Simple yet Effective Semi-supervised Knowledge Distillation from Vision-Language Models via Dual-Head Optimization

Seongjae Kang[1*] **Dong Bok Lee**[1*] **Hyungjoon Jang**[2] **Sung Ju Hwang**[1,3]
[1]KAIST [2]VUNO Inc. [3]DeepAuto.ai
{tjdwo2744,markhi,sjhwang}@kaist.ac.kr
hyungjoon.jang@vuno.co

Abstract

Semi-supervised learning (SSL) has emerged as a practical solution for addressing data scarcity challenges by leveraging unlabeled data. Recently, vision-language models (VLMs), pre-trained on massive image-text pairs, have demonstrated remarkable zero-/few-shot performance that often surpasses SSL approaches due to their exceptional generalization capabilities. This gap motivates us to question: how can we effectively *harness the powerful generalization capabilities of VLMs into task-specific models*? Knowledge distillation (KD) offers a natural framework for transferring VLM capabilities, but we identify that it suffers from *gradient conflicts* between supervised and distillation losses. To address this challenge, we propose Dual-Head Optimization (DHO), which introduces dual prediction heads for each distinct signal. We observe that DHO mitigates *gradient conflicts*, enabling improved feature learning compared to single-head KD baselines, with practical benefits of minimal computational overhead and test-time hyperparameter tuning without retraining. Extensive experiments across 15 datasets show that DHO consistently outperforms KD baselines, often outperforming teacher models with smaller student models. DHO also achieves new state-of-the-art performance on both in-distribution ImageNet semi-supervised learning and out-of-distribution generalization across ImageNet variants. Code and model checkpoints are publicly available at https://github.com/erjui/DHO/tree/main.

1 Introduction

Vision-language models (VLMs), which learn joint vision-language representations through large-scale pre-training, have shown remarkable zero-shot capabilities across diverse tasks [1, 2]. Building upon these strong foundational capabilities, recent work has explored various adaptation strategies, including parameter-efficient approaches such as linear probing [3, 4], lightweight adapters [5, 6], and prompt-based fine-tuning methods [7, 8, 9], demonstrating the potential of VLMs for data-limited visual recognition tasks.

In parallel, semi-supervised learning (SSL) has emerged as a practical approach to address data scarcity by leveraging both labeled and unlabeled data [10, 11, 12, 13]. While these methods have shown success to leverage large amounts of unlabeled data, they often struggle to match the impressive zero- and few-shot capabilities of large pre-trained VLMs [14]. This discrepancy highlights a fundamental limitation: traditional semi-supervised methods, despite their theoretical appeal, remain suboptimal compared to the rich representations learned by foundation models through massive-scale pre-training.

*Equal contribution.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

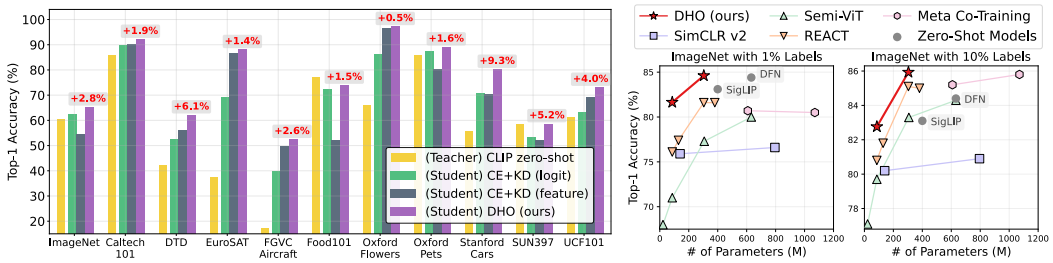

Figure 1: **(Left)**: DHO consistently outperforms single-head baselines on 11 datasets under 16-shot semi-supervised setting. The improvements are evaluated in comparison to the second-best one. **(Right)**: DHO achieves new SoTA on ImageNet in both 1% and 10% labeled data setting, with fewer parameters.

VLMs excel at zero- and few-shot generalization but may lack the fine-grained discriminative power needed for specific tasks, while models trained on limited labeled data capture task-specific patterns but generalize poorly. This complementary nature motivates us to integrate generalist VLM knowledge with task-specific supervision in semi-supervised learning settings. Therefore, the challenge naturally arises: *how can we effectively transfer the powerful capabilities of large VLMs to task-specific models in semi-supervised settings?*

Knowledge distillation [KD; 15] emerges as a natural solution to the challenge, as it offeres an efficient framework to transfer knowledge from large VLMs to student models while simultaneously leveraging task-specific patterns from limited labeled data. However, existing VLM distillation methods primarily focus on general-purpose training [16, 17, 18, 19] or employ multi-stage pipelines with unsupervised pre-distillation [20, 21], which require additional task-specific fine-tuning. Conventional single-stage KD methods, such as logit distillation [15, 22], offer a more direct approach, but we find them *suboptimal* in semi-supervised settings.

Through analysis, we identify that this stems from a fundamental problem: *gradient conflicts* between the supervised loss (from limited labeled data) and the distillation loss (from teacher predictions). This misalignment is particularly severe in semi-supervised settings, where the strong and consistent distillation signal from the teacher can overwhelm the weak and potentially noisy signal from scarce labeled data. Such gradient conflicts (and in Fig. 3) are well-documented to impair effective feature learning [23, 24, 25], preventing the model from finding an optimal balance between task-specific adaptation and general knowledge transfer. We formally characterize when this conflict arises—deriving geometric conditions under which the supervised and KD gradients oppose each other in both the classifier head (Proposition 1) and the shared feature extractor (Proposition 2)—and show that DHO removes the structural guarantee of conflict (Proposition 3).

To address the above issue, we propose a *simple yet effective* distillation framework, DHO (Dual-Head Optimization), which jointly leverages labeled samples and the probabilistic outputs of the teacher model. Specifically, it learns two distinct heads, each optimized with a separate loss: the supervised and the KD loss, respectively. Our analysis reveals that DHO **mitigates gradient conflicts** both in the classification head and the shared feature extractor (in Fig. 3), arising from the two different training signals and **improves feature representations** compared to baselines (Tab. 5 and Fig. 8). Our framework also enables controlling the relative influence of supervised and teacher predictions through **linear combination** of both head outputs, whose effectiveness is demonstrated in Fig. 6. Additionally, DHO enables **post-training hyperparameter adjustment** by tuning the linear combination weights at inference (Fig. 9), eliminating costly training-time hyperparameter search.

We conduct extensive experiments across 15 different datasets including ImageNet [26]. The experimental results demonstrate the **consistent improvement of** DHO over conventional KD methods across all evaluated datasets (Fig. 1-Left). DHO sometimes even **outperforms the zero/few-shot teacher models** with smaller student models with the same labeled data, demonstrating effective task-specific enhancement beyond teacher capabilities (Figs. 4 and 5). Furthermore, DHO achieves **new state-of-the-art (SoTA)** performance on ImageNet semi-supervised setting, improving accuracy by 3% and 0.1% with 1% and 10% labeled data, respectively, while using fewer parameters (Fig. 1-Right). Notably, DHO can be seamlessly **integrated with existing adaptation techniques** with minimal computational overhead, achieving new **SoTA on Out-of-Distribution (OOD) tasks** across ImageNet distribution-shifted variants (Tab. 4 and Appendix F.10).

Our contributions and findings are summarized as follows:

- We firstly **identify** *gradient conflict* when integrating VLMs' general knowledge with task-specific supervision from limited data. To address this, we propose Dual-Head Optimization (DHO), which optimizes the supervised and distillation objectives in separate heads.

- `DHO` effectively resolves *gradient conflicts* both in the classification head and the shared feature extractor, leading to improved **feature representations**. Our framework enables flexible **post-training adjustment** of dual head output weights with **minimal computational overhead**.
- In extensive experiments on 15 datasets, `DHO` **consistently outperforms** baselines and sometimes **surpasses teacher model** performance. It establishes new **SoTA** for **in-distribution ImageNet** (*e.g.*, +3%/+0.1% at 1%/10% labels) with fewer parameters (76M/767M), and also achieves **SoTA OOD generalization** across ImageNet variants when integrated with adaptation methods.

2 Method

2.1 Preliminaries

We begin with preliminaries: a brief background on VLMs, the problem formulation for few-/low-shot learning, and single-head KD baselines. We defer related work—**vision–language pretraining, data-limited adaptation of VLMs, knowledge distillation, and dual-head methods**—to Appendix B.

Background on VLMs. Our work is based on VLMs such as CLIP [1] and ALIGN [2]. These models consist of multimodal encoders: an image encoder $f_{\mathcal{X}} : \mathcal{X} \to \mathbb{R}^d$ and a text encoder $f_{\mathcal{T}} : \mathcal{T} \to \mathbb{R}^d$ where $\mathcal{X}$ and $\mathcal{T}$ denote the domains of images and texts, respectively. For zero-shot classification of VLMs across C classes, we use predefined prompt templates, *e.g.*, "a photo of a `[CLASS]`", where `[CLASS]` is the name of class. Given a set of C target classes, *i.e.*, $y \in \{1, \dots, C\}$, we generate prompted text descriptions $\{t_1, t_2, \dots, t_C\}$. We obtain the categorical probability vector p using the cosine similarity $\texttt{CosSim}(x, y) = \frac{x^\top y}{||x||_2 \cdot ||y||_2}$ over $\{t_1, t_2, \dots, t_C\}$, *i.e.*, $p := \sigma([\texttt{CosSim}(f_{\mathcal{T}}(x), f_{\mathcal{T}}(t_1))/\zeta, \dots, \texttt{CosSim}(f_{\mathcal{T}}(x), f_{\mathcal{T}}(t_C))/\zeta])$, where σ is the softmax function, $\zeta \in \mathbb{R}_{>0}$ is the temperature scaling [15], and final classification is determined by $\arg\max_{c \in \{1,\dots,C\}} p_c$.

Problem Formulation. We focus on transferring knowledge from VLMs to task-specific models under few-shot or low-shot semi-supervised learning scenarios, where both labeled and unlabeled data are utilized. Specifically, given a K-shot and C-class classification problem, we are provided with a labeled dataset $\mathcal{D}^{(l)} = \{(x_n^{(l)}, y_n)\}_{n=1}^N$, where $N = K \times C$ is the total number of labeled examples, and $y_n \in \{1, \dots, C\}$ denotes the class labels. Additionally, we have access to an unlabeled dataset $\mathcal{D}^{(u)} = \{x_m^{(u)}\}_{m=1}^M$ consisting of M unlabeled images. Low-shot learning represents a more realistic setting than traditional few-shot learning where only a small fraction, *e.g.*, 1% ($\frac{N}{N+M} \approx 0.01$) or 10% ($\frac{N}{N+M} \approx 0.1$) of the total dataset is labeled. Our goal is then to develop a student model by leveraging $\mathcal{D}^{(l)}$ and $\mathcal{D}^{(u)}$, guided by the knowledge of the VLM encoders $f_{\mathcal{X}}$ and $f_{\mathcal{T}}$. The student model consists of a **feature extractor** $g(x)$ **parameterzied by** θ and **a linear prediction head** $h(z) = Wz + b \in \mathbb{R}^c$, where $W \in \mathbb{R}^{C \times d}, b \in \mathbb{R}^C$, followed by **the softmax function** σ.

Single-head optimization (SHO) of KD. Our method builds on *logit distillation* in semi-supervised settings [15, 22] that combines supervised loss $\mathcal{L}_{\text{CE}}$ on the labeled dataset $\mathcal{D}^{(l)}$ with KD loss $\mathcal{L}_{\text{KD}}$ on both labeled and unlabeled datasets $\mathcal{D}^{(l)} \cup \mathcal{D}^{(u)}$, *i.e.*, $\lambda\mathcal{L}_{\text{CE}} + (1-\lambda)\mathcal{L}_{\text{KD}}$, where $\lambda \in [0, 1]$ is a loss balancing hyperparameter. Specifically, the supervised loss $\mathcal{L}_{\text{CE}}$ and the KD loss $\mathcal{L}_{\text{KD}}$ are defined as follows:

$$\mathcal{L}_{\text{CE}} = \frac{1}{N} \sum_n \ell\left(\sigma(h(z_n^{(l)})), y_n\right), \tag{1}$$

$$\mathcal{L}_{\text{KD}} = \frac{1}{N} \sum_n D_{\text{KL}}\left[p_n^{(l)} \| \sigma(h(z_n^{(l)}))\right] + \frac{1}{M} \sum_m D_{\text{KL}}\left[p_m^{(u)} \| \sigma(h(z_m^{(u)}))\right]. \tag{2}$$

where ℓ denotes D_{KL} represent the cross-entropy and Kullback-Leibler divergence, respectively. $z_n^{(l)} = g(x_n^{(l)})$ and $z_m^{(u)} = g(x_m^{(u)}) \in \mathbb{R}^d$ are feature representations obtained by the feature extractor g. $p_n^{(l)}$ and $p_m^{(u)}$ are the categorical probability vectors of labeled $x_n^{(l)}$ and unlabeled data $x_m^{(u)}$, respectively, obtained by teacher VLM encoders $f_{\mathcal{X}}$ and $f_{\mathcal{T}}$. Another well-studied single-head KD baseline is *feature distillation*, which leverages mean squared error (MSE) loss to directly align feature representations extracted by the student encoder g and the teacher image encoder $f_{\mathcal{X}}$. We defer the details of feature distillation for VLMs to CLIP-KD [16].

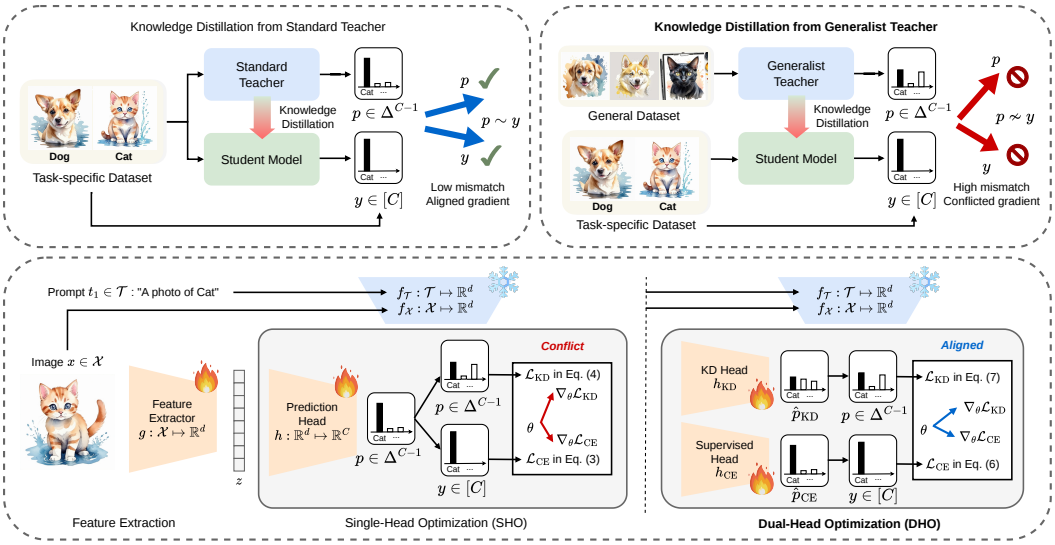

Figure 2: **Conceptual illustration. (Top)** Unlike conventional KD where teacher and student share the dataset, a *generalist* VLM teacher (*e.g.*, CLIP) produces predictions misaligned with task labels, yielding *conflicting* CE/KD gradients (Appendix F.1). **(Bottom)** Single-Head Optimization (SHO) routes both losses through one classifier and inherits this conflict (Fig. 3); `Dual-Head Optimization` (`DHO`) decouples the heads and **mitigates it**.

2.2 `Dual-Head Optimization` (`DHO`)

Gradient conflicts in SHO. Logit distillation in SHO, as described in §2.1, provides a simple approach for transferring knowledge from VLMs to task-specific models. However, we find that its performance gain is *sub-optimal*, which we attribute to **gradient conflicts** between the supervised and KD loss signals which hinder effective feature learning [23, 24, 25]. As illustrated in — and — of Fig. 3, both **the classifier weight W of classifier h and the parameter vector θ of feature extractor g** suffer from gradient conflicts: the cosine similarity between their respective gradients turns negative, *i.e.*, $\texttt{CosSim}(\nabla_\theta \mathcal{L}_{\text{CE}}, \nabla_\theta \mathcal{L}_{\text{KD}}) < 0$ and $\texttt{CosSim}(\nabla_W \mathcal{L}_{\text{CE}}, \nabla_W \mathcal{L}_{\text{KD}}) < 0$, indicating misaligned optimization directions.

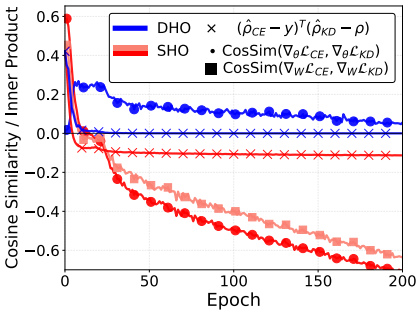

Figure 3: **The average cosine similarity and inner product** over 10 datasets.

To understand this, we first analyze the gradient with respect to (w.r.t) the weight W of a linear head h. The gradients w.r.t the classifier weight W are $\nabla_W \mathcal{L}_{\text{CE}} = (\hat{p} - y)z^\top$ and $\nabla_W \mathcal{L}_{\text{KD}} = (\hat{p} - p)z^\top$, respectively. Their cosine similarity is proportional to $(\hat{p} - y)^\top(\hat{p} - p) \cdot \|z\|^2$:

$$\texttt{CosSim}(\nabla_W \mathcal{L}_{\text{CE}}, \nabla_W \mathcal{L}_{\text{KD}}) \propto (\hat{p} - y)^\top(\hat{p} - p) \cdot \|z\|^2, \tag{3}$$

which misaligns when $(\hat{p} - y)^\top(\hat{p} - p) < 0$ due to prediction mismatch, falling below zero during training (-✕- in Fig. 3), leading to gradient conflicts (— in Fig. 3). In Appendix C.1, we prove that this inner product equals $\|\hat{p} - m\|^2 - r^2$ where $m = \frac{y+p}{2}$ and $r = \frac{\|y-p\|}{2}$, so **gradient conflict is guaranteed whenever $\hat{p}$ lies inside the ball $\mathcal{B}(m, r)$** with the segment $[y, p]$ as its diameter (Proposition 1). In particular, if $\hat{p} = (1-t)\, y + t\, p$ for $t \in (0, 1)$, then $(\hat{p}-y)^\top(\hat{p}-p) = -t(1-t)\|p-y\|^2 < 0$ (Corollary 1).

We further show that **gradient conflicts propagate to the feature extractor g.** Let the gradients w.r.t to the feature representation z be $\nabla_z \mathcal{L}_{\text{CE}} = W^\top(\hat{p} - y)$ and $\nabla_z \mathcal{L}_{\text{KD}} = W^\top(\hat{p} - p)$. Applying the chain rule, the cosine similarity for the feature extractor parameters θ becomes:

$$\texttt{CosSim}(\nabla_\theta \mathcal{L}_{\text{CE}}, \nabla_\theta \mathcal{L}_{\text{KD}}) \propto (\hat{p} - y)^\top W \left(\frac{\partial z}{\partial \theta}\right)^\top \frac{\partial z}{\partial \theta} W^\top (\hat{p} - p) \tag{4}$$

Since $A = W \left(\frac{\partial z}{\partial \theta}\right)^\top \frac{\partial z}{\partial \theta} W^\top \succeq 0$ is PSD, the gradient conflict is also guaranteed under the interpolation condition (Proposition 2), which is empirically confirmed (— in Fig. 3).

Dual-head architecture. To mitigate this issue, we propose Dual Head Optimization (DHO) to decouple $\mathcal{L}_{\text{CE}}$ and $\mathcal{L}_{\text{KD}}$ via *two independent prediction heads*: $h_{\text{CE}}(z) = W_{\text{CE}}z + b_{\text{CE}}$ and $h_{\text{KD}}(z) = W_{\text{KD}}z + b_{\text{KD}}$, with $W_{\text{CE}}, W_{\text{KD}} \in \mathbb{R}^{C\times d}$ and $b_{\text{CE}}, b_{\text{KD}} \in \mathbb{R}^{C}$. The corresponding losses are:

$$\mathcal{L}_{\text{CE}} = \frac{1}{N}\sum_{n} \ell\left(\sigma(h_{\text{CE}}(z_n^{(l)})), y_n\right), \tag{5}$$

$$\mathcal{L}_{\text{KD}} = \frac{1}{N}\sum_{n} D_{\text{KL}}\left[p_n^{(l)} \| \sigma(h_{\text{KD}}(z_n^{(l)}))\right] + \frac{1}{M}\sum_{m} D_{\text{KL}}\left[p_m^{(u)} \| \sigma(h_{\text{KD}}(z_m^{(u)}))\right], \tag{6}$$

where the final loss combines both objectives as $\lambda\mathcal{L}_{\text{CE}} + (1-\lambda)\mathcal{L}_{\text{KD}}$.

