LoCA: Spatially-Aware Low-Rank Convolutional Adaptation of Vision Foundation Models

Sojung An[1*], Junha Lee[2,3*], Sujeong You[2], Nam Ik Cho[3], Donghyun Kim[1†]

[1] Korea University, Seoul, Republic of Korea
[2] Korea Institute of Industrial Technology, Ansan, Republic of Korea
[3] Seoul National University, Seoul, Republic of Korea

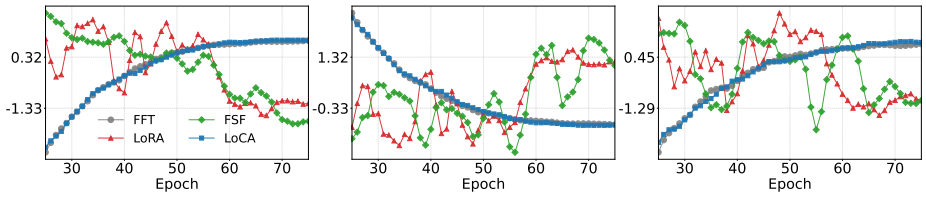
*Equal contribution †Corresponding author: d_kim@korea.ac.kr
https://github.com/ssojungan/loca

Abstract. Pre-trained Vision Foundation Models (VFMs) provide strong visual representations for diverse downstream tasks. The key challenge of VFM adaptation stems from the prohibitive costs of full fine-tuning and catastrophic forgetting. To address this, Low-Rank Adaptation (LoRA) has emerged as the prevailing paradigm for Parameter-Efficient Fine-Tuning (PEFT). However, LoRA is typically designed for transformer self-attention layers parameterized by 2D matrices. Since convolutional kernels inherently couple spatial and channel information within a 4D tensor, forcing them into a monolithic 2D matrix disrupts the inherent spatial topology. In this paper, we propose Low-Rank Convolutional Adaptation (LoCA), a convolution-aware PEFT framework that addresses spatial-channel entanglement by decoupling channel and spatial adaptation. LoCA introduces a low-rank channel adaptation for dense cross-channel mixing and refines spatial bases extracted from pre-trained kernels via Singular Value Decomposition (SVD). Experimental results show that LoCA preserves pre-trained spatial priors and achieves competitive or state-of-the-art performance across fine-grained classification, domain-generalized semantic segmentation, and generative benchmarks.

Keywords: Parameter-Efficient Fine-Tuning · Low-Rank Adaptation · Spatial Inductive Bias · Convolutional Networks

1 Introduction

Vision Foundation Models (VFMs) enable a wide array of general vision tasks [10, 26, 35]. These models learn rich multi-scale representations from large-scale data and transfer effectively to downstream problems through fine-tuning or frozen feature extraction [22, 25]. However, full fine-tuning of large-scale VFMs not only incurs prohibitive computational costs but also leads to catastrophic forgetting of pre-trained knowledge. Parameter-Efficient Fine-Tuning (PEFT) addresses this by optimizing only a small subset of parameters while freezing the original weights [14, 39]. Among these, Low-Rank Adaptation (LoRA) has emerged



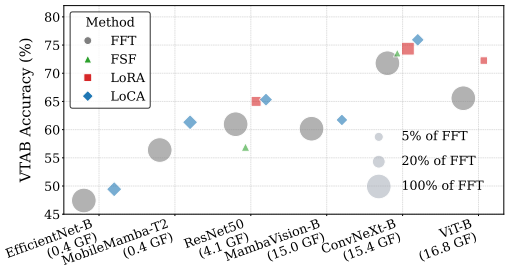
(a) Radial Spread (↑) (b) Center Energy Ratio (↓) (c) Low-Freq Energy Ratio (↑)

Fig. 1: The training dynamics of Effective Receptive Field (ERF) across convolutional layers (*dwconv*) for FFT, LoRA, FSF, and LoCA. (a) Radial Spread: Average distance of gradients from the map center. (b) Center Energy Ratio: Gradient ratio in the 3×3 center. (c) Low-Freq Energy: Fourier low-frequency power ratio. LoRA and FSF exhibit transient ERF growth followed by a reconvergence toward localized patterns, which limit sustained global context. Spatial reconvergence necessitates a structure-preserving design to achieve an expansive receptive field comparable to FFT.

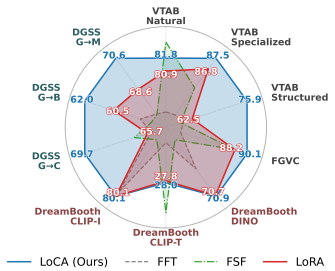
as the de facto standard, offering high representational capacity with minimal trainable parameters [9, 19, 42, 45].

While LoRA is typically designed for linear projections in transformers, convolutional operators in VFM backbones remain relatively underexplored. This gap is critical because convolutional operators remain fundamental across modern VFM backbones. ConvNeXt is a modern convolutional backbone that emphasizes convolution as a sliding-window, weight-sharing strategy that encodes spatial inductive biases [38]. Self-attention in ViT [10] enables global token-to-token interactions but provides limited built-in spatial inductive bias. This prompts hybrid designs that incorporate convolutional components to encode spatial priors [5]. Mamba-based State Space Model (SSM) vision backbones have gained recent attention [12, 15]. Some architectures retain convolutional blocks in early high-resolution stages for local feature extraction. In addition to visual perception, convolutional networks are widely used in generative models such as Stable Diffusion [33] in the U-Net backbone. Given that convolutional operators remain fundamental across modern VFM backbones, extending LoRA beyond Transformer linear layers to convolutional layers becomes an important problem in PEFT.

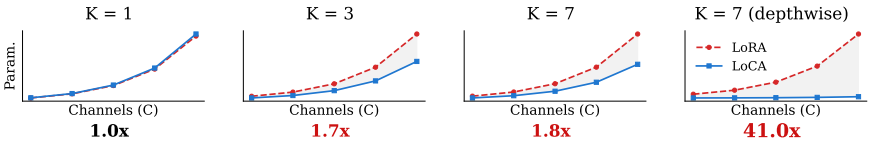
However, directly applying LoRA's low-rank updates to convolution layers is suboptimal, as they operate over spatial regions. Naive LoRA flattens convolutional kernels (4D tensor) into a two-dimensional matrix for low-rank updates [8]. Flattening the convolutional kernel collapses the inherent spatial topology by enforcing cross-channel mixing within a single low-rank parameterization. This structural mismatch limits the preservation of spatial priors, including locality and directionality, and reduces adaptation gains reported in prior work [4, 46]. Recent filter subspace approaches address spatial–channel entanglement by decomposing convolutional filters into spatial bases and channel coefficients [4]. Sparse coding approximates pre-trained kernels within this decomposed subspace. Such approximation inevitably modifies pre-trained representations prior



(a) Parameter-efficient scaling behavior. The x-axis is sorted by GFLOPs.



