon	Rendered-SST2	Resisc-45	StanfordCars	VOC 2007	*Average*
CLIP†	-	-	88.9	90.8	68.2	22.8	44.8	54.7	48.5	24.3	88.7	43.5	58.1	27.0	52.0	69.4	89.0	54.0	60.9	65.6	64.8	83.7	60.0
CoOp	-	S	91.44	91.30	73.01	22.83	69.82	80.19	54.46	42.01	93.31	89.47	73.87	52.40	56.87	91.44	90.69	62.79	59.55	83.83	79.52	74.61	$71.67_{\pm 0.2}$
VPT	-	S	92.84	91.39	75.98	21.11	68.56	87.37	56.77	42.12	89.22	89.04	85.34	56.60	53.54	89.88	90.71	60.30	57.66	84.05	74.95	78.88	$72.32_{\pm 0.6}$
UPT	-	S	92.58	92.05	76.61	23.37	67.68	88.98	56.87	42.46	89.59	89.64	82.72	56.87	47.87	89.11	91.24	60.41	59.03	83.32	76.40	81.20	$72.40_{\pm 0.3}$
MCoOp	-	M	91.53	91.67	73.01	23.12	69.82	81.69	54.46	42.01	93.44	89.47	74.38	58.40	56.87	91.44	90.69	64.91	61.63	84.03	**79.52**	78.45	$72.53_{\pm 0.6}$
MVPT	-	M	92.84	93.54	76.39	21.42	68.56	89.15	56.77	42.12	89.22	89.04	85.34	58.20	53.54	89.88	91.01	66.53	58.14	84.05	74.95	80.69	$73.07_{\pm 0.9}$
MUPT	-	M	92.58	93.38	76.61	23.37	67.68	88.98	56.94	42.46	89.59	**89.64**	82.72	58.13	55.41	89.91	**91.24**	63.36	61.34	83.32	76.40	81.20	$73.21_{\pm 0.7}$
MCoOp	M	M	92.09	91.59	72.63	**23.52**	**70.41**	81.70	54.85	42.34	**93.61**	89.14	72.74	58.40	47.73	90.21	89.61	68.92	**64.89**	**84.39**	79.43	79.55	$72.39_{\pm 0.5}$
MVPT	M	M	**93.46**	93.72	**77.38**	20.79	69.43	**92.23**	**57.07**	**42.57**	88.80	87.78	**89.62**	55.53	**62.07**	**93.08**	91.04	69.69	57.50	84.35	74.20	**82.21**	**74.13**$_{\pm 0.3}$
MUPT	M	M	92.19	**93.75**	75.39	23.45	65.99	90.17	56.06	41.19	89.34	89.38	81.66	**59.00**	57.20	91.38	90.30	**69.74**	62.29	83.40	76.66	79.29	$73.39_{\pm 0.6}$
Δ			**+0.62**	**+1.70**	**+1.40**	**+0.69**	**+0.59**	**+4.86**	**+0.30**	**+0.45**	**+0.30**	**+0.00**	**+4.28**	**+2.13**	**+8.53**	**+3.20**	**+0.00**	**+9.33**	**+5.34**	**+0.56**	**+0.00**	**+3.33**	**+1.81**

for generic objects classification; OxfordPets [66], StanfordCars [45], Flowers102 [65], Food101 [6] and FGVCAircraft [63] for fine-grained classification; SUN397 [92] for scene recognition; UCF101 [78] for action recognition; DTD [14] for texture classification; and, EuroSAT [34] for satellite imagery recognition. In Section 4.2 and 4.3, we use ELEVATER benchmark, which originally cover 20 image classification tasks which includes the 8 overlapped tasks as Caltech101, OxfordPets, StanfordCars, Flowers102, Food101, FGVCAircraft, DTD and EuroSAT, and the rest 12 non-overlapped tasks as Hateful Memes [43], PatchCamelyon [86], Rendered-SST2 [67], KITTI Distance [22], FER 2013 [1], CIFAR-10/100 [46], VOC 2007 Classification [20], Country-211 [67], MNIST [16], GTSRB [79], and Resisc-45 [12].

Baselines We compare our approach against the following methods: (i) **Zero-shot CLIP [67]**[2] . This baseline uses does not involve any prompt-learning strategies as mentioned in Section 3.1. (ii) **Single Task Prompt Tuning** methods, including CoOp [105], VPT [39], UPT [97] for vision, language, and vision-language prompt tuning method.