Mitigation of gradient conflict in DHO. In DHO, gradient conflicts in **the classification head naturally disappear by decoupling the optimization** of W_{CE} and W_{KD}. This separation enables each head to learn distinct signals without interference, reducing prediction mismatch (-×- of Fig. 3). Let $\hat{p}_{\text{CE}}=\sigma(h_{\text{CE}}(z))$ and $\hat{p}_{\text{KD}}=\sigma(h_{\text{KD}}(z))$, then $\nabla_z\mathcal{L}_{\text{CE}}=W_{\text{CE}}^\top(\hat{p}_{\text{CE}}-y)$ and $\nabla_z\mathcal{L}_{\text{KD}}=W_{\text{KD}}^\top(\hat{p}_{\text{KD}}-p)$. The cosine similarity between gradients w.r.t θ in DHO is defined as:

$$\texttt{CosSim}(\nabla_\theta\mathcal{L}_{\text{CE}}, \nabla_\theta\mathcal{L}_{\text{KD}}) \propto (\hat{p}_{\text{CE}}-y)^\top W_{\text{CE}}\left(\frac{\partial z}{\partial\theta}\right)^\top \frac{\partial z}{\partial\theta}W_{\text{KD}}^\top(\hat{p}_{\text{KD}}-p), \tag{7}$$

In Proposition 3, we prove that the cross-head matrix $W_{\text{CE}}\left(\frac{\partial z}{\partial\theta}\right)^\top \frac{\partial z}{\partial\theta}W_{\text{KD}}^\top$ is **not PSD** in general, breaking the structural guarantee of negative alignment that causes conflicts in SHO. Empirically, DHO maintains positive gradient alignment throughout training (*i.e.*, Eq. 7 > 0; -●- of Fig. 3), enabling conflict-free representation learning and **better feature representations** compared to SHO (Tab. 5). We defer an algorithm that describes the full training procedure of DHO to Alg. 1.

2.3 Dual-head Interpolation

After training, we find that using only one of the two heads at inference is *suboptimal* (see Fig. 6). Motivated by the mixture-of-experts paradigm [27], we adopt a *simple yet effective* inference rule for DHO that linearly interpolates the output probability vectors of two heads:

$$\hat{p}_{\texttt{DHO}} = \alpha\,\sigma\big(h_{\text{CE}}(z)\big) + (1-\alpha)\,\sigma\big(h_{\text{KD}}(z)/\beta\big), \tag{8}$$

where $\alpha \in [0,1]$ balances the supervised and KD heads, $\beta > 0$ is a temperature that softens the KD logits, and the final prediction is $\arg\max_{c\in[C]} \hat{p}_{\texttt{DHO}}$. In practice, $(\alpha,\beta)=(0.5, 1)$ already outperforms SHO without tuning (Tab. 17); α, β can also be tuned on a validation split for further gains. See Alg. 2 for the full inference procedure.

Effect of α and β. We now demonstrate the effect of tuning α and β using a validation set. Under mild assumptions, DHO ε-approximates SHO in ℓ_1 by setting $\alpha = \lambda$ and $\beta = 1$:

Assumption 1 (ε-convergence)**.** *Assume that, after sufficient training, both heads converge to their respective target distributions with ℓ_1-bounded error:*

$$\sup_x \|\sigma(h_{\text{CE}}(z)) - y\|_1 \le \varepsilon, \quad \sup_x \|\sigma(h_{\text{KD}}(z)) - p\|_1 \le \varepsilon, \quad \text{where} \quad \varepsilon \in \mathbb{R}_{>0}. \tag{9}$$

Theorem 1 (Inference equivalence)**.** *Under Assumption 1, by setting $\alpha = \lambda$ and $\beta = 1$, then $\|\hat{p}_{\texttt{DHO}} - \hat{p}_{\text{SHO}}\|_1 \le \varepsilon$, where $\hat{p}_{\text{SHO}}$ is the output of SHO optimally trained with λ.*

Details, proofs, and a discussion of Assumption 1 (Remark 3) are deferred to Appendix C. Theorem 1 implies that $\hat{p}_{\text{SHO}}$ trained with any λ can be approximated by the dual-head interpolation in Eq. 8, by setting $\alpha = \lambda$ and $\beta = 1$. Here, λ is a **training** hyperparameter of SHO, while α and β are **inference** hyperparameters of DHO, allowing it to **emulate SHO hyperparameter tuning** without retraining.

Language-aware initialization for VLM students In the case of VLM-to-VLM distillation, we leverage the text encoder $f_{\mathcal{T}}$ of teachers when initializing the dual heads h_{CE} and h_{KD}. Following prior work [3], we initialize the weights as $W_{\text{CE}}, W_{\text{KD}} \leftarrow [f_{\mathcal{T}}(t_1), \ldots, f_{\mathcal{T}}(t_C)]^\top \in \mathbb{R}^{C\times d}$. We further align the prediction logic of KD head h_{KD}, with the cosine similarity-based approach of the teacher VLMs as follows:

$$h_{\text{KD}} = \frac{1}{\zeta}[\texttt{CosSim}(g(x), w_1), \ldots, \texttt{CosSim}(g(x), w_C)]^\top \in \mathbb{R}^C, \tag{10}$$

where $w_c \in \mathbb{R}^d$ denotes the c-th row of W_{KD}.

Table 1: Results on **ImageNet** under few-shot semi-supervision using **ResNet-18** and **ResNet-50**. `DHO` **consistently outperforms all baselines and even the teacher** with ResNet-50 (*e.g.*, +0.7/1.9/2.2/3.4/4.4% with a zero-shot teacher; +1.3/1.5/1.4/1.7/1.4% with a few-shot teacher). See Appendix B for comparison with prior dual-head methods and Appendix D.2 for student initialization rationale (orthogonal to `DHO`).

	ResNet-18 trained from scratch						Self-supervised ResNet-50					
Method	0-shot	1-shot	2-shot	4-shot	8-shot	16-shot	0-shot	1-shot	2-shot	4-shot	8-shot	16-shot
Single-head KD methods												
KD	51.0	-	-	-	-	-	61.0	-	-	-	-	-
CE	-	0.7	1.1	1.8	3.4	8.2	-	11.4	17.3	26.4	36.7	47.0
CE+KD (feature)	-	17.1	23.5	28.0	32.2	33.8	-	23.0	32.3	41.3	48.2	54.3
CE+KD (logit)	-	50.5	50.6	50.6	51.0	51.2	-	60.4	60.8	61.2	61.6	62.3
Dual-head KD methods												
SSKD [28]	-	42.5	46.2	48.0	50.6	52.0	-	55.2	58.1	60.0	62.3	64.0
DHKD [29]	-	19.7	23.5	23.4	23.7	26.8	-	25.6	34.8	42.7	49.2	55.2
`DHO` (Ours)	-	51.8	52.4	52.6	53.3	54.5	-	61.0	62.1	62.5	63.7	64.7
`DHO-F` (Ours)	-	**53.7**	**54.2**	**54.8**	**56.2**	**57.7**	-	**62.3**	**63.1**	**63.9**	**65.5**	**66.8**
Teacher Models (Resnet-50)												
CLIP	60.3	-	-	-	-	-	60.3	-	-	-	-	-
Tip-Adapter-F	-	61.0	61.6	62.5	63.8	65.4	-	61.0	61.6	62.5	63.8	65.4

3 Experiments

3.1 Experimental Setups

Datasets. For in-distribution evaluation, we use ImageNet [26] and 10 fine-grained datasets [30, 31, 32, 33, 34, 35, 36, 37, 38, 39]. To assess out-of-distribution (OOD) generalization, we use four ImageNet variants [40, 41, 42, 43]. See Appendix E for details.

Baselines. We compare `DHO` with conventional single-head KD baselines; **CE**: training only on labeled dataset $\mathcal{D}^{(l)}$ with cross entropy loss (Eq. 1), **KD (logit)**: on unlabeled dataset $\mathcal{D}^{(u)}$ with logit distillation (Eq. 2), and **KD (feature)**: on $\mathcal{D}^{(u)}$ with feature distillation [16]. We train on both $\mathcal{D}^{(l)}$ and $\mathcal{D}^{(u)}$ with **CE+KD (logit)** or **CE+KD (feature)**: combining CE with each KD variant using balancing hyperparameter λ. We also consider dual-head KD approaches **SSKD** [28] and **DHKD** [29], though for different purposes as detailed in Appendix B.

For in-distribution evaluation on ImageNet with low-shot settings, we compare against **self and semi-supervised learning** [22, 11, 44, 12, 13], **CLIP-based-training** [3, 14], **co-training** [45], **KD** [22], and **zero-shot VLMs** [46, 47].

For OOD evaluation, we compare against VLM adaptation methods, including **VPT** [48], **CoOp** [7], **PromptSRC** [49], and **CasPL** [21].

Implementation details. We evaluate across few-shot (1/2/4/8/16-shot) and low-shot (1%/10%) settings, treating remaining data as unlabeled. We adopt Tip-Adapter-F [5] for few-shot teachers, denoting this variant as `DHO-F`. On ImageNet, to avoid label leakage from supervised pretraining, students are ResNet-18 trained from scratch or DINO-pretrained ResNet-50 [50], paired with a CLIP ResNet-50 zero-shot teacher; on the 10 fine-grained datasets the leakage concern does not apply, so we additionally include MobileNetV2 [51] (see Appendix D.2). All students are trained for 20 epochs with AdamW (learning rate 1×10^{-3}, weight decay 1×10^{-2}, batch size 512 split evenly between labeled and unlabeled streams). For a fair comparison, we use the same hyperparameters for `DHO` and SHO baselines (except α, β); for other methods, we report the published results. When validation data is unavailable (*e.g.*, ImageNet), we fix $\beta = 0.5$ across all settings. We heuristically set $\alpha = 0.4$ for zero-shot teachers and $\alpha = 0.2$ for few-shot teachers, reflecting the latter's higher reliability. In low-shot settings, we use $\alpha = 0.5$ due to increased label availability. See Appendix D for details.

3.2 Main Results

Effectiveness of `DHO` compared to conventional KD baselines. Tab. 1 presents results on ImageNet under few-shot semi-supervision using ResNet-18/ResNet-50 student models and ResNet-50 VLM teachers. `DHO` **consistently outperforms** all baselines across all settings. Other dual-head methods (*i.e.*, SSKD/DHKD), not designed for few-shot semi-supervision, even underperform CE+KD on logits. Notably, with ResNet-50, `DHO` **outperforms the teacher in every few-shot setting** (*e.g.*, +0.7/1.8/2.2/3.4/4.4% or +1.3/1.5/1.4/1.7/1.4% with zero-shot or few-shot teachers).

Next, we evaluate `DHO` on 10 additional datasets using the ResNet-18 student and ResNet-50 VLM teachers.

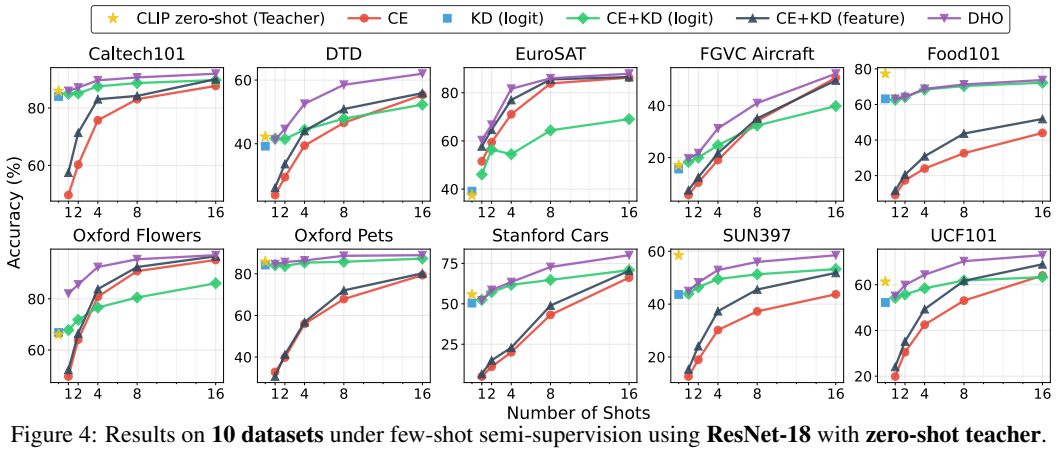

Figure 4: Results on **10 datasets** under few-shot semi-supervision using **ResNet-18** with **zero-shot teacher**.

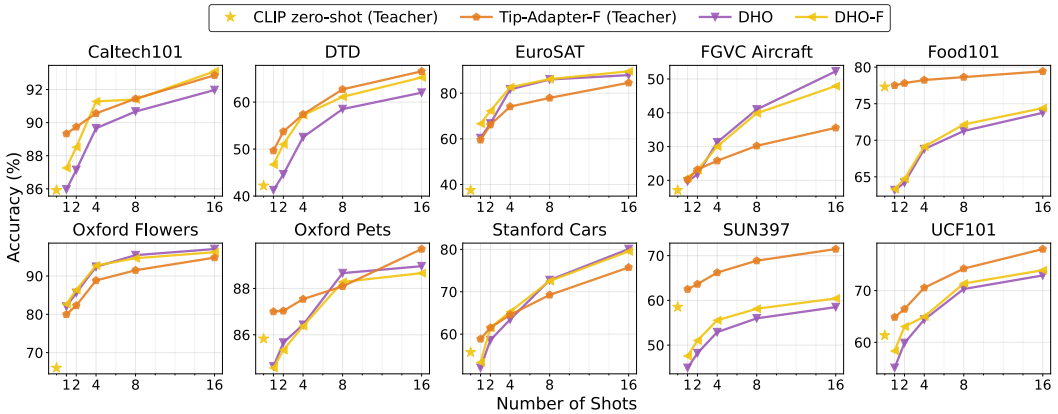

Figure 5: Results on **10 datasets** using **ResNet-18** with either zero- or **few-shot teacher**.

On 10 additional datasets, DHO also **consistently outperforms all baselines** (Fig. 4); the few-shot teacher is more effective than the zero-shot one (Fig. 5), and **the ResNet-18 student trained with DHO beats the ResNet-50 teacher** in most cases.

DHO achieves SoTA performance on ImageNet under low-shot settings. On ImageNet under 1%/10% labeled data, DHO with ViT-L/14 **surpasses the previous SoTA by +3.0%/+0.1% using fewer parameters (76M/767M)** (Tab. 2); all baselines are taken from published papers except DHO and CE+KD (logit). Notably, CE+KD (logit) outperforms semi-supervised methods of the same size, showing they are **suboptimal compared to methods leveraging VLM representations**.

DHO achieves SoTA performance on ImageNet OOD benchmarks. **DHO consistently improves** VLM adaptation methods (Tab. 4) and outperforms CasPL across in-distribution (Val) and OOD (V2/Sketch/R/A) benchmarks, **establishing new SoTA OOD results** in semi-supervised setting—joint labeled-supervision + teacher-distillation is more effective than CasPL's sequential approach, consistent with DHO's gradient-conflict resolution. See Appendix F.10 for fully trained models.

Table 2: Results on **ImageNet under low-shot settings**. For CT/MCT methods, numbers in parentheses indicate the number of architectures for co-training.

Method	Architecture	Params (M)	1% (%)	10% (%)
Self and Semi-supervised Learning				
MSN	ViT-B/4	86	75.7	80.2
Semi-ViT	ViT-L/14	307	77.3	83.3
Semi-ViT	ViT-H/14	632	80.0	84.3
CLIP-based Training				
CLIP	ViT-B/16	86	74.3	80.4
REACT	ViT-B/16	86	76.1	80.8
REACT (Gated-Image)	ViT-B/16	129	77.4	81.8
CLIP	ViT-L/14	304	80.5	84.7
REACT	ViT-L/14	304	81.6	85.1
REACT (Gated-Image)	ViT-L/14	380	81.6	85.0
Co-training based Methods				
CT	Multi-arch (2)	608	80.1	85.1
MCT	Multi-arch (2)	608	80.7	85.2
CT	Multi-arch (4)	1071	80.0	84.8
MCT	Multi-arch (4)	1071	80.5	85.8
Knowledge Distillation				
SimCLR v2 distill	ResNet-50 (2Œ+SK)	140	75.9	80.2
SimCLR v2 self-distill	ResNet-154 (3Œ+SK)	795	76.6	80.9
CE + KD (logit)	ViT-B/16	86	79.8	80.4
CE + KD (logit)	ViT-L/14	304	83.1	83.6
DHO	ViT-B/16	86	81.6	82.8
DHO	ViT-L/14	304	**84.6**	**85.9**
Zero-shot VLMs				
SigLIP	ViT-SO400M/14	400	83.1	-
DFN	ViT-H/14	632	84.4	-

Table 4: Results on **ImageNet and its distribution-shifted variants**. †: backbone frozen during training. `DHO` **consistently improves** VLM adaptation methods and **achieves SoTA results** on OOD benchmarks.

Method	Labeled Data	Teacher Model	Unlabeled Data	Val	V2	Sketch	R	A	Avg
CLIP Zero-shot									
CLIP	-	-	-	66.7	60.8	46.2	74.0	47.8	59.1
Linear Evaluation									
Linear Evaluation	1%	-	-	72.8	64.2	47.0	74.6	48.6	61.4
`DHO`†	1%	✓	-	73.4	65.3	48.3	75.8	49.4	62.4
`DHO`†	1%	✓	✓	**74.6**	**66.4**	**49.2**	**76.1**	**49.8**	**63.2**
Visual Prompt Tuning									
VPT	1.25%	-	-	73.6	64.6	47.7	75.2	48.7	62.0
VPT+`DHO`	1.25%	✓	-	73.6	65.3	48.7	75.9	49.1	62.5
VPT+`DHO`	1.25%	✓	✓	**75.1**	**66.8**	**50.0**	**76.9**	**50.5**	**63.9**
VLM Text-encoder Prompt Tuning									
CoOp	1.25%	-	-	71.5	64.2	48.0	75.2	49.7	61.7
CoOp+CasPL	1.25%	✓	✓	71.9	64.3	48.3	76.0	-	-
CoOp+`DHO`	1.25%	✓	-	72.8	65.5	49.3	76.4	49.5	62.7
CoOp+`DHO`	1.25%	✓	✓	**73.4**	**66.2**	**49.5**	**77.0**	**50.5**	**63.3**
VLM Multimodal Prompt Tuning									
PromptSRC	1.25%	-	-	71.3	64.4	49.6	77.8	50.9	62.8
PromptSRC+CasPL	1.25%	✓	✓	72.8	65.7	49.7	77.9	-	-
PromptSRC+`DHO`	1.25%	✓	-	73.0	65.3	49.5	77.8	**51.3**	63.4
PromptSRC+`DHO`	1.25%	✓	✓	**73.6**	**66.1**	**49.8**	**78.1**	51.0	**63.7**

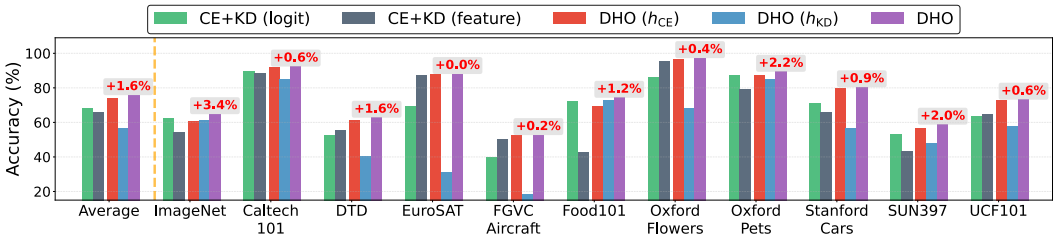

Figure 6: Results of ablation studies on **dual-heads interpolation strategy** in Eq. 8 of `DHO`.

Additional results. `DHO` also outperforms **PCGrad** [52] (Appendix F.3); see Appendix F for **MobileNetV2** and **category-aware KD methods** [53, 54].

3.3 Analysis

Minimal computational overhead of `DHO`. As shown in Tab. 3, `DHO` introduces **negligible computational overhead with minimal parameter increase** on the ImageNet with 1000 classes, which can be further reduced down for datasets with fewer classes ($C < 1000$). We further provide inference computational overhead for other models in Tab. 9. The gain therefore is not from added capacity nor pseudo-labeling: with a specialist teacher (no gradient conflict), `DHO` matches SHO (Appendix F.1).

Table 3: **Inference overhead** using RTX 4090.

