Transferable Model-agnostic Vision-Language Model Adaptation for Efficient Weak-to-Strong Generalization

Jihwan Park[1], **Taehoon Song**[2], **Sanghyeok Lee**[2], **Miso Choi**[1], **Hyunwoo J. Kim**[2*]
[1]Korea University
[2]KAIST
{jseven7071, miso8070}@korea.ac.kr
{taehoons, sanghyeoklee, hyunwoojkim}@kaist.ac.kr

Abstract

Vision-Language Models (VLMs) have been widely used in various visual recognition tasks due to their remarkable generalization capabilities. As these models grow in size and complexity, fine-tuning becomes costly, emphasizing the need to reuse **adaptation knowledge** from 'weaker' models to efficiently enhance 'stronger' ones. However, existing adaptation transfer methods exhibit limited transferability across models due to their model-specific design and high computational demands. To tackle this, we propose **Trans**ferable **M**odel-agnost**i**c adap**ter** (**TransMiter**), a light-weight adapter that improves vision-language models 'without backpropagation'. TransMiter captures the knowledge gap between pre-trained and fine-tuned VLMs, in an 'unsupervised' manner. Once trained, this knowledge can be seamlessly transferred across different models without the need for backpropagation. Moreover, TransMiter consists of only a few layers, inducing a negligible additional inference cost. Notably, supplementing the process with a few labeled data further yields additional performance gain, often surpassing a fine-tuned stronger model, with a marginal training cost. Experimental results and analyses demonstrate that TransMiter effectively and efficiently transfers adaptation knowledge while preserving generalization abilities across VLMs of different sizes and architectures in visual recognition tasks.

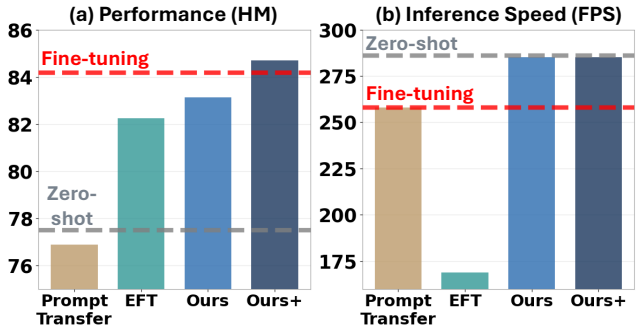

Figure 1: **Comparison of adaptation knowledge transfer methods.** Performance is averaged over 11 visual recognition tasks in a base-to-novel setting. **TransMiter** (ours) outperforms other adaptation transfer approaches, including Prompt Transfer (Su et al. 2022) and EFT (Mitchell et al. 2024), while maintaining inference speed nearly identical to a zero-shot model (gray), which serves as the upper-bound. With a small amount of labeled data, **TransMiter+** (ours+) surpasses its supervised counterpart (Khattak et al. 2023b), *e.g.,* Fine-tuning (red).

1 Introduction

The rapid evolution of vision-language models (VLMs) (Jia et al. 2021; Radford et al. 2021) has driven significant advancements in computer vision. These models are typically built with two modality-specific encoders, one for image and one for text, and are trained on large datasets of image-text pairs using contrastive loss. Here, modality-specific encoders learn to map each modality input into a joint embedding space, enabling zero-shot capabilities across a wide range of visual recognition tasks. The broad and generalized knowledge embedded in VLMs enables efficient adaptation on downstream tasks with few labeled data, as explored in (Yang et al. 2025b; Wang et al. 2024; Zhou et al. 2022b,a; Khattak et al. 2023b,a; Li et al. 2024). However, as VLMs are scaled (Chen et al. 2024; Cherti et al. 2023) or equipped with more complex architectures (Yu et al. 2022), fine-tuning them requires massive memory and computational costs.

In this context, an important research question arises; "How can the knowledge gained from adapting smaller, weaker models, *i.e., adaptation knowledge*, be effectively leveraged to improve larger and more advanced models?" One common approach (Su et al. 2022; Han et al. 2024; Liu et al. 2024; Mitchell et al. 2024) involves directly reusing a subset of parameters from a weaker model to a stronger one. However, due to semantic discrepancy or dimensional mismatch between different models, additional training is often necessary (Su et al. 2022; Han et al. 2024) to align their internal representations properly. Some works (Lu et al. 2023; Liu et al. 2024; Mitchell et al. 2024; Lu et al. 2024) mitigate this issue by enforcing semantic consistency (*e.g.,* model prediction), reducing the need for re-training but at the cost of slower inference due to multiple model usage. Alternatively, knowledge distillation (Yuan et al. 2020; Nasser, Gupte, and Sethi 2024; Burns et al. 2024; Li et al. 2024) has

*Corresponding author.

been demonstrated as an effective way for smaller models to guide larger ones. Still, this approach requires expensive retraining whenever a new model is introduced, limiting its practicality for rapidly evolving models.

In this work, we aim to design an adaptation module to efficiently transfer **adaptation knowledge** from the weaker to stronger VLMs. Our design is guided by three key objectives: (1) **model-agnostic compatibility**, (2) **computationally efficient transfer**, and (3) **minimal additional inference cost**. To this end, we propose **Trans**ferable **M**odel-agnost**i**c adap**ter** (**TransMiter**), a light-weight adapter that efficiently transfers the adaptation knowledge across different models 'without backpropagation'. Instead of directly transferring knowledge, we adopt a novel learning paradigm **"knowledge extraction and transfer"**. TransMiter first extracts adaptation knowledge by learning to capture the knowledge gap through the differences in predicted logits between pre-trained and fine-tuned weak models in an unsupervised manner. Once extracted, TransMiter enables seamless transfer to any stronger models by leveraging the consistent dimensionality and semantics of logits.

To effectively harness the adaptation knowledge, we utilize auxiliary classes alongside task classes to increase the dimensionality of logit vectors. It enhances the expressive power of the logits, thereby boosting the acquisition of adaptation knowledge and transferability across models. Additionally, we propose a basis change to further align models within the latent space. The basis for the transferred model is derived through a closed-form solution, thus eliminating the need for computation-intensive backpropagation. By incorporating the above two methodologies, TransMiter achieves exceptional transferability. After the adaptation knowledge transfer process, TransMiter itself becomes a strong initialization point, enabling a small amount of labeled data to deliver substantial performance boosts, often surpassing fine-tuned stronger counterparts, while incurring only minimal additional training cost. Notably, its light-weight design, consisting of simple projection matrices and a single MLP layer, ensures minimal additional inference cost, making it both efficient and scalable.