(b) Generalization performance of PEFT methods across diverse vision benchmarks



(c) Trainable parameter comparison across convolution kernel sizes (K). Detailed analysis is provided in Appendix A.

Fig. 2: Performance comparison of PEFT-based methods on downstream tasks

to fine-tuning. Freezing cross-channel mixing coefficients limits the adaptation of inter-channel correlations in new domains.

As shown in Fig. 1, we analyze the Effective Receptive Field (ERF) [16] and low-frequency retention during fine-tuning. Higher Radial Spread with lower Center Energy Ratio indicates outward gradient propagation and receptive field growth, while higher Low-Freq Energy reflects stronger retention of informative low-frequency structures. FFT and LoCA show consistent increases in spatial coverage and low-frequency energy throughout training. In contrast, LoRA [19] and Filter Subspace Fine-Tuning (FSF) [4] exhibit only transient spatial expansion, followed by reconvergence toward localized update patterns and unstable frequency behavior. These observations suggest that preserving convolutional spatial structure is crucial for maintaining broad receptive fields, motivating our spatial–channel disentanglement approach with joint adaptation.

To this end, we propose Low-Rank Convolutional Adaptation (LoCA), a structured low-rank reparameterization that adapts convolutional layers by decoupling channel mixing from spatial basis refinement. We first introduce a low-rank channel adaptation process that captures dense cross-channel mixing while mitigating spatial–channel entanglement. Such channel adaptation prevents the topological collapse induced by naive kernel flattening and the structural inaccuracies of conventional weight decomposition. We then design a spatial adaptation mechanism that preserves pre-trained priors through structural bases derived from Singular Value Decomposition (SVD). Additionally, we introduce a hierarchical rank scheduling for convolutional foundation models. LoCA preserves the spatial inductive bias of pre-trained representations and achieves robust performance across diverse downstream tasks.

We summarize the contributions of this work as follows:
- We propose the LoCA framework to address spatial-channel entanglement by decoupling channel and spatial adaptation.
- We introduce SVD-based spatial basis refinement to preserve pre-trained spatial inductive biases effectively.
- We propose a hierarchical rank scheduling tailored to convolutional foundation models.
- Extensive experiments demonstrate that LoCA achieves competitive or state-of-the-art performance across fine-grained classification, domain-generalized segmentation, and generative benchmarks (see Fig. 2).

2 Related Work

2.1 Parameter-Efficient Fine-Tuning

PEFT has emerged as a practical paradigm for adapting large-scale VFMs to downstream tasks by updating only a small subset of parameters while freezing pre-trained backbones [19, 20]. Existing approaches include adapter-based methods [3, 18], which insert lightweight trainable modules into network blocks; prompt-based methods [20,24], which add learnable tokens to the input sequence; selective fine-tuning methods [1, 13], which update specific components such as bias terms; and reparameterization-based methods [9, 19, 42, 45], which optimize implicit low-rank structures for seamless merging into the original weights at inference.

Most PEFT techniques are formulated for Transformer architectures that operate on token sequences with multi-head self-attention [30]. Extending these paradigms to vision tasks such as detection and segmentation remains challenging because pixel-level prediction relies on spatial inductive biases and multi-scale hierarchies.

2.2 Parameter-Efficient Fine-Tuning for Convolutional Layers

Convolution preserves 2D image structure by leveraging inherent spatial inductive biases [26, 27]. Early convolution-specific PEFT methods such as Conv-Adapter [3] introduce trainable modules into convolutional blocks and increase inference-time computation. Flattening-based extensions convert 2D convolutional kernels parameterized as 4D tensors into 2D matrices to reuse linear LoRA formulations [8]. This dimensional collapse entangles spatial topology with cross-channel mixing, compromising locality and weight sharing [4, 46].

Filter subspace methods constrain updates by decomposing pre-trained kernels into spatial atoms and mixing coefficients [4]. This decomposition reconstructs the pre-trained weights only approximately while introducing accumulation error. Reconstructing weights from these decomposed atoms can also restrict cross-channel flexibility by freezing mixing coefficients. These limitations motivate a convolution-aware PEFT that preserves the original weights while enabling spatially structured low-rank updates.

2.3 Low-Rank Adaptation and Singular Value Decomposition

LoRA represents weight adaptation using low-rank factors. To maximize parameter efficiency, subsequent methods have explored rank adaptation. For example, AdaLoRA dynamically adjusts the rank budget across different layers based on importance scores [45]. Other works leverage SVD to replace random initialization with informed low-rank initialization. PiSSA [28], SoMA [42], and SoRA [9] decompose pre-trained weights and use principal or minor components to initialize low-rank factors, improving convergence and knowledge retention. These works suggest that singular components capture reusable structure in pre-trained weights. Building upon this idea, we employ SVD to extract spatial bases from convolutional kernels.

3 Preliminaries

LoRA. LoRA freezes pre-trained weights and approximates updates using low-rank matrices. Motivated by the hypothesis that weight changes during model adaptation possess a low intrinsic rank, LoRA parameterizes the incremental update via the product of two low-rank matrices [19]. For a pre-trained weight matrix $W_0 \in \mathbb{R}^{d_{out} \times d_{in}}$, where d_{in} and d_{out} denote the input and output dimensions respectively, LoRA decomposes the update $\Delta W \in \mathbb{R}^{d_{out} \times d_{in}}$ into BA, where $B \in \mathbb{R}^{d_{out} \times r}$ and $A \in \mathbb{R}^{r \times d_{in}}$ are low-rank matrices with rank $r \ll \min(d_{out}, d_{in})$. and α is a constant scaling factor. Consequently, the fine-tuned weight W' is formulated as:

$$W' = W_0 + \Delta W = W_0 + \frac{\alpha}{r} BA \tag{1}$$

where W_0 remains frozen during training and α denotes a constant scaling factor. One factor is zero-initialized, yielding $\Delta W = 0$ at initialization.

A 2D convolutional layer is parameterized by a 4D weight tensor $W_0 \in \mathbb{R}^{C_{out} \times C_{in} \times k_h \times k_w}$ with C_{out} output channels, C_{in} input channels, and spatial kernel size $k_h \times k_w$. LoRA is formulated for matrix multiplication, so a naive extension to convolution reshapes the kernel tensor into a matrix. This reshaping merges the spatial dimensions into the input-channel axis and produces the flattened weight $W_0^\flat \in \mathbb{R}^{C_{out} \times (C_{in} k_h k_w)}$. The low-rank update is computed in the flattened space by applying Eq. (1) to $W_0^\flat$:

$$W'^\flat = W_0^\flat + \frac{\alpha}{r} BA, \tag{2}$$

where $B \in \mathbb{R}^{C_{\text{out}} \times r}$ and $A \in \mathbb{R}^{r \times (C_{in} k_h k_w)}$.