Model	Params (M)	FLOPs (G)	Throughput (im/s)
ResNet-18	11.69	1.83	3525.7
+ `DHO`	12.20 (+4.4%)	1.83 (+0.0%)	3518.6 (-0.20%)
ResNet-50	25.56	4.14	1018.4
+ `DHO`	27.61 (+8.0%)	4.15 (+0.2%)	1016.4 (-0.20%)

Effectiveness of dual-head interpolation. We evaluate the effectiveness of dual-head interpolation (Eq. 8) by comparing `DHO` with CE+KD (logit), CE+KD (feature), and ablations `DHO` (h_{CE}) and `DHO` (h_{KD}), which predict using only one head (*i.e.*, $\alpha = 1$ or 0). As shown in Fig. 6, `DHO` outperforms `DHO` (h_{CE}) by an average of 1.6% across 11 datasets, with a maximum gain of +3.4% on ImageNet and no degradation on any dataset. Since α and β are inference-time hyperparameters, dual-head interpolation introduces minimal overhead while consistently **improving or maintaining performance**. We also investigate the effectiveness of our adaptive weighting strategy in Appendix F.5. Fig. 7 illustrates three challenging examples: CE head (h_{CE}) is correct in the first, KD head (h_{KD}) in the second, and both fail in the third

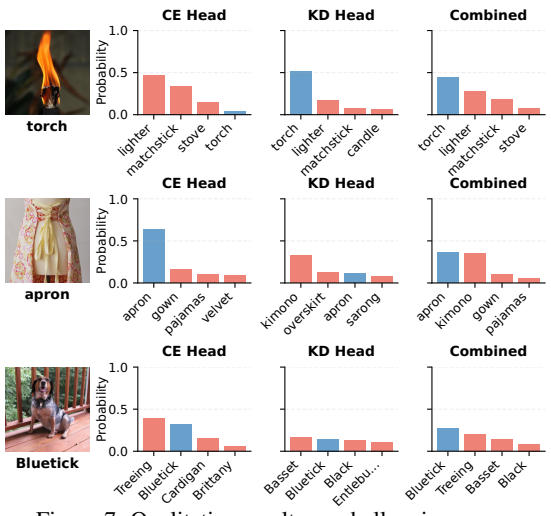

Figure 7: Qualitative results on challenging cases.

case, yet the proposed combined prediction is correct—demonstrating the ability of DHO to resolve individual head failures. See Appendix G.4 for additional analysis on theses challenging examples.

Enhanced feature representation of DHO. To validate our claim that mitigating gradient conflicts improves feature representations, we evaluate features using the standard *linear evaluation* protocol [22]. We train CE+KD (feature), CE+KD (logit), and DHO under the 16-shot semi-supervised setting on ImageNet, freeze the feature extractor g, and train a new prediction linear head h_{LE} on the top of g using fully labeled data. As shown in Tab. 5, DHO achieves higher Top-1 and Top-5 accuracy than other methods (*e.g.*, +0.9% and 0.5%, respectively). To further assess feature quality, we visualize embeddings z using t-SNE [55] in Fig. 8. Compared to the CE+KD (logit) baseline, DHO produces more compact and class-separated feature clusters. These results support our claim that **DHO enhances feature representations** by mitigating gradient conflicts; this improvement leads to better performance of DHO compared to SHO, as discussed through Tab. 1, Figs. 4, 5, 10 and 11.

Table 5: Linear evaluation results.

Method	Top-1 (%)	Top-5 (%)
CE+KD (feature) [16]	62.3	85.0
CE+KD (logit) [22]	66.2	88.8
DHO	**67.1**	**89.3**

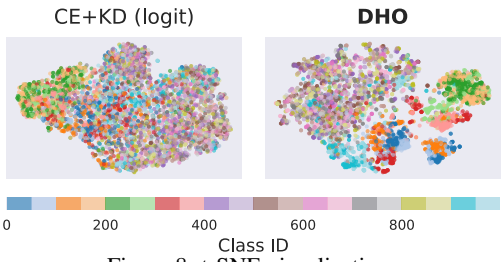

Figure 8: t-SNE visualization.

Effectiveness of Init. and Align in §2.3. To assess the effect of the proposed language-aware initialization and KD-head alignment in §2.3, we run ablations that apply language-aware initialization (**Init.**) to either h_{CE} or h_{KD}, and optionally enable KD-head alignment (**Align.**). We evaluate on ImageNet with 1% labels using a ViT-B/16 student and ViT-L/14 teacher for computational efficiency. As shown in Tab. 6, applying Init. to either head independently improves accuracy (*e.g.*, +0.2/+0.3%), and adding Align. yields additional gains (*e.g.*, +0.4%).

Table 6: Results on **language-aware initialization** and **KD-head alignment** for VLM students on ImageNet with 1% labeled data.

Init (h_{CE}/h_{KD}).	Align.	Accuracy (%)
✗ / ✗	✗	78.3
✓ / ✗	✗	78.5 (+0.2)
✓ / ✓	✗	78.6 (+0.3)
✓ / ✓	✓	**78.7** (+0.4)

Effect of α and β. In Theorem 1, we show that under mild assumptions (Assumption 1), DHO ϵ-approximates the SHO baseline, namely CE+KD (logit), in the ℓ_1 norm by setting $\alpha{=}\lambda$ and $\beta{=}1$. This enables DHO to emulate SHO hyperparameter tuning at inference time without retraining. To study how α and β affect performance, we visualize a grid search on ImageNet with ResNet-50 under the 16-shot setting (Fig. 9). With balanced heads ($\alpha{\approx}0.5$), performance is stable for $\beta \in [0.1, 1]$ and degrades only at extreme α values regardless of β. The accuracy peaks at 65.5% near $(\alpha, \beta){\approx}(0.5, 0.3)$, close to our heuristic setting (64.7%); even with fixed $(\alpha, \beta){=}(0.5, 1)$, DHO exceeds SHO by +1.6 to +9.5% across 1–16 shots (Tab. 17).

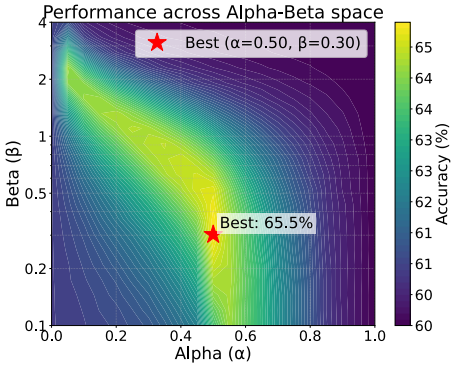

Figure 9: **Grid search results for α and β.**

4 Conclusion, Limitation, and Future Work

We identify *gradient conflicts* between teacher and labeled-data signals as a fundamental obstacle to transferring VLM zero-/few-shot capabilities to task-specific models under limited supervision. We propose DHO (Dual-Head Optimization), a *plug-and-play* framework that decouples the two signals into separate classification heads, **mitigating gradient conflicts** in both heads and the shared feature extractor while **enabling post-training hyperparameter tuning** via linear interpolation of head outputs. Across 15 datasets, DHO **outperforms conventional KD methods** and achieves **state-of-the-art** on ImageNet semi-supervised learning and on OOD benchmarks; see Appendix A for limitations and future work.

References

[1] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[2] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[3] Chunyuan Li, Haotian Liu, Liunian Li, Pengchuan Zhang, Jyoti Aneja, Jianwei Yang, Ping Jin, Houdong Hu, Zicheng Liu, Yong Jae Lee, et al. Elevater: A benchmark and toolkit for evaluating language-augmented visual models. *Advances in Neural Information Processing Systems*, 35:9287–9301, 2022.

[4] Yunshi Huang, Fereshteh Shakeri, Jose Dolz, Malik Boudiaf, Houda Bahig, and Ismail Ben Ayed. Lp++: A surprisingly strong linear probe for few-shot clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23773–23782, 2024.

[5] Renrui Zhang, Rongyao Fang, Wei Zhang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021.

[6] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2):581–595, 2024.

[7] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

[8] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16816–16825, 2022.

[9] Marc Lafon, Elias Ramzi, Clément Rambour, Nicolas Audebert, and Nicolas Thome. Gallop: Learning global and local prompts for vision-language models. In *European Conference on Computer Vision*, pages 264–282. Springer, 2025.

[10] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020.

[11] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Armand Joulin, Nicolas Ballas, and Michael Rabbat. Semi-supervised learning of visual features by non-parametrically predicting view assignments with support samples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8443–8452, 2021.

[12] Zhaowei Cai, Avinash Ravichandran, Paolo Favaro, Manchen Wang, Davide Modolo, Rahul Bhotika, Zhuowen Tu, and Stefano Soatto. Semi-supervised vision transformers at scale. *Advances in Neural Information Processing Systems*, 35:25697–25710, 2022.

[13] Mingkai Zheng, Shan You, Lang Huang, Chen Luo, Fei Wang, Chen Qian, and Chang Xu. Simmatchv2: Semi-supervised learning with graph consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16432–16442, 2023.

[14] Haotian Liu, Kilho Son, Jianwei Yang, Ce Liu, Jianfeng Gao, Yong Jae Lee, and Chunyuan Li. Learning customized visual models with retrieval-augmented knowledge. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15148–15158, 2023.

[15] Geoffrey Hinton. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[16] Chuanguang Yang, Zhulin An, Libo Huang, Junyu Bi, Xinqiang Yu, Han Yang, Boyu Diao, and Yongjun Xu. Clip-kd: An empirical study of clip model distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15952–15962, 2024.

[17] Pavan Kumar Anasosalu Vasu, Hadi Pouransari, Fartash Faghri, Raviteja Vemulapalli, and Oncel Tuzel. Mobileclip: Fast image-text models through multi-modal reinforced training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15963–15974, 2024.

[18] Vishaal Udandarao, Nikhil Parthasarathy, Muhammad Ferjad Naeem, Talfan Evans, Samuel Albanie, Federico Tombari, Yongqin Xian, Alessio Tonioni, and Olivier J Hénaff. Active data curation effectively distills large-scale multimodal models. *arXiv preprint arXiv:2411.18674*, 2024.

[19] Kaicheng Yang, Tiancheng Gu, Xiang An, Haiqiang Jiang, Xiangzi Dai, Ziyong Feng, Weidong Cai, and Jiankang Deng. Clip-cid: Efficient clip distillation via cluster-instance discrimination. *arXiv preprint arXiv:2408.09441*, 2024.

[20] Raviteja Vemulapalli, Hadi Pouransari, Fartash Faghri, Sachin Mehta, Mehrdad Farajtabar, Mohammad Rastegari, and Oncel Tuzel. Knowledge transfer from vision foundation models for efficient training of small task-specific models. In *International Conference on Machine Learning (ICML)*, 2024.

[21] Ge Wu, Xin Zhang, Zheng Li, Zhaowei Chen, Jiajun Liang, Jian Yang, and Xiang Li. Cascade prompt learning for vision-language model adaptation. In *European Conference on Computer Vision*, pages 304–321. Springer, 2024.

[22] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.

[23] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33:5824–5836, 2020.

[24] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34: 18878–18890, 2021.

[25] Jie Chen and Meng Joo Er. Mitigating gradient conflicts via expert squads in multi-task learning. *Neurocomputing*, 614:128832, 2025.

[26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.

[27] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[28] Lingxiao He, Wu Liu, Jian Liang, Kecheng Zheng, Xingyu Liao, Peng Cheng, and Tao Mei. Semi-supervised domain generalizable person re-identification. *arXiv preprint arXiv:2108.05045*, 2021.

[29] Penghui Yang, Chen-Chen Zong, Sheng-Jun Huang, Lei Feng, and Bo An. Dual-head knowledge distillation: Enhancing logits utilization with an auxiliary head. *arXiv preprint arXiv:2411.08937*, 2024.

[30] 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 *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[31] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

[32] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

[33] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008.

[34] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part VI 13*, pages 446–461. Springer, 2014.

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

[36] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[37] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

[38] 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*, 12(7):2217–2226, 2019.

[39] K Soomro. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[40] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pages 5389–5400. PMLR, 2019.

[41] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

[42] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021.

[43] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15262–15271, 2021.

[44] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Mike Rabbat, and Nicolas Ballas. Masked siamese networks for label-efficient learning. In *European Conference on Computer Vision*, pages 456–473. Springer, 2022.

[45] Jay C Rothenberger and Dimitrios I Diochnos. Meta co-training: Two views are better than one. *arXiv preprint arXiv:2311.18083*, 2023.

[46] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023.

[47] Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks. *arXiv preprint arXiv:2309.17425*, 2023.

[48] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022.

[49] Muhammad Uzair Khattak, Syed Talal Wasim, Muzammal Naseer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Self-regulating prompts: Foundational model adaptation without forgetting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15190–15200, 2023.

[50] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[51] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[52] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33:5824–5836, 2020.

[53] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 11953–11962, 2022.

[54] Jiaming Lv, Haoyuan Yang, and Peihua Li. Wasserstein distance rivals kullback-leibler divergence for knowledge distillation. *Advances in Neural Information Processing Systems*, 37: 65445–65475, 2024.

[55] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[56] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

[57] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

[58] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[59] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[60] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[61] Xiao Wang, Guangyao Chen, Guangwu Qian, Pengcheng Gao, Xiao-Yong Wei, Yaowei Wang, Yonghong Tian, and Wen Gao. Large-scale multi-modal pre-trained models: A comprehensive survey. *Machine Intelligence Research*, 20(4):447–482, 2023.

[62] Fei-Long Chen, Du-Zhen Zhang, Ming-Lun Han, Xiu-Yi Chen, Jing Shi, Shuang Xu, and Bo Xu. Vlp: A survey on vision-language pre-training. *Machine Intelligence Research*, 20 (1):38–56, 2023.

[63] Zhe Gan, Linjie Li, Chunyuan Li, Lijuan Wang, Zicheng Liu, Jianfeng Gao, et al. Vision-language pre-training: Basics, recent advances, and future trends. *Foundations and Trends® in Computer Graphics and Vision*, 14(3–4):163–352, 2022.

[64] Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[65] Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, et al. Maskclip: Masked self-distillation advances contrastive language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10995–11005, 2023.

[66] Yuting Gao, Jinfeng Liu, Zihan Xu, Jun Zhang, Ke Li, Rongrong Ji, and Chunhua Shen. Pyramidclip: Hierarchical feature alignment for vision-language model pretraining. *Advances in neural information processing systems*, 35:35959–35970, 2022.

[67] 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.

[68] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23390–23400, 2023.

[69] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling vision transformers to 22 billion parameters. In *International Conference on Machine Learning*, pages 7480–7512. PMLR, 2023.

[70] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[71] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2818–2829, 2023.

[72] Quan Sun, Jinsheng Wang, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, and Xinlong Wang. Eva-clip-18b: Scaling clip to 18 billion parameters. *arXiv preprint arXiv:2402.04252*, 2024.

[73] Qingpei Guo, Furong Xu, Hanxiao Zhang, Wang Ren, Ziping Ma, Lin Ju, Jian Wang, Jingdong Chen, and Ming Yang. M^2-encoder: Advancing Bilingual Image-Text Understanding by Large-scale Efficient Pretraining. *arXiv preprint arXiv:2401.15896*, 2024.

[74] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18123–18133, 2022.

[75] Hieu Pham, Zihang Dai, Golnaz Ghiasi, Kenji Kawaguchi, Hanxiao Liu, Adams Wei Yu, Jiahui Yu, Yi-Ting Chen, Minh-Thang Luong, Yonghui Wu, et al. Combined scaling for zero-shot transfer learning. *Neurocomputing*, 555:126658, 2023.

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

[77] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19113–19122, 2023.

[78] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15659–15669, 2023.

[79] Cristina Menghini, Andrew Delworth, and Stephen Bach. Enhancing clip with clip: Exploring pseudolabeling for limited-label prompt tuning. *Advances in Neural Information Processing Systems*, 36:60984–61007, 2023.

[80] Cairong Zhao, Yubin Wang, Xinyang Jiang, Yifei Shen, Kaitao Song, Dongsheng Li, and Duoqian Miao. Learning domain invariant prompt for vision-language models. *IEEE Transactions on Image Processing*, 2024.

[81] Shuvendu Roy and Ali Etemad. Consistency-guided prompt learning for vision-language models. *arXiv preprint arXiv:2306.01195*, 2023.

[82] Ji Zhang, Shihan Wu, Lianli Gao, Heng Tao Shen, and Jingkuan Song. Dept: Decoupled prompt tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12924–12933, 2024.

[83] Tao Yu, Zhihe Lu, Xin Jin, Zhibo Chen, and Xinchao Wang. Task residual for tuning vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10899–10909, 2023.

[84] Julio Silva-Rodriguez, Sina Hajimiri, Ismail Ben Ayed, and Jose Dolz. A closer look at the few-shot adaptation of large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23681–23690, 2024.

[85] Raphael Gontijo Lopes, Stefano Fenu, and Thad Starner. Data-free knowledge distillation for deep neural networks. *arXiv preprint arXiv:1710.07535*, 2017.

[86] Akisato Kimura, Zoubin Ghahramani, Koh Takeuchi, Tomoharu Iwata, and Naonori Ueda. Few-shot learning of neural networks from scratch by pseudo example optimization. *arXiv preprint arXiv:1802.03039*, 2018.

[87] Gaurav Kumar Nayak, Konda Reddy Mopuri, Vaisakh Shaj, Venkatesh Babu Radhakrishnan, and Anirban Chakraborty. Zero-shot knowledge distillation in deep networks. In *International Conference on Machine Learning*, pages 4743–4751. PMLR, 2019.

[88] Jaemin Yoo, Minyong Cho, Taebum Kim, and U Kang. Knowledge extraction with no observable data. *Advances in Neural Information Processing Systems*, 32, 2019.

[89] Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3514–3522, 2019.

[90] Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8715–8724, 2020.

[91] Gongfan Fang, Jie Song, Xinchao Wang, Chengchao Shen, Xingen Wang, and Mingli Song. Contrastive model inversion for data-free knowledge distillation. *arXiv preprint arXiv:2105.08584*, 2021.

[92] Dang Nguyen, Sunil Gupta, Kien Do, and Svetha Venkatesh. Black-box few-shot knowledge distillation. In *European Conference on Computer Vision*, pages 196–211. Springer, 2022.

[93] Gaurav Patel, Konda Reddy Mopuri, and Qiang Qiu. Learning to retain while acquiring: Combating distribution-shift in adversarial data-free knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7786–7794, 2023.

[94] Shikang Yu, Jiachen Chen, Hu Han, and Shuqiang Jiang. Data-free knowledge distillation via feature exchange and activation region constraint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24266–24275, 2023.

[95] He Liu, Yikai Wang, Huaping Liu, Fuchun Sun, and Anbang Yao. Small scale data-free knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6008–6016, 2024.

[96] Minh-Tuan Tran, Trung Le, Xuan-May Le, Jianfei Cai, Mehrtash Harandi, and Dinh Phung. Large-scale data-free knowledge distillation for imagenet via multi-resolution data generation. *arXiv preprint arXiv:2411.17046*, 2024.

[97] Yuzheng Wang, Dingkang Yang, Zhaoyu Chen, Yang Liu, Siao Liu, Wenqiang Zhang, Lihua Zhang, and Lizhe Qi. De-confounded data-free knowledge distillation for handling distribution shifts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12615–12625, 2024.

[98] Pan Du, Suyun Zhao, Zisen Sheng, Cuiping Li, and Hong Chen. Semi-supervised learning via weight-aware distillation under class distribution mismatch. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16410–16420, 2023.

[99] Jing Yang, Xiatian Zhu, Adrian Bulat, Brais Martinez, and Georgios Tzimiropoulos. Knowledge distillation meets open-set semi-supervised learning. *International Journal of Computer Vision*, pages 1–20, 2024.

[100] Zhiyuan Fang, Jianfeng Wang, Lijuan Wang, Lei Zhang, Yezhou Yang, and Zicheng Liu. Seed: Self-supervised distillation for visual representation. *arXiv preprint arXiv:2101.04731*, 2021.

[101] Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Compress: Self-supervised learning by compressing representations. *Advances in Neural Information Processing Systems*, 33:12980–12992, 2020.

[102] K L Navaneet, Soroush Abbasi Koohpayegani, Ajinkya Tejankar, and Hamed Pirsiavash. Simreg: Regression as a simple yet effective tool for self-supervised knowledge distillation. In *British Machine Vision Conference (BMVC)*, 2021.

[103] Kai Wang, Fei Yang, and Joost van de Weijer. Attention distillation: self-supervised vision transformer students need more guidance. *arXiv preprint arXiv:2210.00944*, 2022.

[104] Haohang Xu, Jiemin Fang, Xiaopeng Zhang, Lingxi Xie, Xinggang Wang, Wenrui Dai, Hongkai Xiong, and Qi Tian. Bag of instances aggregation boosts self-supervised distillation. *arXiv preprint arXiv:2107.01691*, 2021.

[105] Aditya Singh and Haohan Wang. Simple unsupervised knowledge distillation with space similarity. In *European Conference on Computer Vision*, pages 147–164. Springer, 2025.

[106] Zhiyuan Fang, Jianfeng Wang, Xiaowei Hu, Lijuan Wang, Yezhou Yang, and Zicheng Liu. Compressing visual-linguistic model via knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1428–1438, 2021.