Extensive experiments demonstrate that TransMiter effectively transfers adaptation knowledge across models of varying sizes and architectures, confirming its superior transferability in diverse visual recognition tasks. The contributions of TransMiter can be summarized as:

- We propose **Trans**ferable **M**odel-agnost**i**c adap**ter** (**TransMiter**), a light-weight adapter for enhancing vision-language models without backpropagation.
- We incorporate two key techniques, auxiliary class expansion and basis change, that significantly boost the transferability of TransMiter, while the addition of a few labeled data further unlocks its full potential.
- We demonstrate TransMiter's effectiveness and efficiency in transferring adaptation knowledge across a wide range of datasets and models.

2 Related Works

Vision-Language Model Adaptation. Vision-language models (VLMs), such as CLIP (Radford et al. 2021), have demonstrated strong generalization abilities across various visual recognition tasks. Leveraging this strength, numerous studies have investigated fine-tuning VLMs on visual recognition tasks, such as prompt learning (Yang et al. 2025b; Wang et al. 2024; Zhou et al. 2022b,a; Zhu et al. 2023; Lee et al. 2023; Cho, Kim, and Kim 2023; Khattak et al. 2023b,a; Li et al. 2024; Yang et al. 2025a), low-rank adaptation (Zanella and Ayed 2024), linear probing (Ouali et al. 2023), and regularization (Park, Ko, and Kim 2024; Zhu et al. 2023; Khattak et al. 2023b). However, existing VLM adaptation methods often neglect the models' rapid evolution in terms of size and complexity, resulting in the inconvenience of retraining (Su et al. 2022) whenever a new model comes. To address this limitation, we focus on developing a transferable adaptation module for VLMs, enabling the efficient transfer of adaptation knowledge acquired during fine-tuning across different VLMs, regardless of their size or architectural design.

Weak-to-Strong Generalization. In an era of rapidly evolving VLMs, "weak-to-strong generalization", where stronger models leverage the knowledge of weaker models, is emerging as a new research challenge for enabling efficient model adaptation. Previous methods try to transfer either a portion of a model's internal parameters (Su et al. 2022; Chijiwa 2024; Han et al. 2024) or the entire model itself (Lu et al. 2023; Liu et al. 2024; Mitchell et al. 2024) to other models. However, differences in architecture often hinder parameter reuse, and even when feasible, aligning representation spaces requires costly retraining. While full-model transfer alleviates the problem using predicted logits, it requires running multiple models, leading to significant inference cost. Some other studies (Yuan et al. 2020; Nasser, Gupte, and Sethi 2024; Burns et al. 2024; Li et al. 2024) have shown that weaker models can effectively supervise stronger ones, a concept known as reverse knowledge distillation (Nasser, Gupte, and Sethi 2024). While these methods incur no additional inference cost, they still require expensive retraining whenever a new model is introduced, making it impractical for frequent model updates. In our work, we design an adaptation module for high cross-model transferability and fast inference speed. Additionally, we employ "knowledge extraction and transfer" approach, where adaptation knowledge is extracted only "once" from a weaker model and efficiently transferred to a stronger model.

3 Preliminaries

Vision-Language Model. A pre-trained Vision-Language Model (VLM), denoted as $\theta_{\text{pt}} = \{\mathcal{V}_{\text{pt}}, \mathcal{T}_{\text{pt}}\}$, is trained on large-scale image-text datasets using a contrastive loss, where $\mathcal{V}_{\text{pt}}$ and $\mathcal{T}_{\text{pt}}$ denote the pre-trained image encoder and text encoder, respectively. Given an input image x and task classes C_{task} are processed by the image and text encoders, producing an image feature and a set of text features, one for each class $c \in C_{\text{task}}$. These text features serve as "anchors", acting as reference points for classification. To de-

termine the predicted class $\hat{y}$, the image feature is compared to each class's text feature using a similarity measure. The class with the highest similarity to the image feature is selected as the predicted class. This process is expressed as:

$$\begin{aligned} z_{\text{pt}}(c;x) &= \textbf{sim}\left(\mathcal{V}_{\text{pt}}(x), \mathcal{T}_{\text{pt}}(c)\right), \forall c \in C_{\text{task}} \\ \hat{y} &= \arg\max_{c \in C_{\text{task}}} z_{\text{pt}}(c;x), \end{aligned} \tag{1}$$

where **sim** denotes the cosine similarity. We will omit the $(c;x)$ terms in logits for brevity.

To incorporate task knowledge into VLMs, the model can be fine-tuned using contrastive loss between image and text features, yielding a fine-tuned model $\theta_{\text{ft}} = \{\mathcal{V}_{\text{ft}}, \mathcal{T}_{\text{ft}}\}$.

Problem Setting. Suppose that we have a pre-trained source model $\theta_{\text{pt-s}}$ (*e.g.,* weak model) and its fine-tuned counterpart $\theta_{\text{ft-s}}$. Our goal is to capture the knowledge gap, *i.e.,* adaptation knowledge, between the pre-trained and fine-tuned source models, then transfer this knowledge to the pre-trained target model $\theta_{\text{pt-t}}$ (*e.g.,* strong model) without using any labeled data. Specifically, the extraction of adaptation knowledge δ_{s} can be expressed as:

$$\delta_{\text{s}} = \mathcal{E}\left(\theta_{\text{ft-s}}, \theta_{\text{pt-s}}\right), \tag{2}$$

where $\mathcal{E}$ is the adaptation knowledge extraction operation.

Once the adaptation knowledge is extracted, it is transferred to the pre-trained target model, resulting in a knowledge-enhanced target model θ_{t}^* as:

$$\theta_{\text{t}}^* = \mathcal{J}\left(\theta_{\text{pt-t}}, \delta_{\text{s}}\right), \tag{3}$$

where $\mathcal{J}$ is the adaptation knowledge transfer operation.

The enhanced target model θ_{t}^* is expected to obtain the effects of fine-tuning in two perspectives: (1) preservation of its pre-trained knowledge, and (2) superiority over the fine-tuned source model for its usefulness. We can formalize our objective as follows:

$$\max\left(\mathcal{P}(\theta_{\text{pt-t}}), \mathcal{P}(\theta_{\text{ft-s}})\right) \leqslant \mathcal{P}(\theta_{\text{t}}^*) \lesssim \mathcal{P}(\theta_{\text{ft-t}}), \tag{4}$$

where $\mathcal{P}(\theta)$ refers to evaluation performance of model θ. The target model fine-tuned with labeled data, $\mathcal{P}(\theta_{\text{ft-t}})$, serves as a theoretical upper bound (Burns et al. 2024).

4 Method

We introduce the **TRANS**ferable **M**odel-agnost**I**c adap**TER** (**TransMiter**), a light-weight adapter which efficiently enhances stronger model by leveraging weaker model's adaptation knowledge without backpropagation. We provide TransMiter's architectural design and the adaptation knowledge extraction process $\mathcal{E}$ in Section 4.1. We then present its forward-only adaptation knowledge transfer process $\mathcal{J}$, which facilitates efficient adaptation transfer, in Section 4.2.

