Parameter-Efficient Fine-Tuning for Vision-Language Models: The Post-Transformer Evolution

Patalee Narasinghe
Department of Computer Science and Engineering
University of Moratuwa
Sri Lanka
patalee.21@cse.mrt.ac.lk

B.H. Sudantha
Department of Information Technology
University of Moratuwa
Sri Lanka
sudanthabh@uom.lk

***Abstract*—The rapid proliferation of large-scale Vision-Language Models (VLMs) has revolutionized multimodal artificial intelligence, enabling unprecedented capabilities in cross-modal understanding and generation. However, the substantial computational and memory requirements for fine-tuning these billion-parameter models present significant deployment challenges, particularly for resource constrained environments like mobile robots and edge devices. This survey provides a comprehensive analysis of Parameter-Efficient Fine-Tuning (PEFT) techniques tailored for VLMs in the post-Transformer era (2021-2025). PEFT methods are systematically categorized into three mechanistic paradigms: input-level adaptation (prompting), feature-level adaptation (adapters), and weight-level adaptation (reparameterization). Representative techniques, including CoOp, CoCoOp, MaPLe, CLIP-Adapter, Tip-Adapter, LoRA, DoRA, and PiSSA, are analyzed. These methods are critically evaluated regarding parameter efficiency, convergence, latency, catastrophic forgetting, and alignment preservation. Comparative benchmarking on ImageNet, VQAv2, and MMBench is used to isolate optimal strategies for distinct application needs. This analysis extends into specialized fields, critically examining adaptations for medical imaging, remote sensing, and video understanding, alongside privacy preserving federated learning. It is concluded that while competitive performance is offered by methods like DoRA and Tip-Adapter-F, the optimal strategy is critically dependent on the specific architecture, task complexity, and deployment constraints.**

***Index Terms*—Vision-Language Models, Parameter-Efficient Fine-Tuning, Edge AI, Multimodal Learning, Low-Rank Adaptation, Prompt Tuning**

I. Introduction

The proliferation of large-scale foundation models has fundamentally restructured the landscape of artificial intelligence. In the domain of multimodal learning, the convergence of Computer Vision (CV) and Natural Language Processing (NLP) has catalyzed the development of powerful Vision-Language Models (VLMs) such as CLIP [1], BLIP [2], LLaVA [3], and Flamingo [4]. These models, typically trained on billions of image-text pairs sourced from the web, serve as general-purpose perception engines, capable of aligning visual semantics with linguistic structures in a shared embedding space. They exhibit unprecedented capabilities in zero-shot classification, visual question answering (VQA), and cross-modal reasoning [5].

However, the sheer scale of these architectures which are ranging from hundreds of millions to hundreds of billions of parameters, presents a formidable barrier to their practical deployment. Fine-tuning large scale LLMs or models such as Flamingo-80B on a downstream task using traditional Full Fine-Tuning (FFT) requires updating every weight in the network. This process presents three fundamental challenges: (1) *Computational prohibitions*, requiring massive clusters of high-end GPUs; (2) *Storage inefficiency*, as a separate copy of the massive model must be saved for each specific task; and (3) *Catastrophic forgetting*, where the delicate cross-modal alignment learned during pre-training is disrupted by task-specific overfitting [6]. For such large models, the parameter scale creates a definitive physical barrier; for instance, FFT of a 65B model requires over 780GB of VRAM, far exceeding even multi-GPU 80GB A100 clusters, necessitating PEFT approaches [7].

Addressing this bottleneck, Parameter-Efficient Fine-Tuning (PEFT) has emerged as a functional necessity. PEFT methodologies rely on the hypothesis that task-specific updates occupy a low intrinsic dimension relative to total parameters. By freezing the pre-trained backbone and optimizing only a small subset of parameters (often fewer than 1%), PEFT achieves a trifecta of benefits: drastic reductions in memory footprints, mitigation of catastrophic forgetting, and modularity that allows multiple task-specific adapters to coexist on a single base model [8].