Training Details Our implementation is based on CoOp.[3] Throughout the experiments, we use CLIP as our vision-language model (*i.e.*, ViT-B/16 for all the experiments except for the scaling ablation). Following CoOp [105] and VPT [39], we use a context length of 16 for both CoOp and VPT throughout the study. We empirically find a shorter context length of 4 leads to better performance for UPT, and we use 4 contexts for UPT only. (This design choice is discussed in more detail in the Appendix). The resulting prompt vectors of CoOp/MCoOp, VPT/MVPT, UPT/MUPT account for 0.01%, 0.11%, 0.45% total parameters of the ViT-B/16 (124M parameters) model. All the prompt vectors for CoOp, VPT or UPT are randomly initialized without using the pretrained word embeddings of "a photo of a" for initialization in [105] for a fair comparison. All the methods are trained with a batch size of 32 for 200 epochs following [105]. All the image input size is set to 224×224. We use Adam optimizer and cosine learning rate schedule. All the learning rate is set as 2e-3, and the warmup period is set as 1 epoch following [105]. All the few-shot experiments are averaged with 3 runs. For each experiment, we select the best prompt checkpoint using the validation set that consists 20% splits from the few-shot sampled training set.

4.1. Cross-task Generalization

We examine the efficacy of the proposed multitask prompt initialization in MVLPT via cross-task generalization. Specifically, we use all the 11 tasks in [105] as *source tasks* and the non-overlapped 12 tasks in ELEVATER as *target tasks*. We perform multitask learning on all the *source tasks* to learn the shared prompt vectors. The resulting shared prompt vectors will be used as the prompt initialization for single-task adaptation on each *target task*. We evaluate across 1, 5, 20 shots as suggested in the ELEVATER [50]

[2] https://github.com/openai/CLIP.
[3] https://github.com/KaiyangZhou/CoOp.

benchmark. The shot number is adopted for both multitask prompt initialization and single *target task* adaptation, respectively. It means that for 1 shot, we will sample 1 instance for each image class of all the *source tasks* for multitask prompt initialization and then adapt the learned prompt initialization to 1-shot learning for each *target task*. The baseline prompt learning method CoOp, CoCoOp, VPT and UPT are using random initialized prompt as in [39, 105] for single *target task* adaptation. The results are summarized in Table 1, showing that multitask prompt initialization variants MCoOp, MVPT and MUPT mostly outperform the baseline prompt learning counterparts by a significant margin. (averaged over 3 runs). The improvement is also consistent across different numbers of shots. It is also interesting that the most effective task of multitask prompt initialization differs for each prompt learning method. Specifically, MCoOp benefits the task where the class names are distinct the most like Resisc-45, while MVPT/MUPT favors the task where the images are more separable like VOC 2007 Classification. We further analyze this different preference in Section 4.3. Nevertheless, we note that multitask prompt initialization does not always guarantee performance improvement when the number of *source task* shots is extremely small as 1 and the *target task* needs extreme fine-grained or specialized classification like 211-way classification in Country-211.

4.2. Few-shot ELEVATER

We measure the effectiveness of the proposed multitask prompt adaptation in MVLPT on all 20 few-shot ELEVATER tasks. We set the number of shots as 20 in each setting. Specifically, versus adapting the learned prompt initialization to each *target task* independently (single-task prompt adaptation), we group several *target tasks* as in Figure 2 and perform multitask learning in each group to learn shared prompt vectors during prompt adaptation. We determine which tasks should be grouped for each prompt learning method based on the transferability map shown in Figure 4, which is discussed in more details in Section 4.3. The detailed results are shown in Table 2. It clearly demonstrates that multitask prompt adaptation variants exhibit better transferability than single *target task* prompt adaptation counterparts. Comparing single-task prompt adaption and multitask prompt adaption, multitask adaption boosts the averaged performance on CoOp, VPT, UPT by 0.86%, 0.75% and 0.81%, respectively. Using 20 ELEVATER tasks as *source tasks* can further improve the results for MVPT and MUPT. For MCoOp, multitask prompt initialization may make the class name distribution less separable for the task has distinct categories like KITTI Distance, which effaces the improvement on other tasks. The resulting MVPT achieves **74.13**% the new state-of-the-art on 20 shot ELEVATER benchmark for ViT-B/16 model comparing to 64.41% in [57]. We also observe that there exist tasks that are not improved using multitask prompt adaptation. We attribute that to some tasks like FGVCAircraft with distant and specialized categories may not be able to leverage useful cross-task knowledge from other ELEVATER tasks during prompt adaptation.