4.1 Transferable Model-Agnostic Adapter

Previous adaptation techniques (Zhou et al. 2022b; Khattak et al. 2023a; Zhou et al. 2022a; Zhu et al. 2023; Khattak et al. 2023b; Li et al. 2024; Ouali et al. 2023) rely on internal model parameters that differ across VLMs, making direct transfer across different VLMs challenging. To mitigate this problem, our adapter uses the predicted logits from the VLM as input. The logits exhibit two key properties: (1) a fixed dimensionality that aligns with the number of classes, and (2) semantic consistency for each logit element. These properties facilitate effective communication (Moschella et al. 2023) between different models, enabling seamless reuse of our adapter. Moreover, our adapter does not require access to the parameters of VLM, thereby achieving high efficiency.

Prediction-based Adapter. Given an image x, we start with the input logits $z_{\text{pt-s}} \in \mathbb{R}^{N_{\text{task}}}$ obtained from the pre-trained weak VLM $\theta_{\text{pt-s}}$, where N_{task} refers to the number of task classes. These logits are projected into a D-dimensional latent space by multiplying them with the projection matrix $W_{\text{in}} \in \mathbb{R}^{N_{\text{task}} \times D}$, producing the input feature h_{s}. Then the input feature h_{s} is passed through a transformation function $f : \mathbb{R}^D \to \mathbb{R}^D$, which consists of a single MLP layer with a residual connection, yielding $\hat{h}_{\text{s}}$. Finally, the output feature $\hat{h}_{\text{s}}$ is projected back into the original logit space by multiplying them with a reconstruction matrix $W_{\text{out}} \in \mathbb{R}^{D \times N_{\text{task}}}$. As a result, we can obtain refined logits of the weak VLM $\hat{z}_{\text{s}}$, which can be written as:

$$\begin{aligned} z_{\text{pt-s}} &= [\textbf{sim}(\mathcal{V}_{\text{pt-s}}(x), \mathcal{T}_{\text{pt-s}}(c))]_{c \in C_{\text{task}}} \in \mathbb{R}^{N_{\text{task}}}, \\ h_{\text{s}} &= z_{\text{pt-s}} W_{\text{in}} \in \mathbb{R}^D, \\ \hat{h}_{\text{s}} &= f(h_{\text{s}}) \in \mathbb{R}^D, \\ \hat{z}_{\text{s}} &= \hat{h}_{\text{s}} W_{\text{out}} \in \mathbb{R}^{N_{\text{task}}}, \end{aligned} \tag{5}$$

where $f(h) = h + \text{MLP}(h)$.

A key consideration is adapting to the task while preserving the generalized knowledge in the original model. Since TransMiter employs simple 2D matrices to facilitate transitions between the logit and latent space, we enforce an inverse relationship between the projection and reconstruction matrices. If the transformation function f operates as an identity function (*e.g.,* , $f(h) = h$), the output remains identical to the original logits, making an inverse relationship between the matrices reasonable. Here, we adopt orthogonality for the two matrices, *i.e.,* $W_{\text{in}} W_{\text{in}}^\intercal = W_{\text{out}}^\intercal W_{\text{out}} = I$, and set the two matrices in a transposed relationship, *i.e.,* $W_{\text{in}} = W_{\text{out}}^\intercal$. These properties provide the projection and reconstruction matrices with an implicit regularization effect (Liu et al. 2021; Bansal, Chen, and Wang 2018; Qiu et al. 2023) and establish an inverse relationship, *i.e.,* $W_{\text{in}} W_{\text{out}} = I$, between them. For simplicity, we will denote W_{out} as W_{s}, representing the transition matrix for the weak VLM. We can rewrite (5) in a simplified form as follows:

$$\hat{z}_{\text{s}} = f(z_{\text{pt-s}} W_{\text{s}}^\intercal) W_{\text{s}} \in \mathbb{R}^{N_{\text{task}}}. \tag{6}$$

The training objective for TransMiter is to ensure that our refined prediction $\hat{z}_{\text{s}}$ closely mimics the prediction of the fine-tuned weak VLM, $z_{\text{ft-s}}$. By doing so, we can obtain the adaptation knowledge δ_{s} from the fine-tuned model in an unsupervised manner, leveraging unlabeled data $\mathcal{D}_{\text{u}}$. With the KL divergence loss, the training objective can be written as:

$$\begin{aligned} z_{\text{ft-s}} &= [\textbf{sim}(\mathcal{V}_{\text{ft-s}}(x), \mathcal{T}_{\text{ft-s}}(c))]_{c \in C_{\text{task}}} \in \mathbb{R}^{N_{\text{task}}}, \\ \mathcal{L} &= \text{KL}\left(\sigma(z_{\text{ft-s}}/\tau_{\text{ft-s}}), \sigma(\hat{z}_{\text{s}}/\tau_{\text{pt-s}})\right), \end{aligned} \tag{7}$$