[107] Kan Wu, Houwen Peng, Zhenghong Zhou, Bin Xiao, Mengchen Liu, Lu Yuan, Hong Xuan, Michael Valenzuela, Xi Stephen Chen, Xinggang Wang, et al. Tinyclip: Clip distillation via affinity mimicking and weight inheritance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21970–21980, 2023.

[108] Ximeng Sun, Pengchuan Zhang, Peizhao Zhang, Hardik Shah, Kate Saenko, and Xide Xia. Dime-fm: Distilling multimodal and efficient foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15521–15533, 2023.

[109] Gyeongman Kim, Doohyuk Jang, and Eunho Yang. Promptkd: Distilling student-friendly knowledge for generative language models via prompt tuning. *arXiv preprint arXiv:2402.12842*, 2024.

[110] Yifan Chen, Xiaozhen Qiao, Zhe Sun, and Xuelong Li. Comkd-clip: Comprehensive knowledge distillation for contrastive language-image pre-traning model. *arXiv preprint arXiv:2408.04145*, 2024.

[111] Marco Mistretta, Alberto Baldrati, Marco Bertini, and Andrew D Bagdanov. Improving zero-shot generalization of learned prompts via unsupervised knowledge distillation. In *European Conference on Computer Vision*, pages 459–477. Springer, 2025.

[112] Jang Hyun Cho and Bharath Hariharan. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4794–4802, 2019.

[113] Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5191–5198, 2020.

[114] Yichen Zhu and Yi Wang. Student customized knowledge distillation: Bridging the gap between student and teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5057–5066, 2021.

[115] Tao Huang, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge distillation from a stronger teacher. *Advances in Neural Information Processing Systems*, 35:33716–33727, 2022.

[116] Xin-Chun Li, Wen-Shu Fan, Bowen Tao, Le Gan, and De-Chuan Zhan. Exploring dark knowledge under various teacher capacities and addressing capacity mismatch. *arXiv preprint arXiv:2405.13078*, 2024.

[117] Rongzhi Zhang, Jiaming Shen, Tianqi Liu, Jialu Liu, Michael Bendersky, Marc Najork, and Chao Zhang. Do not blindly imitate the teacher: Using perturbed loss for knowledge distillation. *arXiv preprint arXiv:2305.05010*, 2023.

[118] Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

[119] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[120] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[121] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. Mixed precision training. *arXiv preprint arXiv:1710.03740*, 2017.

[122] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your vit but faster. *arXiv preprint arXiv:2210.09461*, 2022.

[123] Xudong Wang, Zhirong Wu, Long Lian, and Stella X Yu. Debiased learning from naturally imbalanced pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14647–14657, 2022.

[124] Jimmy Lei Ba. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[125] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[126] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Appendix Overview

This appendix provides supplementary material to support the main paper and is organized as follows:

- **Limitations and Future Work** (Appendix A): discusses the scope of foundation models and tasks considered in this work and outlines directions for future research.
- **Related Work** (Appendix B): discusses previous work relevant to ours, including vision-language pre-training, data-limited adaptation of VLMs, knowledge distillation (KD), and dual-head approaches.
- **Theoretical Analysis** (Appendix C): provides formal analysis of gradient conflicts in SHO with sufficient conditions for their occurrence (Appendix C.1), followed by inference equivalence guarantees for `DHO` (Appendix C.2).
- **Algorithms and Implementation** (Appendix D): presents detailed pseudocode (Appendix D.1), implementation specifics (Appendix D.2), and computational overhead analysis (Appendix D.3).
- **Datasets** (Appendix E): describes the datasets used in our experiments, including statistics and preprocessing details.
- **Additional Experiments** (Appendix F): presents motivation (Appendix F.1), MobileNet (Appendix F.2), gradient surgery (Appendix F.3), additional KD methods (Appendix F.4), adaptive weighting (Appendix F.5), SHO hyperparameters and (α, β) (Appendix F.6), 2-stage baselines (Appendix F.8), orthogonality to SSL methods (Appendix F.9), full-fine-tuning OOD results (Appendix F.10), repeated-run reliability (Appendix F.11), and additional VLM teachers (Appendix F.12).
- **Additional Analysis** (Appendix G): contains feature-quality analysis (Appendix G.1), per-layer gradient conflicts (Appendix G.2), non-linear head design and regularization (Appendix G.3), and dual-head and qualitative results (Appendix G.4).
- **The Use of LLMs** (Appendix H): describes our use of LLMs during paper preparation.

A Limitations and Future Work

While our main focus is to address the core problem of *gradient conflicts* arising from **general knowledge of VLMs and task-specific patterns from labeled data**, we acknowledge several limitations that present opportunities for future research.

Foundation-model scope. We primarily focus on VLMs as general knowledge source, as they provide strong zero-shot and few-shot capabilities in visual recognition tasks. However, we believe the fundamental problem of gradient conflicts between general foundational knowledge and task-specific objectives extends beyond VLMs. This conflict likely emerges in various scenarios where large pre-trained foundation models (such as instruction-tuned language models [56, 57, 58, 59, 60]) are adapted to specialized downstream tasks. Exploring how `DHO` performs across diverse foundation models and modalities remains an important direction for future work.

Task scope. Our implementation is limited to visual recognition tasks, as they represent the most fundamental domain in computer vision and provide an ideal testbed for analyzing gradient conflicts in knowledge transfer. Also, VLMs' strong zero-shot and few-shot capabilities in visual recognition tasks make them natural candidates for knowledge distillation. However, extending our approach to more complex visual understanding tasks such as object detection and segmentation would be a promising direction with dedicated architectural adaptations.

B Related Work

Vision-language pre-training. The emergence of vision-language pre-training has marked a significant breakthrough, enabling the use of extensive image-text pairs collected from the web [61, 62] to train powerful vision encoders transferable to various vision tasks [63, 64]. Early works such as CLIP [1] and ALIGN [2] leveraged contrastive learning techniques to align images and text

into a joint representation space, facilitating zero-shot transfer via language prompts. Building on these foundations, subsequent research has focused on improving vision-language models through enhanced training methodologies [65, 66, 67, 46], as well as scaling models and datasets [67, 68, 69, 70, 71, 47, 72, 73] with their zero-shot transfer capabilities [2, 74, 75, 14]. In contrast, our work focuses specifically on target tasks with compact models, aiming to distill knowledge from these large VLMs effectively.

Data-limited adaptation of VLMs. To preserve pretrained semantic features of VLMs during adaptation with limited data, several approaches have been proposed. Prompt tuning [76], initially designed for language models, has been successfully extended to vision tasks. Various methods [48, 7, 8, 77, 78, 49, 79, 80, 81, 82, 9] have demonstrated the effectiveness of training learnable prompts while keeping the base model frozen. Adapters [6, 5, 83, 84] provide an alternative approach by introducing lightweight, trainable modules while maintaining the pre-trained backbone intact. LP++ [4] has shown that simple linear layers can effectively adapt CLIP representations in data-limited settings. Note that our work is orthogonal to these approaches: we aim to distill the knowledge of pretrained VLMs into compact models under data-scarce scenarios, making these adaptation methods complementary and applicable to both teacher VLMs in our framework and student models when they are also VLMs.

Knowledge Distillation [**KD**; 15] enables transferring knowledge from large teacher models to compact student architectures, particularly in data-constrained settings. Researchers have explored synthetic data generation [85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97], semi-supervised [22, 28, 98, 99], and unsupervised KD using self-supervised teachers [100, 101, 102, 103, 104, 105]. In the VLM domain, recent works [106, 107, 108, 16, 17, 18, 19] distill from large-scale vision-language models into smaller architectures, often using transductive [109, 110] or multi-stage unsupervised strategies [20, 21, 111]. Meanwhile, KD remains challenging due to numerous issues, including model capacity gaps [112, 113, 114, 115, 116] and inconsistencies between soft and hard targets [117]. These challenges are further complicated by misalignment between labeled data and foundational knowledge, especially in few-shot learning scenarios where limited labeled examples may not fully capture the rich semantic understanding of foundation models.

Dual-head approaches. We also consider existing dual-head KD approaches. **SSKD** [28] trains separate heads for labeled and unlabeled data, assuming different data distributions, while **DHKD** [29] introduces a binary KD loss to alleviate neural collapse [118]. While these previous KD methods adopt dual-head architectures, they do not target distillation from foundation models or combine predictions at inference. Furthermore, both methods *infer using only the single supervised head* h_{CE} and do not address gradient conflicts arising from zero-/few-shot VLM teachers, whose prediction distributions can significantly differ from the limited labeled data. In contrast, `DHO` explicitly mitigates such gradient conflicts (Fig. 3) by training h_{KD} on both labeled and unlabeled data, and further provides *dual-head interpolation at inference time*, enabling flexible aggregation of supervised and teacher signals. As shown in Theorem 3 and Fig. 9, this interpolation can emulate tuning KD hyperparameters *without retraining*, giving `DHO` both improved performance and minimal hyperparameter tuning cost.

C Theoretical Analysis

In this section, we provide a theoretical analysis of our `Dual-Head Optimization` (`DHO`) framework. We first establish formal conditions under which gradient conflicts arise in single-head knowledge distillation (Appendix C.1), proving that conflicts are structurally guaranteed when the student prediction lies between the label and teacher distributions. We then show that `DHO` breaks this structural guarantee by decoupling the head parameters. Next, we prove that post-training, the optimal prediction from the dual-head model—formulated as a weighted combination of the heads' outputs—is mathematically equivalent to the optimal solution of conventional single-head distillation (Appendix C.2). Note that in this section we slightly abuse the notation of the main paper for clarity, *e.g.*, we denote p_τ as teacher predictions with temperature scaling τ.

C.1 Analysis of Gradient Conflicts

We formalize the gradient conflict phenomenon observed empirically in Fig. 3 and provide sufficient conditions under which it is guaranteed. We adopt the notation from the main paper: $\hat{p} = \sigma(h(z))$ is

the student prediction, y is the one-hot ground-truth label, p is the teacher soft prediction, W is the classifier weight, $z = g(x)$ is the feature representation, and θ parameterizes the feature extractor g.

C.1.1 Gradient Conflict in SHO: Classifier Head

Recall from the main paper that the inner product of the CE and KD gradients with respect to the classifier weight W is:

$$\langle \mathrm{vec}(\nabla_W \mathcal{L}_{\mathrm{CE}}),\ \mathrm{vec}(\nabla_W \mathcal{L}_{\mathrm{KD}}) \rangle = (\hat{p} - y)^\top (\hat{p} - p)\ \|z\|^2. \tag{11}$$

Since $\|z\|^2 > 0$, the sign of the gradient inner product is determined entirely by $(\hat{p} - y)^\top (\hat{p} - p)$. The following proposition gives a geometric characterization of when this term is negative.

Proposition 1 (Geometric condition for gradient conflict in SHO)**.** *Define the midpoint $m = \frac{y+p}{2}$ and the radius $r = \frac{\|y-p\|}{2}$. Then:*

$$(\hat{p} - y)^\top (\hat{p} - p) = \|\hat{p} - m\|^2 - r^2. \tag{12}$$

Consequently, gradient conflict in the classifier head occurs if and only if $\hat{p} \in \mathcal{B}(m, r)$, where $\mathcal{B}(m, r) = \{q : \|q - m\| < r\}$ is the open ball centered at m with radius r.

Proof. Expanding $(\hat{p} - y)^\top (\hat{p} - p)$:

$$(\hat{p} - y)^\top (\hat{p} - p) = \|\hat{p}\|^2 - \hat{p}^\top (y + p) + y^\top p \tag{13}$$

$$= \|\hat{p}\|^2 - 2\hat{p}^\top m + \|m\|^2 - \|m\|^2 + y^\top p \tag{14}$$

$$= \|\hat{p} - m\|^2 - \left(\|m\|^2 - y^\top p\right). \tag{15}$$

We compute the remaining term:

$$\|m\|^2 - y^\top p = \frac{\|y + p\|^2}{4} - y^\top p = \frac{\|y\|^2 + 2y^\top p + \|p\|^2}{4} - y^\top p \tag{16}$$

$$= \frac{\|y\|^2 - 2y^\top p + \|p\|^2}{4} = \frac{\|y - p\|^2}{4} = r^2. \tag{17}$$

Substituting into Eq. 15 yields $(\hat{p} - y)^\top (\hat{p} - p) = \|\hat{p} - m\|^2 - r^2$. This is negative if and only if $\|\hat{p} - m\| < r$, *i.e.*, $\hat{p} \in \mathcal{B}(m, r)$. □

Proposition 1 establishes that gradient conflict occurs whenever the student prediction lies inside the ball whose diameter is the segment from y to p. The following corollary shows that a natural interpolation condition is sufficient.

Corollary 1 (Interpolation implies gradient conflict)**.** *If the student prediction lies on the line segment between y and p, i.e., $\hat{p} = (1 - t)\, y + t\, p$ for some $t \in (0, 1)$, and $p \neq y$, then:*

$$(\hat{p} - y)^\top (\hat{p} - p) = -t(1 - t)\, \|p - y\|^2 < 0. \tag{18}$$

Proof. Under the interpolation condition: $\hat{p} - y = t(p - y)$ and $\hat{p} - p = -(1 - t)(p - y)$. Therefore, $(\hat{p} - y)^\top (\hat{p} - p) = -t(1 - t)\, \|p - y\|^2 < 0$, since $t \in (0, 1)$ and $p \neq y$. Alternatively, one can verify that $\|\hat{p} - m\| = |t - \frac{1}{2}|\, \|p - y\| < \frac{1}{2}\|p - y\| = r$ for $t \in (0, 1)$, confirming $\hat{p} \in \mathcal{B}(m, r)$. □

C.1.2 Gradient Conflict Propagates to the Feature Extractor

We now show that gradient conflicts in the classifier head propagate to the feature extractor parameters θ under the same interpolation condition.

Proposition 2 (Gradient conflict in SHO: feature extractor)**.** *Let $J = \frac{\partial z}{\partial \theta} \in \mathbb{R}^{d \times P}$ be the Jacobian of the feature extractor. The inner product of the CE and KD gradients with respect to θ is:*

$$\langle \nabla_\theta \mathcal{L}_{CE},\ \nabla_\theta \mathcal{L}_{KD} \rangle = (\hat{p} - y)^\top\, A\, (\hat{p} - p), \tag{19}$$

where $A = W\, J\, J^\top W^\top \in \mathbb{R}^{C \times C}$ is positive semi-definite (PSD). Under the interpolation condition $\hat{p} = (1 - t)\, y + t\, p$ with $t \in (0, 1)$ and $p \neq y$:

$$\langle \nabla_\theta \mathcal{L}_{CE},\ \nabla_\theta \mathcal{L}_{KD} \rangle = -t(1 - t)\, (p - y)^\top A\, (p - y) \leq 0. \tag{20}$$

Equality holds only in the degenerate case $A(p - y) = 0$, i.e., when the teacher-label disagreement lies in the null space of A.

Proof. From the chain rule, $\nabla_\theta \mathcal{L}_{\text{CE}} = J^\top W^\top (\hat{p} - y)$ and $\nabla_\theta \mathcal{L}_{\text{KD}} = J^\top W^\top (\hat{p} - p)$. Their inner product is:

$$(\hat{p} - y)^\top W J J^\top W^\top (\hat{p} - p) = (\hat{p} - y)^\top A (\hat{p} - p). \tag{21}$$

A is PSD since for any $v \in \mathbb{R}^C$: $v^\top A\, v = v^\top W J J^\top W^\top v = \|J^\top W^\top v\|^2 \geq 0$. Under the interpolation condition:

$$(\hat{p} - y)^\top A\, (\hat{p} - p) = t(p - y)^\top A \cdot \big(-(1-t)(p-y)\big) = -t(1-t)\,(p-y)^\top A\,(p-y) \leq 0, \tag{22}$$

where the inequality follows from $A \succeq 0$. □

Remark 1 (Prevalence of gradient conflicts during training). The interpolation condition $\hat{p} \approx (1 - t)\, y + t\, p$ naturally arises during joint CE+KD optimization. At the start of training, the student prediction $\hat{p}$ is approximately uniform; as training progresses, the student is simultaneously pulled toward the one-hot label y by $\mathcal{L}_{\text{CE}}$ and toward the teacher distribution p by $\mathcal{L}_{\text{KD}}$. The resulting trajectory passes through the conflict region $\mathcal{B}(m, r)$, which encompasses a large portion of the probability simplex when $\|y - p\|$ is substantial—as is typically the case for zero-shot VLM teachers on fine-grained tasks. Fig. 3 provides empirical confirmation: the inner product $(\hat{p} - y)^\top (\hat{p} - p)$ (—×—) turns negative early in training and remains so, producing sustained gradient conflicts (—●—).

C.1.3 DHO Eliminates Structural Gradient Conflict

We now show that `DHO` removes the structural mechanism that causes gradient conflicts.

Proposition 3 (DHO eliminates structural gradient conflict)**.** *In `DHO`:*

(a) ***Classifier heads:*** *Since $\mathcal{L}_{CE}$ depends only on W_{CE} and $\mathcal{L}_{KD}$ depends only on W_{KD}, there is no gradient interaction: $\nabla_{W_{CE}} \mathcal{L}_{KD} = 0$ and $\nabla_{W_{KD}} \mathcal{L}_{CE} = 0$. Gradient conflict in the classifier head is eliminated by construction.*

(b) ***Feature extractor:*** *The gradient inner product with respect to θ becomes:*

$$\langle \nabla_\theta \mathcal{L}_{CE},\ \nabla_\theta \mathcal{L}_{KD} \rangle = (\hat{p}_{CE} - y)^\top\, B\, (\hat{p}_{KD} - p), \tag{23}$$

*where $B = W_{CE}\, J J^\top W_{KD}^\top \in \mathbb{R}^{C \times C}$. Unlike the SHO matrix $A = W J J^\top W^\top$ which is PSD, the cross-head matrix B is **not necessarily PSD**. Therefore, the structural guarantee of negative gradient inner product (Proposition 2) is broken.*

Proof. Part (a) is immediate from the independence of W_{CE} and W_{KD} in the respective loss functions. For part (b), consider arbitrary $v \in \mathbb{R}^C$:

$$v^\top B\, v = v^\top W_{\text{CE}} J J^\top W_{\text{KD}}^\top v = (W_{\text{CE}}^\top v)^\top (J J^\top)(W_{\text{KD}}^\top v). \tag{24}$$

This equals $(J^\top W_{\text{CE}}^\top v)^\top (J^\top W_{\text{KD}}^\top v) = \langle u_1, u_2 \rangle$, where $u_1 = J^\top W_{\text{CE}}^\top v$ and $u_2 = J^\top W_{\text{KD}}^\top v$ are generally different vectors. Since $\langle u_1, u_2 \rangle$ can be positive, negative, or zero depending on the angle between u_1 and u_2, the matrix B is indefinite in general. In contrast, the SHO matrix A always satisfies $v^\top A v = \|J^\top W^\top v\|^2 \geq 0$. The PSD structure that guarantees conflict in SHO (Proposition 2) is thus absent in `DHO`, which is consistent with the empirical observation of positive gradient alignment (—●— in Fig. 3). □

Remark 2 (Summary: mechanism of gradient conflict resolution). In SHO, gradient conflict arises from a structural interaction: the shared head W creates a PSD matrix $A = W J J^\top W^\top$ that, combined with the opposing error vectors $(\hat{p} - y)$ and $(\hat{p} - p)$, systematically produces negative gradient inner products (Propositions 1 and 2). `DHO` resolves this by (a) eliminating head-level conflicts entirely through decoupled parameters, and (b) replacing the PSD matrix A with an indefinite cross-head matrix B, which prevents the systematic negative alignment from propagating to the feature extractor.

C.2 Inference Equivalence

We now prove that, despite training with separate heads, the combined prediction of `DHO` at inference time recovers the theoretical optimum of SHO.

C.2.1 Optimal Distribution for Single-Head Optimization

We begin by considering two target probability distributions: the ground truth label distribution y and the teacher's softened distribution p_τ for input $x \in \mathcal{X}$, where:

- y represents the ground truth label distribution, typically one-hot encoded vectors where $y_c = 1$ for the true class c and 0 elsewhere
- p_τ denotes the teacher's softened distribution with temperature scaling: $p_\tau = \sigma(z_t/\tau)$, where z_t represents the teacher's logits and σ is the softmax function

Theorem 2 (Optimal Distribution for Single-Head Optimization)**.** *The distribution $\hat{p}^*$ that minimizes the weighted combination of cross-entropy loss with respect to y and Kullback-Leibler divergence with respect to p_τ:*

$$\mathcal{L}(\hat{p}) = \lambda \ell(\hat{p}, y) + (1-\lambda) D_{\mathrm{KL}}(p_\tau \| \hat{p}) \tag{25}$$

is given by the weighted arithmetic mean:

$$\hat{p}^* = \lambda y + (1-\lambda) p_\tau \tag{26}$$

where $\lambda \in [0,1]$ is the weighting hyperparameter.