4.3. Task Transferability

To understand the cross-knowledge [8, 69, 80, 84, 102] in vision-language prompt tuning, we conduct a large-scale study on task transferability with 20 ELEVATER tasks in 400 combinations for each prompt tuning method, following [82]. We use checkpoints from each task in ELEVATER after 20-shot learning on 3 different seeds as the *source*. Then, we perform zero-shot adaptation to the rest of the tasks. We normalized the scores by dividing the transfer performance with the best one on that task and presented the results in Figure 4. To select groups for multitask adaptation, we select the top 1 and 2 transferability with respect to each *target task*. We jointly train such group of 2 and 3 tasks and select the best checkpoint based on the the validation performance for each task, respectively.

We also report the performance with different grouping strategies for multitask prompt adaptation on 20-shot ELEVATER in Table 4, where Best M stands for using the aforementioned grouping strategy and Worst M stands for grouping the most dissimilar tasks from the transferability map.[4] It directly suggests that the transferability map could serve as a principal way to group the relevant tasks and failing to do that leads to significant performance degradation.

We additionally try two other task grouping methods, exploring task similarity encoded in learned prompts and unsupervised grouping. We first mimic ATTEMPT [3] and SPoT [87] to calculate cosine similarity between learned prompts. Adapting to our method, we choose tasks with the highest similarity based on the attention map and apply to multitask adaptation. For the second grouping method, we extract the feature of all training set images using CLIP. The features are clustered into 20 groups using K-Means, which gives the task grouping proposals. The results are shown in Table 3. As we adopt the common grouping method in NLP (Prompt Sim), we find that the result is slightly lower, while the cost is similar to ours. The unsupervised grouping (K-Means) does not require training prompts before grouping. It is efficient, but the performances are mostly lower than the single task baselines (Single). An unsupervised method can induce error during task grouping, which hurts the performance in return.

5. Discussion

Source Tasks There is rich literature [10, 30, 31, 33] to use ImageNet1K to pretrain vision backbones for various downstream vision tasks (object detection [27], semantic seg-

[4]We provide detailed task group information in Appendix.

Table 3. Averaged results of multitask adaptation on ELEVATER with different task grouping methods.

Method	CoOp	VPT	UPT
Single	$71.67_{\pm 0.2}$	$72.32_{\pm 0.6}$	$72.40_{\pm 0.3}$
Ours	$72.39_{\pm 0.5}$	$74.13_{\pm 0.3}$	$73.39_{\pm 0.6}$
Prompt Sim	$72.15_{\pm 0.3}$	$73.94_{\pm 0.2}$	$72.99_{\pm 0.4}$
K-Means	$70.98_{\pm 0.2}$	$73.16_{\pm 0.3}$	$72.25_{\pm 0.3}$

Table 4. Ablation of prompt adaptation strategies for MVLPT.

Model	Source	Adaptation	Averaged ELEVATER
MCoOp	M	S	$70.93_{\pm 0.3}$
MVPT	M	S	$73.16_{\pm 0.3}$
MUPT	M	S	$72.25_{\pm 0.5}$
MCoOp	M	Best M	**72.39**$_{\pm 0.5}$
MVPT	M	Best M	**74.13**$_{\pm 0.3}$
MUPT	M	Best M	**73.39**$_{\pm 0.6}$
MCoOp	M	Worst M	$70.13_{\pm 0.7}$
MVPT	M	Worst M	$71.81_{\pm 0.2}$
MUPT	M	Worst M	$69.94_{\pm 1.0}$

Table 5. Ablation of source tasks for MCoOp, MVPT and MVLPT.