This work specifically focuses on the "Post-Transformer Era" (2021–2025), characterized by the widespread adoption of Transformer-based backbones (e.g., Vision Transformer (ViT) [9] and large language models such as LLaMA [10]), and a broader shift from task-specific architectures to unified foundation models. Unlike early transfer learning, which mostly involved linear probing or tuning the final layers of a ResNet, modern PEFT operates deep within the self-

979-8-3195-4220-5/26/$31.00

attention mechanisms of the network. Methods are grouped by mechanistic principles beyond simple enumeration: Input Space (Prompting), Feature Space (Adapters), and Weight Space (Reparameterization). Special emphasis is placed on emerging second-generation PEFT methods like DoRA [11] and PiSSA [12], which address theoretical limitations of earlier approaches like LoRA [13] regarding weight magnitude and direction coupling.

II. Theoretical Foundations of PEFT for VLMs

Before analyzing specific methodologies, the theoretical foundations of PEFT for VLMs must be established. Efficacy stems from geometric properties of the pre-training loss landscape, cross-modal alignment dynamics, and spectral properties of weight updates.

A. The Intrinsic Dimension Hypothesis

The primary justification for PEFT stems from the *Intrinsic Dimension Hypothesis* [14], [15], which posits that the optimization of over-parameterized models occurs on a low-dimensional manifold embedded within the high-dimensional parameter space.

Let $\Theta_{pre} \in \mathbb{R}^D$ denote the pre-trained VLM parameters. The hypothesis posits that task-specific solutions can be found in a lower-dimensional subspace $\mathbb{R}^d$ ($d \ll D$) via a projection $P \in \mathbb{R}^{D \times d}$:

$$\Theta_{task} = \Theta_{pre} + P\theta_d \tag{1}$$

where $\theta_d \in \mathbb{R}^d$ represents the learnable parameters in the intrinsic dimension. This theoretical framework justifies LoRA's approach of restricting the weight update $\Delta W = BA$ to a low-rank manifold where $r \ll \min(d_{out}, d_{in})$.

B. Cross-Modal Alignment and the Modality Gap

Theoretical analysis of VLMs must consider multi-modal embedding geometry beyond single-manifold optimization. A critical challenge is the *modality gap* [16]: the conical separation between image and text embeddings in shared latent space.

PEFT modules serve as alignment bridges, learning transformations $\mathcal{T}$ that minimize alignment error without manifold distortion. Uniform adaptation risks exacerbating this gap via text overfitting, necessitating modality-specific adapters.

C. The Regularization-Forgetting Paradox

The impact of PEFT on *catastrophic forgetting* remains a subject of theoretical debate.

The Classical View (Implicit Regularization): Traditionally, PEFT is viewed as an implicit regularizer. By freezing the majority of parameters Θ_{frozen}, the model is forced to find a solution Θ_{peft} that remains close to the pre-trained initialization Θ_0. This is often formulated as a constrained optimization problem where the Frobenius norm of the update is bounded, theoretically limiting the drift from generalizable features [17].

The Spectral View (Intruder Dimensions): Recent spectral analyses [18] challenge the notion that parameter efficiency inherently preserves pre-trained features. Specifically, LoRA updates often introduce *intruder dimensions*. These are defined as high-rank singular vectors in the update matrix ΔW that remain orthogonal to the singular vectors of the pre-trained weights W_0.

Compared to full fine-tuning, LoRA's low-rank constraint can restrict the optimization path. This restriction often forces the model to learn shortcut features that lie outside the original pre-training manifold. Empirically, these intruder dimensions have been linked to higher rates of catastrophic forgetting, particularly in continual learning settings. This suggests that while PEFT is parameter-efficient, it is not necessarily *spectrally* consistent with the base model's original feature space.

III. Taxonomy of PEFT Methods for VLMs

Current Parameter-Efficient Fine-Tuning (PEFT) methodologies can be systematically categorized based on the locus of information modification: *Input-Space* (Prompting), *Function-Space* (Adapters), and *Parameter-Space* (Reparameterization).

A. Input-Level Adaptation: Prompt Learning

Prompt learning shifts the optimization burden from model parameters to the input context. This paradigm is particularly effective for aligning pre-trained VLMs to downstream tasks without modifying the frozen backbone.