Proof. We begin by expanding the objective function:

$$\begin{aligned}
\mathcal{L}(\hat{p}) &= \lambda \ell(\hat{p}, y) + (1-\lambda) D_{\mathrm{KL}}(p_\tau \| \hat{p}) && (27)\\
&= -\lambda \sum_{c=1}^{C} y_c \log \hat{p}_c + (1-\lambda) \sum_{c=1}^{C} p_{\tau,c} \log \frac{p_{\tau,c}}{\hat{p}_c} && (28)\\
&= -\lambda \sum_{c=1}^{C} y_c \log \hat{p}_c + (1-\lambda) \sum_{c=1}^{C} p_{\tau,c} \log p_{\tau,c} - (1-\lambda) \sum_{c=1}^{C} p_{\tau,c} \log \hat{p}_c && (29)\\
&= -\sum_{c=1}^{C} [\lambda y_c + (1-\lambda) p_{\tau,c}] \log \hat{p}_c + (1-\lambda) \sum_{c=1}^{C} p_{\tau,c} \log p_{\tau,c} && (30)
\end{aligned}$$

Since the last term is constant with respect to $\hat{p}$, the optimization problem reduces to minimizing:

$$\mathcal{L}'(\hat{p}) = -\sum_{c=1}^{C} [\lambda y_c + (1-\lambda) p_{\tau,c}] \log \hat{p}_c \tag{31}$$

Subject to the probability constraints:

$$\sum_{c=1}^{C} \hat{p}_c = 1, \quad \hat{p}_c \geq 0 \quad \forall c \in \{1, 2, \ldots, C\} \tag{32}$$

Applying the method of Lagrange multipliers with multiplier μ:

$$\mathcal{L}(\hat{p}, \mu) = -\sum_{c=1}^{C} [\lambda y_c + (1-\lambda) p_{\tau,c}] \log \hat{p}_c + \mu \left(\sum_{c=1}^{C} \hat{p}_c - 1 \right) \tag{33}$$

Taking the partial derivative with respect to $\hat{p}_c$ and setting it to zero:

$$-\frac{\lambda y_c + (1-\lambda) p_{\tau,c}}{\hat{p}_c} + \mu = 0 \tag{34}$$

Solving for $\hat{p}_c$:

$$\hat{p}_c = \frac{\lambda y_c + (1-\lambda) p_{\tau,c}}{\mu} \tag{35}$$

Using the constraint $\sum_{c=1}^{C} \hat{p}_c = 1$, and observing that $\sum_{c=1}^{C} y_c = 1$ and $\sum_{c=1}^{C} p_{\tau,c} = 1$ (both being probability distributions):

$$\sum_{c=1}^{C} \hat{p}_c = \sum_{c=1}^{C} \frac{\lambda y_c + (1-\lambda) p_{\tau,c}}{\mu} = 1 \tag{36}$$

$$\frac{1}{\mu}[\lambda \sum_{c=1}^{C} y_c + (1-\lambda) \sum_{c=1}^{C} p_{\tau,c}] = 1 \tag{37}$$

$$\frac{1}{\mu}[\lambda + (1-\lambda)] = 1 \implies \mu = 1 \tag{38}$$

Therefore, the optimal solution is:

$$\hat{p}_c^* = \lambda y_c + (1-\lambda) p_{\tau,c} \tag{39}$$

This weighted arithmetic mean of the two target distributions is the optimal solution that minimizes our objective function. □

C.2.2 Inference Equivalence of Dual-Head Optimization

In our proposed Dual-Head Optimization (DHO) framework, we extract shared features $g(x)$ from input x and apply two specialized classification heads:

- $h_{\text{CE}}(z) = W_{\text{CE}} z + b_{\text{CE}}$: optimized exclusively to match ground truth labels using cross-entropy loss $\ell(\sigma(h_{\text{CE}}(z)), y)$
- $h_{\text{KD}}(z) = W_{\text{KD}} z + b_{\text{KD}}$: optimized exclusively to match teacher predictions using KL divergence $D_{\text{KL}}(p_\tau \| \sigma(h_{\text{KD}}(z)/\beta))$

where $z = g(x)$ is the feature representation, and the parameter β controls the temperature during inference, while a fixed temperature of 1 is used during training of the knowledge distillation head.

Assumption 2 (ε-Convergence)**.** We assume that after sufficient training, both heads have converged to their respective target distributions with bounded error:

$$\sup_x \|\sigma(h_{\text{CE}}(z)) - y\|_1 \leq \varepsilon, \quad \sup_x \|\sigma(h_{\text{KD}}(z)/\beta) - p_\tau\|_1 \leq \varepsilon \tag{40}$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm and $\varepsilon > 0$ is a small constant.

Theorem 3 (Inference Equivalence Under ε-Convergence)**.** *Under Assumption 2, by combining the outputs of both heads as:*

$$\hat{p}_{\texttt{DHO}} = \alpha \cdot \sigma(h_{CE}(z)) + (1-\alpha) \cdot \sigma(h_{KD}(z)/\beta), \quad \textit{where } \alpha = \lambda \tag{41}$$

we obtain a prediction that approximates the optimal single-head solution with bounded error:

$$\|\hat{p}_{\texttt{DHO}} - \hat{p}^*\|_1 \leq \varepsilon \tag{42}$$

Proof. We analyze the ℓ_1 distance between the DHO prediction and the optimal solution:

$$\begin{aligned}
\|\hat{p}_{\texttt{DHO}} - \hat{p}^*\|_1 &= \|\alpha \cdot \sigma(h_{\text{CE}}(z)) + (1-\alpha) \cdot \sigma(h_{\text{KD}}(z)/\beta) - \lambda y - (1-\lambda) p_\tau\|_1 && (43)\\
&= \|\lambda(\sigma(h_{\text{CE}}(z)) - y) + (1-\lambda)(\sigma(h_{\text{KD}}(z)/\beta) - p_\tau)\|_1 && (44)\\
&\leq \lambda \|\sigma(h_{\text{CE}}(z)) - y\|_1 + (1-\lambda)\|\sigma(h_{\text{KD}}(z)/\beta) - p_\tau\|_1 && (45)\\
&\leq \lambda\varepsilon + (1-\lambda)\varepsilon = \varepsilon && (46)
\end{aligned}$$

where we applied the triangle inequality for the ℓ_1 norm and used Assumption 2.

Therefore, we have established that:

$$\hat{p}_{\texttt{DHO}} \approx_\varepsilon \hat{p}^* \tag{47}$$

where $\approx_\varepsilon$ denotes approximation with ℓ_1 error bound ε. □

Lemma 1 (Temperature Matching via KL Divergence)**.** *Assume the knowledge distillation head is trained to minimize KL divergence with respect to the teacher's predictions at temperature 1, such that:*

$$D_{\mathrm{KL}}(p_1 \| \sigma(h_{KD}(z))) \leq \delta \tag{48}$$

Then, setting the temperature parameter $\beta = \tau$ at inference time allows the KD head to approximate the teacher's prediction at temperature τ with error bound:

$$\|\sigma(h_{KD}(z)/\beta) - p_\tau\|_1 \leq \sqrt{2\delta} \tag{49}$$

Proof. When logits are properly scaled and under appropriate conditions of the softmax function, we can reasonably approximate:

$$D_{\mathrm{KL}}(p_\tau \| \sigma(h_{\mathrm{KD}}(z)/\tau)) \approx D_{\mathrm{KL}}(p_1 \| \sigma(h_{\mathrm{KD}}(z))) \leq \delta \tag{50}$$

Applying Pinsker's inequality, which establishes a relationship between KL divergence and the L1 norm difference between probability distributions:

$$\|p_\tau - \sigma(h_{\mathrm{KD}}(z)/\tau)\|_1 \leq \sqrt{2 D_{\mathrm{KL}}(p_\tau \| \sigma(h_{\mathrm{KD}}(z)/\tau))} \leq \sqrt{2\delta} \tag{51}$$

To ensure ε-convergence between the KD head at temperature τ and the teacher's prediction at temperature τ, it is sufficient to guarantee:

$$\sqrt{2\delta} \leq \varepsilon \Rightarrow \delta \leq \frac{\varepsilon^2}{2} \tag{52}$$

□

Corollary 2 (Optimal `DHO` Configuration)**.** *With proper training ensuring ε-convergence of both heads, dual-head optimization with temperature parameter $\beta = \tau$ and mixing parameter $\alpha = \lambda$ approximates the optimal single-head objective with error bounded by ε:*

$$\hat{p}_{\texttt{DHO}} \approx_\varepsilon \hat{p}^* = \lambda y + (1-\lambda) p_\tau \tag{53}$$

This demonstrates that our `DHO` approach achieves the same theoretical optimality as SHO.

Remark 3 (On the role of Assumption 2). The supremum form of Assumption 2 is the cleanest sufficient condition under which Theorem 3 delivers a uniform ℓ_1 bound; an average-error variant $\mathbb{E}_x\|\sigma(h_{\mathrm{CE}}(z)) - y\|_1 \leq \varepsilon$, $\mathbb{E}_x\|\sigma(h_{\mathrm{KD}}(z)) - p\|_1 \leq \varepsilon$ preserves Theorem 3 in expectation by the same triangle-inequality argument. Two clarifications about the role of this assumption. (i) Theorem 3 establishes a *floor on what `DHO` recovers*: post-training tuning of (α, β) subsumes the family of SHO objectives parametrized by λ, replacing N training-time grid points of SHO with a single training run of `DHO`. The practical contribution—test-time tuning instead of train-time tuning—survives any reasonable relaxation of the assumption. (ii) Theorem 3 does *not* explain why `DHO` outperforms SHO empirically; that gain comes from gradient-conflict-free feature learning (Proposition 3, supported by Tab. 5 and Fig. 8), not from the inference-time interpolation itself.

D Algorithms and Implementation

D.1 Pseudocode

We present the pseudocode for `DHO` in Algs. 1 and 2 for training and inference, respectively.

Algorithm 1 `DHO` Training with zero-shot CLIP [1] teacher

Input: labeled set $\mathcal{D}^{(l)} = \{(x_i^{(l)}, y_i)\}_{i=1}^{N}$, unlabeled set $\mathcal{D}^{(u)} = \{x_j^{(u)}\}_{j=1}^{M}$,
student feature extractor g, prediction heads $h_{\text{CE}}, h_{\text{KD}}$, teacher encoders $f_{\mathcal{X}}, f_{\mathcal{T}}$,
prompt template "A photo of `[CLASS]`", temperature scaling factors ζ, η,
balancing hyperparameter λ,
supervised mini-batch size B, and unsupervised mini-batch size B'.
while not converged **do**
  Sample mini-batch $\mathcal{B}^{(l)} = \{(x_b^{(l)}, y_b)\}_{b=1}^{B}$ from $\mathcal{D}^{(l)}$, $\mathcal{B}^{(u)} = \{x_{b'}^{(u)}\}_{b'=1}^{B'}$ from $\mathcal{D}^{(l)} \cup \mathcal{D}^{(u)}$.
  // Process labeled data
  for each $(x_b^{(l)}, y_b) \in \mathcal{B}^{(l)}$ **do**
    $z_b^{(l)} \leftarrow g(x_b^{(l)})$
    $\hat{p}_{\text{CE},b}^{(l)} \leftarrow \sigma(h_{\text{CE}}(z_b^{(l)}))$
    $\hat{p}_{\text{KD},b}^{(l)} \leftarrow \sigma(\frac{1}{\eta} h_{\text{KD}}(z_b^{(l)}))$
    $p_b^{(l)} \leftarrow \sigma\left(\frac{1}{\zeta \cdot \eta}[\texttt{CosSim}(f_{\mathcal{X}}(x_b^{(l)}), f_{\mathcal{T}}(t_1)), \ldots, \texttt{CosSim}(f_{\mathcal{X}}(x_b^{(l)}), f_{\mathcal{T}}(t_C))]^{\top}\right)$
  end for
  // Process unlabeled data
  for each $x_{b'}^{(u)} \in \mathcal{B}^{(u)}$ **do**
    $z_{b'}^{(u)} \leftarrow g(x_{b'}^{(u)})$
    $\hat{p}_{\text{KD},b'}^{(u)} \leftarrow \sigma(\frac{1}{\eta} h_{\text{KD}}(z_{b'}^{(u)}))$
    $p_{b'}^{(u)} \leftarrow \sigma\left(\frac{1}{\zeta \cdot \eta}[\texttt{CosSim}(f_{\mathcal{X}}(x_{b'}^{(u)}), f_{\mathcal{T}}(t_1)), \ldots, \texttt{CosSim}(f_{\mathcal{X}}(x_{b'}^{(u)}), f_{\mathcal{T}}(t_C))]^{\top}\right)$
  end for
  // Compute losses and update
  $\mathcal{L}_{\text{CE}} \leftarrow \frac{1}{B} \sum_{b=1}^{B} \ell(\hat{p}_{\text{CE},b}^{(l)}, y_b)$
  $\mathcal{L}_{\text{KD}} \leftarrow \frac{1}{B} \sum_{b=1}^{B} D_{\text{KL}}(\hat{p}_{\text{KD},b}^{(l)} || p_b^{(l)}) + \frac{1}{B'} \sum_{b'=1}^{B'} D_{\text{KL}}(\hat{p}_{\text{KD},b'}^{(u)} || p_{b'}^{(u)})$
  $\mathcal{L} \leftarrow \lambda \mathcal{L}_{\text{CE}} + (1 - \lambda) \mathcal{L}_{\text{KD}}$
  Update parameters of g, h_{CE}, h_{KD} using $\nabla \mathcal{L}$
end while

Algorithm 2 Dual-Head Optimization Inference

Input: an image x, feature extractor g, prediction heads $h_{\text{CE}}, h_{\text{KD}}$, linear coefficient α, temperature scaling β
$z \leftarrow g(x)$
$\hat{p}_{\text{CE}} \leftarrow \sigma(h_{\text{CE}}(z))$
$\hat{p}_{\text{KD}} \leftarrow \sigma(h_{\text{KD}}(z)/\beta)$
$\hat{p} \leftarrow \alpha \cdot \hat{p}_{\text{CE}} + (1 - \alpha) \cdot \hat{p}_{\text{KD}}$
$\hat{y} \leftarrow \arg\max_c(\hat{p}_c)$
Return: $\hat{y}$

D.2 Implementation Details

Architecture choices for teacher and student models. For zero-shot teachers, we use CLIP ResNet-50 [1] in few-shot settings and VIT-H/14 from DFN [47] in low-shot settings. For few-shot teachers, we adopt Tip-Adapter-F [5], a learnable adapter model, and denote the corresponding variant as `DHO-F`. On ImageNet, to avoid label leakage, we either train ResNet-18 from scratch or use a self-supervised ResNet-50 from DINO [50]. For other datasets, we use ResNet-18 and MobileNetV2 [51] without such concerns. In low-shot settings, we use CLIP ViT-B/16 and ViT-L/14 [1].

Few-/low-shot settings. Tab. 7 provides a comprehensive overview of the implementation details for our experiments on 1) few-shot semi-supervised settings on ImageNet and 10 datasets, 2) low-shot semi-supervised settings on ImageNet, and 3) VLM-based adaptation methods.

Table 7: Implementation details for our experiments across different settings.

Few-shot Semi-supervised Settings on ImageNet

Model Configuration	Student Training Details
• **Student:** ResNet18 [119] from scratch or ResNet50 from DINO [50] • **Input size:** 224×224 • **Zero-shot Teacher:** ResNet50 from CLIP [1] • **Few-shot Teacher:** ResNet50 from Tip-Adapter-F [5] • **Teacher input size:** 224×224 • **labeled data:** $K \in \{1, 2, 4, 8, 16\}$ shots • ζ, η, **and** λ: $0.01, 2, 0.5$ • α **and** β**:** $\alpha = 0.4$, $\beta = 0.5$ (zero-shot); $\alpha = 0.2$, $\beta = 0.5$ (few-shot)	• **Epochs:** 20 • **Optimizer:** AdamW (β_1=0.9, β_2=0.999) • **Learning rate:** 1×10^{-3}, weight decay: 1×10^{-2} • **Batch size:** 512 (labeled: 256, unlabeled: 256) • **Scheduler:** Cosine decay without warmup • **Augmentation:** Random crops (x0.5-1.0), horizontal flips

Few-shot Semi-supervised Settings on 10 Fine-Grained Datasets

Model Configuration	Student Training Details
• **Student:** ResNet18 [119] or MobileNet [51] pre-trained on ImageNet under supervision • **Input size:** 224×224 • **Zero-shot Teacher:** ResNet50 from CLIP [1] • **Few-shot Teacher:** ResNet50 from Tip-Adapter-F [5] • **Teacher input size:** 224×224 • **labeled data:** $K \in \{1, 2, 4, 8, 16\}$ shots • ζ, η, **and** λ: $0.01, 2, 0.5$ • α **and** β**:** determined by validation	• **Epochs:** 200 • **Optimizer:** AdamW (β_1=0.9, β_2=0.999) • **Learning rate:** 1×10^{-3}, weight decay: 1×10^{-2} • **Batch size:** 128 (labeled: 64, unlabeled: 64) • **Scheduler:** Cosine decay without warmup • **Augmentation:** Random crops (x0.5-1.0), horizontal flips

Low-shot Semi-supervised Settings on ImageNet

Model Configuration	Student Training Details
• **Student:** CLIP ViT-B/16 or ViT-L/14 [1] • **Input size:** 224×224 (ViT-B/16) or 336×336 (ViT-L/14) • **Zero-shot Teacher:** CLIP ViT-L/14 or ViT-H/14 [47] • **Teacher input size:** 336×336 (ViT-L/14) or 378×378 (ViT-H/14) • **Few-shot Teacher:** N/A • **labeled data:** 1% ($\frac{N}{N+M} \approx 0.01$) or 10% ($\frac{N}{N+M} \approx 0.1$) of training data • ζ, η, **and** λ: $0.01, 2, 0.5$ • α **and** β**:** $\alpha = 0.5$, $\beta = 0.5$	• **Epochs:** 32 • **Optimizer:** AdamW (β_1=0.9, β_2=0.999) • **Learning rate:** 5×10^{-5}, weight decay: 5×10^{-2} • **Batch size:** 512 (labeled: 256, unlabeled: 256) • **Scheduler:** Cosine warmup decay (5000 steps) • **Augmentation:** Random crops (x0.5-1.0), horizontal flips

OOD settings. Tab. 8 provides the implementation details for our out-of-distribution (OOD) generalization experiments on 1) full training model evaluation and 2) adaptation methods including linear evaluation, visual prompt tuning, and VLM-based methods.

Table 8: Implementation details for our out-of-distribution generalization experiments.

Full Training

Model Configuration	**Training Details**
• **Student:** CLIP ViT-B/16 [1] • **Student input size:** 224×224 • **Zero-shot Teacher:** CLIP ViT-L/14 [1] • **Teacher input size:** 336×336 • **Labeled data:** 1% and 10% ImageNet • ζ, η, **and** λ: $0.01, 2, 0.5$ • α **and** β**:** 0.5 and 1	• **Epochs:** 32 • **Optimizer:** AdamW (β_1=0.9, β_2=0.999) • **Learning rate:** 5×10^{-5}, weight decay: 5×10^{-2} • **Batch size:** 512 (labeled: 256, unlabeled: 256) • **Scheduler:** Cosine warmup decay (5000 steps) • **Augmentation:** Random crops (x0.5-1.0), horizontal flips

Adaptation Methods (Linear Evaluation & Visual Prompt Tuning)

Method Configuration	**Training Details**
• **Linear evaluation** [50] • **Visual prompt tuning** [48] • **Frozen backbone:** CLIP ViT-B/16 [1] • **Input size:** 224×224 • **Zero-shot Teacher:** CLIP ViT-L/14 [1] • **Teacher input size:** 336×336 • **Labeled data:** 1% and 10% ImageNet • ζ, η, **and** λ: $0.01, 2, 0.5$ • α **and** β**:** 0.5 and 1	• **Epochs:** 20 • **Optimizer:** AdamW (β_1=0.9, β_2=0.999) • **Learning rate:** 5×10^{-5}, weight decay: 5×10^{-2} • **Batch size:** 512 (labeled: 256, unlabeled: 256) • **Scheduler:** Cosine warmup decay (5000 steps) • **Augmentation:** Random crops (x0.5-1.0), horizontal flips

Adaptation Methods (Prompt Tuning)

Method Configuration	**Training Details**
• **Prompt tuning:** CoOp [7], PromptSRC [49] • **Frozen backbone:** CLIP ViT-B/16 [1] • **Input size:** 224×224 • **Zero-shot Teacher:** CLIP ViT-L/14 [1] • **Teacher input size:** 336×336 • **Labeled data:** 1% and 10% ImageNet • ζ, η, **and** λ: $0.01, 2, 0.5$ • α **and** β**:** 0.5 and 1	• **Prompt tuning:** Following PromptSRC [49] configurations • **Comparison:** CasPL [21] with domain-specific unlabeled data

D.3 Computational Costs

Inference overhead of DHO over SHO. Tab. 9 presents computational overheads at inference time introduced by DHO over SHO for all the architectures in this paper, such as MobileNetV2 [51], ResNet-18 [119], ResNet-50 [119], ViT-B/16 [120], and ViT-L/16 [120].