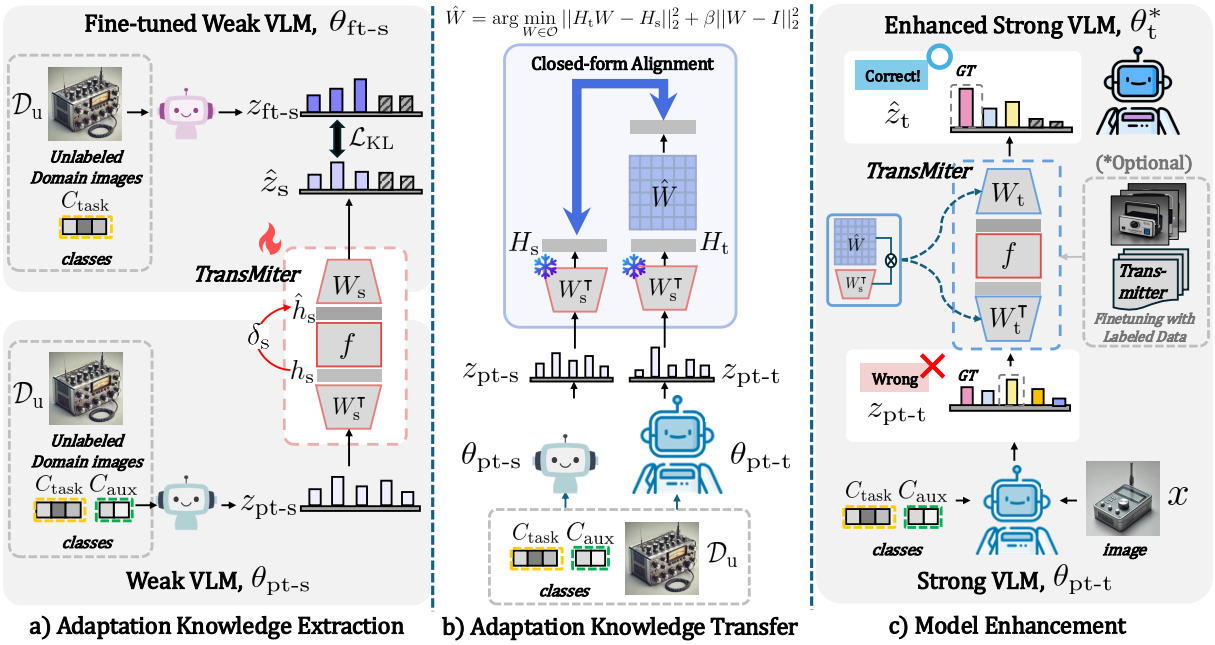

Figure 2: **Overall pipeline. (a) Adaptation Knowledge Extraction.** Given pre-trained $\theta_{\text{pt-s}}$ and fine-tuned weak VLMs $\theta_{\text{ft-s}}$, TransMiter captures the adaptation knowledge δ_{s}, by minimizing the distance between the refined logits $\hat{z}_{\text{s}}$ and the fine-tuned weak VLM logits $z_{\text{ft-s}}$. The adapter takes the zero-shot model logits as input and incorporates both task classes C_{task} and auxiliary classes C_{aux}. **(b) Adaptation Knowledge Transfer.** Once the strong VLM $\theta_{\text{pt-t}}$ is available, the mapping matrix $\hat{W}$ is computed using a closed-form solution to align the input features between the weak (H_{s}) and strong (H_{t}) VLMs, replacing the original transition matrix W_{s} with $W_{\text{t}} = \hat{W}^{\intercal}W_{\text{s}}$. **(c) Model Enhancement.** During the inference with the target VLM using TransMiter θ_{t}^{*}, the pre-trained target VLM logits $z_{\text{pt-t}}$ are passed through the adapter, resulting in enhanced predictions. Subsequently, as TransMiter offers a strong initial point, it can be fine-tuned with labeled data to maximize its capability.

where σ and τ denote the softmax function and temperature scaling factor, respectively.

Auxiliary Class Expansion. As described in Section 3, the text features for each class serve as anchors. Here, logits can be viewed as a relative representation (Moschella et al. 2023) of the image, where each logit element represents the relative distance between the image feature and the text features of corresponding anchor classes. As emphasized in (Moschella et al. 2023), employing a sufficiently large number of anchors is crucial to facilitate effective transfer.

Building on this insight, we incorporate additional auxiliary classes C_{aux} alongside the task classes C_{task} to construct the input logits. We collect the candidates of auxiliary classes from OpenImages (Kuznetsova et al. 2018), which includes a broad range of real-world objects. With the dimensionality of the logit space M, where $M > N_{\text{task}}$, we randomly sample auxiliary classes in a total number of $N_{\text{aux}} = M - N_{\text{task}}$. We compute logits that include auxiliary classes and pass them through the prediction adapter, which is written as:

$$z_{\text{pt-s}} = [\textbf{sim}(\mathcal{V}_{\text{pt-s}}(x), \mathcal{T}_{\text{pt-s}}(c))]_{c \in C_{\text{task}} \cup C_{\text{aux}}} \in \mathbb{R}^{M}. \quad (8)$$

Note that the transition matrix W_{s} is re-shaped to $\mathbb{R}^{D \times M}$, where D refers to the dimensionality of the latent space. With the reshaped transition matrix W_{s}, the output logits will have a dimensionality of M. Here, only the output logits corresponding to the task classes are used.

4.2 Forward-only Adapter Transfer

The primary goal of TransMiter is to transfer adaptation knowledge acquired from weaker to stronger models without fine-tuning it directly. A straightforward transfer is directly using the logits from the pre-trained strong VLM $z_{\text{pt-t}}$ as TransMiter's input, which can be written as:

$$\begin{aligned} z_{\text{pt-t}} &= [\textbf{sim}(\mathcal{V}_{\text{pt-t}}(x), \mathcal{T}_{\text{pt-t}}(c))]_{c \in C_{\text{task}} \cup C_{\text{aux}}} \in \mathbb{R}^{M}, \\ \hat{z}_{\text{t}} &= f(z_{\text{pt-t}}W_{\text{s}}^{\intercal})W_{\text{s}} \in \mathbb{R}^{M}, \\ \hat{z}_{\text{t}} &:= [\hat{z}_{\text{t}}(c)]_{c \in C_{\text{task}}} \in \mathbb{R}^{N_{\text{task}}}, \end{aligned} \quad (9)$$

where $\mathcal{V}_{\text{pt-t}}$ and $\mathcal{T}_{\text{pt-t}}$ denote the image and text encoder of pre-trained strong VLM, respectively. To maintain semantic consistency, the same classes ($C_{\text{task}} \cup C_{\text{aux}}$) must be used to compute the logits for the pre-trained strong VLM as those used for weak VLM.

However, since our adapter is initially trained on the distribution of weak VLM's logits, discrepancies between the logit distributions of weak and strong VLMs lead to suboptimal transferability. To address this, we focus on aligning the input features h_{s} and h_{t} from weak and strong VLMs, resulting in strong VLM's transition matrix, *i.e.,* basis.

