HIERARCHICAL ENSEMBLE ADAPTER TUNING FOR FEW-SHOT ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Parameter-efficient few-shot adaptation usually trains one lightweight adapter of fixed bottleneck rank. That single rank commits adaptation to one hypothesis, which often lacks the robustness required for few-shot scenarios. In this paper, we propose Hierarchical Ensemble Adapter Tuning (HEAT), a framework that trains an ensemble of lightweight adapters with varying bottleneck ranks, each imposing a different structural inductive bias. Based on multiple adaptation hypotheses, HEAT expands the model's representational capacity by learning nested, rank-constrained features: a shared core of dominant task directions that each higher-rank adapter refines. To leverage this hierarchical structure at inference time, we introduce an uncertainty-aware aggregation strategy that dynamically weights each adapter's output according to its instance-wise predictive entropy. Analysis of the learned representations reveals that the adapters form nested subspaces where each of them sharing a common core subspace and that entropy weighting systematically exploits this structure, amplifying the most confident adapter on a per-sample basis. Across eleven few-shot benchmarks, HEAT achieves state-of-the-art accuracy all within the same parameter budget as a single CLIP-Adapter.

1 INTRODUCTION

The emergence of large-scale pre-trained models has established the "pre-train, then adapt" paradigm as the foundation of modern machine learning. Vision-language models such as CLIP (Radford et al., 2021b; Jia et al., 2021; Yuan et al., 2021; Yu et al., 2022) learn robust cross-modal representations from vast image–text corpora, providing strong zero-shot baselines across diverse recognition tasks. Yet, adapting these models to specialized downstream domains remains a significant challenge. Full fine-tuning is often computationally prohibitive and risks catastrophic forgetting (Kirkpatrick et al., 2017; Luo et al., 2025), where the model's general-purpose knowledge degrades, particularly in data-constrained few-shot regimes (Hu et al., 2022; Raffel et al., 2020; Khan et al., 2023a; Naeem et al., 2023).

To mitigate these issues, Parameter-Efficient Fine-Tuning (PEFT) has gained prominence by freezing the backbone and optimizing only a small fraction of the parameters (Houlsby et al., 2019; Jia et al., 2022; Zhang et al., 2021; Zhou et al., 2022a; Yu et al., 2023a). These techniques, specifically those employing lightweight bottleneck modules or adapters (Houlsby et al., 2019; Gao et al., 2024) strike a balance between adaptability and efficiency. However, existing adapter-based methods rely on a *single* adaptation hypothesis (e.g., one fixed bottleneck rank) to capture all task-relevant features. We argue that this reliance on a single rank adapter is inherently brittle. To be specific, a single rank commits the model to one fixed subspace of task directions, even though the directions worth adapting differ across tasks and samples.

In this paper, we propose Hierarchical Ensemble Adapter Tuning (HEAT), a framework designed to capture a multi-level feature hierarchy through structured rank variation. Instead of relying on a single bottleneck configuration, HEAT constructs an ensemble of lightweight adapters with varying ranks. Lower-rank adapters, constrained by a tighter information bottleneck, capture only the dominant task directions whereas higher-rank adapters retain this shared core and add directions that smaller ones cannot represent. This design contrasts sharply with equal-rank ensembles, which we find converge to disjoint subspaces (Sec. 5.1 and App. Fig. 6). Because all HEAT adapters share the same initialization, the rank constraint becomes the primary source of differentiation, producing

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

nested subspaces in which each smaller adapter's subspace is approximately contained within the larger ones. This structured hierarchy, which we term *hierarchical ensembling*, is what enables effective aggregation across adapters.

To effectively integrate these nested features at inference time, we introduce a parameter-free, uncertainty-aware aggregation strategy. During inference, HEAT dynamically weights each adapter's prediction based on instance-specific predictive entropy, amplifying confident adapters while down-weighting uncertain ones. Our analysis shows that this weighting naturally exploits the capacity hierarchy (Sec. 5.2): on difficult samples the high-rank adapter's sharper predictions receives greater weight, while on easier samples the weights converge toward uniformity. We attribute the effectiveness of this aggregation to the nested subspace structure (Secs. 5.1 and 5.2). Since the adapters share a common core and differ primarily in how many directions beyond this core they capture, their confidence levels on a given sample carry a consistent, comparable signal that entropy weighting can exploit.

Our contributions are summarized as follows. (1) We propose HEAT, a novel ensemble-based PEFT framework for few-shot learning that leverages structured-rank variation under shared initialization to induce a hierarchy of nested adapter subspaces. (2) We design a parameter-free, uncertainty-aware aggregation strategy that performs instance-wise entropy weighting, exploiting the hierarchical structure for robust integration of rank-specific features. (3) We conduct extensive experiments across eleven few-shot benchmarks, showing that HEAT achieves state-of-the-art performance for $K \geq 4$ shots and generalizes across backbones (ResNet-50, SigLIP-B/16) and under distribution shift. (4) Through comprehensive analysis we demonstrate that HEAT adapters construct hierarchical nested subspaces with a shared core, providing a more structured and collaborative representation than the disjoint subspaces produced by equal-rank ensembles.

2 Related Work

Adapting large-scale vision–language models (VLMs), such as CLIP (Radford et al., 2021a; Jia et al., 2021), face numerous challenges: full fine-tuning is expensive, memory-intensive, and vulnerable to overfitting or catastrophic forgetting (Kumar et al., 2022; Wortsman et al., 2022; Goyal et al., 2023; Khan et al., 2023b). These limitations have driven widespread interest in few-shot adaptation strategies that preserve the strong priors.