Filter Subspace Adaptation. Filter Subspace Fine-tuning (FSF) was proposed to represent each convolution filter as a linear combination of spatial elements referred to as filter atoms. Formally, Chen et al. [4] decompose a pre-trained convolutional layer $W_0 \in \mathbb{R}^{C_{out} \times C_{in} \times k_h \times k_w}$ into a filter atom layer $\mathbf{D} \in \mathbb{R}^{m \times k_h \times k_w}$ and an atom coefficient layer $\boldsymbol{\alpha} \in \mathbb{R}^{C_{out} \times C_{in} \times m}$:

$$W_0 = \boldsymbol{\alpha} \times \mathbf{D}. \tag{3}$$

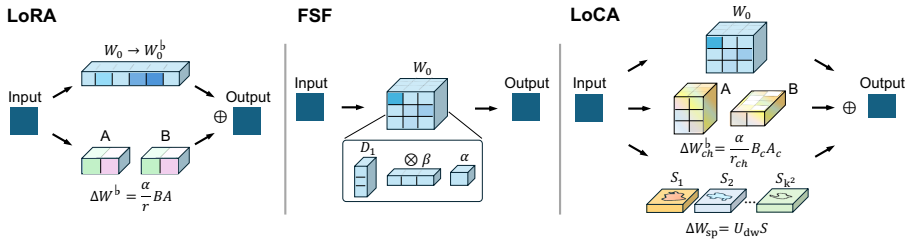


Fig. 3: Convolutional kernel adaptation architectures. (a) LoRA: Decomposes weights into a frozen W_0 and a low-rank update $\Delta W = BA$. (b) FSF: Decomposes W_0 into spatial atom bases (D_1), intra-channel mixing components (β), and cross-channel atom coefficients (α). (c) LoCA (ours): Learns only ΔW while freezing W_0, integrating spatial information via learned basis $\mathcal{S}$ for stable channel–spatial structural adaptation.

This indicates that each filter slice $W_0^{i,j} \in \mathbb{R}^{k_h \times k_w}$ is constructed by a linear combination of the filter atoms: $W_0^{i,j} = \sum_{l=1}^{m} \boldsymbol{\alpha}^{i,j,l} \mathbf{d}_l$. $\mathbf{D} = \{\mathbf{d}_l\}_{l=1}^{m}$ denotes a set of m filter atoms for spatial convolution and $\boldsymbol{\alpha}$ controls spatially invariant cross-channel mixing. Sparse coding initializes these components by minimizing reconstruction error on the pre-trained weights. Each filter atom $\mathbf{d}_l$ can be recursively decomposed into an overcomplete atom set $\mathbf{D}_1 \in \mathbb{R}^{(m \cdot m_1) \times k_h \times k_w}$ using intra-channel mixing coefficients $\boldsymbol{\beta} \in \mathbb{R}^{(C_{in} \cdot m) \times m_1}$. FSF updates only the spatial atoms ($\mathbf{D}$ or $\mathbf{D}_1$) while freezing $\boldsymbol{\alpha}$ to retain pre-trained generalization.

4 Low-Rank Convolutional Adaptation

This section introduces LoCA as a framework that preserves spatial inductive bias during convolutional layer adaptation. LoCA decouples the adaptation into low-rank channel adaptation (Sec. 4.1) and spatial basis refinement (Sec. 4.2). The low-rank channel adaptation isolates dense cross-channel mixing to resolve spatial–channel entanglement, while spatial basis refinement optimizes SVD-derived structural bases to preserve the inherent spatial topology. The independent channel and spatial paths are then composed to form the decoupled LoCA design (Sec. 4.3). Finally, we introduce hierarchical rank scheduling for convolutional vision backbones (Sec. 4.4).

4.1 Low-Rank Channel Adaptation

To explicitly resolve the spatial-channel entanglement, we establish a dedicated low-rank channel adaptation mechanism that isolates dense cross-channel dependencies from spatial topology. Building upon the flattened formulation in Eq. (2), we define the low-rank channel adaptation term. Let r_{ch} denote the channel rank and α the scaling factor. The update is defined as:

$$\Delta W_{\text{ch}}^{\flat} = \frac{\alpha}{r_{\text{ch}}} B_c A_c, \tag{4}$$

where $B_c \in \mathbb{R}^{C_{\text{out}} \times r_{\text{ch}}}$ and $A_c \in \mathbb{R}^{r_{\text{ch}} \times (C_{\text{in}} k_h k_w)}$. We then reshape $\Delta W_{\text{ch}}^{\flat}$ back to its original 4D spatial structure, denoted as $\Delta W_{\text{ch}} \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}} \times k_h \times k_w}$. To guarantee functional equivalence to the pre-trained model at initialization, A_c is initialized using a Kaiming uniform distribution, and B_c is initialized to zero. This zero-initialization ensures $\Delta W_{\text{ch}} = 0$ at the start of training, preserving the original pre-trained representations without requiring kernel replacement.

4.2 SVD-based Spatial Basis Refinement

Naive flattening collapses the $k_h \times k_w$ spatial structure, making it difficult to isolate spatial modulation from dense channel transformations. We instead parameterize spatial adaptation with a compact set of pre-trained spatial bases. In practice, we reshape each pre-trained kernel slice into a length-$k_h k_w$ vector and apply zero-mean and unit-variance standardization to obtain $W_{\text{norm}} \in \mathbb{R}^{C_{\text{out}} C_{\text{in}} \times k_h k_w}$. We then form a spatial covariance matrix $C_{\text{sp}} = W_{\text{norm}}^{\top} W_{\text{norm}} \in \mathbb{R}^{k_h k_w \times k_h k_w}$. Specifically, we compute the SVD of the spatial covariance induced by the pre-trained kernel, set the spatial rank as $r_{\text{sp}} = k_h k_w$, and obtain an initial basis tensor $\mathcal{S} \in \mathbb{R}^{r_{\text{sp}} \times k_h \times k_w}$. The deterministic basis $\mathcal{S}$ is initialized from this SVD and treated as a learnable parameter for refinement. Each slice $\mathcal{S}_m \in \mathbb{R}^{k_h \times k_w}$ corresponds to the m-th principal spatial pattern, such as edges or textures. Channel-specific coefficients are learned with $U_{\text{dw}} \in \mathbb{R}^{D \times r_{\text{sp}}}$ where $D = \min(C_{\text{out}}, C_{\text{in}})$. The spatial update $\Delta W_{\text{sp}}^{\text{diag}}[i] \in \mathbb{R}^{k_h \times k_w}$ for channel i is defined by a basis expansion:

$$\Delta W_{\text{sp}}^{\text{diag}}[i] = \sum_{m=1}^{r_{\text{sp}}} U_{\text{dw}}[i, m] \cdot \mathcal{S}_m. \tag{5}$$

The spatial tensor is defined on the depthwise diagonal with the Kronecker delta δ_{ij} (i.e., $\delta_{ij} = 1$ if $i = j$ and 0 otherwise). Since $i = j$ implicitly guarantees $i \leq \min(C_{\text{out}}, C_{\text{in}}) = D$, the index is safely bounded:

$$\Delta W_{\text{sp}}[i, j, :, :] = \delta_{ij} \, \Delta W_{\text{sp}}^{\text{diag}}[i]. \tag{6}$$

Diagonal parameterization separates spatial refinement from cross-channel mixing.