1) Static vs. Dynamic Prompting: **Static Soft Prompts:** The foundational approach, *Context Optimization (CoOp)* [19], replaces rigid manual templates with learnable continuous vectors. The input sequence is formulated as $[V]_1, [V]_2, \ldots, [V]_M, [\text{CLASS}]_i$, where the context vectors $[V]_k$ are optimized via backpropagation while the class token $[\text{CLASS}]_i$ remains fixed. *Critique:* While CoOp automates prompt engineering, it learns a *static* context that is class-agnostic. This leads to overfitting on "base" (seen) classes and degradation on "novel" (unseen) classes, as the prompt cannot adapt to instance-specific visual variations.

Conditional/Dynamic Prompts: To resolve the static limitation, *Conditional Context Optimization (CoCoOp)* [20] generates instance-specific residuals via a meta-network $\pi(\cdot)$:

$$v(x) = v_{static} + \pi(x_{img}) \tag{2}$$

This mechanism creates a dynamic decision boundary for each instance, significantly improving zero-shot transfer. However, it adds inference latency as the prompts must be recomputed for every forward pass.

2) Regularization and Generalization Strategies: To mitigate CoOp's overfitting without incurring the inference overhead of CoCoOp, recent works focus on regularization and knowledge distillation.

Gradient and Knowledge Guidance: *ProGrad* [21] prevents the "forgetting" of general knowledge by updating prompts only in directions that align with the original zero-shot gradients (gradient projection). Similarly, *KgCoOp* [22]

explicitly minimizes the Euclidean distance between the embeddings of the learnable prompts and the hand-crafted manual templates, forcing the learned prompts to remain semantically coherent.

Mutual Information Maximization: Addressing the trade-off between template robustness and learnable accuracy, Zhang et al. [23] recently proposed treating soft prompts and manual templates as dual views. They optimize a joint objective:

$$\mathcal{L}_{total} = \mathcal{L}_{task} + \lambda \mathcal{L}_{MI}(P_{soft}, P_{manual}) \tag{3}$$

Mutual information maximization between soft and manual prompts is a strong approach to balance template robustness with task adaptation, especially in a few-shot settings with base versus novel classes.

3) Shallow vs. Deep Multimodal Alignment: Early methods (CoOp/CoCoOp) are typically uni-modal, modifying only the text input. *Multi-modal Prompt Learning (MaPLe)* [24] argues that this fails to bridge the "modality gap" in deeper network layers. MaPLe implements **Deep Prompting**: it injects learnable tokens into both the vision and language branches simultaneously across multiple transformer blocks. Crucially, a coupling function ensures semantic consistency between the modalities, enabling the model to adjust its feature hierarchy globally rather than just at the input level.

4) Visual-Specific Adaptation (VPT & APT): For visual-heavy tasks, *Visual Prompt Tuning (VPT)* [25] prepends learnable "pixel tokens" to the ViT patch sequence. While effective for static images, video VLMs face the "token explosion" problem. Variants like *Attention Prompt Tuning (APT)* mitigate the quadratic cost of attention by injecting prompts specifically into non-local temporal blocks, balancing temporal reasoning with computational efficiency.

B. Feature-Level Adaptation: Adapter Modules

Unlike prompting, which manipulates inputs, adapters manipulate intermediate feature maps. This category has evolved significantly from static bottleneck layers to dynamic, routing-based architectures that balance memory efficiency with out-of-distribution (OOD) robustness.

1) Parametric Adaptation: Dynamic and Reconstruction Architectures: While early adapters followed a static bottleneck architecture ($h' = h + W_{up}\sigma(W_{down}h)$), recent research [17] identifies that fixed adapters struggle to balance task-specific learning with the preservation of pre-trained generalization.

a) Mixture-of-Experts (MoE) Adapters: To address multi-task interference and capacity limitations, approaches like *MoE-Adapters* [26] replace the single feed-forward network with a sparse mixture of experts.

A routing gate $G(x)$ dynamically selects a top-k subset of experts for each input token, enabling the model to specialize parameters for distinct visual concepts (e.g., separating texture from shape) without increasing inference latency:

$$h' = h + \sum_{i \in \text{Top}k} G(x)_i \cdot E_i(h) \tag{4}$$

where E_i represents the i-th expert network and $G(x)_i$ is the sparse gating weight.

b) Reconstruction-Based Adapters: To mitigate catastrophic forgetting of general knowledge, *RMAdapter* [27] introduces a dual-branch design. Alongside the standard fine-tuning branch, a reconstruction branch forces the model to map adapted features back to the original pre-trained latent space.