Parameter-Efficient Fine-Tuning (PEFT). PEFT adapts large models by freezing the backbone and training a small set of parameters (Houlsby et al., 2019; Xin et al., 2024; Khan et al., 2023b). Some recent works (Tian et al., 2024; Valipour et al., 2023; Al Rahhal et al., 2025) have introduced novel LoRA modules; however, these methods update intermediate activations and therefore require gradient flow through the backbone. In particular, adapter-based PEFT (Houlsby et al., 2019) inserts lightweight bottlenecks into the frozen backbone, enabling black-box adaptation. In the context of few-shot VLM adaptation, CLIP-Adapter (Gao et al., 2024) attaches residual adapters to CLIP features, TIP-Adapter (Zhang et al., 2021) augments with cache-based token–feature matching, and TaskRes (Yu et al., 2023b) adds residual task pathways. Despite efficiency, these methods usually adopt a *single* architectural hypothesis (e.g., fixed rank), restricting adaptation to a single subspace and increasing brittleness under domain shift or hyperparameter misalignment (App. E). CLAP (Silva-Rodriguez et al., 2024) mitigates this overfitting with a class-adaptive penalty but still lacks model-level diversity. HEAT complements adapter-based PEFT by ensembling rank-diverse adapters and using uncertainty-aware weighting for robust few-shot adaptation. Finally, work in Bai et al. (2024) is most closely aligned with our motivation. Unlike Bai et al. (2024), which mainly combines multiple adapter branches or multi-scale attention paths to improve CLIP retrieval, HEAT introduces a structured ensemble where diversity comes from systematically varying adapter bottleneck ranks. This rank hierarchy induces nested subspaces, in which higher-rank adapters refine a core shared by all. Moreover, HEAT uses instance-wise entropy-based aggregation, rather than static averaging, to dynamically emphasize the most confident adapter at inference time.

Deep Ensembles Networks. Deep ensembles are known to improve accuracy, calibration, and robustness (Lakshminarayanan et al., 2017) but are impractical for billion-parameter VLMs due to training and inference cost. Efficient variants such as BatchEnsemble (Wen et al., 2020), Fast Deep Ensembles (Fort et al., 2020), and dynamic expansion approaches (Yan et al., 2021; Wang et al., 2022) reduce overhead by sharing parameters or reusing modules, yet typically rely on randomness

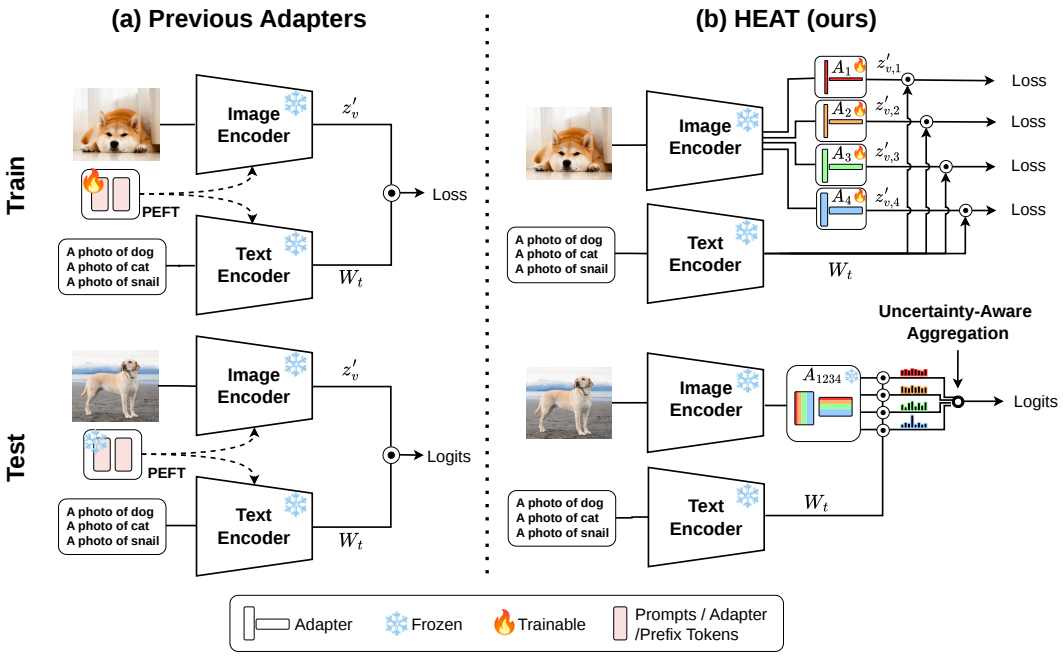

Figure 1: **Overview of Our Framework (HEAT).** (a) Previous adapters train a single PEFT module and apply it directly at test time. (b) HEAT trains multiple ReLU adapters with varying ranks to capture hierarchical knowledge, then combines their features at inference using an uncertainty-aware aggregation strategy.

or data splits, which are not tailored to frozen-backbone adaptation. HEAT performs *structured* ensembling at the adapter level: diversity arises from systematically varying bottleneck rank, yielding nested subspaces that a lightweight, uncertainty-aware aggregator can exploit.

3 Proposed Method

This section describes HEAT, a few-shot adaptation method that learns several hypotheses inside a single adapter budget (Fig. 1). Differences among the hypotheses are structurally driven by the bottleneck rank, rather than by variation in initialization. At test time the hypotheses stay fixed, and only their relative influence changes per sample through uncertainty-aware aggregation.