Model	Source	Adaptation	Averaged 12 *target tasks*
CoOp	ImageNet1K	S	$66.36_{\pm 0.5}$
VPT	ImageNet1K	S	$68.80_{\pm 0.9}$
UPT	ImageNet1K	S	$67.45_{\pm 0.7}$
MCoOp	*10 source tasks*	S	$66.51_{\pm 0.5}$
MVPT	*10 source tasks*	S	$70.31_{\pm 1.1}$
MUPT	*10 source tasks*	S	$70.08_{\pm 0.9}$
MCoOp	*11 source tasks*	S	**67.60**$_{\pm 0.5}$
MVPT	*11 source tasks*	S	**71.73**$_{\pm 0.6}$
MUPT	*11 source tasks*	S	**72.15**$_{\pm 0.7}$

mentation [32]). In Table 5, we study the impact of *source tasks* using ImageNet1K, 10 *source tasks* in [105] excluding ImageNet1K and 11 *source tasks* in [105] including ImageNet1K using the 20-shot cross-task generalization setting. It shows that ImageNet1K servers as a strong *source task* for prompt initialization while performing multitask prompt initialization from the diverse 10 *source tasks* leads to noticeable improvement especially for MVPT/MUPT. Besides, combing ImageNet1K with the 10 *source tasks* gives the best performance, which may suggest the potential to scale our MVLPT to more diverse set as *source tasks* like even more than thousands of tasks [13, 89] in NLP community.

Scaling We conduct scaling experiments in Figure 1 to analyze how MVLPT performs with increasing pretrained model sizes. It is based on 20-shot cross-task generalization setting except for 0-shot CLIP, These results show that

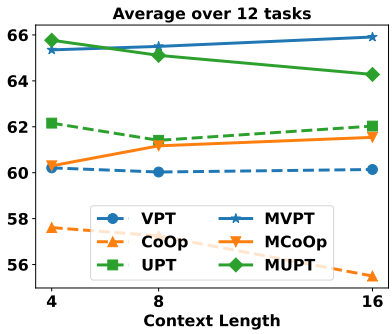


Figure 5. Ablation on context length.

our MVPT, MCoOp, MUPT is not only able to achieve the same parameter efficiency but also effective across model scales ranging from ViT-B/32 to ViT-L/14. ViT-B/32 (2.59 GFLOPs, 125M parameters) to ViT-B/16 (11.27 GFLOPs) and ViT-L/14 (51.90 GFLOPs, 390M parameters).

Context Length The ablation study on context length is also carried out in the cross-task generalization setting. Following [105], we study 4, 8 and 16 context tokens and use random initialization for all. In Figure 5, we see consistent improvement for longer context length of MVPT, MCoOp and marginal performance difference for VPT, CoOp, UPT. For MUPT, we observe longer context length turns out to hurt the performance, which we assume could be potentially attribute to the context length discussion in CoOp [105].

Limitations As discussed in the Section 3, the improvements of *multitask prompt initialization* accompanies the cost of extra compute for multitask prompt tuning on *source tasks*. Even though the procedure is conducted once like pretraining then the learned prompt can be reused as initialization for various target tasks. It sums up to $\frac{N_{\text{source}}}{N_{\text{target}}}$ more compute (N_{source}, N_{target} stands for number of *source tasks*, *target tasks*, respectively). The extra compute caused by *multitask prompt adaptation* is marginal except for evaluating the zero-shot task transferability for task grouping guidance.

6. Conclusion

In this paper, we propose multitask vision-language prompt tuning (MVLPT). We demonstrate that MVLPT exhibits strong generalizability and few-shot learning performance compared to baseline prompt learning methods. The most performant MVLPT sets the new state-of-the-art performance on the ELEVATER benchmark. We also study task transferability across 20 vision tasks and provide a guideline for multitask prompt learning.

References

[1] FER 2013: Kaggle challenges in representation learning facial expression recognition. https://www.kaggle.

com/. 6

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022. 2

[3] Akari Asai, Mohammadreza Salehi, Matthew E Peters, and Hannaneh Hajishirzi. Attempt: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6655–6672, 2022. 3, 7

[4] Akari Asai, Mohammadreza Salehi, Matthew E Peters, and Hannaneh Hajishirzi. Attentional mixtures of soft prompt tuning for parameter-efficient multi-task knowledge sharing. 2022. 2, 3, 4

[5] Amir Bar, Yossi Gandelsman, Trevor Darrell, Amir Globerson, and Alexei A Efros. Visual prompting via image inpainting. In *NeurIPS*, 2022. 3