The objective function is modified to penalize deviations from the original manifold:

$$\mathcal{L} = \mathcal{L}_{task} + \lambda ||f_{recon}(f_{adapter}(x)) - f_{original}(x)||_2^2 \tag{5}$$

This geometric regularization ensures that while the adapter learns downstream tasks, it retains the structural integrity of the original CLIP feature space.

2) Non-Parametric (Training-Free) Adaptation: The "Cache Model" paradigm, initiated by *Tip-Adapter* [28], introduces a non-parametric alternative to gradient-based adapter training. It constructs a key-value cache from the few-shot support set, where visual features extracted by frozen CLIP serve as keys and corresponding one-hot class labels as values. During inference, test sample features retrieve nearest neighbors from this cache via cosine similarity, combining retrieved cache logits with original CLIP logits through a residual connection:

$$\hat{y} = \text{softmax}\left(\text{CLIP}(x) + \lambda \cdot \text{Retrieve}(\text{CLIP}(x))\right) \tag{6}$$

Unlike parametric adapters requiring backpropagation, standard Tip-Adapter remains completely training-free after cache construction. The variant *Tip-Adapter-F* further bridges the gap to full fine-tuning by unfreezing *only* the cache keys as learnable parameters for brief fine-tuning ($\sim$20 epochs), while keeping values fixed. This hybrid approach outperforms prompt tuning methods by achieving comparable few-shot accuracy with $\sim$10× fewer training epochs (20 vs 200), while avoiding backpropagation through the backbone.

C. Weight-Level Adaptation: Reparameterization

Reparameterization methods inject trainable parameters directly into the weight update computation. These are currently among the most widely adopted approaches for Generative VLMs (e.g., LLaVA) as they offer **Zero-Latency Inference** via weight merging.

1) Low-Rank Adaptation (LoRA) and Optimization variants: LoRA hypothesizes that weight updates have a low intrinsic rank. It freezes the pre-trained weights W_0 and optimizes low-rank matrices $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$ where $r \ll d$:

$$W' = W_0 + \Delta W = W_0 + \frac{\alpha}{r}(B \cdot A) \tag{7}$$

Recent analysis by *LoRA+* [29] reveals that standard LoRA suffers from suboptimal convergence because matrices A and B play distinct roles (B dictates the update direction, A extracts features). LoRA+ introduces distinct learning rates ($\eta_B = \lambda \eta_A$ with $\lambda > 1$), proving that the adapter matrices require disparate optimization scales to reach full fine-tuning performance.

2) Structural Decomposition: DoRA (Weight-Decomposed LoRA) [11] addresses a theoretical limitation in LoRA; the coupling of magnitude and direction. In full fine-tuning, these two properties can change independently, but LoRA forces them to co-evolve.

DoRA decomposes the weight matrix into a magnitude vector m and a directional matrix V:

$$W = m\frac{V}{\|V\|_c}, \quad \text{where } V = W_0 + B \cdot A \tag{8}$$

By applying LoRA only to the *directional* component V and fully training the magnitude m, DoRA demonstrates improved stability and alignment, which correlates with reduced hallucinations on benchmarks such as POPE and VQAv2.

3) Spectral Awareness: Standard LoRA initializes A with Gaussian noise and B with zeros. *PiSSA* [12] argues this is inefficient as it forces the model to learn updates from scratch. Instead, PiSSA employs **Principal Singular Values**:

$$W_0 = U\Sigma V^T \approx \underbrace{U_{[:r]}\Sigma_{[:r]}^{1/2}}_{B_{init}} \cdot \underbrace{\Sigma_{[:r]}^{1/2}V_{[:r]}^T}_{A_{init}} + W_{res} \tag{9}$$

The adapter matrices A and B are initialized with the top-r principal components of W_0 (the principal variance directions), which often correspond to important feature directions and accelerate convergence compared to random initialization, while the residual W_{res} is frozen.

4) Quantization-Aware Scaling: QLoRA and LoftQ: To enable training of 70B+ parameter models on consumer hardware, *QLoRA* [7] utilizes a frozen backbone in 4-bit NormalFloat (NF4).