3.1 Preliminaries: Revisiting CLIP and CLIP-Adapter

We first revisit the foundational models upon which our work is built. CLIP (Radford et al., 2021b) jointly trains a visual encoder $f_v(\cdot)$ and a text encoder $f_t(\cdot)$ on hundreds of millions of image-text pairs with a contrastive objective. For zero-shot classification on a downstream task with C classes, text prompts (e.g., "a photo of a {class}") are encoded as classifier weights $W_t = \{W_{t,c}\}_{c=1}^{C}$, where $W_{t,c} = f_t(\text{prompt}_c)$. The visual embedding $z_v = f_v(I)$ of an image I is classified via $p(y = c|I) = \exp(\text{sim}(z_v, W_{t,c})/\tau)/\sum_{j=1}^{C} \exp(\text{sim}(z_v, W_{t,j})/\tau)$, where $\text{sim}(\cdot,\cdot)$ is the cosine similarity and τ is a temperature parameter. CLIP-Adapter (Gao et al., 2024) adapts CLIP for few-shot learning by inserting a lightweight adapter module A after the visual and/or text encoder: a two-layer bottleneck MLP with a residual connection (Fig. 1a), which produces $z'_v = (1-\alpha)z_v + \alpha \cdot \text{A}(z_v)$, where α is a residual ratio hyperparameter that balances pre-trained and task-specific knowledge. However, relying on a single, homogeneous adapter fails to capture diverse features of the dataset and can lead to overfitting which also hurts domain generalization (Tab. 2).

3.2 Training an Ensemble of Hierarchical Adapters

Our approach, depicted in Fig. 1b, diverges from prior methods by training an ensemble of adapters instead of a single one. The key insight is that different bottleneck dimensions implicitly lead adapters to learn nested representations: a shared core of dominant directions that each higher-rank adapter extends (see Sec. 5 for theoretical intuition and empirical validation.).

We construct an ensemble of N independent adapters, $\{A_1, A_2, \ldots, A_N\}$ with heterogeneous ranks. Specifically, each k-th adapter A_k is a bottleneck module defined by a down-projection matrix $W_{\text{down},k} \in \mathbb{R}^{r_k \times D}$ and an up-projection matrix $W_{\text{up},k} \in \mathbb{R}^{D \times r_k}$, where D is the feature dimension of the CLIP encoder, and r_k is the bottleneck dimension or rank such that $r_k \ll D$.

We intentionally assign a different rank r_k to each adapter in the ensemble. So that different ranks act as an inductive bias: each rank imposes a different information bottleneck, limiting how many task directions each adapter can represent. As we show in Sec. 5, the resulting subspaces are hierarchically nested, the low-rank adapter's subspace is approximately contained within the high-rank adapter's, creating structured diversity that an entropy-based aggregation can exploit.

The forward pass through an adapter A_k applied to a feature $z_v \in \mathbb{R}^D$ is $A_k(z_v) = \text{ReLU}(z_v W_{\text{down},k}^\top) W_{\text{up},k}^\top$ giving us $z_{v,k} = A_k(z_v)$. The final adapted feature $z'_{v,k}$ is then produced as $z'_{v,k} = (1-\alpha) z_v + \alpha \cdot z_{v,k}$. We keep α fixed and initialize it with $1 - (\text{Acc}_{zs}/100)$ where Acc_{zs} is the zero shot accuracy on training set.

Independent Training Protocol. Crucially, all N adapters are trained *independently* on the *entire* few-shot training dataset. This forces each adapter to learn a complete, albeit different, solution for the task, with the differences arising purely from their intrinsic inductive biases. The training for each k-th adapter A_k minimizes the cross-entropy of its own logits on the full few-shot set, $\mathcal{L}_k = \text{CE}\big(\text{sim}(z'_{v,k}, W_t), y\big)$, where y is the class label.

3.3 Uncertainty-Aware Test-Time Aggregation

During inference, we combine the predictions of the heterogeneous adapters $\{A_1, \ldots, A_N\}$. Naively averaging their logits would treat confident and uncertain predictions equally and fail to exploit the adapters' specialist nature. We therefore introduce an *Uncertainty-Aware Aggregation* mechanism that weights each adapter by its predictive confidence on each sample. Because adapters of different capacity exhibit systematically different confidence patterns: the high-capacity adapter produces lower entropy on hard samples, earning it a larger share of the weight, while on easy samples all adapters' confidences converge and the weighting approaches uniformity. Entropy weighting implicitly rebalances each adapter's contribution per sample, producing the characteristic fan-shaped weight pattern shown in Sec. 5.2.

Given a test image, the frozen CLIP encoders produce the visual feature z_v and classifier weights W_t. For each adapter A_k, we compute its modified visual feature $z'_{v,k}$, obtain logits $s_k = \text{sim}(z'_{v,k}, W_t)$ via cosine similarity with W_t, and convert them into a probability distribution p_k with a softmax. We quantify each adapter's uncertainty with the Shannon Entropy (Shannon, 1948) $H(p_k) = -\sum_{c=1}^{C} p_{k,c} \log(p_{k,c})$, where $p_{k,c}$ is the probability assigned to class c by adapter k; low entropy indicates a peaked (confident) prediction and high entropy a uniform (uncertain) one. Confidence weights are obtained via a softmax over negative entropies, $w_k = \exp(-H(p_k)/\beta) / \sum_{j=1}^{N} \exp(-H(p_j)/\beta)$, where β is a temperature hyperparameter that controls the sharpness of the weighting, so confident adapters receive large weights while uncertain ones are down-weighted. The final logits are the weighted average $s_{\text{final}} = \sum_{k=1}^{N} w_k \cdot s_k$, and the prediction is $\arg\max s_{\text{final}}$. Because adapters of different capacity exhibit systematically different confidence patterns, entropy weighting implicitly rebalances each adapter's contribution per sample.

3.4 Efficient Parallel Adapter Inference

Sequentially evaluating each adapter introduces unnecessary computational overhead as inference from multiple adapters would otherwise require repeated forward passes. To mitigate this, we implement a parallelized adapter forward pass that fuses shared computation across the ensemble.