4.3 Channel-Spatial Composition

We compose the independent channel and spatial paths to synthesize our decoupled design into a unified adaptation framework without distorting pre-trained knowledge. The combined update tensor $\Delta W_{cs} \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}} \times k_h \times k_w}$ adds the spatial update on the depthwise diagonal and uses ΔW_{ch} for cross-channel mixing:

$$\Delta W_{cs}[i, j, :, :] = \Delta W_{\text{ch}}[i, j, :, :] + \delta_{ij} \Delta W_{\text{sp}}^{\text{diag}}[i] \tag{7}$$

Initializing U_{dw} and B_c to zero ensures $\Delta W_{cs} = 0$, thereby preserving exact functional equivalence to the frozen pre-trained model. The final adapted weight W' is defined as follows:

$$W' = W_0 + \Delta W_{cs}. \tag{8}$$

4.4 Hierarchical Rank Scheduling

Convolution-based VFMs adopt a hierarchical architecture that encodes features through progressively increasing channel dimensions across stages [33, 38]. Early stages extract local features with narrower channels, while deeper stages encode global semantics with wider channels. A fixed rank applied uniformly across stages limits the ability to capture this hierarchical diversity. To address this, we introduce hierarchical rank scheduling to determine the channel rank $r_{\text{ch}}^{(s)}$ based on the stage-specific width $C_{\text{out}}^{(s)}$. We compute a base rank as $\lfloor R \cdot \frac{C_{\text{out}}^{(s)}}{\sum_i C_{\text{out}}^{(i)}} \rfloor$, where R is the global rank budget. By aligning adaptation capacity with layer width, this strategy ensures that the narrower stem receives smaller ranks while deeper layers are allocated larger ranks to model complex semantic features. While the channel rank is scaled across stages, the spatial rank r_{sp} (as defined in Sec. 4.2) remains fixed to the localized spatial kernel size ($k_h k_w$). This strategy improves parameter efficiency without sacrificing adaptation performance. Empirical validation is provided in Sec. 5.4.

5 Experiment

In this section, we evaluate LoCA's effectiveness through experiments across three distinct tasks: (1) fine-grained visual adaptation (Sec. 5.1) using the VTAB-1k and FGVC datasets; (2) generative generalization performance (Sec. 5.2) via the DreamBooth dataset; and (3) domain generalized semantic segmentation (DGSS).

5.1 Fine-grained Adaptation on VTAB-1k and FGVC

Experimental Setup. Our evaluation utilizes the FGVC and VTAB-1k benchmarks. The FGVC comprises four fine-grained recognition tasks: CUB-200-2011 [37], Stanford Dogs [21], Stanford Cars [23], and NABirds [36]. The VTAB-1k benchmark partitions downstream tasks into Natural, Specialized, and Structured semantic domains.

Results. We compare the proposed LoCA with existing PEFT methods on VTAB-1k and FGVC benchmarks. As shown in Tab. 1, we report LoCA results on ConvNeXt-B and ResNet-50. On VTAB-1k, rank-16 (r16) generally achieves the highest accuracy, while ConvNeXt-B exhibits only marginal gains from increasing the rank. The marginal gains on ConvNeXt-B stem from the smaller dimensionality of convolutional kernels compared with linear layers. LoCA achieves the best VTAB-1k average accuracy with rank-4 on ConvNeXt-B and rank-16 on ResNet-50. On ConvNeXt-B, LoCA reaches this performance with only 0.97 M trainable parameters. In Tab. 2, LoCA with rank-16 outperforms the other PEFT methods on FGVC using both ConvNeXt-B and ResNet-50. LoRA applies low-rank approximation after kernel flattening, which increases the number of additional parameters to 17.58 M on ConvNeXt-B while yielding lower accuracy. Due to architectural differences between ConvNeXt (`Conv. Seq.`) and ResNet

Table 1: Performance comparison on the VTAB-1k visual classification benchmark using ConvNeXt-B and ResNet-50 backbones. We compare LoCA (in blue) with full Fine-Tuning (FFT) (in gray), Linear Probing (LP), Partial [40], MLP [18], Bias Tuning (Bias) [1], Visual Prompt Tuning (VPT) [20], Conv-Adapter [3], and Filter Subspace Fine-Tuning (FSF) [4]. Param. represents the trainable parameters (M).

Tuning	Param.	ConvNeXt-B				Param.	ResNet-50			
		Natural	Specialized	Structured	Average		Natural	Specialized	Structured	Average
# Tasks	-	7	4	8	19	-	7	4	8	19
FFT	87.62	80.52	87.54	63.85	74.98	23.61	65.58	82.0	52.32	63.45
LP	1.68	74.48	81.50	34.76	59.23	0.48	63.75	77.60	30.96	52.89
Partial-1 [40]	4.72	73.76	81.64	39.55	61.01	2.10	64.34	78.64	45.78	59.51
MLP-3 [18]	2.45	73.78	81.36	35.68	59.33	3.51	61.79	70.77	33.97	51.97
Bias [1]	1.76	69.07	72.81	25.29	51.42	0.49	63.51	77.22	33.39	53.85
VPT [20]	1.75	78.48	83.00	44.64	65.18	0.49	66.25	77.32	37.52	56.09
LoRA [19]	17.32	80.89	86.80	62.46	74.36	1.90	65.06	82.53	**56.21**	65.01
CA [3]	6.83	80.62	86.29	64.88	75.18	1.37	64.20	81.33	52.74	64.78
CoLoRA [32]	4.57	76.1	83.1	58.2	70.0	1.40	66.6	82.6	51.9	63.8
FSF [4]	1.11	**82.96**	85.53	59.59	73.59	0.72	62.64	80.25	36.50	56.91
LoCA (r4)	0.97	82.22	**87.54**	64.74	**75.98**	0.47	66.08	82.41	52.44	63.77
LoCA (r16)	5.01	81.81	87.51	**65.01**	75.93	1.51	**67.21**	**83.61**	54.52	**65.32**

Table 2: Average Top-1 accuracy (%) on FGVC datasets

Tuning	ConvNeXt-B		ResNet-50	
	# Param.	Average	# Param.	Average
FFT	87.87	79.73	24.14	75.73
LP	0.31	77.55	0.62	45.39
Bias [1]	0.44	64.98	0.67	48.85
LoRA [19]	17.58	88.18	2.43	80.96
CA [3]	6.13	89.28	2.23	83.48
CoLoRA [32]	4.57	86.11	0.99	76.44
FSF [4]	1.11	88.04	2.52	81.07
LoCA (r16)	3.70	**90.06**	1.94	**83.67**

Table 3: Quantitative comparison of subject alignment

Methods	DINO↑	CLIP-I↑	CLIP-T↑
Pretrained	0.320	0.643	0.267
Real Images	0.711	0.857	–
Textual Inversion [11]	0.564	0.739	0.213
DreamBooth [34]	0.642	0.794	0.236
LoRA [19]	0.637	0.792	0.239
↪ w/ Conv	0.707	0.801	0.278
FSF [4]	0.572	0.715	**0.313**
LoCA (r16)	**0.709**	**0.801**	0.280

(`Res. Par.`), we use sequential CA insertion for ConvNeXt and parallel $k \times k$ CA adaptation for ResNet, following the best-performing placements reported in Conv-Adapter [3].