[6] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *ECCV*, 2014. 6

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 3

[8] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997. 2, 7

[9] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *arXiv preprint arXiv:2205.13535*, 2022. 3

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 3, 7

[11] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022. 2

[12] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 2017. 6

[13] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022. 2, 3, 8

[14] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 6

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[16] Li Deng. The MNIST database of handwritten digit images for machine learning research. *IEEE signal processing magazine*, 2012. 6

[17] Mohammad Mahdi Derakhshani, Enrique Sanchez, Adrian Bulat, Victor Guilherme Turrisi da Costa, Cees GM Snoek, Georgios Tzimiropoulos, and Brais Martinez. Variational prompt tuning improves generalization of vision-language models. *arXiv preprint arXiv:2210.02390*, 2022. 3

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 3

[19] Mohamed Elhoseiny, Babak Saleh, and Ahmed Elgammal. Write a classifier: Zero-shot learning using purely textual descriptions. In *ICCV*, 2013. 2

[20] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 2010. 6

[21] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR-W*, 2004. 5

[22] Jannik Fritsch, Tobias Kuehnl, and Andreas Geiger. A new performance measure and evaluation benchmark for road detection algorithms. In *ITSC*. IEEE, 2013. 6

[23] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. *NeurIPS*, 2013. 2

[24] Andreas Fürst, Elisabeth Rumetshofer, Viet Tran, Hubert Ramsauer, Fei Tang, Johannes Lehner, David Kreil, Michael Kopp, Günter Klambauer, Angela Bitto-Nemling, et al. Cloob: Modern hopfield networks with infoloob outperform clip. *arXiv preprint arXiv:2110.11316*, 2021. 2

[25] Zhe Gan, Linjie Li, Chunyuan Li, Lijuan Wang, Zicheng Liu, and Jianfeng Gao. Vision-language pre-training: Basics, recent advances, and future trends. *arXiv preprint arXiv:2210.09263*, 2022. 2

[26] Tianyu Gao, Adam Fisch, and Danqi Chen. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*, 2020. 3

[27] Ross Girshick. Fast r-cnn. In *ICCV*, 2015. 7

[28] Lluis Gomez, Yash Patel, Marçal Rusiñol, Dimosthenis Karatzas, and CV Jawahar. Self-supervised learning of visual features through embedding images into text topic spaces. In *CVPR*, 2017. 2

[29] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 1

[30] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 16000–16009, 2022. 7

[31] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 3, 7

[32] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 8

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 7

[34] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 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*, 2019. 6

[35] Olivier J. Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Aäron van den Oord. Data-efficient image recognition with contrastive predictive coding. In *ICML*, 2020. 3

[36] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, pages 2790–2799. PMLR, 2019. 1

[37] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *NeurIPS*, 2021. 1

[38] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 1, 3

[39] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022. 1, 2, 3, 4, 6, 7

[40] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *ACL*, 2020. 3

[41] Armand Joulin, Laurens Van Der Maaten, Allan Jabri, and Nicolas Vasilache. Learning visual features from large weakly supervised data. In *ECCV*, 2016. 2

[42] Chen Ju, Tengda Han, Kunhao Zheng, Ya Zhang, and Weidi Xie. Prompting visual-language models for efficient video understanding. *arXiv preprint arXiv:2112.04478*, 2021. 3

[43] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *NeurIPS*, 2020. 6

[44] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, pages 5583–5594. PMLR, 2021. 2

[45] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV-W*, 2013. 6

[46] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[47] Jimmy Lei Ba, Kevin Swersky, Sanja Fidler, et al. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *ICCV*, 2015. 2

[48] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021. 1, 3

[49] Ang Li, Allan Jabri, Armand Joulin, and Laurens van der Maaten. Learning visual n-grams from web data. In *ICCV*, 2017. 2

[50] Chunyuan Li, Haotian Liu, Liunian Harold Li, Pengchuan Zhang, Jyoti Aneja, Jianwei Yang, Ping Jin, Yong Jae Lee, Houdong Hu, Zicheng Liu, et al. Elevater: A benchmark and toolkit for evaluating language-augmented visual models. In *NeurIPS*, 2022. 1, 2, 5, 6

[51] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2

[52] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, volume 34, pages 9694–9705, 2021. 2