Fused Down-Projection. Instead of executing each adapter independently, we concatenate all down-projection matrices along the bottleneck dimension: $W_{\text{down}}^{\text{concat}} = \text{cat}_r(W_{\text{down},1}, \ldots, W_{\text{down},N}) \in \mathbb{R}^{(\sum_k r_k) \times D}$, where cat_r denotes concatenation along the bottleneck dimension. Given the visual feature $z_v \in \mathbb{R}^D$ extracted from the frozen backbone, all bottleneck activations are computed using a single matrix multiplication: $b = \text{ReLU}\big(z_v (W_{\text{down}}^{\text{concat}})^\top\big) \in \mathbb{R}^{\sum_k r_k}$.

The resulting vector b contains the concatenated bottleneck activations of all adapters and is then split into segments $\{b_1, b_2, \ldots, b_N\}$ corresponding to the bottleneck dimensions $\{r_1, r_2, \ldots, r_N\}$.

Independent Up-Projection. To maintain same FLOP cost, we keep each up-projection independent. Each bottleneck segment $b_k \in \mathbb{R}^{r_k}$ is passed through its corresponding up-projection matrix: $z_{v,k} = b_k W_{\text{up},k}^{\top} \in \mathbb{R}^D$. Our implementation avoids padding on up-projection and performs the up-projection for each adapter directly on its corresponding chunk. This eliminates unnecessary floating-point operations while preserving parallelism in the dominant down-projection stage. Tab. 3 shows that, while we incur a small inference time on adapter-only forward passes (Adapter), there is no extra inference time incurred when calculated over backbone and adapter combined (Full) when compared with single CLIP-Adapter.

In practice, the concatenated down-projection matrix $W_{\text{down}}^{\text{concat}}$ is cached after adapter initialization, ensuring that the fused computation introduces no additional preprocessing overhead at test-time.

4 Experiments

We evaluate HEAT across 11 standard few-shot recognition benchmarks spanning general, fine-grained, and domain-specific visual tasks.

4.1 Experimental Setup

Datasets and Backbones. We utilize OpenAI-CLIP (Radford et al., 2021a) as our foundation model and evaluate on 11 diverse datasets: ImageNet (Deng et al., 2009), Caltech101 (Fei-Fei et al., 2004), OxfordPets (Parkhi et al., 2012), StanfordCars (Krause et al., 2012), Flowers102 (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014), FGVCAircraft (Maji et al., 2013), SUN397 (Xiao et al., 2010), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2018), and UCF101 (Soomro et al., 2012). We further evaluate generality on SigLIP-B/16 (Zhai et al., 2023) in Sec. 4.4 and under distribution shift (ImageNet $\rightarrow$ ImageNet-Sketch/-A/-R) in Tab. 2. For few-shot training, we randomly sample $K \in \{1, 2, 4, 8, 16\}$ examples per class using fixed seeds to ensure reproducibility and fair comparison with prior work (Silva-Rodriguez et al., 2024; Yu et al., 2023b).

Architectural Configurations. To maintain a strict parameter budget equivalent to a single CLIP-Adapter ($r = 256$), we distribute the capacity across our ensemble. For the ResNet-50 (He et al., 2016) backbone, we use $N = 4$ adapters with ranks $r = \{28, 36, 64, 128\}$. For ViT-B/16 (Dosovitskiy et al., 2021), we use $N = 5$ adapters with $r = \{28, 36, 64, 128, 256\}$, which matches the total parameter count of the ResNet-50 ensemble due to the smaller output dimensionality of the ViT backbone. Detailed parameter counts are provided in Tab. 3. Performance is stable across other rank allocations with the same budget (within 0.3 points, App. Tab. 14).

Table 1: **Avg. few-shot accuracy (%) across 11 datasets.** Best in **bold**, second best underlined. HEAT and CLAP show best overall performance, with HEAT achieving state-of-the-art accuracy from $K \geq 4$ shots onward.

Method	K=1	K=2	K=4	K=8	K=16
CoOp Zhou et al. (2022b)	59.6	61.8	66.5	69.9	73.3
CLIP-Ad. Gao et al. (2024)	58.4	62.5	66.2	69.9	73.4
CrM-LP Lin et al. (2023)	62.2	64.5	66.7	70.4	73.7
TaskRes Yu et al. (2023b)	61.4	65.3	68.4	71.7	74.4
CLAP Silva-Rodriguez et al. (2024)	**62.8**	**66.1**	69.1	72.1	74.6
HEAT *(Ours)*	62.2	65.9	**69.2**	**72.4**	**75.5**

4.2 Main Results

Average Performance and Scaling. As shown in Tab. 1, HEAT achieves state-of-the-art (SOTA) average accuracy across most shot counts. While CLAP shows a marginal advantage in extremely low-data regimes ($K \leq 2$), HEAT demonstrates a superior scaling trajectory. Starting at $K = 4$, our method widens its lead, outperforming the best previous adapter by 0.9% at $K = 16$.

Comparison with CLIP-Adapter. The most significant comparison is against CLIP-Adapter, as both methods share an identical parameter budget. HEAT outperforms CLIP-Adapter on all shots (Tab. 1). The gains are largest in the low-shot regime, reaching $+3.8\%$ at K=1 and $+3.4\%$ at K=2, and remain consistent at $+3.0\%$ for both K=4 and K=8. Even at K=16, HEAT keeps a $+2.2\%$ margin, for an average improvement of about 3.1% across all shots. This consistent margin confirms that redistributing parameters into a hierarchical ensemble is a strictly more powerful strategy than committing to a single adaptation hypothesis.

Domain Generalization Results. We test domain generalization by training on 8-shot ImageNet with ViT-B/16 and evaluating on ImageNet-Sketch, -A, and -R (Tab. 2). HEAT improves over zero-shot CLIP on the source and on every target domain. A single CLIP-Adapter raises source accuracy but falls well below zero-shot on the targets, so the same parameter budget spent on one adapter overfits the source domain. Detailed comparison is in App. E.