Generalization across Backbones. The performance of LoCA is evaluated against FFT on convolution-based VFM backbones, including ResNet, ConvNeXt, MambaVision, EfficientNet, and MobileMamba. All experiments are conducted using base models. Fig. 4 shows that LoCA outperforms FFT across diverse backbones. On MobileMamba, LoCA improves Top-1 accuracy over FFT, in-

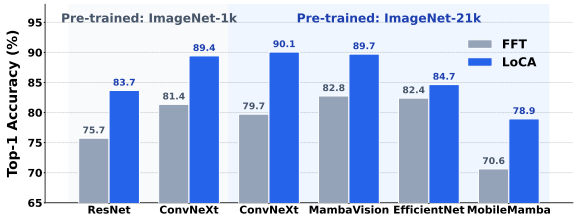


Fig. 4: Performance comparison with FFT across various backbone architectures

Fig. 5: Qualitative results of subject-driven task. We visualize the results to compare PEFT methods: LoRA [19], FSF [4], and LoCA.

creasing it from 70.6% to 78.9%. The results demonstrate consistent gains across architectures that incorporate convolution operations. Detailed results are provided in Appendix B.

5.2 Generative Generalization with DreamBooth

Experimental Setup. For the generative task, comparative experiments involving DreamBooth [34], LoRA [19], and FSF [4] all utilize Stable Diffusion v1.4 [33] under identical training configurations. Performance analysis follows the DreamBooth evaluation protocol by using images generated from 25 prompts. Quantitative assessment focuses on text alignment, where DINO [2] and CLIP-I [31] measure subject fidelity while CLIP-T [31] evaluates text prompt fidelity. CLIP-I and DINO calculate the average cosine similarity between the embeddings of generated and ground truth images based on CLIP and ViT-S/16-based DINO, respectively. Similarly, CLIP-T computes the average cosine similarity between the text prompt and the generated image embeddings.

Results. We evaluate LoCA against PEFT approaches on subject-driven generation. Tab. 3 shows that LoCA achieves strong subject alignment and competitive text alignment. LoCA outperforms LoRA by achieving the best DINO score and tying for the best CLIP-I score, while maintaining a competitive CLIP-T score. Convolution-based adaptation incorporates spatial inductive biases for subject alignment, whereas LoRA applies low-rank updates to linear projections.

Visualization. Fig. 5 shows that LoCA better retains the structural identity of the reference subject. For qualitative evaluation, we visualize results from LoRA [19], FSF [4], and LoCA across four classes. LoRA preserves subject identity but demonstrates limited reflection of textual attributes (e.g., 'chef outfit' or 'city in the background'). FSF [4] captures textual attributes but struggles to preserve

Table 4: Performance comparison on synthetic-to-real DGSS using various backbones and model sizes. Models are trained on GTAV and evaluated on Cityscapes, BDD100K, and Mapillary.

Method	Backbone	Param.	Trainable Param.	GFLOPs	Citys.	BDD	Map.	Avg.
DINO Pre-trained								
FFT	ViT-B	86.5M	86.5M	216	60.84	52.98	62.12	58.65
SoMA	ViT-B	86.5M	2.3M	216	66.71	57.48	67.34	63.84
FFT	ConvNeXt-B	87.56M	87.56M	81	62.18	57.01	65.00	61.40
LoRA (Linear)	ConvNeXt-B	87.56M	2.9M	81	63.90	57.87	65.53	62.43
LoRA	ConvNeXt-B	87.56M	17.2M	81	64.17	56.98	65.74	62.30
SoMA	ConvNeXt-B	87.56M	2.9M	81	64.99	57.65	65.67	62.77
CA	ConvNeXt-B	87.56M	2.3M	81	59.94	56.47	63.48	59.96
CoLoRA	ConvNeXt-B	87.56M	2.2M	81	61.87	55.70	64.04	60.53
FSF	ConvNeXt-B	87.56M	0.6M	81	60.15	56.98	63.70	60.27
LoCA	ConvNeXt-B	87.56M	3.4M	81	**66.46**	**58.53**	66.29	**63.76**
LoCA‡	ConvNeXt-B	87.56M	3.4M	81	65.56	58.02	**66.39**	63.32
FFT	ConvNeXt-L	196.2M	196.2M	152	65.50	59.10	67.01	63.87
LoRA (Linear)	ConvNeXt-L	196.2M	4.3M	152	66.95	60.46	68.45	65.29
LoRA	ConvNeXt-L	196.2M	25.9M	152	65.74	60.50	68.62	64.95
SoMA	ConvNeXt-L	196.2M	4.3M	152	68.85	60.27	69.26	66.13
CA	ConvNeXt-L	196.2M	4.5M	152	63.16	58.41	67.32	62.96
CoLoRA	ConvNeXt-L	196.2M	3.6M	152	64.51	59.56	67.20	63.59
FSF	ConvNeXt-L	196.2M	0.8M	152	66.57	58.52	66.96	64.02
LoCA	ConvNeXt-L	196.2M	5.0M	152	68.54	61.60	69.33	66.49
LoCA‡	ConvNeXt-L	196.2M	5.0M	152	**69.73**	**62.03**	**70.62**	**67.46**
ImageNet21k Pre-trained								
FFT	ResNet101	42.3M	42.3M	42	41.29	44.29	48.79	44.79
SoMA†	ResNet101	42.3M	2.5M	42	41.23	45.57	**49.71**	45.50
LoCA	ResNet101	42.3M	2.8M	42	**44.82**	**46.13**	49.21	**46.72**
FFT	MambaVision-B	96.7M	96.7M	211	36.05	30.13	31.39	32.52
LoCA	MambaVision-B	96.7M	2.5M	211	**45.21**	**41.68**	**45.70**	**44.20**
FFT	MambaVision-L3	737.5M	737.5M	1,556	52.88	45.87	56.07	51.61
LoCA	MambaVision-L3	737.5M	9.0M	1,556	**59.94**	**50.61**	**61.39**	**57.31**

† PEFT via linearization of patch-level convolutions and weights.
‡ Channel mixing path initialization follows the SVD-based approach of SoMA [42].

subject identity and capture fine-grained visual details. For the vase-class prompt 'a [V] floating on top of water', FSF generates a bulky bottle-like object rather than the intended subject. In contrast, LoCA preserves subject identity while better reflecting textual attributes. Detailed qualitative comparisons appear in Appendix C.