Source → Target (Strategy)	Model	Method	FLOPs (G)	Avg.	Image Net	Caltech	Pets	Cars	Flowers	Food	Air craft	SUN	DTD	Euro SAT	UCF
RN50 → ViT-B/16 (CoOP)	Source	Fine-tuning	0	73.39	62.91	91.56	86.25	73.20	94.74	74.57	32.05	68.96	63.44	83.33	76.32
	Target	Zero-shot	0	65.33	66.72	93.31	89.10	65.54	70.77	**85.88**	24.81	62.57	44.09	48.38	67.46
		Prompt Transfer	0	62.74	64.27	90.63	86.43	59.03	63.37	85.00	4.13	55.63	38.10	78.40	65.10
		EFT	12.27	75.78	66.55	94.32	88.51	74.64	94.07	81.63	**33.90**	70.60	64.30	**85.82**	79.19
		TransMiter (ours)	0.01	**77.16**	**69.68**	**95.21**	**91.60**	**75.92**	**95.13**	84.52	33.69	**72.76**	**67.38**	82.62	**80.30**
		Fine-tuning	0	79.92	71.86	95.48	91.94	82.65	97.37	84.22	43.46	74.86	68.68	85.44	83.12
RN50→ ViT-L/14 (CoOP)	Source	Fine-tuning	0	73.39	62.91	91.56	86.25	73.20	94.74	74.57	32.05	68.96	63.44	83.33	76.32
	Target	Zero-shot	0	72.54	73.46	95.13	93.49	76.88	79.46	90.91	32.55	67.66	53.07	60.33	74.99
		Prompt Transfer	0	77.28	75.00	96.07	**94.00**	75.83	85.70	**90.97**	32.17	72.60	61.20	87.03	79.50
		EFT	12.27	80.27	73.99	96.05	91.31	**82.12**	95.70	87.91	**40.01**	73.99	68.75	**89.24**	**83.89**
		TransMiter (ours)	0.01	**80.39**	**75.60**	**96.52**	93.42	81.39	**96.06**	89.50	39.52	**75.21**	**69.90**	83.77	83.36
		Fine-tuning	0	84.57	78.24	97.00	94.41	88.88	98.98	89.79	56.28	77.76	72.97	88.57	87.39
ViT-B/16→ViT-L/14 (CoOP)	Source	Fine-tuning	0	79.92	71.86	95.48	91.94	82.65	**97.37**	84.22	43.46	74.86	68.68	85.44	83.12
	Target	Zero-shot	0	72.54	73.46	95.13	93.49	76.88	79.46	90.91	32.55	67.66	53.07	60.33	74.99
		Prompt Transfer	0	78.50	76.77	96.27	**94.83**	78.27	86.37	**91.30**	36.33	73.73	62.40	**87.27**	80.00
		EFT	35.16	81.04	74.90	96.59	93.20	**84.47**	96.14	88.49	41.93	75.01	69.98	86.17	84.51
		TransMiter (ours)	0.01	**82.05**	**76.78**	**96.96**	94.24	84.27	97.25	90.15	**44.38**	**76.72**	**72.04**	84.72	**85.01**
		Fine-tuning	0	84.57	78.24	97.00	94.41	88.88	98.98	89.79	56.28	77.76	72.97	88.57	87.39
ViT-B/16 → ViT-L/14 (PromptSRC)	Source	Fine-tuning	0.37	81.74	72.81	95.77	93.85	80.89	97.04	87.50	**45.20**	76.73	73.31	**91.02**	85.05
	Target	Zero-shot	0	72.54	73.46	95.13	93.49	76.88	79.46	90.91	32.55	67.66	53.07	60.33	74.99
		Prompt Transfer	1.34	73.41	73.23	92.73	91.43	75.03	80.57	90.00	30.37	68.77	54.27	75.90	75.20
		EFT	36.84	81.34	75.98	96.09	94.22	82.80	95.51	90.85	41.56	75.20	72.58	85.64	84.26
		TransMiter (ours)	0.01	**82.51**	**77.11**	**97.03**	**94.93**	**83.11**	**97.39**	**91.66**	42.71	**77.14**	**74.09**	86.95	**85.52**
		Fine-tuning	1.31	85.17	79.05	97.38	95.07	86.85	98.66	91.96	53.07	79.97	75.73	90.92	88.23

Table 1: **Performance on base-to-base adaptation transfer.** We combine source and target models among RN50, ViT-B/16, and ViT-L/14. Gray-colored rows represent the performance of the target model when fine-tuned, serving as the upper bound.

Basis Change. After training TransMiter, unlabeled images $x \in \mathcal{D}_{\text{u}}$ are used to extract the input feature of weak VLM h_{s} by multiplying $z_{\text{pt-s}}$ with $W_{\text{s}}^{\intercal}$. The same images are used to compute strong VLM logits $z_{\text{pt-t}}$ and obtain the input feature of strong VLM h_{t} by also multiplying $z_{\text{pt-t}}$ with $W_{\text{s}}^{\intercal}$. Our goal is to find $\hat{W}$, a mapping between the input feature of weak and strong VLMs, by minimizing the distance between h_{s} and h_{t} of all the images. With a regularization term, the objective can be written as:

$$\begin{aligned} H_{\text{s}} &= [z_{\text{pt-s}}\left(C_{\text{task}} \cup C_{\text{aux}}; x\right) W_{\text{s}}^{\intercal}]_{x \in \mathcal{D}_{\text{u}}} \in \mathbb{R}^{|\mathcal{D}_{\text{u}}| \times D}, \\ H_{\text{t}} &= [z_{\text{pt-t}}\left(C_{\text{task}} \cup C_{\text{aux}}; x\right) W_{\text{s}}^{\intercal}]_{x \in \mathcal{D}_{\text{u}}} \in \mathbb{R}^{|\mathcal{D}_{\text{u}}| \times D}, \\ \hat{W} &= \underset{W \in \mathcal{O}}{\arg\min} ||H_{\text{t}} W - H_{\text{s}}||_2^2 + \beta ||W - I||_2^2, \end{aligned} \tag{10}$$

where $\mathcal{O} = \{W \in \mathbb{R}^{D \times D}, W^{\intercal} W = I_D\}$ denotes the orthogonality constraint. The mapping matrix $\hat{W}$ can be derived as the solution to the Orthogonal Procrustes problem (Schönemann 1966) via singular value decomposition, making backpropagation unnecessary. The solution can be written as:

$$\begin{aligned} U, S, V &= \textbf{SVD}(H_{\text{t}}^{\intercal} H_{\text{s}} + \beta I), \\ \hat{W} &= U V^{\intercal}, \end{aligned} \tag{11}$$

where β refers to the regularization weight.

Here, the projection matrix for strong VLM is updated to $W_{\text{s}}^{\intercal} \hat{W}$. Likewise, the reconstruction matrix for strong VLM is updated to $\hat{W}^{\intercal} W_{\text{s}}$. By doing so, the inverse relationship between the projection and reconstruction matrices and their orthogonality are preserved. This process ensures that strong VLM retains its pre-trained knowledge while absorbing adaptation knowledge from weak VLM. The final outputs of the enhanced strong VLM can be written as:

$$\begin{aligned} \hat{z}_{\text{t}} &= f(z_{\text{pt-t}} W_{\text{t}}^{\intercal}) W_{\text{t}} \in \mathbb{R}^{M}, \\ \hat{z}_{\text{t}} &:= [\hat{z}_{\text{t}}(c)]_{c \in C_{\text{task}}} \in \mathbb{R}^{N_{\text{task}}}. \end{aligned} \tag{12}$$

5 Experiments

5.1 Experimental setup

Settings. During the adaptation knowledge transfer, we need a fine-tuned weak VLM. Based on how this model was fine-tuned, we set up two main evaluation setups: (1) **Base-to-base adaptation transfer**, where the weak model is trained on all the task classes, and (2) **Base-to-novel adaptation transfer**, where the weak model is fine-tuned on a subset of classes (*i.e.,* seen classes), and tested on both seen and the remaining classes (*i.e.,* unseen classes). Through these configurations, we aim to evaluate how effectively the methods transfer task-specific adaptation knowledge from weaker to stronger models while preserving their generalized knowledge. We evaluate our method on 11 visual recognition datasets for both settings. See Supplement for more details.