Table 2: ImageNet → Sketch/A/R on ViT-B/16. CLIP-Adapter and HEAT are trained on 8-shot ImageNet. Avg. is over the three target domains.

Method	ImageNet	-Sketch	-A	-R	Avg.
Zero-Shot	66.72	46.10	47.83	73.97	55.97
CLIP-Adapter	69.66	23.75	34.46	64.85	41.02
HEAT	**70.52**	**47.38**	**48.81**	**74.65**	**56.95**

4.3 Efficiency and Parameter Budget

While an ensemble might suggest increased inference time, our parallelized inference scheme (Sec. 3.4) minimizes this overhead. Tab. 3 reports inference latency under two measurements: *Adapter-only* latency, which measures the time required for the adapters to process a feature vector, and *Full* latency measures the end-to-end time (Backbone + Adapter). Sequential inference, which does a simple forward pass through one adapter after the other, incurs considerable latency. HEAT (Parallel) incurs a small inference time cost on adapter-only forward passes. There is no extra inference time cost incurred when calculated over backbone and adapter combined (Full) as compared to single CLIP-Adapter baseline. This demonstrates that in practical deployment, HEAT with our parallelization scheme incurs no extra FLOP or inference time cost while maintaining same number of parameters as Clip-Adapter.

Table 3: **Latency and parameters.** Per-sample time over 8100 samples. Sequential runs adapters one by one while Parallel fuses them (Sec. 3.4). Parallel HEAT matches CLIP-Adapter's parameter count with negligible full-pass overhead.

Method	Ranks r	Params	Adapter-only (μs)	Full (μs)
CLIP-Adapter	[256]	0.524M	0.61	232.4
HEAT (Sequential)	[28, 36, 64, 128]	0.524M	2.22	403.2
HEAT (Parallel)	[28, 36, 64, 128]	0.524M	1.66	238.9

4.4 Hierarchical vs Standard Ensembling

HEAT departs from standard ensembling in two ways: (i) it varies adapter ranks instead of replicating one architecture, and (ii) it shares a common initialization seed instead of using independent seeds. Tab. 4 isolates these factors by comparing four ensembles, all with the same total parameter budget as a single CLIP-Adapter, that differ in whether ranks and seeds differ ($\neq$) or match ($=$). The equal-rank baselines, Ens 2×128 (two adapters, r=128) and Ens 4×64 (four adapters, r=64), use independent seeds following standard ensemble practice. HEAT with shared initialization outperforms all other configurations at every shot count, confirming that both rank diversity and shared seeds are necessary. Notably, with different initialization HEAT *underperforms* the equal-rank ensembles despite having the same rank diversity: adapters of different rank converge to unrelated optima (App. Fig. 6c), which we expect to make their confidences less comparable, producing incoherent entropy signals that undermine confidence-based aggregation. Rank diversity helps only when paired with shared initialization, which produces the nested subspaces (Sec. 5.1). Per-dataset results for both initializations are in App. Tab. 7.

Table 4: Average accuracy (%) over 11 datasets. HEAT's combination of varying ranks with shared initialization outperform equal-rank and different-init ensembles at every shot count.

Method	Ranks	Seeds	K=1	K=2	K=4	K=8	K=16
Ens 2×128	$=$	$\neq$	61.20	64.93	68.48	71.90	75.22
Ens 4×64	$=$	$\neq$	61.79	65.21	68.69	71.91	75.02
HEAT (diff. init)	$\neq$	$\neq$	61.40	63.88	67.21	71.05	74.20
HEAT	$\neq$	$=$	**62.18**	**65.85**	**69.21**	**72.44**	**75.47**

Siglip Backbone. We compare three SigLIP-B/16 configurations with the same parameter budget: one rank-256 adapter, an equal-rank ensemble of four rank-64 adapters (Ens 4×64), and HEAT (ranks 28, 36, 64, and 128). Ens 4×64 uses independent initialization seeds, while HEAT shares one seed. Tab. 5 reports the mean accuracy over the 10 datasets excluding ImageNet, averaged over 3 seeds. HEAT is highest at every shot. Over the Clip-Adapter the gains are highest for $K = 1, 2$ with 0.50, 0.68. Ens 4×64 itself exceeds the Clip-Adapter at every shot, and the HEAT shared-initialization with rank diversity adds further gain on top of that ensemble. This validates that our approach is generalizable to other frameworks. We share full SigLIP per-dataset results in App. I.

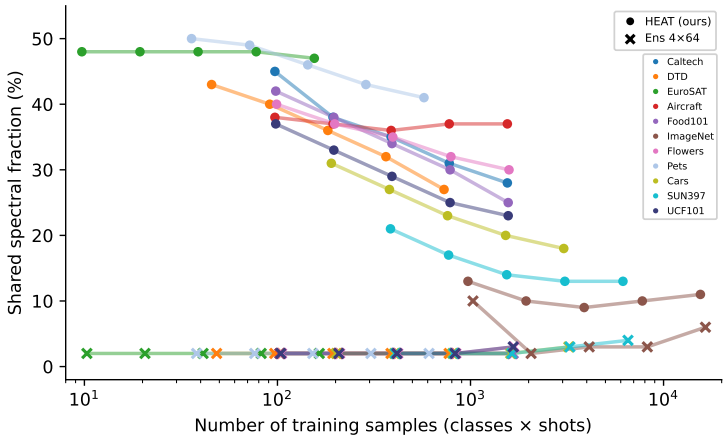

Figure 2: Shared spectral fraction (%) vs. number of training samples (classes × shots), averaged over seeds. Each color denotes a dataset; circles mark HEAT, crosses mark ensemble of 4×64 adapters. HEAT adapters share a large fraction of spectral directions (up to ∼50%), indicating nested subspaces whose overlap decreases as more training data allows greater specialization. In contrast, Ens 4×64 remains near ∼2–3%, indicating nearly disjoint subspaces.