However, direct quantization introduces initialization errors. To mitigate this, *LoftQ* [30] (LoRA-Fine-Tuning-Aware Quantization) simultaneously quantizes W_0 and initializes the LoRA adapters to approximate the quantization error:

$$W_0 \approx Q(W_0) + B_{init}A_{init} \tag{10}$$

This ensures the starting point of the quantized model is mathematically equivalent to the full-precision model, preventing the "quantization gap" often seen in early QLoRA training steps.

IV. Critical Analysis and Performance Evaluation

A. The Modality Gap and Locus of Adaptation

A pivotal debate in VLM adaptation concerns where to inject trainable parameters within architectures like LLaVA (Vision Encoder $\rightarrow$ Projector $\rightarrow$ LLM). Three primary loci have been identified:

1) **Projector Tuning (Feature Alignment):** This approach tunes *only* the connector (e.g., **LLaVA Pretraining Stage 1**) while keeping the LLM frozen to align visual features with the language embedding space [3]. While highly parameter-efficient, recent studies indicate that keeping the LLM frozen can limit the model's ability to follow complex formatting instructions, as the visual tokens alone may not sufficiently regularize the LLM's output style [31].
2) **Backbone Tuning:** Methods like **LLaVA-1.5** [31] address these limitations by unfreezing and fine-tuning the entire LLM backbone alongside the projector during the instruction tuning stage. This allows the model to learn complex multimodal interactions and precise response formatting (e.g., short-form VQA answers) that projector tuning alone often fails to capture. Complementary efficiency improvements come from architectures like **MoE-LLaVA** [32], which employ a sparse mixture-of-experts design to reduce computational costs while maintaining performance comparable to dense backbones.
3) **Vision Encoder Tuning:** Although typically avoided to prevent the degradation of robust, general-purpose visual features, tuning the encoder becomes strictly necessary in specialized domains like medical imaging or remote sensing; in these fields, the distribution shift is so severe that pre-trained natural image features are insufficient for discrimination [33].

B. Benchmarking: ImageNet and Generative Tasks

TABLE I: Comparison of few-shot learning methods on ImageNet (16-shot). **Note:** Accuracy metrics are reported for the **ViT-B/16** backbone. MaPLe demonstrates high efficiency, converging in just 5 epochs.

Method	Mechanism	Epochs	ImageNet Acc.
Zero-Shot CLIP [1]	Baseline	-	68.35%
CoOp [19]	Soft Prompt	200	71.92%
CoCoOp [20]	Cond. Prompt	10	73.10%
CLIP-Adapter [34]	MLP Adapter	200	71.13%
Tip-Adapter [28]	Cache Lookup	-	70.75%
Tip-Adapter-F [28]	Tuned Cache	20	73.69%
MaPLe [24]	Deep Prompts	**5**	73.47%

TABLE II: Visual Instruction Tuning Performance (Generative VLMs). Evaluation results of LLaVA-1.5-7B DoRA, LoRA, and Full Fine-Tuning (FT) with visual instruction tuning data. Note: The **Avg** column represents the average score across these 7 vision language tasks (VQAv2, GQA, VisWiz, SQA, VQA-T, POPE, and MMBench). [11], [31]

Method	VQAv2	MMBench	Avg (7 tasks)	Params Tuned
LLaVA-FFT	78.5	64.3	66.5	100%
LLaVA-LoRA	79.1	66.1	66.9	4.61%
LLaVA-DoRA	78.6	66.1	67.6	4.63%

C. Efficiency Trade-offs

Deployment architecture is strictly dictated by the method chosen:

- **Inference Latency (Adapters):** Adapter layers impose a **15–30% latency penalty** [13] because they insert extra sequential steps into the forward pass. This delay is

TABLE III: Efficiency Metrics Comparison (7B Model)

Method	VRAM	Inf. Latency	Storage/Task
Full Fine-Tuning [7]	$>$ 80GB	1.0x	$\sim$13 GB
Adapter Layers [13], [35]	$\sim$16GB	$\sim$1.15x	$\sim$100 MB
LoRA [13]	$\sim$24GB	**1.0x (Merged)**	$\sim$100 MB
QLoRA [7]	$\sim$**6GB**	**1.0x (Merged)**	$\sim$100 MB

often unacceptable for real-time applications like high-frequency trading or conversational agents.