5.3 Domain Generalization for Semantic Segmentation

Experimental Setup. Experiments utilize convolution-based backbones, specifically ResNet [17], and ConvNeXt [38]. Evaluation further includes recent vision foundation models with hybrid transformer–convolution architectures, including DINOv3-ConvNeXt [35] and MambaVision [15]. All models employ a consistent Mask2Former [6] decoder for DGSS and DGOD tasks. A fixed rank of 16 across

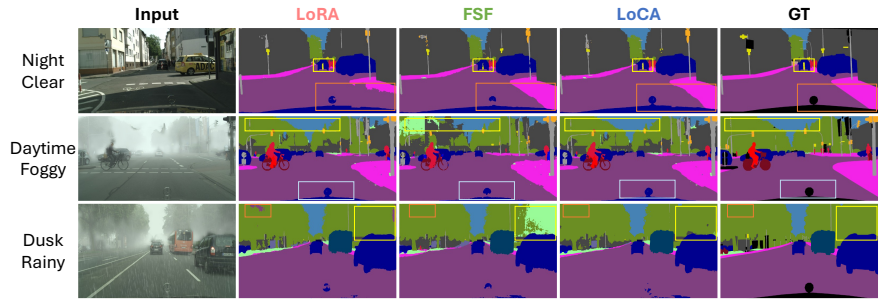


Fig. 6: Qualitative domain adaptation results (GTAV → Cityscapes)

all experiments ensures a fair comparison. For DGSS and DGOD tasks, models are trained on the GTAV dataset and evaluated on three real-world benchmarks: Cityscapes [7], BDD100K [41], and Mapillary [29]. Appendix D presents detailed experimental settings and performance analysis for different backbones.

Results. As shown in Tab. 4, we evaluate the generalization capability of LoCA on DGSS by training on GTAV and testing on real-world out-of-distribution (OOD) benchmarks. LoCA with DINO-pretrained ConvNeXt backbones [35] demonstrates competitive performance compared to other PEFT methods on OOD datasets. LoRA-based adaptation flattens convolution kernels and requires learning a large number of parameters. LoCA achieves strong performance with substantially fewer trainable parameters than LoRA. Initializing low-rank components with the principal singular components following SoMA boosts performance by approximately +1.0 on average across three datasets in ConvNeXt-L. Conversely, zero initialization outperforms this method for ConvNeXt-B. The gain observed in ConvNeXt-L suggests that larger models retain more salient eigenvalue structures from pre-trained weights. Benchmarking against SoMA results on ViT-B [10], we evaluated ConvNeXt-B with a comparable parameter count. ConvNeXt-B achieves performance parity with ViT-B at 81 GFLOPs, representing a 2.6× reduction from the 216 GFLOPs required by ViT-B. The ConvNeXt-B-based LoCA maintains high accuracy while mitigating computational overhead. For DGSS, models are trained on GTAV and evaluated on Cityscapes, BDD100K, and Mapillary. Detailed DGOD results are provided in Appendix D.

Visualization. Qualitative evaluations utilize ConvNeXt-B to compare LoRA [19], FSF [4], and LoCA on night clear, foggy, and rainy Cityscapes scenes [7]. These challenging environments serve as a benchmark for robustness. As shown in Fig. 6, our method generates fine-grained segmentation maps in both rainy and foggy scenes. LoCA preserves sharp object boundaries and captures fine-grained details in the boxed regions. LoCA exhibits stronger robustness to weather-induced noise than LoRA and FSF.

Generalization across Backbones. Recent vision foundation models adopt convolutional architectures such as Mamba-based designs [12]. We adapt convolutional components during fine-tuning of MambaVision [15], ResNet [17], and

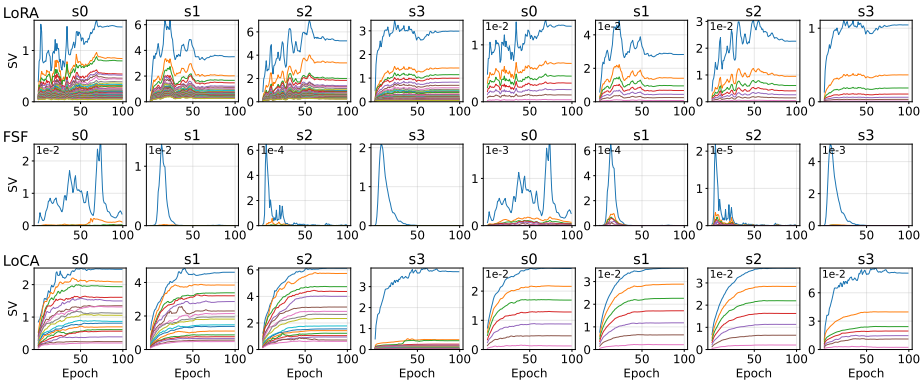


Fig. 7: Singular value evolution of ConvNeXt-B depthwise convolution weights across three PEFT methods: LoRA, FSF, and LoCA. The y-axis denotes singular values. Columns 1–4: channel-wise SVD of depthwise weights; columns 5–8: spatial SVD. Stable distributions prevent spectral tail growth and ensure controlled effective rank behavior.

ConvNeXt [26]. On MambaVision-B, LoCA improves the average score from 32.52 to 44.20, corresponding to +11.68 points or a 35.9% relative gain over FFT. Tab. 4 further shows that LoCA achieves competitive performance while using only about 2.5% of the parameters.

5.4 Ablation Studies

Evolution of Representational Capacity. The evolution of singular value spectra during training reflects representational capacity during adaptation. A distributed singular value spectrum can indicate effective representation learning, with information distributed across multiple components rather than concentrated in a few dominant ones [43]. To analyze this spectral evolution, we track the singular value spectra of a ConvNeXt-B depthwise convolution weight matrix (dwconv) on VTAB-1k Oxford Pets. As shown in Fig. 7, LoRA and FSF exhibit limited variation in their basis components across training epochs. In contrast, LoCA exhibits progressively diverse basis components with smooth singular value growth across both leading and trailing components. This spectral evolution reflects kernel importance identified through covariance analysis and enables joint channel–spatial adaptation.