Table 6: Mean pairwise CKA between adapter outputs (all shots/seeds). HEAT shows higher inter-adapter similarity than Ens 4×64 on every dataset.

Dataset	CKA	
	HEAT	4×64
Caltech	.83	.67
DTD	.78	.55
EuroSAT	.94	.88
Aircraft	.48	.40
Food	.81	.61
ImageNet	.61	.41
Flowers	.82	.64
Pets	.87	.71
Cars	.74	.49
SUN397	.67	.43
UCF101	.78	.53
Mean	**.76**	**.57**

5 The Role of Hierarchical Adapters

To understand how rank diversity and shared initialization shape the learned adapters, we conduct three complementary analyses. Having established in Sec. 4.4 that both rank diversity and shared initialization are necessary for HEAT's gains, we now examine *how* these design choices shape the learned representations. Unless stated otherwise, results are reported at 16 shots and averaged over 3 seeds.

5.1 Hierarchical Adapters Form a Nested Subspace

We analyze how adapters' learned representations relate to one another. For each adapter's down-projection matrix we perform SVD and compute the *shared subspace fraction* $1 - r_{.99}^{\text{concat}} / \sum_k r_{.99}^{(k)}$, where $r_{.99}$ counts the singular-value directions capturing 99% of spectral energy. Values near 100% indicate overlapping subspaces while near 0% indicates disjoint ones. We complement this with pairwise linear Centered Kernel Alignment (CKA) Kornblith et al. (2019), a representation similarity measure, computed on test-time outputs.

Table 5: Ablation on SigLIP. Average accuracy (%) over 10 datasets (excluding ImageNet due to computational cost). HEAT uses ranks 28, 36, 64, and 128 with one shared initialization. Ens 4×64 uses four rank-64 adapters with different seed init.

Method	Ranks	Seeds	K=1	K=2	K=4	K=8	K=16
Zero-shot	–	–	73.51	73.51	73.51	73.51	73.51
Clip-Ad. (1×256)	–	–	77.26	78.73	81.05	83.36	85.44
Ens 4×64	=	≠	77.66	79.32	81.38	83.61	85.62
HEAT	≠	=	**77.76**	**79.41**	**81.49**	**83.82**	**85.69**

We can see in Fig. 2 that, despite sharing same training data, HEAT adapters converge onto a substantial core of shared spectral directions (27% at 16 shots) whereas equal-rank adapters scatter into nearly disjoint subspaces (∼3% overlap). This nesting arises from the bottleneck hierarchy where the d=28 adapter learns the most essential discriminative directions, d=64 recovers these same directions plus additional detail, and d=128 preserves finer task structure.

Because each adapter's subspace is approximately contained within the next larger one, the ensemble builds a layered representation: smaller adapters, with limited capacity, capture only the dominant (high-energy) directions, while larger adapters retain these core directions and add lower-energy, task-specific ones.

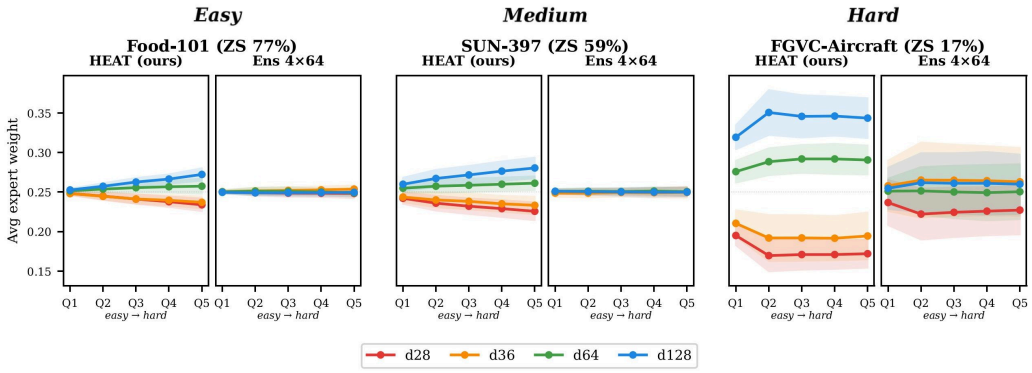

Figure 4: Mean entropy weight per adapter vs. zero-shot confidence quintile, averaged over 3 seeds and shots $K \in \{1, 2, 4, 8, 16\}$. Panels are Food-101 (ZS 77%), SUN-397 (59%), and FGVC Aircraft (17%), with a shared y-axis. Shading is ± 1 std. In HEAT, the rank-128 adapter takes more weight on hard samples and the weights approach uniformity on easy ones; Ens 4×64 stays flat.

Equal-rank adapters, lacking this capacity gradient, converge to unrelated local optima (Fig. 2 and App. Fig. 6a,b). CKA confirms this pattern at the representation level (Tab. 6).

Nested Subspace Evidence. We test subspace containment directly via pairwise comparisons of core singular value directions (Fig. 3). For adjacent adapter pairs, we compare the singular directions leading to cumulative spectral energy τ, with τ varying from 0-50%, 50-70%, 70-90%, 90-99%. The core singular values ($\tau \leq 0.50$) are 86–100% shared, confirming that larger adapters absorb the principal subspace of smaller ones. Containment degrades at fine spectral resolution (0.90−0.99: 0–57%), showing that adapters diverge in low-energy directions while sharing a mutual core:the signature of a nested hierarchy.

Why nesting emerges. To build intuition for why nesting emerges, we formally analyze the adapters in the linear case. If they are all approximating the same underlying target, then increasing an adapter's rank lets it capture additional singular directions while retaining those captured at lower ranks. Its subspace therefore grows with rank, giving a nested hierarchy. See App. A for the formal results. We argue that shared initialization supplies this common target (Remark 2), consistent with nesting collapsing under different seeds (App. Fig. 6c).