- **Serving Complexity (LoRA/QLoRA):** While LoRA achieves **zero-latency** via weight merging ($W_{new} = W_{base} + \Delta W$) [13], it introduces a "Multi-Tenant" serving challenge. Merging weights permanently creates a new model instance (e.g., 13GB for Llama-2-13B). To serve 100 users with different skills, one must either (1) load 100 separate merged models (requiring massive VRAM), or (2) dynamically swap LoRA adapters into the base model at runtime (which introduces loading latency) [36].

V. Domain-Specific Applications

A. Medical and Remote Sensing Adaptation

In specialized domains with extreme distribution shifts from internet-based pre-training data, domain-specific adaptation becomes critical. *BiomedCLIP* [37], trained on 15 million biomedical image-text pairs from PubMed Central, demonstrates that large-scale domain-specific pre-training substantially outperforms generic CLIP on medical tasks including pathology classification and radiology analysis. For remote sensing, *RemoteCLIP* [38] addresses unique challenges through data scaling via box-to-caption and mask-to-box conversion, achieving strong zero-shot and few-shot performance on aerial imagery tasks with improved semantic understanding of satellite imagery concepts.

B. Video Understanding and Temporal Shift

Video VLMs introduce the dimension of time, challenging static PEFT methods. **Temporal Adapters:** Approaches like Attention Prompt Tuning (APT) [39] inject learnable prompts specifically into temporal attention blocks to learn motion patterns without retraining frozen spatial features. **Video Adapters:** MotionLoRA [40] adds LoRA to AnimateDiff's motion module self-attention layers, fine-tuning new motion patterns (pans/zooms) with just 20–50 reference videos using only $\sim$1% of the motion module's parameters.

C. Federated Learning with PEFT

PEFT is a critical enabler for Federated Learning (FL) due to privacy and bandwidth constraints. While standard frameworks aggregate adapter updates (e.g., LoRA) globally, they often struggle with non-IID data distributions. **Handling Heterogeneity:** F^{3}OCUS [41] addresses this by computing "layer importance scores" (based on Neural Tangent Kernels) to dynamically select only the most relevant layers for fine-tuning on each client. This targeted approach ensures the aggregated global model remains robust despite diverse data distributions across edge devices.

VI. Emerging Research Directions

Dynamic Adaptation (Continual & Test-Time): Current PEFT methods often assume static task distributions, leading to catastrophic forgetting in sequential settings. Future work focuses on *Parameter-Efficient Continual Fine-Tuning (PECFT)* [6], which reviews techniques like orthogonal subspace projection, prompt tuning, and adapters to prevent interference between sequentially learned tasks. Similarly, in unsupervised settings, *Test-Time Adaptation (TTA)* methods like *Tent* (Entropy Minimization) [42] allow models to adapt normalization statistics at inference time without labeled data, enhancing robustness to distribution shifts.

Unified & Automated Architectures: Moving beyond single adapters, the Mixture of Adapters paradigm enables dynamic composition of specialized skills through advanced routing mechanisms like top-k gating [43]. Concurrently, AdaLoRA [44] automates adaptation by dynamically allocating parameter budgets (ranks) based on layer sensitivity scores, increasingly critical for VLMs to balance the parameter disparity between compute-intensive visual encoders and linguistic modules.

VII. Conclusion

Parameter-Efficient Fine-Tuning (PEFT) represents a fundamental paradigm shift in VLM adaptation, transitioning from monolithic weight updates to surgical interventions. This survey systematically categorizes methods into input-level prompting (CoOp, MaPLe), feature-level adapters (CLIP-Adapter, MoE-Adapters), and weight-level reparameterization (LoRA, DoRA, QLoRA), benchmarking their efficacy across ImageNet, VQAv2, and MMBench.

Key findings reveal distinct optimal strategies: (1) Cache-based methods like **Tip-Adapter-F** excel in few-shot latency-critical scenarios with minimal training overhead; (2) Advanced LoRA variants like **DoRA** and **PiSSA** achieve near full fine-tuning performance for generative reasoning; (3) Multimodal approaches like **MaPLe** effectively bridge vision-language gaps across domains.

Critical challenges persist in multi-tenant LoRA serving [36] and federated non-IID heterogeneity [41]. As VLMs scale toward trillions of parameters, PEFT remains the essential enabler for efficient, private, and domain-specialized multimodal deployment.