Baselines. We adopt CLIP (Radford et al. 2021) for VLMs, varying with architecture and size, *e.g.,* ResNet-50 (RN50), ViT-B/16, and ViT-L/14. We assume that the target model has stronger generalization capabilities than the source model in terms of its size and architecture. For fine-tuning the source model, we apply CoOp (Zhou et al.

2022b) and PromptSRC (Khattak et al. 2023b). In each setting, we evaluate three methods within our framework: (1) **Source Fine-tuning** $\theta_{\text{ft-s}}$, where the source model is fine-tuned; (2) **Target Zero-shot** $\theta_{\text{pt-t}}$, the zero-shot performance of the pre-trained target model; and (3) **Target TransMiter** θ_{t}^{*}, where TransMiter is applied to the pre-trained target model. As described in Section 3, our objective is to improve the strong model beyond the performance of the fine-tuned weak model (**Source Fine-tuning**) and pre-trained strong model (**Target Zero-shot**). Additionally, we report the performance of **Target Fine-tuning** $\theta_{\text{ft-t}}$, where the strong model is fine-tuned. It serves as the 'upper bound'.

To compare with transferable adaptation methods, we employ the following baselines: **EFT** (Mitchell et al. 2024; Liu et al. 2024) and **Prompt Transfer** (Su et al. 2022). Prompt Transfer transfers learned soft prompts from weaker to stronger models. This approach requires resource-intensive training to align the representation spaces between weak and strong models. To achieve this, we adopt a naive knowledge distillation method (Li et al. 2024; Hinton, Vinyals, and Dean 2015). In contrast, EFT operates in a training-free manner but relies on multiple weak models to assist the strong model during inference, leading to high inference costs. Additional details are provided in the Supplement.

5.2 Experimental Results

Base-to-base adaptation transfer. Table 1 reports the performance of our method and baselines across 11 visual recognition datasets under the base-to-base adaptation transfer setting. To evaluate the inference efficiency of each method, we also reported the additionally introduced GFLOPs compared to the base model.

In all source-target combinations, TransMiter outperforms Target Zero-shot and Source Fine-tuning in average performance across 11 tasks. With ResNet-50 as the source and ViT-B/16 as the target, TransMiter surpasses Target Zero-shot and source Fine-tuning by a margin of 11.77% and 3.77%. As the target model advances, for instance from ViT-B/16 to ViT-L/14 with ResNet-50 as the source, the performance increases further, from 77.16% to 80.39%. This result is not surprising, as stronger zero-shot performance generally corresponds to enhanced fine-tuning capabilities.

Interestingly, using a stronger source model ViT-B/16 with ViT-L/14 as the target further improves the performance of the target model from 80.39% to 82.05%. Furthermore, when a more advanced fine-tuning approach is applied to the source model, CoOp to PromptSRC, the performance of the target model with TransMiter increases from 82.05% to 82.51%. These results demonstrate that more generalized and well-adapted models possess strong capabilities for transferring adaptation knowledge.

When comparing TransMiter with other transferable adaptation methods, it shows the best average accuracy across all source-target combinations. This is especially notable given that Prompt Transfer requires substantial retraining efforts whenever the new model comes, while TransMiter uses a simple forward-only algorithm to update the adapter. Additionally, TransMiter achieves efficient adaptation with minimal computational overhead, requiring only 0.01 GFLOPs, while EFT incurs a significantly higher infer-

Dataset	Set	Source	Target				
		Fine-tuning	Zero-shot	Prompt-transfer	EFT	**TransMiter** (ours)	Fine-tuning
Average on 11 datasets	Base	84.19	76.71	76.07	84.32	**85.27**	87.15
	Novel	75.35	80.85	77.84	80.24	**81.07**	81.59
	HM	79.53	78.73	76.94	82.23	**83.12**	84.28
ImageNet	Base	77.73	79.19	78.03	80.83	**81.73**	83.05
	Novel	70.42	74.03	73.83	74.43	**75.84**	76.97
	HM	73.89	76.52	75.87	77.50	**78.67**	79.90
Caltech	Base	98.19	95.61	92.97	96.82	**98.58**	98.47
	Novel	94.21	**97.71**	95.47	96.72	96.29	97.42
	HM	96.16	96.65	94.20	96.77	**97.42**	97.94
Pets	Base	95.39	95.16	92.93	94.52	**95.80**	96.01
	Novel	95.41	**98.10**	97.37	97.13	97.20	98.68
	HM	95.40	**96.61**	95.10	95.81	96.50	97.33
Cars	Base	78.29	74.51	71.60	79.47	**79.78**	84.06
	Novel	75.13	**84.67**	83.77	83.88	84.01	84.79
	HM	76.68	79.27	77.21	81.62	**81.84**	84.42
Flowers	Base	**97.94**	82.72	81.10	95.16	97.28	98.73
	Novel	76.93	**82.98**	78.77	81.61	82.01	82.48
	HM	86.17	82.85	79.92	87.86	**88.99**	89.88
Food101	Base	90.68	93.75	92.90	93.71	**94.18**	94.29
	Novel	91.20	**94.92**	93.33	93.97	94.63	95.04
	HM	90.94	94.33	93.11	93.84	**94.40**	94.66
Aircraft	Base	42.84	37.21	34.47	44.68	**44.78**	52.32
	Novel	38.11	**44.33**	39.07	44.01	43.39	43.55
	HM	40.34	40.46	36.63	**44.34**	44.07	47.54
SUN397	Base	82.67	73.52	74.20	81.42	**82.69**	85.05
	Novel	78.54	77.72	77.50	78.44	**80.85**	81.42
	HM	80.55	75.56	75.81	79.90	**81.76**	83.19
DTD	Base	**83.37**	61.23	57.63	82.02	**83.37**	84.57
	Novel	58.53	**70.89**	59.10	68.68	69.08	70.41
	HM	68.78	65.71	58.36	74.76	**75.56**	76.84
EuroSAT	Base	92.26	70.93	83.13	92.19	**92.94**	93.29
	Novel	71.71	83.44	77.80	82.69	**85.82**	83.71
	HM	80.70	76.68	80.38	87.18	**89.24**	88.24
UCF	Base	86.73	79.94	77.83	86.66	**86.88**	88.81
	Novel	78.64	80.58	80.27	81.11	**82.64**	83.07
	HM	82.48	80.26	79.03	83.79	**84.71**	85.85

Table 2: **Performance on base-to-novel adaptation transfer.** Source and target models are ViT-B/16 and ViT-L/14, respectively. We adopt PromptSRC as a fine-tuning strategy.