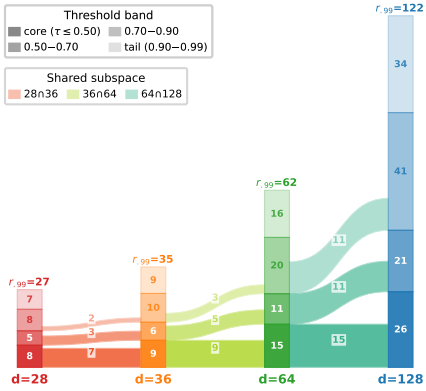

Figure 3: Mutual subspace overlap between adjacent HEAT adapters, decomposed by spectral energy (τ) band. Bar heights $\propto$ band rank; flow widths $\propto$ shared directions. Core bands (0-70% energy) confirm a nested hierarchy, where core directions are shared, while low-spectral areas collect unique details

5.2 How Entropy Weighting Exploits the Hierarchy

The nested subspace structure implies that adapters of different capacity differ mainly in how many directions beyond the shared core they capture. We now ask whether entropy weighting exploits this by examining how adapter weights vary with sample difficulty, measured by zero-shot confidence.

Across all 11 datasets, HEAT exhibits a characteristic *fan pattern* (Fig. 4): the high-capacity adapter (d=128) consistently receives the highest weight on hard samples while smaller adapters gain share on easy ones. Equal-rank adapters produce flat, overlapping curves near 0.25 regardless of difficulty. The effect scales with task difficulty: on FGVC Aircraft (zero-shot accuracy 17%), the weight spread ranges from 0.15 to 0.35; on Food-101 (77%), the fan is barely visible, and all weights approach uniformity. We describe this as *moderated dominance*: r=128 is the most-weighted adapter nearly everywhere, but its advantage diminishes on easier samples as smaller adapters' confidences

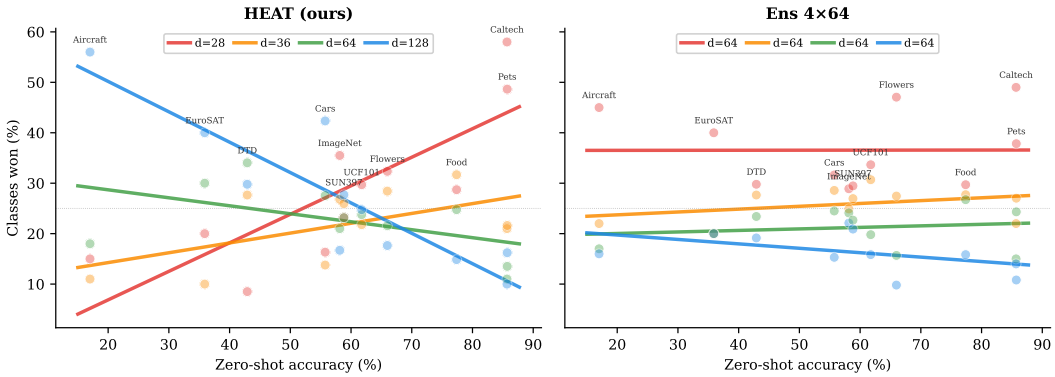

Figure 5: Fraction of classes won by each adapter vs. zero-shot accuracy of the dataset (16 shots, non-trivial classes only). **Left:** In HEAT, the trend lines cross: d=128 dominates on hard datasets (low zero-shot accuracy) while smaller adapters win more classes on easy ones. **Right:** In the equal-rank ensemble, all trend lines are flat and no systematic specialization emerges.

converge. Entropy weighting thus acts as a capacity moderator rather than performing a hard hand-off between adapters. Consistent with this, HEAT outperforms every one of its individual adapters (App. Fig. 8a), so the gain is not explained by selecting the largest adapter alone.

5.3 How Hierarchical Adapters Specialize

Beyond per-sample weighting, we investigate in Fig. 5 whether adapters specialize at the *class* level. For each class we identify the adapter with the highest seed-averaged accuracy. In HEAT (left panel), the trend lines for different bottleneck dimensions cross: on *hard datasets* with low zero-shot accuracy (FGVC Aircraft, EuroSAT, DTD), d=128 wins the majority of classes while smaller adapters contribute little; on *easy datasets* with high zero-shot accuracy (Caltech-101, Oxford Pets), the pattern reverses and smaller adapters win more classes. This crossover indicates that the advantage of high capacity is task-dependent: fine-grained discrimination requires the full bottleneck, while on easier tasks higher-rank adapters appear to hurt performance, likely because their additional capacity captures noisy or irrelevant details that might hurt generalization. In Ens 4×64 (right panel), all trend lines remain flat regardless of dataset difficulty: which adapter dominates is arbitrary and changes with the random seed.

5.4 Discussion, Takeaway and Conclusion

Layered analysis. At the *representation level* (Sec. 5.1 and App. A), varying rank yields a nested subspace hierarchy, with each adapter's subspace approximately contained in the next larger one, so larger adapters refine a core shared by all. At the *per-sample level* (Sec. 5.2), entropy weighting gives more weight to the largest adapter, especially on hard samples. At the *dataset/class level* (Sec. 5.3), while large adapters win on hard datasets, smaller adapters win more on easy ones. This highlight that the benefit of capacity is task-dependent. Lastly, HEAT outperforms every one of its individual adapters (App. Fig. 8a).

Diversity vs hierarchy. More diversity alone does not help, as we can see with equal-rank ensembles that have greater raw diversity but cannot exploit it. The effectiveness of our ensembling approach comes from the nested structure paired with our weighting mechanism.