Table 9: Inference overhead using RTX 4090 across different architecture.

Model	Params (M)	FLOPs (G)	Throughput (im/s)
MobileNetV2	3.50	0.33	2978.4
+ DHO	4.79 (+36.5%)	0.34 (+3.0%)	2971.2 (-0.24%)
ResNet-18	11.69	1.83	3525.7
+ DHO	12.20 (+4.4%)	1.83 (+0.0%)	3518.6 (-0.20%)
ResNet-50	25.56	4.14	1018.4
+ DHO	27.61 (+8.0%)	4.15 (+0.2%)	1016.4 (-0.19%)
ViT-B/16	86.57	16.87	290.2
+ DHO	87.34 (+0.9%)	16.87 (+0.0%)	290.1 (-0.02%)
ViT-L/16	304.33	59.70	255.1
+ DHO	305.35 (+0.3%)	59.70 (+0.0%)	255.6 (+0.18%)

Training time and hardware requirements. Tab. 10 presents the training time required for our experiments. For VLM distillation experiments, which represent the most resource-intensive component of our work, we used 8Œ NVIDIA RTX 4090 GPUs. The ViT-H/14 to ViT-L/14 distillation required approximately 80 hours, while the ViT-H/14 to ViT-B/16 and ViT-L/14 to ViT-B/16 distillations required approximately 40 and 28 hours, respectively. For the ViT-H/14 to ViT-L/14 distillation, we implemented gradient accumulation with 4 steps and mixed precision training [121] to optimize computational efficiency. For ImageNet experiments, we used 4Œ NVIDIA RTX 4090 GPUs, with ResNet-18 and ResNet-50 models requiring approximately 6 and 8 hours of training time, respectively. We provide these details to facilitate reproduction of our results and to give researchers a clear understanding of the computational resources needed to implement our approach at scale.

Table 10: Training time and hardware.

Student	Teacher	Training Time	Hardware
ResNet-18	ResNet-50	≈ 6 hours	4Œ RTX 4090
ResNet-50	ResNet-50	≈ 8 hours	4Œ RTX 4090
ViT-B/16	ViT-L/14	≈ 28 hours	8Œ RTX 4090
ViT-B/16	ViT-H/14	≈ 40 hours	8Œ RTX 4090
ViT-L/14	ViT-H/14	≈ 80 hours	8Œ RTX 4090

Inference overhead improvements with ToMe. To further improve the computational efficiency of our approach, we explored integrating Token Merging (ToMe) [122] with DHO. ToMe is a technique that reduces the number of tokens in ViTs by merging similar tokens to improve the efficiency of ViTs. Tab. 11 shows that combining DHO with ToMe significantly reduces computational costs with minimal impact on performance.

Table 11: Performance and inference overhead of DHO with Token Merging (ToMe) on ImageNet under low-shot semi-supervised settings using RTX 4090.

Method	Labeled	Accuracy (%)	Params (M)	FLOPs (G)	Throughput (im/s)
DHO	1%	81.6	87.22	17.58	243.35
DHO + ToMe	1%	81.4 (–0.2)	87.22	13.12 (–25.4%)	323.39 (+32.9%)
DHO	10%	82.8	87.22	17.58	238.11
DHO + ToMe	10%	82.5 (–0.3)	87.22	13.12 (–25.4%)	308.49 (+29.6%)

E Datasets

Table 12: Overview of datasets used in our experiments, organized into three categories: **(top)** standard classification datasets, **(middle)** ImageNet, and **(bottom)** ImageNet variants for out-of-distribution (OOD) evaluation. For few-shot semi-supervised learning experiments, we report both the absolute number of labeled samples and their percentage relative to the full training set.

Dataset	# Classes	# Train	# Val	# Test	# Labeled (1-shot)	# Labeled (16-shot)
Fine-grained 10 Datasets						
Caltech101 [30]	100	4,128	1,649	2,465	100 (2.42%)	1,600 (38.76%)
OxfordPets [31]	37	2,944	736	3,669	37 (1.26%)	592 (20.11%)
StanfordCars [32]	196	6,509	1,635	8,041	196 (3.01%)	3,136 (48.18%)
Flowers102 [33]	102	4,093	1,633	2,463	102 (2.49%)	1,632 (39.87%)
Food101 [34]	101	50,500	20,200	30,300	101 (0.20%)	1,616 (3.20%)
FGVCAircraft [35]	100	3,334	3,333	3,333	100 (3.00%)	1,600 (48.00%)
SUN397 [36]	397	15,880	3,970	19,850	397 (2.50%)	6,352 (40.00%)
DTD [37]	47	2,820	1,128	1,692	47 (1.67%)	752 (26.67%)
EuroSAT [38]	10	13,500	5,400	8,100	10 (0.07%)	160 (1.19%)
UCF101 [39]	101	7,639	1,898	3,783	101 (1.32%)	1,616 (21.15%)
Coarse-grained Dataset						
ImageNet [26]	1,000	1.28M	-	50,000	1,000 (0.08%)	16,000 (1.25%)
ImageNet OOD Variants						
ImageNet-V2 [40]	1,000	-	-	10,000	-	-
ImageNet-Sketch [41]	1,000	-	-	50,889	-	-
ImageNet-A [43]	200	-	-	7,500	-	-
ImageNet-R [42]	200	-	-	30,000	-	-

We evaluated our approach on 11 diverse datasets, with ImageNet [26] serving as our primary benchmark. The datasets span general object recognition [26, 30], fine-grained classification tasks (vehicles [32, 35], natural entities [33, 31, 34]), and specialized domains (scenes [36], textures [37], remote sensing [38], and human actions [39]). Additionally, we conduct experiments on four out-of-distribution test sets to further validate the model's generalization capabilities. To assess our model's robustness to distribution shifts, we evaluate it on several challenging variants of ImageNet: ImageNet-v2 [40], ImageNet-Sketch [41], ImageNet-A [43], and ImageNet-R [42].

We summarize the overview of datasets used in Tab. 12, these datasets exhibit diversity in their characteristics, with varying numbers of classes and samples per dataset. This diversity enabled us to thoroughly validate our method across different few-shot semi-supervised learning scenarios by systematically varying the ratios between labeled and unlabeled samples.

F Additional Experiments

F.1 Experiments on Motivation

We observe that existing KD methods, such as logit distillation [15], face gradient conflict in **few-/low-shot semi-supervised settings**. To investigate whether gradient conflicts are severe or only present in few-/low-shot semi-supervised settings, we conduct additional experiments by using full labels on ImageNet and 8 datasets using ResNet-18/ResNet-50 as student/teacher. First of all, we observe **no gradient conflicts at all in any fully supervised settings**, *i.e.*, $\texttt{CosSim}(\nabla_\theta \mathcal{L}_{\text{CE}}, \nabla_\theta \mathcal{L}_{\text{KD}}) > 0$ throughout training. This is because the distillation signal from a fully supervised teacher aligns well with the labeled data when the teacher is trained on the same dataset as the student. However, in few-/low-shot semi-supervised settings, this alignment often breaks down, as shown in Fig. 3. Tab. 13 also reports results on ImageNet using a ResNet-18 student and a ResNet-50 teacher. With fully labeled data, the gap is significantly reduced (0.4), while `DHO` outperforms CE+KD (logit) in few-shot settings, validating that `DHO` is specifically tailored to few-/low-shot semi-supervised settings.

Table 13: Results on ImageNet using a ResNet-18 student and a ResNet-50 teacher. 100% denotes using the fully labeled dataset with a fully supervised teacher.

Method	1-shot	2-shot	4-shot	8-shot	16-shot	100%
CE+KD (logit)	50.5	50.6	50.6	51.0	51.2	66.3
`DHO`	**51.8**	**52.4**	**52.6**	**53.3**	**54.5**	**66.7**

F.2 Experiments on MobileNet

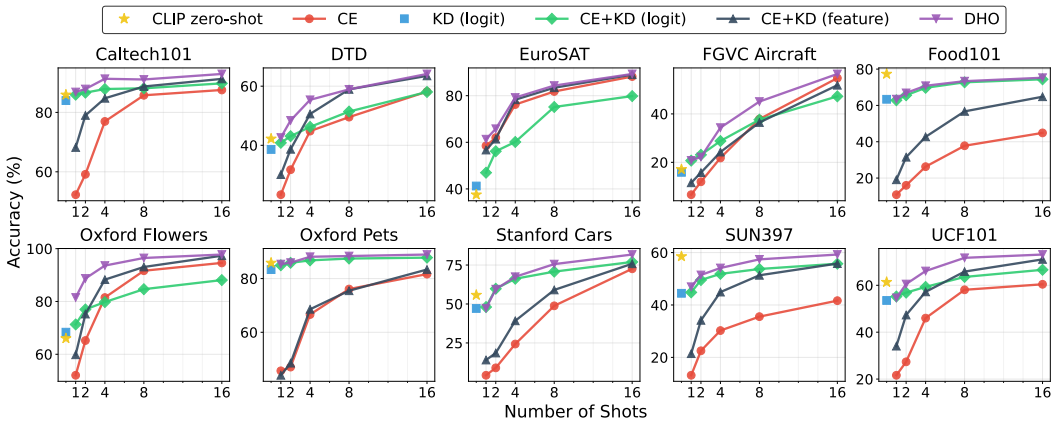

Figure 10: Results on **10 datasets** under few-shot semi-supervision using **MobileNetV2** with **zero-shot teacher** [1].

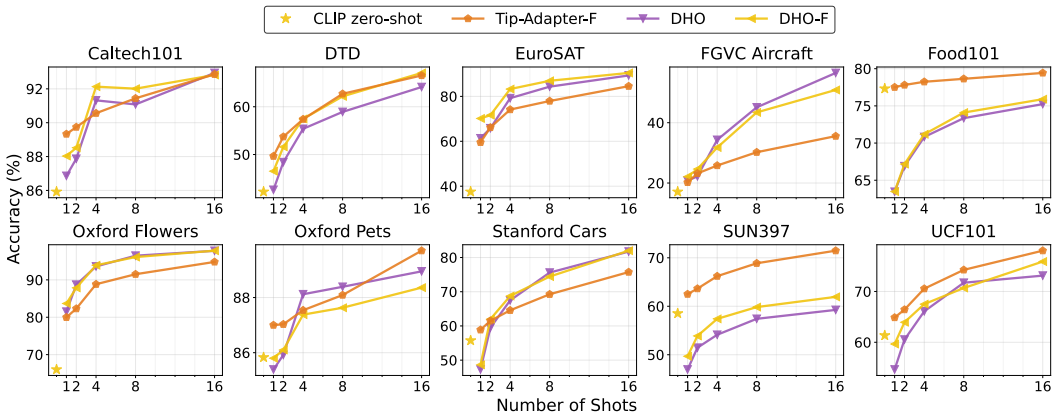

Figure 11: Results on **10 datasets** using **MobileNetV2** with either zero- or **few-shot teacher** [5].

To demonstrate the versatility of our `DHO` approach beyond ResNet [119] and ViT [120] architectures, we extended our experiments to the MobileNetV2 [51], which is specifically designed for real-world applications with compact models. We maintained identical experimental settings as described in Appendix D.2, using MobileNetV2 as the student model while distilling from CLIP ResNet50.

As illustrated in Fig. 10, our `DHO` consistently outperforms all single-head baseline methods, demonstrating its effectiveness on lightweight model architectures along with ResNet18. Furthermore, Fig. 11 reveals patterns similar to our ResNet18 experiments regarding few-shot integration. Our method successfully incorporates few-shot teacher knowledge, although we observe that the few-shot teacher does not consistently yield improvements over the zero-shot teacher. Notably, our distilled MobileNetV2 model sometimes achieves superior performance to the zero- and few-shot teachers (ResNet-50) despite having significantly fewer parameters. This pattern of outperforming both zero-shot and few-shot teachers mirrors the observations from our main experiments, further validating the effectiveness of our approach across different architectural families.

F.3 Additional Results with Gradient Surgery Methods

In this section, we present experimental results comparing `DHO` with PCGrad [52], a well-known gradient surgery method from the multi-task learning literature. PCGrad addresses gradient conflicts by projecting conflicting gradients to resolve conflicts post hoc.

We compare DHO with PCGrad using 8 of the 10 datasets (without Food101, Sun397) with ResNet-18 as the student model. The results are shown in Tab. 14.

Table 14: Performance comparison of PCGrad with DHO on 8 datasets (average) using ResNet-18.

Method	1-shot	2-shot	4-shot	8-shot	16-shot
CE+KD (logit)	56.2	58.9	61.7	65.8	69.8
PCGrad	56.8	60.3	62.1	67.5	71.6
DHO	**60.1**	**63.7**	**70.2**	**75.4**	**79.1**

We observe that PCGrad improves over CE+KD (logit), but still underperforms compared to DHO. Moreover, PCGrad incurs additional memory and computational costs ($\mathcal{O}(|\theta|)$) due to gradient projection and storage, whereas DHO remains lightweight ($\mathcal{O}(d \times C)$) and simple to implement.

The superior performance of DHO can be attributed to its approach of avoiding gradient conflicts at the source by isolating the learning dynamics of each objective via dual heads, rather than resolving conflicts after they arise. Additionally, DHO provides dynamic interpolation capability at inference time between supervised and distillation signals, which standard gradient-based methods do not offer.

F.4 Additional Results with KD Methods

In this section, we present experimental results combining our DHO method with additional distillation approaches: Decoupled Knowledge Distillation (DKD) [53] and Wasserstein Knowledge Distillation (WKD) [54]. Both DKD and WKD are orthogonal to DHO since their losses can be applied directly to the outputs of h_{CE}. DKD decouples the target class from non-target classes, and WKD computes a kernel matrix within each class. Both approaches require ground-truth labels, restricting their use to the small labeled dataset $\mathcal{D}^{(l)}$.

Despite these limitations, we conducted experiments on ImageNet (Tab. 15) and on 8 of the 10 datasets (without Food101, Sun397) as shown in Tab. 16. The results demonstrate that DHO substantially improves the performance of these state-of-the-art KD methods. Notably, WKD+DHO consistently outperforms DHO alone in most settings, demonstrating the extensibility of DHO due to its simplicity.

Table 15: Performance comparison of DKD and WKD with and without DHO on ImageNet. Numbers in parentheses show improvement over the base method.

Method	1-shot	2-shot	4-shot	8-shot	16-shot
DHO	51.8	52.4	52.6	53.3	54.5
DKD	8.9	15.0	20.6	28.2	34.9
DKD+DHO	47.1 (+38.2)	44.3 (+29.3)	40.9 (+20.3)	40.2 (+12.0)	42.4 (+7.5)
WKD	11.0	17.0	17.0	27.7	34.6
WKD+DHO	**53.2** (+42.2)	**53.3** (+36.3)	**53.3** (+36.3)	**54.0** (+26.3)	**54.8** (+20.2)

Table 16: Performance comparison of DKD and WKD with and without DHO on 8 datasets (average). Numbers in parentheses show improvement over the base method.

Method	1-shot	2-shot	4-shot	8-shot	16-shot
DHO	**60.1**	63.7	70.2	75.4	79.1
DKD	28.7	41.3	55.4	66.3	73.1
DKD+DHO	46.2 (+17.5)	51.2 (+9.9)	61.8 (+6.4)	69.8 (+3.5)	74.9 (+1.8)
WKD	30.0	38.2	53.3	66.1	73.6
WKD+DHO	59.6 (+29.6)	**64.5** (+26.3)	**71.2** (+17.9)	**75.7** (+9.6)	**79.6** (+6.0)

F.5 Additional Results with Adaptive Weighting

To further explore the potential of DHO, we implement an entropy-based adaptive weighting mechanism. Let the entropy of a probability vector $p \in \Delta^{C-1}$ be $H(p) = -\sum_{c=1}^{C} p_c \log p_c$. We compute

the adaptive weight α as:

$$\alpha = \frac{\exp(-H(\hat{p}_{\text{CE}}))}{\exp(-H(\hat{p}_{\text{CE}})) + \exp(-H(\hat{p}_{\text{KD}}))} \tag{54}$$

where $\hat{p}_{\text{CE}}$ and $\hat{p}_{\text{KD}}$ are the output probability vectors from h_{CE} and h_{KD}, respectively. The final prediction is then computed as $\hat{p} = \alpha \cdot \hat{p}_{\text{CE}} + (1 - \alpha) \cdot \hat{p}_{\text{KD}}$.

The intuition behind this approach is that lower entropy (higher confidence) predictions should receive higher weights in the final ensemble. When one head produces more confident predictions than the other, the adaptive weighting mechanism automatically emphasizes the more certain prediction. The results show that the entropy-based adaptive weighting method is not proved to be effective in these experiments. While the adaptive weighting does not consistently outperform the fixed interpolation approach, this is likely due to modern neural networks producing overconfident predictions, making entropy an unreliable proxy for uncertainty without proper calibration. However, we believe adaptive weighting could be beneficial with well-calibrated models or alternative uncertainty measures.

F.6 Sensitivity of Single-head KD to Hyperparameters

To investigate the sensitivity of single-head KD baseline, *i.e.*, CE + KD (logit), we conduct additional experiments by varying the loss balancing hyperparameter $\lambda \in \{0.0, 0.1, \ldots, 0.9, 1.0\}$ and the temperature hyperparameter $\tau \in \{0.1, 0.3, 0.5, 1, 2, 4\}$, resulting in total 55 training runs. Fig. 12 shows the interpolated results from these 55 experiments on ImageNet using a ResNet-50 student and a CLIP ResNet-50 teacher, following the same experimental setup in Fig. 9. We observe that the single-head KD baseline is also sensitive to the hyperparameters (λ, τ), requiring intensive training to obtain the best combination. Crucially, the best performance of CE + KD (logit) is 63.0%, which is less than the performance of DHO (64.7%) with heuristically selected (α, β).

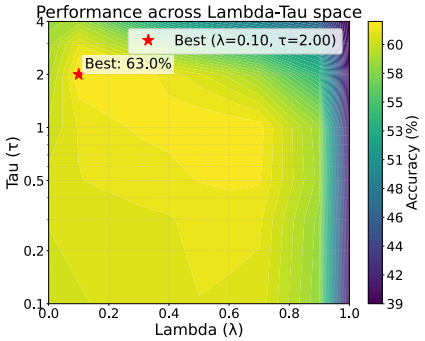

Figure 12: Grid search results for λ and τ for **CE + KD (logit)**.

Validation tuning of (α, β) is optional. We further verify that DHO does not depend on a validation split for hyperparameter selection. Tab. 17 reports DHO with a single *fixed* choice $(\alpha, \beta) = (0.5, 1)$—the value implied by Theorem 1 when $\lambda = 0.5$ and the temperatures are matched—alongside DHO with (α, β) selected on a held-out validation split, and the SHO baseline with its training-time weight $\lambda = 0.5,\ \tau = 1$. Numbers are 8-dataset averages with ResNet-18 student and CLIP ResNet-50 zero-shot teacher.

Table 17: **DHO does not require validation tuning** of (α, β): a single fixed $(0.5, 1)$ already outperforms SHO at every shot, and matches validation-searched DHO from 4-shot onward. 8-dataset averages with ResNet-18 student and CLIP ResNet-50 zero-shot teacher.

Method	1-shot	2-shot	4-shot	8-shot	16-shot
SHO (λ=0.5, τ=1)	55.9	59.4	62.1	67.4	71.2
DHO, fixed (α, β)=(0.5, 1)	57.5 (+1.6)	63.5 (+4.1)	71.6 (+9.5)	**75.3** (+7.9)	**79.2** (+8.0)
DHO, validation-searched (α, β)	**59.1**	**64.3**	**71.7**	75.2	**79.2**

The gap between fixed and searched DHO is $\leq 0.1\%$ from 4-shot onward and only 1.6% at 1-shot—i.e., validation tuning is most valuable exactly where the validation signal is the noisiest (one labeled example per class). Compared with SHO's 55-run grid search over (λ, τ) in Fig. 12 (best 63.0%), DHO with a single fixed (α, β) at 16-shot achieves 79.2% with no grid search and a single training run.

Grid search over (α, β). Fig. 9 (in the main paper) visualizes a (α, β) grid search on ImageNet with ResNet-50 under the 16-shot setting, complementing *Tab.* 17. The peak (65.5%) is at $(\alpha, \beta) \approx (0.5, 0.3)$, close to the heuristic setting (64.7%) and the fixed $(0.5, 1)$ value (Tab. 17). These

gains require no additional training, demonstrating the **efficiency of post-training hyperparameter search** for the dual-head interpolation in §2.3.