Strategy	Method	Base	Novel	HM	FPS
CLIP	Zero-shot	76.71	80.85	78.73	286
CLIP	**TransMiter+**	86.00	80.77	83.30	285
MaPLe	Fine-tuning	85.58	79.35	82.35	261
MaPLe	**TransMiter+**	**87.41**	**81.54**	**84.37**	**285**
PromptSRC	Fine-tuning	87.15	81.59	84.28	258
PromptSRC	**TransMiter+**	**87.59**	**81.97**	**84.69**	**285**
HPT	Fine-tuning	**87.98**	82.13	84.95	250
HPT	**TransMiter+**	87.53	**82.65**	**85.02**	**285**
LwEIB	Fine-tuning	86.86	**82.74**	84.75	148
LwEIB	**TransMiter+**	**87.55**	82.52	**84.96**	**285**

Table 3: **Performance on supervised fine-tuning.** All the methods use ViT-L/14 as a base model. TransMiter+ uses ViT-B/16 with each specified strategy for adaptation knowledge extraction.

Model		Method			Base-to-novel			Base-to-base
		PA	ACE	BC	Base	Novel	HM	
Source	(a)	Zero-shot			69.51	74.29	71.72	65.33
	(b)	✓			82.45	75.90	78.86	79.30
	(c)	✓	✓		83.83	76.21	79.62	81.39
	(d)	Fine-tuning			84.19	75.35	79.28	81.74
Target	(e)	Zero-shot			76.71	80.85	78.63	72.54
	(f)	✓			80.82	79.64	80.09	76.25
	(g)	✓	✓		82.90	79.80	81.24	78.98
	(h)	✓		✓	81.60	80.24	80.80	77.65
	(i)	✓	✓	✓	**85.27**	**81.07**	**83.12**	**82.51**

Table 4: **Ablation study of TransMiter.** PA, ACE, and BC refer to "Prediction-based Adapter", "Auxiliary Class Expansion", and "Basis Change", respectively.

ence cost. This indicates that TransMiter is not only effective in transferring adaptation knowledge but also efficient.

Base-to-novel adaptation transfer. Table 2 presents an experiment under the base-to-novel adaptation transfer setting to verify whether the transferable adaptation methods can maintain generalization ability in transferred models. On average performance across all datasets, TransMiter outperforms all baseline methods in all metrics: Base, Novel, and HM. Notably, TransMiter is the only method that maintains its Novel performance without any degradation compared to Target Zero-shot. These results underscore TransMiter's superior ability to transfer specialized knowledge while maintaining strong generalization capabilities.

In addition to the unsupervised adaptation transfer setting, we further fine-tune TransMiter with labeled data (TransMiter+) to fully assess its capability, and compare it against recent state-of-the-art supervised fine-tuning approaches in Table 3. We integrate additional adaptation strategies, such as MaPLe (Khattak et al. 2023a), HPT (Wang et al. 2024), and LwEIB (Yang et al. 2025b), into TransMiter for a weaker model (*e.g.,* ViT-B/16) adaptation, which serves as a 'fine-tuned weak VLM' during the adaptation knowledge extraction process. Even without the adaptation knowledge extraction phase (row 2), TransMiter+ leads to a substantial improvement over the zero-shot baseline. Moreover, since TransMiter's knowledge extraction and transfer provide a strong initialization, fine-tuning TransMiter with supervision after the adaptation knowledge transfer consistently yields improved performance across various strategies. Importantly, combining any fine-tuning strategy with TransMiter not only surpasses the performance of its fine-tuning counterpart but also lowers inference cost, demonstrating both high effectiveness and efficiency.

5.3 Ablation study

In this section, we evaluate the effectiveness of TransMiter through ablations. Unless otherwise stated, we use ViT-B/16 and ViT-L/14 as the source and target VLMs, respectively, and PromptSRC (Khattak et al. 2023b) for fine-tuning.

Component analysis in TransMiter. Table 4 provides a comparison of TransMiter components when applying to both source and target models, with results reported as the average accuracy of all tasks. When the prediction-based adapter (PA) is applied naively to the source model ((a)→(b)), it significantly improves the source model's zero-shot performance, but still lags behind the performance of the fine-tuned source model ((b) vs. (d)). With the addition of auxiliary classes ((b)→(c)), the prediction adapter better approximates or even surpasses the fine-tuned source model in certain cases ((c) vs. (d)), such as the performance of Novel and HM in the base-to-novel setting. These results indicate that TransMiter can effectively capture the adaptation knowledge from the source model.

Also, naively applying the adapter to the target model ((e)→(f)) leads to an improvement over the zero-shot target model. When auxiliary classes (ACE) are introduced ((f)→(g)), the target model is further enhanced. Additionally, incorporating basis change (BC) with the adapter ((f)→(h)) also contributes to better transferability. When both methods are applied together in the target model ((i)), it achieves the highest performance among the tested configurations. These results demonstrate that using auxiliary classes and basis change methodology has strong complementarity and serves as a highly effective method for enhancing the transferability of TransMiter.

Method	Labels	Transfer		Inference	HM
		Time (s)	Mem. (Mb)	FPS	
EFT	✗	0	0	169	81.62
Prompt Transfer	✗	480	33188	258	77.21
TransMiter	✗	24	3784	**285**	**81.84**
PromptSRC	✓	321	31014	258	84.42
TransMiter+	✓	52	2996	**285**	**85.16**

Table 5: **Efficiency analysis.** Computation costs are measured using a single A6000 GPU on StanfordCars dataset.

Efficiency analysis. In Table 5, we evaluate the computational costs of TransMiter on StanfordCars dataset. Among unsupervised transfer methods, TransMiter achieves strong performance and fast inference, while its backpropagation-free design significantly reduces transfer time and memory. With a few labeled data, TransMiter+ surpasses the performance and inference speed of the fine-tuning strategy (PromptSRC) used in its development. This improvement takes less than a minute of extra training, as it updates only the adapter on top of the VLM, whereas PromptSRC requires almost full-model gradient computation, highlighting TransMiter's superior efficiency.

6 Conclusion

We present TransMiter, a transferable model-agnostic adapter that enhances vision-language models without backpropagation. TransMiter utilizes the model's prediction to capture adaptation knowledge, enabling seamless knowledge transfer from weaker to stronger models, regardless of architecture or size. With its simple structure and forward-only transfer, TransMiter ensures high computational efficiency. Experimental results demonstrate its exceptional transferability across various models, offering a practical solution for scalable and efficient adaptation.