Conclusion We introduced Hierarchical Ensemble Adapter Tuning (HEAT), which replaces the conventional single adapter with an ensemble of lightweight adapters of varying rank. This rank diversity acts as an inductive bias: adapters converge onto a shared core subspace, with each higher-rank adapter adding rank-specific refinements whose value varies across tasks and samples. An uncertainty-aware aggregation mechanism exploits this structured hierarchy by weighting each adapter's prediction by its per-sample confidence. HEAT establishes a new state-of-the-art across eleven benchmarks for $K \in \{4, 8, 16\}$ shots while keeping the same parameter budget as a single CLIP-Adapter, and remains effective across backbones and under distribution shift.

USE OF GENERATIVE AI

In this work, we used generative AI tools to implement code, which the authors verified and tested. We also used them to help formalize theoretical intuitions into mathematical claims and to assist in writing the corresponding proofs. We also used generative AI to improve our figures and plots. We have not used generative AI tools to generate synthetic data, clean or reformat datasets, support qualitative analysis, or assist with translation, and survey or interview instruments are not applicable to this work. Additionally, we used generative AI tools to draft and edit the paper for readability. We reviewed all AI-assisted text, proofs, code, and figures. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The experimental protocol is specified in Sec. 4.1: the 11 datasets, the shot counts $K \in \{1, 2, 4, 8, 16\}$, and the fixed seeds shared with prior work. Adapter ranks, the shared initialization, and the entropy-weighted aggregation are given in Sec. 3. Training hyperparameters are listed in App. Sec. L. The appendix contains the formal nesting argument (Sec. A), per-dataset results (Secs. D and I), the aggregation comparison (Sec. G), and the distribution-shift setup (Sec. E).

REFERENCES

Mohamad Mahmoud Al Rahhal, Yakoub Bazi, and Mansour Zuair. Lora-clip: Efficient low-rank adaptation of large clip foundation model for scene classification. *IEEE Geoscience and Remote Sensing Letters*, 2025.

Yue Bai, Handong Zhao, Zhe Lin, Ajinkya Kale, Jiuxiang Gu, Tong Yu, Sungchul Kim, and Yun Fu. Advancing vision-language models with adapter ensemble strategies. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision (ECCV)*, 2014.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3606–3613, 2014.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*, 2021.

Li Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Worskshops (CVPRW)*, pp. 178–178, 2004.

Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. In *arXiv preprint arXiv:1912.02757*, 2020.

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. *IJCV*, 2024.

Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 4th edition, 2013.

Sachin Goyal, Ananya Kumar, Sankalp Garg, Zico Kolter, and Aditi Raghunathan. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *CVPR*, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Introducing eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 3606–3613, 2018.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2nd edition, 2013.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

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*, pp. 4904–4916. PMLR, 2021.

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*, pp. 709–727. Springer, 2022.

Muhammad Gul Zain Ali Khan, Muhammad Ferjad Naeem, Luc Van Gool, Alain Pagani, Didier Stricker, and Muhammad Zeshan Afzal. Learning attention propagation for compositional zero-shot learning. In *WACV*, 2023a.

Muhammad Gul Zain Ali Khan, Muhammad Ferjad Naeem, Luc Van Gool, Didier Stricker, Federico Tombari, and Muhammad Zeshan Afzal. Introducing language guidance in prompt-based continual learning. In *ICCV*, 2023b.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. Similarity of neural network representations revisited. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 2019. URL http://dblp.uni-trier.de/db/conf/icml/icml2019.html#Kornblith0LH19.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3498–3505, 2012.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-tuning can distort pretrained features and underperform out-of-distribution. *arXiv preprint arXiv:2202.10054*, 2022.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Zhiqiu Lin, Samuel Yu, Zhiyi Kuang, Deepak Pathak, and Deva Ramanan. Multimodality helps unimodality: Cross-modal few-shot learning with multimodal models. In *CVPR*, 2023.

Yun Luo, Zhen Yang, Fandong Meng, Yafu Li, Jie Zhou, and Yue Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *IEEE Transactions on Audio, Speech and Language Processing*, 2025.

S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. In *ArXiv Preprint*, 2013.

Muhammad Ferjad Naeem, Muhammad Gul Zain Ali Khan, Yongqin Xian, Muhammad Zeshan Afzal, Didier Stricker, Luc Van Gool, and Federico Tombari. I2mvformer: Large language model generated multi-view document supervision for zero-shot image classification. In *CVPR*, 2023.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3498–3505, 2012.

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 (ICML)*, pp. 8748–8763, 2021a.

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*, pp. 8748–8763. PmLR, 2021b.

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

Claude E Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

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*, pp. 23681–23690, 2024.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. In *ArXiv Preprint*, 2012.

Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Chengzhong Xu. Hydralora: An asymmetric lora architecture for efficient fine-tuning. *Advances in Neural Information Processing Systems*, 2024.

Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. Dylora: Parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 2023.

Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. In *ECCV*, pp. 398–414, 2022.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: An alternative approach to efficient ensemble and lifelong learning. In *International Conference on Learning Representations (ICLR)*, 2020.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *CVPR*, 2022.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3485–3492, 2010.

Yi Xin, Siqi Luo, Haodi Zhou, Junlong Du, Xiaohong Liu, Yue Fan, Qing Li, and Yuntao Du. Parameter-efficient fine-tuning for pre-trained vision models: A survey. *arXiv e-prints*, pp. arXiv–2402, 2024.

Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *CVPR*, pp. 3014–3023, 2021.

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.

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*, pp. 10899–10909, 2023a.

Tao Yu, Zhihe Lu, Xin Jin, Zhibo Chen, and Xinchao Wang. Task residual for tuning vision-language models. In *CVPR*, 2023b.

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

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *ICCV*. IEEE, 2023.

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.

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*, pp. 16816–16825, 2022a.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 2022b.