Spatial Representation Analysis. We validate the effectiveness of LoCA in convolutional adaptation by visualizing the orthogonality and spatial coverage of rank components. Fig. 8 shows the cosine similarity among rank components and their spatial coverage. LoRA rank components exhibit higher similarity and tend to collapse into redundant feature directions. In contrast, LoCA produces more orthogonal rank components, which encourages diverse subspace representations and improves convolutional expressivity. The increased diversity of rank components becomes more evident at the patch level. The highly activated patches in LoCA are distributed across diverse spatial regions rather than localized to

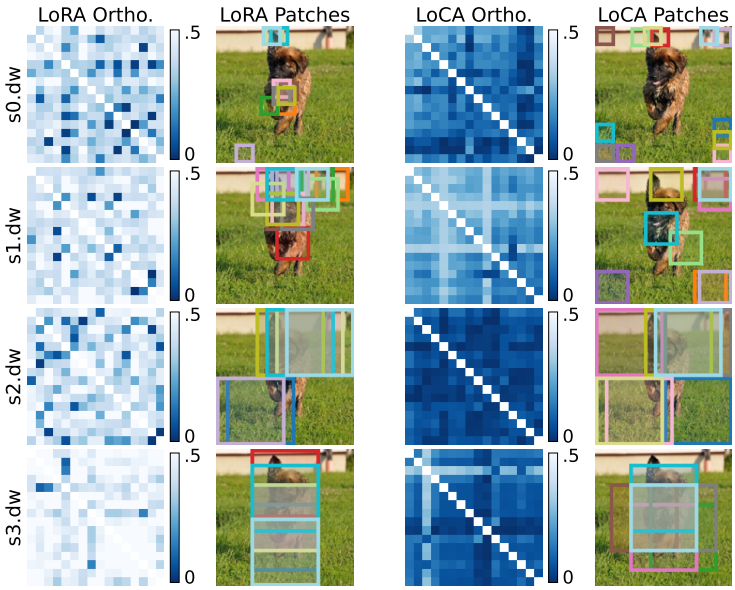


Fig. 8: Absolute pairwise cosine similarity and top-activated localized patches [44] of trained rank components across stages (s0–3). Localized patches denote image regions with the strongest activation responses for each rank component.

a single spatial cue. LoRA focuses on localized regions, whereas LoCA learns orthogonal rank components with broader spatial coverage.

Ablation Analysis of Proposed Methods. We analyze the contribution of each proposed method relative to a LoRA baseline. The analysis compares incremental variants, including standard convolutional LoRA, channel mixing, spatial basis refinement, and hierarchical rank scheduling. As shown in Tab. 5, applying the standard convolutional LoRA formulation decreases the average score by 0.34 points. Naively applying LoRA to convolutional kernels does not improve adaptation performance and may degrade OOD performance. In contrast, the proposed channel mixing increases the average score to 66.16, yielding a +0.87 gain over LoRA Linear. Adding spatial basis refinement further improves the average score to 67.03, yielding a +1.74 gain over LoRA Linear. Hierarchical rank scheduling achieves the best average score of 67.46, corresponding to a +2.17 gain over LoRA Linear. Convolutional architectures encode coarse-to-fine information across stages. Hierarchical rank scheduling aligns the adaptation capacity with this structural property.

Table 5: Ablation study of proposed methods under DGSS benchmarks

Tuning Method	Citys.	BDD	Map.	Avg.
LoRA Linear	66.95	60.46	68.45	65.29
⌞ LoRA Convolution	65.74 ▽1.21	60.50 △0.04	68.62 △0.17	64.95 ▽0.34
⌞ Channel Mixing	68.22 △1.27	61.12 △0.66	69.13 △0.68	66.16 △0.87
⌞ Spatial Basis	69.44 △2.49	61.39 △0.93	70.26 △1.81	67.03 △1.74
⌞ Hierarchical Rank	69.73 △2.78	62.03 △1.57	70.62 △2.17	67.46 △2.17

Analysis of Covariance-SVD Initialization. We compare covariance-SVD initialization with other initialization strategies on VTAB-1k and DGSS data. Across diverse tasks, covariance-SVD initialization consistently outperforms other initialization strategies. The consistent gains suggest that preserving the directional structure of convolutional feature subspaces benefits representation learning. Covariance-SVD preserves pre-trained spatial priors because SVD is applied to spatial covariance rather than a flattened convolution tensor. Covariance-SVD performs best on both VTAB-1k and DGSS (Tab. 6).

Table 6: Ablation study on initialization strategies

Initialization	Zero	Flatten SVD	Uniform	Covariance
VTAB-1k Avg.	75.7	73.0	75.7	**75.9**
DGSS Avg.	65.6	66.2	66.3	**66.5**

6 Conclusions

Although LoRA is the dominant PEFT approach, convolutional adaptation remains underexplored despite the centrality of spatial information to visual adaptation. To this end, we present Low-Rank Convolutional Adaptation (LoCA), a convolution-aware PEFT framework for adapting vision foundation models. LoCA decouples the adaptation into low-rank channel adaptation and spatial basis refinement. This convolution-aware PEFT framework addresses spatial-channel entanglement by decoupling channel and spatial adaptation while preserving pre-trained spatial priors. Furthermore, our hierarchical rank scheduling aligns adaptation capacity with the backbone's hierarchical feature extraction. Experimental results show that LoCA achieves strong performance across tasks and backbones, outperforming existing methods on several benchmarks and remaining competitive on others.

Acknowledgements

This research was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2019-II190079, Artificial Intelligence Graduate School Program (Korea University), 1%; No. RS-2025-25439490, 40%), Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024 (No. RS-2024-00345025, International Collaborative Research and Global Talent Development for the Development of Copyright Management and Protection Technologies for Generative AI, 10%), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2024-00341514, 39%), the Industrial Technology Innovation Program (No. RS-2025-25448266, Development of Humanoid Robots Specialized in Display Manufacturing Processes Based on AI Foundation Models, 10%) grant funded by the Korea government (MOTIE).

References

1. Cai, H., Gan, C., Zhu, L., Han, S.: Tinytl: Reduce memory, not parameters for efficient on-device learning. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 11285–11297. Curran Associates, Inc. (2020), https://proceedings.neurips.cc/paper_files/paper/2020/file/81f7acabd411274fcf65ce2070ed568a-Paper.pdf
2. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
3. Chen, H., Tao, R., Zhang, H., Wang, Y., Li, X., Ye, W., Wang, J., Hu, G., Savvides, M.: Conv-adapter: Exploring parameter efficient transfer learning for convnets. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 1551–1561 (June 2024)
4. Chen, W., Miao, Z., Qiu, Q.: Large convolutional model tuning via filter subspace. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=E5YmIBvOqV
5. Chen, Z., Duan, Y., Wang, W., He, J., Lu, T., Dai, J., Qiao, Y.: Vision transformer adapter for dense predictions. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=plKu2GByCNW
6. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1290–1299 (June 2022)
7. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
8. Ding, C., Cao, X., Xie, J., Fan, L., Wang, S., Lu, Z.: Lora-c: Parameter-efficient fine-tuning of robust cnn for iot devices. arXiv preprint arXiv:2410.16954 (2024)
9. Ding, N., Lv, X., Wang, Q., Chen, Y., Zhou, B., Liu, Z., Sun, M.: Sparse low-rank adaptation of pre-trained language models. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. pp. 4133–4145. Association for Computational Linguistics, Singapore (Dec 2023). https://doi.org/10.18653/v1/2023.emnlp-main.252, https://aclanthology.org/2023.emnlp-main.252/
10. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=YicbFdNTTy
11. Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A.H., Chechik, G., Cohen-Or, D.: An image is worth one word: Personalizing text-to-image generation using textual inversion (2022). https://doi.org/10.48550/ARXIV.2208.01618, https://arxiv.org/abs/2208.01618
12. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. In: First Conference on Language Modeling (2024), https://openreview.net/forum?id=tEYskw1VY2
13. Guo, D., Rush, A.M., Kim, Y.: Parameter-efficient transfer learning with diff pruning. In: Proceedings of the 59th annual meeting of the association for computa-