Acknowledgments

This work was partly supported by Korea Research Institute for defense Technology planning and advancement - Grant funded by Defense Acquisition Program Administration(DAPA)(KRIT-CT-23-021) (30%), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2023R1A2C2005373) (25%), Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No. RS-2024-00443251, Accurate and Safe Multimodal, Multilingual Personalized AI Tutors) (25%), and the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (grant number: HR20C0021) (20%).

References

Bansal, N.; Chen, X.; and Wang, Z. 2018. Can We Gain More from Orthogonality Regularizations in Training Deep Networks? In *NeurIPS*.

Burns, C.; Izmailov, P.; Kirchner, J. H.; Baker, B.; Gao, L.; Aschenbrenner, L.; Chen, Y.; Ecoffet, A.; Joglekar, M.; Leike, J.; Sutskever, I.; and Wu, J. 2024. Weak-to-Strong Generalization: Eliciting Strong Capabilities With Weak Supervision. In *ICML*.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*.

Cherti, M.; Beaumont, R.; Wightman, R.; Wortsman, M.; Ilharco, G.; Gordon, C.; Schuhmann, C.; Schmidt, L.; and Jitsev, J. 2023. Reproducible scaling laws for contrastive language-image learning. In *CVPR*.

Chijiwa, D. 2024. Transferring Learning Trajectories of Neural Networks. In *ICLR*.

Cho, E.; Kim, J.; and Kim, H. J. 2023. Distribution-aware prompt tuning for vision-language models. In *ICCV*.

Han, C.; Xu, J.; Li, M.; Fung, Y.; Sun, C.; Jiang, N.; Abdelzaher, T. F.; and Ji, H. 2024. Word Embeddings Are Steers for Language Models. In *ACL*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.; Parekh, Z.; Pham, H.; Le, Q. V.; Sung, Y.; Li, Z.; and Duerig, T. 2021. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *ICML*.

Khattak, M. U.; Rasheed, H.; Maaz, M.; Khan, S.; and Khan, F. S. 2023a. Maple: Multi-modal prompt learning. In *CVPR*.

Khattak, M. U.; Wasim, S. T.; Naseer, M.; Khan, S.; Yang, M.-H.; and Khan, F. S. 2023b. Self-regulating Prompts: Foundational Model Adaptation without Forgetting. In *ICCV*.

Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J. R. R.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Malloci, M.; Duerig, T.; and Ferrari, V. 2018. The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*.

Lee, D.; Song, S.; Suh, J.; Choi, J.; Lee, S.; and Kim, H. J. 2023. Read-only prompt optimization for vision-language few-shot learning. In *ICCV*.

Li, Z.; Li, X.; Fu, X.; Zhang, X.; Wang, W.; Chen, S.; and Yang, J. 2024. Promptkd: Unsupervised prompt distillation for vision-language models. In *CVPR*.

Liu, A.; Han, X.; Wang, Y.; Tsvetkov, Y.; Choi, Y.; and Smith, N. A. 2024. Tuning Language Models by Proxy. *COLM*.

Liu, W.; Lin, R.; Liu, Z.; Rehg, J. M.; Paull, L.; Xiong, L.; Song, L.; and Weller, A. 2021. Orthogonal Over-Parameterized Training. In *CVPR*.

Lu, X.; Brahman, F.; West, P.; Jung, J.; Chandu, K.; Ravichander, A.; Ammanabrolu, P.; Jiang, L.; Ramnath, S.; Dziri, N.; et al. 2023. Inference-time policy adapters (ipa): Tailoring extreme-scale lms without fine-tuning. In *EMNLP*.

Lu, Z.; Bai, J.; Li, X.; Xiao, Z.; and Wang, X. 2024. Beyond Sole Strength: Customized Ensembles for Generalized Vision-Language Models. In *ICML*.

Mitchell, E.; Rafailov, R.; Sharma, A.; Finn, C.; and Manning, C. D. 2024. An Emulator for Fine-tuning Large Language Models using Small Language Models. In *ICLR*.

Moschella, L.; Maiorca, V.; Fumero, M.; Norelli, A.; Locatello, F.; and Rodolà, E. 2023. Relative representations enable zero-shot latent space communication. In *ICLR*.

Nasser, S. A.; Gupte, N.; and Sethi, A. 2024. Reverse knowledge distillation: Training a large model using a small one for retinal image matching on limited data. In *CVPR*.

Ouali, Y.; Bulat, A.; Matinez, B.; and Tzimiropoulos, G. 2023. Black box few-shot adaptation for vision-language models. In *CVPR*.

Park, J.; Ko, J.; and Kim, H. J. 2024. Prompt learning via meta-regularization. In *CVPR*.

Qiu, Z.; Liu, W.; Feng, H.; Xue, Y.; Feng, Y.; Liu, Z.; Zhang, D.; Weller, A.; and Schölkopf, B. 2023. Controlling Text-to-Image Diffusion by Orthogonal Finetuning. In *NeurIPS*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Schönemann, P. H. 1966. A generalized solution of the orthogonal procrustes problem. *Psychometrika*.

Su, Y.; Wang, X.; Qin, Y.; Chan, C.; Lin, Y.; Wang, H.; Wen, K.; Liu, Z.; Li, P.; Li, J.; Hou, L.; Sun, M.; and Zhou, J. 2022. On Transferability of Prompt Tuning for Natural Language Processing. In *NAACL-HLT*.

Wang, Y.; Jiang, X.; Cheng, D.; Li, D.; and Zhao, C. 2024. Learning hierarchical prompt with structured linguistic knowledge for vision-language models. In *AAAI*.

Yang, C.; Park, J.; Kim, H. J.; et al. 2025a. Visual Diversity and Region-aware Prompt Learning for Zero-shot HOI Detection. In *NeurIPS*.

Yang, L.; Zhang, R.-Y.; Chen, Q.; and Xie, X. 2025b. Learning with Enriched Inductive Biases for Vision-Language Models. *IJCV*.

Yu, J.; Wang, Z.; Vasudevan, V.; Yeung, L.; Seyedhosseini, M.; and Wu, Y. 2022. CoCa: Contrastive Captioners are Image-Text Foundation Models. *TMLR*.

Yuan, L.; Tay, F. E.; Li, G.; Wang, T.; and Feng, J. 2020. Revisiting knowledge distillation via label smoothing regularization. In *CVPR*.

Zanella, M.; and Ayed, I. B. 2024. Low-Rank Few-Shot Adaptation of Vision-Language Models. In *CVPRW*.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022a. Conditional Prompt Learning for Vision-Language Models. In *CVPR*.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022b. Learning to Prompt for Vision-Language Models. *IJCV*.

Zhu, B.; Niu, Y.; Han, Y.; Wu, Y.; and Zhang, H. 2023. Prompt-aligned gradient for prompt tuning. In *ICCV*.