References

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning (ICML)*, 2021, pp. 8748–8763.

[2] J. Li, D. Li, C. Xiong, and S. Hoi, "Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation," in *International Conference on Machine Learning (ICML)*, 2022, pp. 12 888–12 900.

[3] H. Liu *et al.*, "Visual instruction tuning," in *Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 34 892–34 916.

[4] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr *et al.*, "Flamingo: a visual language model for few-shot learning," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 23 716–23 736.

[5] S. Long, H. Lu, X. Li, M. Lu, J. Cao, and Z. Qiu, "Vision-and-language pretrained models: A survey," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, 2023, pp. 5475–5483.

[6] E. N. Coleman, L. Quarantiello, Z. Liu, Q. Yang, S. Mukherjee, J. Hurtado, and V. Lomonaco, "Parameter-efficient continual fine-tuning: A survey," 2025. [Online]. Available: https://arxiv.org/abs/2504.13822

[7] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[8] D. Zhang, T. Feng, L. Xue, Y. Wang, Y. Dong, and J. Tang, "Parameter-efficient fine-tuning for foundation models," 2025. [Online]. Available: https://arxiv.org/abs/2501.13787

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2021.

[10] H. Touvron, T. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama: Open and efficient foundation language models," 2023, arXiv preprint. [Online]. Available: https://arxiv.org/abs/2302.13971

[11] S.-Y. Liu, C.-Y. Wang, H. Yin, P. Molchanov, Y.-C. F. Wang, K.-T. Cheng, and M.-H. Chen, "Dora: Weight-decomposed low-rank adaptation," in *International Conference on Machine Learning*, 2024.

[12] F. Meng, Z. Wang, and M. Zhang, "Pissa: Principal singular values and singular vectors adaptation of large language models," in *Advances in Neural Information Processing Systems*, vol. 37, 2024, pp. 121 038–121 072.

[13] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *International Conference on Learning Representations*, 2022.

[14] C. Li, H. Farkhoor, R. Liu, and J. Yosinski, "Measuring the intrinsic dimension of objective landscapes," in *International Conference on Learning Representations*, 2018.

[15] A. Aghajanyan, S. Gupta, A. Shrivastava, X. Chen, L. Zettlemoyer, and S. Gupta, "Intrinsic dimensionality explains the effectiveness of language model fine-tuning," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, 2021.

[16] W. Liang, Y. Zhang, Y. Kwon, S. Ye, and J. Zou, "Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 17 612–17 625.

[17] Z. Mai, C. Pan, Z. Wang, Z. Liu, A. Vasudevan, Z. Lawrence, F. I.-H. Lee, Z. Jia *et al.*, "Lessons and insights from a unifying study of parameter-efficient fine-tuning (peft) in visual recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, June 2025.

[18] R. Shuttleworth, J. Andreas, A. Torralba, and P. Sharma, "Lora vs full fine-tuning: An illusion of equivalence," 2024. [Online]. Available: https://arxiv.org/abs/2410.21228

[19] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Learning to prompt for vision-language models," 2021. [Online]. Available: https://arxiv.org/abs/2109.01134

[20] ——, "Conditional prompt learning for vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 816–16 825.

[21] B. Zhu, Y. Niu, Y. Han, Y. Wu, and H. Zhang, "Prompt-aligned gradient for prompt tuning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 15 659–15 669.

[22] H. Yao, R. Zhang, and C. Xu, "Visual-language prompt tuning with knowledge-guided context optimization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 6757–6767.

[23] Q. Zhang, "Generalizable prompt tuning for vision-language models," 2024. [Online]. Available: https://arxiv.org/abs/2410.03189

[24] M. U. Khattak, H. Rasheed, M. Maaz, S. Khan, and F. S. Khan, "Maple: Multi-modal prompt learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 19 113–19 122.

[25] M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, and S.-N. Lim, "Visual prompt tuning," in *European Conference on Computer Vision*, 2022, pp. 709–727.

[26] J. Yu, Y. Zhuge, L. Zhang, P. Hu, D. Wang, H. Lu, and Y. He, "Boosting continual learning of vision-language models via mixture-of-experts adapters," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 23 219–23 230.