F.7 Comparison under Few-shot Teacher

In Tab. 1 and Fig. 5, we observe the improvement of DHO under few-shot teachers, *i.e.*, DHO-F. In this section, we also compare DHO-F against single-head KD baselines under few-shot teachers, *i.e.*, CE + KD (logit)-F. Specifically, we train and evaluate CE+KD (logit)-F under the same experimental setup as Fig. 5, and summarize the results in Tab. 18. We observe that the baseline improves with the few-shot teacher; however, DHO-F still outperforms the baseline even under the same few-shot teacher setting.

Table 18: Results on 10 datasets (average) using ResNet-18 with either zero- or **few-shot teachers**.

Method	0-shot	1-shot	2-shot	4-shot	8-shot	16-shot
Zero-shot teacher distillation						
CE + KD (logit)	–	55.6	58.2	61.1	64.8	68.4
DHO	–	**58.9**	**62.2**	**68.4**	**73.1**	**76.5**
Few-shot teacher distillation						
CE + KD (logit)-F	–	59.8	63.9	68.9	72.6	76.2
DHO-F	–	**61.1**	**64.7**	**69.5**	**73.6**	**76.9**
Teacher Models						
Zero-shot teacher	58.8	–	–	–	–	–
Few-shot teacher	–	65.0	67.2	70.4	73.3	76.9

F.8 Comparison to Simple 2-Stage Baseline

One possible way to mitigate gradient conflict in few-/low-shot settings with VLM teachers is to train a single-head classifier in two distinct stages. Specifically, we can (1) train a single-head classifier using only the KD loss $\mathcal{L}_{\text{KD}}$ in Eq. 2 on both the labeled dataset $\mathcal{D}^{(l)}$ and the unlabeled dataset $\mathcal{D}^{(u)}$, and then (2) fine-tune the classifier using only the CE loss $\mathcal{L}_{\text{CE}}$ in Eq. 1 on the labeled dataset $\mathcal{D}^{(l)}$. To investigate the effectiveness of DHO over this simple 2-stage baseline, we conduct additional experiments using the exact setup employed in Fig. 4 and Tab. 7 on the same 10 datasets, with the only exception that we train each stage with 100 epochs to make up a total 200 epochs for fair comparison. We also consider using a representative semi-supervised learning method, *i.e.*, **FixMatch** [10], in the second stage to prevent overfitting. For FixMatch, we follow the original hyperparameters $\tau = 0.95$ and $\lambda_u = 1$, using RandAugment and Cutout as strong augmentations. The only exception is $\mu = 7$, which we set to 1 (*i.e.*, a 1:1 ratio of labeled to unlabeled samples per mini-batch), matching the setup of DHO.

Table 19: Comparison with **simple two-stage baselines** with/without **FixMatch** [10] on 10 datasets (average) using ResNet-18 with a zero-shot teacher.

Method	FixMatch	0-shot	1-shot	2-shot	4-shot	8-shot	16-shot
KD	-	**53.9**	-	-	-	-	-
CE + KD (logit)	-	-	55.6	58.2	61.1	64.8	68.4
2-stage	-	-	53.0	53.6	58.5	63.7	68.6
2-stage	✓	-	53.5	54.6	59.8	66.7	70.5
DHO	-	-	**58.9**	**62.2**	**68.4**	**73.1**	**76.5**
Zero-shot teacher	-	58.8	-	-	-	-	-

As shown in Tab. 19, the two-stage baseline performs similarly to CE + KD (logit) when a comparatively larger number of labeled samples is available (*e.g.*, 8-/16-shot), but it significantly degrades when labeled data is scarce (*e.g.*, 1-/2-/4-shot). This result is intuitive: during the second stage of fine-tuning on the labeled data only, as the amount of labeled data decreases, the classifier becomes more prone to overfitting. As expected, applying FixMatch in the second stage improves performance by preventing overfitting. However, DHO consistently achieves the best performance, even compared to this two-stage baseline. This indicates that DHO is more efficient in practice, since it avoids splitting training into multiple stages and does not require additional fine-tuning cycles, thereby reducing engineering complexity (*e.g.*, tuning τ, λ_u, μ, and strong augmentations of FixMatch).

F.9 DHO is Orthogonal to SSL Methods

Due to the simplicity of DHO, we observe that it can be seamlessly integrated with, and improves upon, recent category-aware KD methods [53, 54], as shown in Appendix F.4. In this section, we further demonstrate that DHO is also orthogonal to semi-supervised learning (SSL) methods such as DebiasedPL [123], which leverages **debiased pseudo labels** to train a classifier. Specifically, we train h_{CE} using DebiasedPL, while h_{KD} is trained with the KD loss of DHO, *i.e.*, $\mathcal{L}_{\text{KD}}$ in Eq. 6.

We conduct experiments on ImageNet with 1% labels on ImageNet using ResNet-50 with a zero-shot VLM teacher. We use a batch size of 384, an EMA teacher model, multi-view augmentation, and 50 training epochs. Because the Google Drive link in the official codebase for the 1% ImageNet split has expired, we randomly sample a new 1% split, which may introduce minor discrepancies relative to the numbers reported in the original paper. As shown in Tab. 20, DHO improves DebiasedPL by **+1.4**, demonstrating that DHO is both simple and complementary to strong SSL approaches.

Table 20: Comparison with DebiasedPL [123].

Method	ImageNet 1%
DebiasedPL	63.9
DebiasedPL + DHO	**65.3 (+1.4)**
Zero-shot VLM teacher	60.3

Table 21: Performance of ViT-B/16 and ViT-L/14 distilled from ViT-H/14 with entropy adaptive weighting under different percentages of labeled data.

Method	ViT-B/16		ViT-L/14	
	1%	10%	1%	10%
DHO	**81.66**	**82.78**	84.59	**85.94**
DHO+Ent	**81.66**	82.65	**84.60**	85.92

F.10 Out-of-Distribution Evaluation upon fully trained model

We provide the evaluation on out-of-distribution datasets with fully trained model in Tab. 22. DHO significantly outperformed zero-shot baselines on similar-distribution variants (ImageNet-V2, ImageNet-Sketch) across both ViT-B/16 and ViT-L/14 architectures, but showed performance degradation on out-of-distribution datasets (ImageNet-R, ImageNet-A), suggesting increased distribution overfitting from full model training. Interestingly, ViT-B/16 models distilled from ViT-L/14 handled shifted distributions better than those taught by the larger ViT-H/14 DFN [47], despite the latter's superior performance on shifted distributions such as ImageNet-R and ImageNet-A. We attribute this to the shared training background between ViT-B/16 and ViT-L/14 in the CLIP framework [1], which appears to better preserve generalization capabilities during the adaptation. This points to an important insight: **our method works best on out-of-distributions when the teacher and student models share similar training distributions**, suggesting that successful knowledge distillation also depends on the alignment between teacher and student than just the teacher's raw capabilities.

Table 22: Accuracy(%) of DHO with full training model on the ImageNet distribution-shifted variants.

Student Model	**Params (M)**	**Labeled Data**	**Teacher Model**	**Val**	**V2**	**Sketch**	**R**	**A**
ViT-B/16 Student								
ViT-B/16 [1]	86M	zero-shot	-	66.7	60.8	46.2	**74.0**	**47.0**
ViT-B/16	86M	1%	ViT-L/14	78.7	70.1	48.0	70.9	41.1
ViT-B/16	86M	10%	ViT-L/14	80.8	71.3	47.4	71.7	41.4
ViT-B/16	86M	1%	ViT-H/14	81.6	72.6	50.6	65.5	35.6
ViT-B/16	86M	10%	ViT-H/14	**82.8**	**73.6**	**50.7**	67.7	37.8
ViT-L/14 Student								
ViT-L/14 [1]	304M	zero-shot	-	75.3	68.3	59.2	**86.5**	**74.6**
ViT-L/14	304M	1%	ViT-H/14	84.6	77.0	61.5	79.9	60.8
ViT-L/14	304M	10%	ViT-H/14	**85.9**	**77.8**	**61.7**	82.8	64.4
Zero-shot VLM								
ViT-H [47]	632M	zero-shot	-	83.6	77.2	71.7	92.3	77.4

F.11 Repeated-Run Reliability

We evaluate the variability of DHO in the 16-shot semi-supervised setting on eight datasets using three seeds (1, 2, and 3). The student is an ImageNet-pretrained ResNet-18 with linear CE and KD heads, and the teacher is zero-shot OpenAI CLIP RN50. Each seed controls both labeled-example sampling and training randomness; the test split is fixed. These measurements therefore capture both sources of variability.

Protocol. For this supplementary study, we train for 200 epochs using AdamW with learning rate 10^{-3}, weight decay 10^{-2}, batch size 64 per labeled/unlabeled stream, distillation temperature 2, and loss weight $\lambda = 0.5$. The checkpoint and inference parameters are selected using validation accuracy: $\alpha \in \{0, 0.05, \ldots, 1\}$ and $\beta \in \{0.5, 0.75, 1, 1.25, 1.5, 2, 4\}$. The selected model is then evaluated on the test set. Each run uses one RTX 5090; the 24 runs require 4.38 GPU-hours of recorded runtime.

Table 23: Three-seed test accuracy (%) of `DHO` under 16-shot semi-supervision. SD is the sample standard deviation across seeds; both training and labeled-split seeds vary.

Dataset	Seed 1	Seed 2	Seed 3	Mean $\pm$ SD
Caltech-101	92.54	92.54	93.51	92.86 ± 0.56
DTD	61.23	60.40	61.23	60.95 ± 0.48
EuroSAT	88.26	89.15	90.44	89.28 ± 1.10
FGVC Aircraft	54.16	52.78	52.93	53.29 ± 0.76
Oxford Flowers	97.48	97.24	97.85	97.52 ± 0.31
Oxford Pets	88.20	88.14	88.83	88.39 ± 0.38
Stanford Cars	81.21	81.54	81.51	81.42 ± 0.18
UCF-101	73.25	73.41	72.32	72.99 ± 0.59
Macro average	79.54	79.40	79.83	79.59 ± 0.22

Aggregation and scope. We report arithmetic means and sample standard deviations (SDs, with denominator $n - 1$) across the three runs. For the macro row, we first average accuracy equally across the eight datasets within each seed, then compute the mean and SD of those three macro averages. The resulting accuracy is $79.59 \pm 0.22\%$. These SDs describe run-to-run variation, not confidence intervals or significance tests. This study covers the eight-dataset protocol above; it does not provide uncertainty estimates for the single-run ImageNet 1%/10% results.

F.12 Additional VLM Teachers

We evaluate `DHO` with two additional teacher checkpoints: OpenCLIP RN50 pretrained on YFCC15M (`RN50-quickgelu/yfcc15m`) and SigLIP ViT-B/16 pretrained on WebLI (`ViT-B-16-SigLIP/webli`) [46]. OpenCLIP provides another CLIP pretraining source, whereas SigLIP extends the evaluation beyond the CLIP family.

Controlled comparison. We use the eight datasets, ImageNet-pretrained ResNet-18 student, 16-shot supervision, and 200-epoch protocol in Appendix F.11. Each CE+KD/DHO pair uses the same teacher checkpoint, labeled and test splits, optimizer settings, and training/split seed (1). CE+KD applies both losses to one head; `DHO` uses separate linear heads and validation-selected interpolation. Checkpoints are selected by validation accuracy for both methods. The table reports one run per method and dataset, with equal-weight macro averages; no three-seed mean is mixed into this comparison. The 32 runs require 7.44 GPU-hours of recorded runtime, with one RTX 5090 per run.

Results. `DHO` improves over CE+KD on all eight datasets for both teachers. The macro gains are 31.14 percentage points with OpenCLIP and 6.88 points with SigLIP. These controlled comparisons support applicability to the additional checkpoints tested; they do not establish statistical significance or performance across all VLM teachers.

G Additional Analysis

G.1 Experiments on Feature Quality

In this section, we provide more insights into the impact of gradient conflict on **feature learning**. To measure feature quality, we introduce the following definition:

Definition 1 (Feature quality)**.** *Let $Z \in \mathbb{R}^{N \times d}$ be the feature matrix of X (i.e., $Z = [g(x_1), \ldots, g(x_N)]^\top$) and $Y \in \mathbb{R}^{N \times C}$ the one-hot encoded target matrix. For $\lambda \geq 0$, we define*

Table 24: Test accuracy (%) with additional VLM teachers under 16-shot semi-supervision. Both methods use seed 1 and matched teacher/data settings within each comparison.

	OpenCLIP RN50		SigLIP ViT-B/16	
Dataset	CE+KD	DHO	CE+KD	DHO
Caltech-101	65.92	**87.79**	93.83	**94.16**
DTD	34.46	**58.63**	58.39	**66.37**
EuroSAT	33.49	**87.37**	71.64	**91.73**
FGVC Aircraft	17.22	**45.87**	47.76	**61.45**
Oxford Flowers	69.71	**96.22**	94.64	**98.54**
Oxford Pets	49.66	**72.23**	88.96	**90.19**
Stanford Cars	19.66	**52.20**	86.83	**89.07**
UCF-101	23.71	**62.68**	68.78	**74.41**
Macro average	39.23	**70.37**	76.36	**83.24**

*the **feature quality** of Z as the expected risk (↓) of ridge regression*:

$$\mathcal{L} = \mathbb{E}_{z,y}\left[\|y - z^\top \hat{W}_\lambda\|_2^2\right], \quad \text{where } \hat{W}_\lambda = (Z^\top Z + \lambda N I_d)^{-1} Z^\top Y.$$

We adopt the expected risk of ridge regression (↓) as the measure of feature quality for simplicity (while linear probing is widely used and we also use it in Tab. 5).

Let $g_1 := \nabla_\theta \mathcal{L}_{\text{CE}}(\theta)$, $g_2 := \nabla_\theta \mathcal{L}_{\text{KD}}(\theta)$, $g := \nabla_\theta \mathcal{L}(\theta)$, and $G = \frac{1}{2}g_1 + \frac{1}{2}g_2$. We assume:

Assumption 3 (Gradient cone)**.**

$$g = \beta_1 g_1 + \beta_2 g_2 + r_\perp, \qquad \beta_1, \beta_2 \geq 0, \qquad r_\perp^\top g_1 = r_\perp^\top g_2 = 0.$$

This means that the probe gradient lies within the non-negative cone spanned by g_1 and g_2, up to an orthogonal residual $r_\perp$—that is, both g_1 and g_2 **contribute positively to improving feature quality**.

Taking a vanilla SGD step ($\theta - \eta G$, $\eta > 0$), with first-order Taylor expansion:

$$\mathcal{L}(\theta - \eta G) = \mathcal{L}(\theta) - \eta\, g^\top G + \mathcal{O}(\eta^2) \approx \mathcal{L}(\theta) - \eta g^\top G. \tag{55}$$

We neglect the $\mathcal{O}(\eta^2)$ term (*e.g.*, $\eta < 10^{-3}$ makes it negligible). Under the Assumption 3:

$$\begin{aligned} g^\top G &= \tfrac{1}{2}\, g^\top (g_1 + g_2) \\ &= \tfrac{1}{2}\, (\beta_1 g_1 + \beta_2 g_2 + r_\perp)^\top (g_1 + g_2) \\ &= \tfrac{1}{2}\big(\beta_1 \|g_1\|_2^2 + \beta_2 \|g_2\|_2^2\big) + \tfrac{1}{2}(\beta_1 + \beta_2)\, g_1^\top g_2. \end{aligned} \tag{56}$$

If gradients are aligned ($g_1^\top g_2 > 0$), then Eq. 56 is positive, accelerating the decrease in Eq. 55, and thus **the expected probe risk decreases more**.

To empirically validate this, we report $\hat{\mathcal{L}} := \frac{1}{M}\sum_{i=1}^{M} \|y_i - z_i^\top \hat{W}_\lambda\|_2^2$ on the test set over 200 epochs with the same setup as Fig. 4:

Table 25: **Feature quality** throughout training, computed using Definition 1.

10 Datasets (Average)	1	11	21	31	41	51	61	71	81	91	101
CE+KD (logit)	4.974	4.664	4.638	4.608	4.601	4.598	4.595	4.595	4.594	4.595	4.595
`DHO`	4.672	4.435	4.413	4.391	4.390	4.390	4.390	4.393	4.393	4.394	4.395
10 Datasets (Average)	111	121	131	141	151	161	171	181	191		
CE+KD (logit)	4.595	4.596	4.596	4.596	4.597	4.598	4.598	4.599	4.599		
`DHO`	4.396	4.396	4.398	4.399	4.400	4.400	4.400	4.401	4.401		

Tab. 25 shows that across all epochs, $\hat{\mathcal{L}}$ in Definition 1 of `DHO` is lower than that of CE+KD (logit), **confirming our analysis on feature quality**.

G.2 Per-Layer Gradient Conflict Analysis

Fig. 3 reports the gradient cosine similarity computed over the entire flattened parameter vector $\nabla_\theta \mathcal{L}$. A natural concern is that this single high-dimensional inner product may average out per-layer structure—for instance, conflict could be concentrated only in the classifier head while early backbone layers remain neutral, which would weaken the propagation argument in Proposition 2. We address this concern by decomposing the cosine similarity per ResNet stage.

Setup. We train SHO (CE+KD logit, single head) and `DHO` from scratch on 8 datasets (Caltech-101, DTD, EuroSAT, FGVC-Aircraft, Oxford Flowers, Oxford Pets, Stanford Cars, UCF101) under the 16-shot setting using a ResNet-18 student and a CLIP ResNet-50 zero-shot teacher, with the same hyperparameters as Fig. 4. During training, every 20 epochs we record $\nabla_\theta \mathcal{L}_{\mathrm{CE}}$ and $\nabla_\theta \mathcal{L}_{\mathrm{KD}}$ separately and compute their cosine similarity restricted to the parameters of each ResNet stage `layer1`–`layer4` (and, for SHO, the shared classifier head). For `DHO`, the classifier-head similarity is undefined by construction: h_{CE} receives only $\nabla \mathcal{L}_{\mathrm{CE}}$ and h_{KD} receives only $\nabla \mathcal{L}_{\mathrm{KD}}$, so the two never meet at the head—this architectural decoupling is precisely the design intent.

Conflict in SHO is uniform across the backbone and deepens with depth. Tab. 26 reports the 8-dataset average cosine similarity at the final epoch. SHO exhibits negative alignment at *every* backbone stage and the magnitude grows monotonically with depth (`layer1`: -0.39 $\rightarrow$ `layer4`: -0.45, classifier: -0.76). This is the strongest empirical confirmation of Proposition 2: the conflict at the classifier head propagates backward through the Jacobian and registers at every depth, not only near the head. `DHO`, in contrast, maintains near-zero / mildly positive alignment uniformly across `layer1`–`layer4`, with the gap to SHO exceeding $+0.46$ at every stage.

Table 26: **Per-layer gradient cosine similarity** between $\nabla_\theta \mathcal{L}_{\mathrm{CE}}$ and $\nabla_\theta \mathcal{L}_{\mathrm{KD}}$ at the final epoch, averaged over 8 datasets (16-shot, ResNet-18 student, CLIP ResNet-50 zero-shot teacher). SHO conflict is uniform across the backbone and deepens monotonically with depth; `DHO` maintains near-zero / mildly positive alignment at every stage. The classifier-head similarity for `DHO` is undefined by construction (separate heads receive disjoint gradients).

Layer	SHO	DHO	Δ (DHO $-$ SHO)
`layer1` (early)	-0.3924	$+0.0738$	$+0.4662$
`layer2`	-0.3931	$+0.0695$	$+0.4626$
`layer3`	-0.4048	$+0.0642$	$+0.4690$
`layer4` (deepest)	-0.4526	$+0.0459$	$+0.4985$
classifier head	-0.7570	–	–

Per-dataset breakdown at `layer4`. Tab. 27 reports the per-dataset cosine similarity at `layer4` (the deepest backbone stage, where conflict is strongest). SHO shows negative alignment on *every* dataset, and `DHO` resolves the conflict on *every* dataset. Notably, the datasets with the most severe SHO conflict (EuroSAT: -0.80, DTD: -0.67, FGVC-Aircraft: -0.62) are the same datasets on which `DHO` achieves the largest improvements over SHO in Fig. 4, suggesting that the per-layer alignment metric tracks the magnitude of the realized accuracy gain.

Implication for the empirical claim. The per-layer decomposition makes the empirical statement in §2.3—that `DHO` maintains positive gradient alignment throughout training—precise and robust. It is not an artifact of averaging a high-dimensional vector: positivity (or, conservatively, non-negativity) holds at every backbone stage and on every dataset evaluated. Conversely, the corresponding statement for SHO—negative alignment forced by the PSD form $A = J^\top W^\top W J$ in Proposition 2—is realized at every layer rather than only at the classifier head, supporting the propagation argument.