tional linguistics and the 11th international joint conference on natural language processing (volume 1: Long papers). pp. 4884–4896 (2021)
14. Han, Z., Gao, C., Liu, J., Zhang, J., Zhang, S.Q.: Parameter-efficient fine-tuning for large models: A comprehensive survey. Transactions on Machine Learning Research (2024), https://openreview.net/forum?id=lIsCS8b6zj
15. Hatamizadeh, A., Kautz, J.: Mambavision: A hybrid mamba-transformer vision backbone. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 25261–25270 (2025)
16. He, H., Zhang, J., Cai, Y., Chen, H., Hu, X., Gan, Z., Wang, Y., Wang, C., Wu, Y., Xie, L.: Mobilemamba: Lightweight multi-receptive visual mamba network. In: Proceedings of the computer vision and pattern recognition conference. pp. 4497–4507 (2025)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)
18. Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., Attariyan, M., Gelly, S.: Parameter-efficient transfer learning for NLP. In: Proceedings of the 36th International Conference on Machine Learning (2019)
19. Hu, E.J., yelong shen, Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-rank adaptation of large language models. In: International Conference on Learning Representations (2022), https://openreview.net/forum?id=nZeVKeeFYf9
20. Jia, M., Tang, L., Chen, B.C., Cardie, C., Belongie, S., Hariharan, B., Lim, S.N.: Visual prompt tuning. In: Computer Vision – ECCV 2022. pp. 709–727. Springer, Springer Nature Switzerland, Cham (2022)
21. Khosla, A., Jayadevaprakash, N., Yao, B., Fei-Fei, L.: Novel dataset for fine-grained image categorization. In: First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition. Colorado Springs, CO (June 2011)
22. Kim, D., Wang, K., Sclaroff, S., Saenko, K.: A broad study of pre-training for domain generalization and adaptation. In: Computer Vision – ECCV 2022. pp. 621–638. Springer, Springer Nature Switzerland, Cham (2022)
23. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops (June 2013)
24. Li, X.L., Liang, P.: Prefix-tuning: Optimizing continuous prompts for generation. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 4582–4597. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.acl-long.353, https://aclanthology.org/2021.acl-long.353/
25. Li, Y., Mao, H., Girshick, R., He, K.: Exploring plain vision transformer backbones for object detection. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. pp. 280–296. Springer Nature Switzerland, Cham (2022)
26. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11976–11986 (June 2022)

27. Luo, W., Li, Y., Urtasun, R., Zemel, R.: Understanding the effective receptive field in deep convolutional neural networks. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 29. Curran Associates, Inc. (2016), https://proceedings.neurips.cc/paper_files/paper/2016/file/c8067ad1937f728f51288b3eb986afaa-Paper.pdf
28. Meng, F., Wang, Z., Zhang, M.: Pissa: Principal singular values and singular vectors adaptation of large language models. In: Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., Zhang, C. (eds.) Advances in Neural Information Processing Systems. vol. 37, pp. 121038–121072. Curran Associates, Inc. (2024). https://doi.org/10.52202/079017-3846, https://proceedings.neurips.cc/paper_files/paper/2024/file/db36f4d603cc9e3a2a5e10b93e6428f2-Paper-Conference.pdf
29. Neuhold, G., Ollmann, T., Rota Bulo, S., Kontschieder, P.: The mapillary vistas dataset for semantic understanding of street scenes. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) (Oct 2017)
30. Prottasha, N.J., Chowdhury, U.R., Mohanto, S., Nuzhat, T., Sami, A.A., Ali, M.S., Sobuj, M.S.I., Raman, H., Kowsher, M., Garibay, O.O.: Peft a2z: parameter-efficient fine-tuning survey for large language and vision models. arXiv preprint arXiv:2504.14117 (2025)
31. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (18–24 Jul 2021), https://proceedings.mlr.press/v139/radford21a.html
32. Ran, W., Zhang, W., Pang, S., Zhu, Q., Liu, J., Liu, J., Cao, X., Li, Q., Yan, Y., Ma, C.: Correlated low-rank adaptation for convnets. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2026), https://openreview.net/forum?id=3pF7rt9fQM
33. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10684–10695 (June 2022)
34. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 22500–22510 (June 2023)
35. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., Massa, F., Haziza, D., Wehrstedt, L., Wang, J., Darcet, T., Moutakanni, T., Sentana, L., Roberts, C., Vedaldi, A., Tolan, J., Brandt, J., Couprie, C., Mairal, J., Jégou, H., Labatut, P., Bojanowski, P.: DINOv3 (2025), https://arxiv.org/abs/2508.10104
36. Van Horn, G., Branson, S., Farrell, R., Haber, S., Barry, J., Ipeirotis, P., Perona, P., Belongie, S.: Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 595–604 (2015)
37. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset. Tech. rep., California Institute of Technology (2011)
38. Woo, S., Debnath, S., Hu, R., Chen, X., Liu, Z., Kweon, I.S., Xie, S.: Convnext v2: Co-designing and scaling convnets with masked autoencoders. In: Proceedings of

the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16133–16142 (June 2023)
39. Xu, L., Xie, H., Qin, S.J., Tao, X., Wang, F.L.: Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–20 (2026). https://doi.org/10.1109/TPAMI.2026.3657354
40. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K. (eds.) Advances in Neural Information Processing Systems. vol. 27. Curran Associates, Inc. (2014), https://proceedings.neurips.cc/paper_files/paper/2014/file/532a2f85b6977104bc93f8580abbb330-Paper.pdf
41. Yu, F., Chen, H., Wang, X., Xian, W., Chen, Y., Liu, F., Madhavan, V., Darrell, T.: Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
42. Yun, S., Chae, S., Lee, D., Ro, Y.: Soma: Singular value decomposed minor components adaptation for domain generalizable representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 25602–25612 (June 2025)
43. Yunis, D., Patel, K.K., Wheeler, S., Savarese, P.H.P., Vardi, G., Frankle, J., Livescu, K., Maire, M., Walter, M.: Rank minimization, alignment and weight decay in neural networks. In: High-dimensional Learning Dynamics 2024: The Emergence of Structure and Reasoning (2024), https://openreview.net/forum?id=u3sssLLu4y
44. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 818–833. Springer International Publishing, Cham (2014)
45. Zhang, Q., Chen, M., Bukharin, A., He, P., Cheng, Y., Chen, W., Zhao, T.: Adaptive budget allocation for parameter-efficient fine-tuning. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=lq62uWRJjiY
46. Zhong, Z., Tang, Z., He, T., Fang, H., Yuan, C.: Convolution meets loRA: Parameter efficient finetuning for segment anything model. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=ezscMer8L0