[27] X. Lin, W. Li, S. Guo, L. Wang, and D. Huang, "Rmadapter: Reconstruction-based multi-modal adapter for vision-language models," 2025. [Online]. Available: https://arxiv.org/abs/2512.06811

[28] R. Zhang, W. Zhang, C. Xie, Y. Chai, Z. Cao, D. Song, and H. Li, "Tip-adapter: Training-free clip-adapter for better vision-language modeling," in *European Conference on Computer Vision*, 2022, pp. 494–510.

[29] S. Hayou, N. Ghosh, and B. Yu, "LoRA+: Efficient low rank adaptation of large models," in *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

[30] Y. Li, Y. Yu, C. Liang, P. He, N. Karampatziakis, W. Chen, and T. Zhao, "LoftQ: LoRA-fine-tuning-aware quantization for large language models," in *International Conference on Learning Representations (ICLR)*, 2024.

[31] H. Liu, C. Li, Y. Li, Y. J. Lee, and Y. J. Freivalds, "Improved baselines with visual instruction tuning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 26 296–26 308.

[32] B. Lin, Z. Tang, Y. Ye, J. Cui, B. Zhu, P. Jin, J. Zhang, M. Ning, and L. Yuan, "Moe-llava: Mixture of experts for large vision-language models," 2024.

[33] S. Aburass, O. Dorgham, J. Al Shaqsi, M. Abu Rumman, and O. Al-Kadi, "Vision transformers in medical imaging: a comprehensive review," *Journal of Imaging Informatics in Medicine*, vol. 38, no. 6, pp. 3928–3971, 2025.

[34] P. Gao, S. Geng, R. Zhang, T. Ma, R. Fang, Y. Zhang, H. Li, and Y. Qiao, "Clip-adapter: Better vision-language models with feature adapters," *International Journal of Computer Vision (IJCV)*, vol. 132, pp. 581–595, 2024.

[35] N. Houlsby, A. Giurgiu, S. Jastrzębski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for nlp," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 2790–2799.

[36] Y. Sheng, Z. Gu, Z. Dong, S. Qin, H. Wang, Z. Zhao, W. Zhang, and Z. Li, "S-lora: Serving thousands of concurrent lora adapters," *arXiv preprint arXiv:2311.03285*, 2023.

[37] S. Zhang, Y. Xu, N. Usuyama *et al.*, "Biomedclip: A multimodal biomedical foundation model pretrained from fifteen million scientific image-text pairs," https://arxiv.org/abs/2303.00915, 2023.

[38] F. Liu, D. Chen, Z. Guan, X. Zhou, J. Zhu, Q. Ye, L. Fu, and J. Zhou, "Remoteclip: A vision language foundation model for remote sensing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–16, 2024.

[39] W. G. C. Bandara and V. M. Patel, "Attention prompt tuning: Parameter-efficient adaptation of pre-trained models for spatiotemporal modeling," https://arxiv.org/abs/2403.06978, 2024.

[40] Y. Guo, C. Yang, A. Rao, Z. Liang, Y. Wang, Y. Qiao, M. Agrawala, D. Lin, and B. Dai, "Animatediff: Animate your personalized text-to-image diffusion models without specific tuning," in *International Conference on Learning Representations (ICLR)*, 2024.

[41] P. Saha, D. Wagner *et al.*, "F^{3}ocus: Federated finetuning of vision-language foundation models with optimal client layer updating strategy via multi-objective meta-heuristics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 20 006–20 017.

[42] D. Wang, E. Shelhamer, S. Sha, O. Khurana, E. Geng, P. Abbeel, and T. Darrell, "Tent: Fully test-time adaptation by entropy minimization," in *International Conference on Learning Representations (ICLR)*, 2021.

[43] J. Pfeiffer, S. Ruder, I. Vulić, and E. M. Ponti, "Modular deep learning," *Transactions on Machine Learning Research (TMLR)*, 2023, iSSN 2835-8856.

[44] Q. Zhang, M. Chen, A. Bukharin, P. He, Y. Cheng, W. Chen, and T. Zhao, "Adalora: Adaptive budget allocation for parameter-efficient fine-tuning," in *International Conference on Learning Representations (ICLR)*, 2023.