G.3 Additional Analysis on Non-Linear Head Design

To further investigate the architectural advantages of dual head optimization, we conducted experiments with non-linear head designs, replacing the linear heads used in our main experiments. We design a non-linear classifier with a sequence of layers: an initial linear projection layer, followed

Table 27: **Per-dataset gradient cosine similarity at `layer4`** (16-shot, ResNet-18 student, CLIP ResNet-50 zero-shot teacher). `DHO` resolves conflict on every dataset; the datasets with the strongest SHO conflict (EuroSAT, DTD, FGVC-Aircraft) coincide with the largest DHO accuracy gains in Fig. 4.

Dataset	SHO	DHO	Δ (DHO − SHO)
EuroSAT	−0.8035	+0.0040	+0.8075
DTD	−0.6682	+0.0615	+0.7297
FGVC-Aircraft	−0.6213	+0.0223	+0.6437
Oxford Flowers	−0.4542	+0.0572	+0.5114
UCF101	−0.3752	+0.0372	+0.4124
Stanford Cars	−0.2728	+0.0500	+0.3228
Oxford Pets	−0.2434	+0.0672	+0.3106
Caltech-101	−0.1822	+0.0772	+0.2594
Average	**−0.4526**	**+0.0459**	**+0.4985**

by layer normalization [124], GELU activation [125], dropout [126], and a final linear classification layer. We compared `DHO` with three non-linear configurations; **`DHO+NL-Head-CE`**: non-linear CE head, **`DHO+NL-Head-KD`**: non-linear KD head, and **`DHO+NL-Head-CE+KD`**: non-linear both CE and KD heads. All experiments followed the few-shot semi-supervised setting detailed in Appendix D.2.

Table 28: Results of different **non-linear head configurations** on **11 datasets** including **ImageNet** under few-shot semi-supervision using **ResNet-18** with **zero-shot teacher** [1]. We report averaged accuracy for 10 visual recognition datasets except the ImageNet.

	ImageNet					Average of 10 tasks				
Configuration	1-shot	2-shot	4-shot	8-shot	16-shot	1-shot	2-shot	4-shot	8-shot	16-shot
`DHO` (base)	61.7	62.2	62.6	63.8	65.1	58.9	**62.2**	68.4	**73.1**	**76.5**
`DHO`+NL-Head-CE	61.7	61.9	62.2	63.1	64.8	**59.3**	62.1	68.0	72.7	76.2
`DHO`+NL-Head-KD	**62.1**	**62.6**	**62.9**	**64.0**	**65.9**	58.3	62.1	67.8	72.4	**76.5**
`DHO`+NL-Head-CE+KD	62.0	62.3	62.6	63.8	65.4	59.1	**62.2**	**68.6**	72.4	**76.5**

Performance Analysis. Our experiments revealed key findings regarding head architecture (Tab. 28, Fig. 13). On ImageNet, non-linear KD heads consistently outperformed linear ones, suggesting complex architectures better capture teacher predictions. Conversely, non-linear CE heads degraded performance, likely due to overfitting on limited labeled data. While dual non-linear heads outperformed fully linear configurations, they were less effective than non-linearity in the KD head alone.

On the other 10 datasets, optimal configurations varied considerably with no consistently superior approach. This highlights that non-linear transformation effectiveness depends strongly on dataset characteristics and head functionality. Given comparable performance but superior computational efficiency, we adopted linear head architectures for all subsequent experiments.

Table 29: Results on **dual-heads interpolation strategy** of different **non-linear head configurations** on **ImageNet** under 16-shots semi-supervised setting.

Configuration	CE Head	KD Head	Combined
`DHO` (base)	60.64	61.55	65.37
`DHO`+NL-Head-CE	60.18	61.39	64.91
`DHO`+NL-Head-KD	60.95	61.76	**65.97**
`DHO`+NL-Head-CE+KD	**61.66**	**61.81**	65.59

Head Decomposition Analysis. Analysis under the 16-shot semi-supervised setting revealed complex relationships between architectural choices and head-wise performance as shown in Tab. 29. Non-linear CE branches decreased CE head performance from 60.64% to 60.18% despite increased parameters. Conversely, non-linear KD heads improved both heads: CE accuracy increased to 60.95% and KD prediction to 61.76%. However, dual non-linear heads reduced combined performance from 65.97% to 65.59%, suggesting head specialization may compromise joint feature representation. These findings highlight the complex interplay between architectural decisions and multi-head learning dynamics.

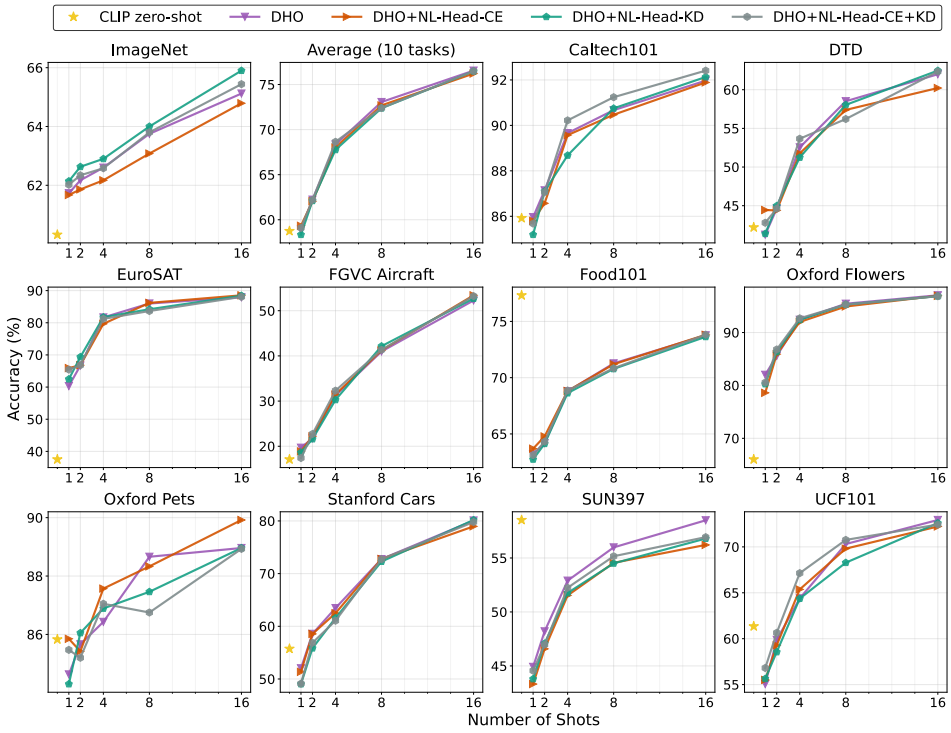

Figure 13: Results of different **non-linear head configurations** on **11 datasets** including **ImageNet** under few-shot semi-supervision using **ResNet-18** with **zero-shot teacher** [1].

G.3.1 Regularization of the Nonlinear CE Head

To examine whether the degradation of nonlinear CE heads depends on regularization, we compare the linear head, the default MLP, and five MLP configurations with stronger dropout and/or CE-head weight decay. The KD head remains linear. We use the eight-dataset, 16-shot ResNet-18/OpenAI CLIP RN50 protocol in Appendix F.11, with training and split seed 1 for every configuration. All other training settings are fixed; the 56 configuration–dataset runs require 10.07 GPU-hours of recorded runtime, including the linear baseline also used in Appendix F.11.

Validation and reporting. For each configuration and dataset, we select the checkpoint and interpolation parameters using validation accuracy, then evaluate that model on the test set. Tab. 30 reports the equal-weight macro validation and test accuracies of all seven configurations. The validation column uses the recorded combined-head accuracy at the selected checkpoint. We retain all rows as an exploratory comparison; the row with the highest test accuracy is not a validation-selected architectural recommendation.

Table 30: Nonlinear CE-head regularization under 16-shot semi-supervision. Validation and test accuracies (%) are macro averages over eight datasets, using one seed per configuration. Bold marks the highest value in each accuracy column.

CE head	Dropout	CE-head WD	Validation	Test
Linear	—	0.01	82.54	79.54
MLP (default)	0.5	0.01	**82.83**	79.30
MLP	0.7	0.01	82.73	79.07
MLP	0.5	0.05	82.38	78.97
MLP	0.5	0.10	82.48	79.06
MLP	0.7	0.05	82.76	79.10
MLP	0.7	0.10	82.74	**79.71**

Regularization sensitivity. The default MLP has the highest macro validation accuracy (82.83%). The MLP with dropout 0.7 and weight decay 0.10 has the highest observed test accuracy (79.71%), exceeding the linear head (79.54%) by 0.17 percentage points. The other four stronger regularizers do not exceed the linear head. Thus, the results indicate sensitivity to regularization, rather than general superiority of nonlinear heads. This is a single-seed study and does not establish statistical significance for the small observed difference.

G.4 Additional Dual-Head Analysis

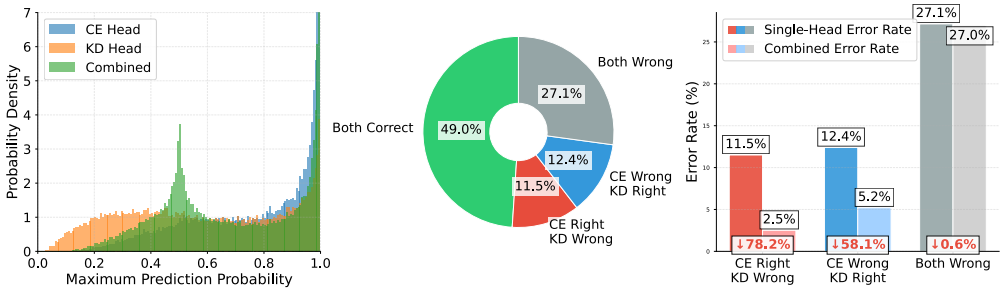

Figure 14: Analysis of DHO on the ImageNet under 16-shot semi-supervised setting. **(Left)** Maximum probability distributions for predictions from CE head, KD head, and their combined output. **(Middle)** Prediction agreement diagram analysis, categorizing cases where both heads are correct, only one head is correct, and both heads are incorrect. **(Right)** Error reduction analysis comparing single-head failure cases against improvements achieved through combined predictions.

In this section, we further analyze the prediction behavior of DHO. As shown in Fig. 14 (left), despite sharing feature representations, the CE head (h_{CE}), trained on labeled data, produces sharper predictions, whereas the KD head (h_{KD}), guided by teacher distillation, generates smoother distributions. Prediction agreement analysis (Fig. 14, middle) shows that the two heads agree in 76.2% of cases while complementing each other: the CE head correctly classifies 11.5% of cases where the KD head fails, and vice versa for 12.4%. Error reduction analysis (Fig. 14, right) further demonstrates that our combined approach reduces failure rates from 11.5% to 2.5% for the KD head and from 12.4% to 5.2% for the CE head, confirming the effectiveness of DHO.

We also present additional qualitative results of DHO, both on ImageNet (see Figs. 15 and 16) and on other 10 datasets (see Figs. 17 and 18).



Figure 15: Additional qualitative results on ImageNet under 16-shot semi-supervised setting.

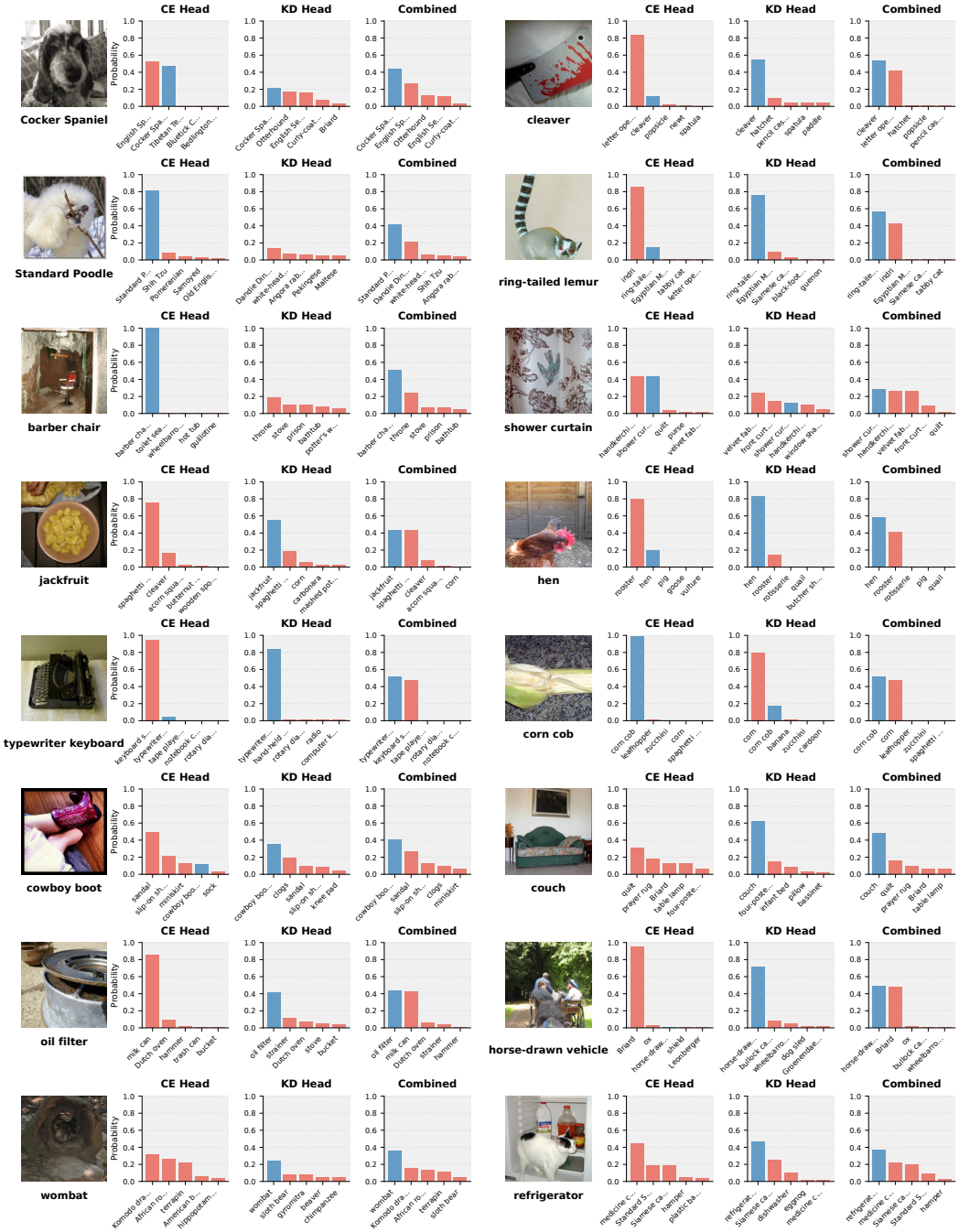

Figure 16: Additional qualitative results on ImageNet under 16-shot semi-supervised setting.

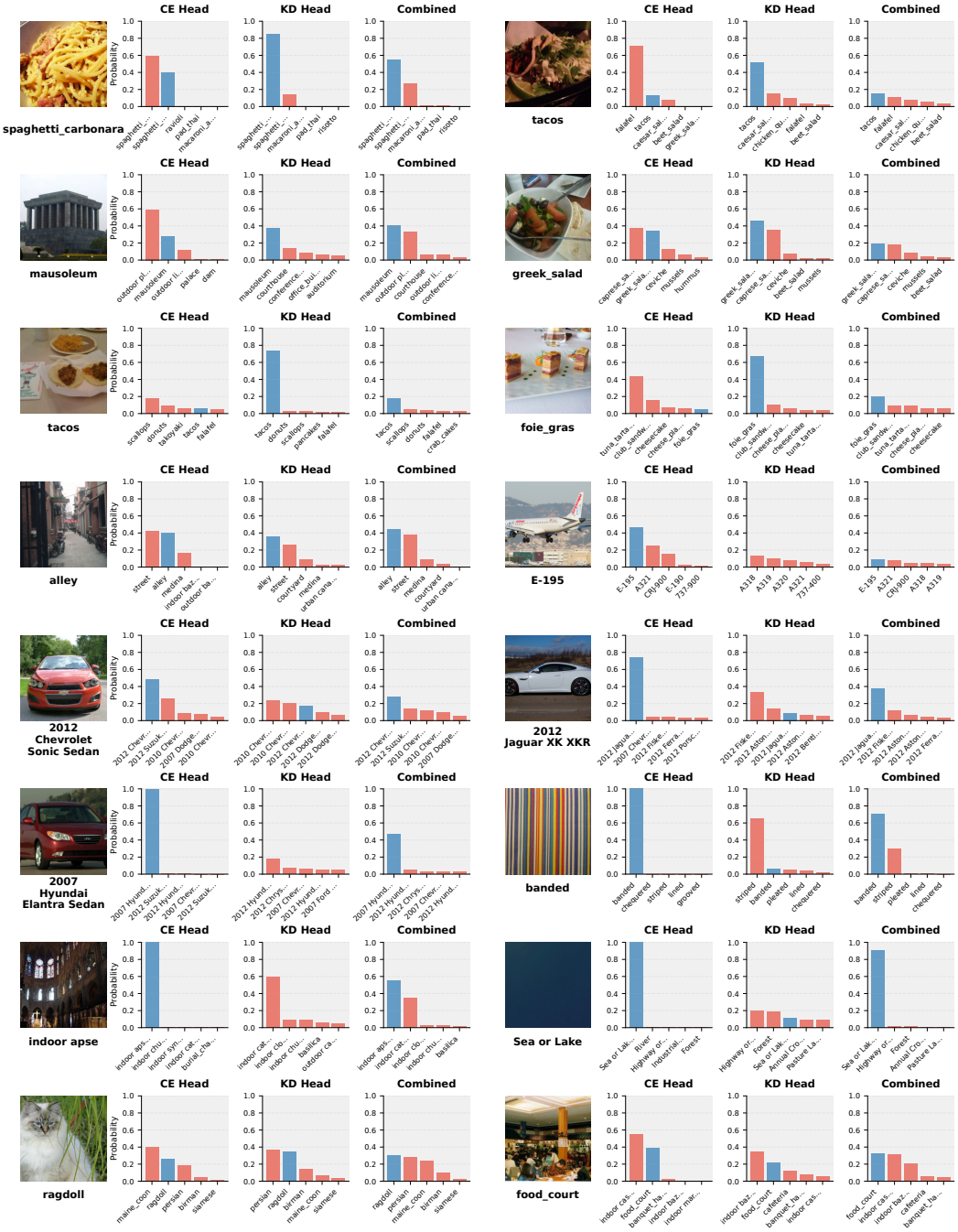

Figure 17: Additional qualitative results on other 10 datasets for models trained under 16-shot semi-supervised setting.



Figure 18: Additional qualitative results on other 10 datasets for models trained under 16-shot semi-supervised setting.

H The Use of LLMs

We used LLMs solely for light editing such as correcting grammatical errors and polishing some words. They did not contribute to research ideation, experiments, analysis, or substantive writing.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes. Our main contributions are also detailed in §1. Our gradient conflict analysis is presented in §2.2 with formal theoretical guarantees in Appendix C.1. We also provide post-hoc analysis on dual-head combination in Appendix C.2 with empirical analysis in Appendix G.4.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes, we discussed limitations in §4 and Appendix A (foundation-model scope and task scope). We also reported computational cost in training in Appendix D.2, and analyzed the computational efficiency of our method in §3.2 (Tab. 3) and Appendix D.3.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Yes, we provided the full set of assumptions and complete proofs for gradient conflict analysis (Appendix C.1, Propositions 1–3) and inference equivalence (Appendix C.2, Theorems 2–3) in Appendix C.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We used publicly available datasets and models for our experiments. Our proposed method is also simple and easy to implement, with described details in §3. We will also release our code and instructions for reproducing the results.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, once the blind review is over.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (*e.g.*, data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We specified all the training and test details (*e.g.*, data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results in §3.1 and Appendix D.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: We report three-seed means and sample standard deviations for the eight-dataset study in Appendix F.11. Both training randomness and labeled-split sampling vary across seeds; the test split is fixed. However, repeated runs do not cover all main results, including ImageNet 1%/10% comparisons. The additional-teacher comparisons and nonlinear-head regularization study each use one seed per setting. We therefore do not claim statistical significance for those results or the small reported system-level margins.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provided type of hardware, and time of execution needed to reproduce the experiments in Tab. 10 inside Appendix D.2 for experiments taking the most time. Also we provided extensive information of computational efficiency of our method in §3.2 and Appendix D.3.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: Yes, we followed the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines.

   Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Our work uses only standard academic datasets like ImageNet for improving semi-supervised learning methods. We don't use any private or personal data, and our techniques don't present obvious negative societal impacts.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our work focuses on semi-supervised learning methods using standard academic datasets and does not pose high risks for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly credit all creators of assets used in our work.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: Our work does not release any new assets such as datasets, code, or models.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our work does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Our work does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: Our work does not involve any LLMs as important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2026/LLM) for what should or should not be described.