[53] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 2

[54] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*, 2020. 2

[55] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL*, 2021. 1, 3

[56] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *arXiv preprint arXiv:2110.05208*, 2021. 2

[57] Feng Liang, Yangguang Li, and Diana Marculescu. Supmae: Supervised masked autoencoders are efficient vision learners. *arXiv preprint arXiv:2205.14540*, 2022. 7

[58] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *NeurIPS*, 2022. 2

[59] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*, 2021. 3

[60] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *ACL*, pages 61–68, 2022. 3

[61] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. Gpt understands, too. In *ICML*, 2021. 1, 3

[62] Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5206–5215, 2022. 3

[63] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 6

[64] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*, 2022. 3

[65] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008. 6

[66] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *CVPR*, 2012. 6

[67] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1, 3, 6

[68] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019. 3

[69] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21(140):1–67, 2020. 7

[70] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *CVPR*, 2022. 3

[71] Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. In *ICLR*, 2021. 2, 3

[72] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022. 3

[73] Timo Schick and Hinrich Schütze. It's not just size that matters: Small language models are also few-shot learners. In *NAACL*, 2021. 3

[74] Sheng Shen, Chunyuan Li, Xiaowei Hu, Yujia Xie, Jianwei Yang, Pengchuan Zhang, Anna Rohrbach, Zhe Gan, Lijuan Wang, Lu Yuan, et al. K-lite: Learning transferable visual models with external knowledge. In *NeurIPS*, 2022. 2

[75] Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can clip benefit vision-and-language tasks? In *ICLR*, 2022. 2

[76] Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *EMNLP*, 2020. 3

[77] Richard Socher, Milind Ganjoo, Hamsa Sridhar, Osbert Bastani, Christopher D Manning, and Andrew Y Ng. Zero-shot learning through cross-modal transfer. In *NeurIPS*, 2013. 2

[78] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 6

[79] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *IJCNN*, 2011. 6

[80] Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *ICML*, pages 9120–9132. PMLR, 2020. 2, 7

[81] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019. 2

[82] Yusheng Su, Xiaozhi Wang, Yujia Qin, Chi-Min Chan, Yankai Lin, Zhiyuan Liu, Peng Li, Juanzi Li, Lei Hou, Maosong Sun, et al. On transferability of prompt tuning for natural language understanding. *arXiv preprint arXiv:2111.06719*, 2021. 7

[83] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *EMNLP*, 2019. 2

[84] Sebastian Thrun. Is learning the n-th thing any easier than learning the first? *NeurIPS*, 8, 1995. 2, 7

[85] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[86] Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *MICCAI*, 2018. 6

[87] Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou, and Daniel Cer. Spot: Better frozen model adaptation through soft prompt transfer. In *ACL*, pages 5039–5059, 2022. 3, 4, 7

[88] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022. 2

[89] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *EMNLP*, 2022. 3, 8

[90] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. In *ICLR*, 2022. 2

[91] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *ICLR*, 2021. 3

[92] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010. 6

[93] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. Filip: Fine-grained interactive language-image pre-training. In *ICLR*, 2022. 2

[94] Yuan Yao, Ao Zhang, Zhengyan Zhang, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Cpt: Colorful prompt tuning for pre-trained vision-language models. *arXiv preprint arXiv:2109.11797*, 2021. 3

[95] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022. 3

[96] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021. 1, 3

[97] Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. Unified vision and language prompt learning. *arXiv preprint arXiv:2210.07225*, 2022. 4, 6

[98] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *CVPR*, 2021. 2

[99] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. *arXiv preprint arXiv:2112.02413*, 2021. 3

[100] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 3

[101] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D Manning, and Curtis P Langlotz. Contrastive learning of medical visual representations from paired images and text. *arXiv preprint arXiv:2010.00747*, 2020. 2

[102] Yu Zhang and Qiang Yang. A survey on multi-task learning. *TKDE*, 2021. 2, 7

[103] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *ICML*, pages 12697–12706. PMLR, 2021. 3

[104] Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [mask]: Learning vs. learning to recall. In *NAACL*, 2021. 3

[105] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *arXiv preprint arXiv:2109.01134*, 2021. 1, 2, 3, 5, 6, 7, 8

[106] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, pages 16816–16825, 2022. 1, 3