This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Parameter Efficient Mamba Tuning via Projector-targeted Diagonal-centric Linear Transformation

Seokil Ham Hee-Seon Kim Sangmin Woo Changick Kim
Korea Advanced Institute of Science and Technology (KAIST)
Daejeon, South Korea
{gkatjrdlf, hskim98, smwoo95, changick}@kaist.ac.kr

Abstract

*Despite the growing interest in Mamba architecture as a potential replacement for Transformer architecture, parameter-efficient fine-tuning (PEFT) approaches for Mamba remain largely unexplored. In our study, we introduce two key insights-driven strategies for PEFT in Mamba architecture: (1) While state-space models (SSMs) have been regarded as the cornerstone of Mamba architecture, then expected to play a primary role in transfer learning, our findings reveal that Projectors—not SSMs—are the predominant contributors to transfer learning. (2) Based on our observation, we propose a novel PEFT method specialized to Mamba architecture: **Pro**jector-targeted **Dia**gonal-centric **L**inear Transformation (**ProDiaL**). ProDiaL focuses on optimizing only the pretrained Projectors for new tasks through diagonal-centric linear transformation matrices, without directly fine-tuning the Projector weights. This targeted approach allows efficient task adaptation, utilizing less than 1% of the total parameters, and exhibits strong performance across both vision and language Mamba models, highlighting its versatility and effectiveness.*

1. Introduction

Recently, Mamba architecture [5, 13, 33, 51] has been garnered attention as a promising alternative to the Transformer architecture [44]. While Transformers have become the foundational architecture for deep learning across various fields, they suffer from a significant limitation: their inference time increases quadratically with the length of input tokens. To overcome this limitation, Mamba architecture introduces a selective mechanism within state-space models (SSMs) [10, 14] and hardware-aware operations, allowing dynamic and linear computation with respect to input size.

As an alternative to Transformers, Mamba architecture has become widely used in large-scale models [5, 9, 12, 13, 27, 39, 42] due to its computational efficiency. However, despite this efficiency, large-scale models based on Mamba still face significant computational and memory costs during full fine-tuning. While existing parameter-efficient fine-tuning (PEFT) methods [3, 20, 22, 30, 46, 48] are designed primarily for Transformer-based architectures, leaving PEFT methods tailored to Mamba largely unexplored.

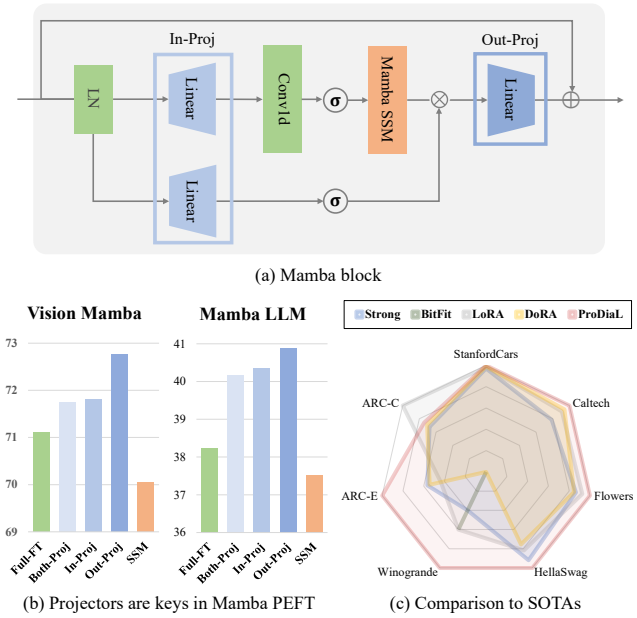


Figure 1. **Overview of Mamba Architecture and Performance Comparison.** (a) The Mamba block structure, illustrating key components including the Input-Projector (In-Proj), Output-Projector (Out-Proj), and State-Space Model (SSM). (b) Performance analysis in Fine-Tuning for Vision Mamba and Mamba LLM, showing that projectors are essential for effective downstream task performance. (c) The radar chart illustrates the relative performance of our proposed method (ProDiaL) compared to other leading PEFT methods (Strong [17], BitFit [46], LoRA [20], DoRA [32]) across multiple benchmarks, demonstrating powerful performance in both vision and language tasks.

In this work, we first investigate the Mamba architecture to determine the components best suited for PEFT. Among the key operations of the Mamba architecture——SSM, 1D-Convolution, Embedding, and Linear Projectors——previous study [17] has focused on fine-tuning the SSM, assuming it to be critical for downstream task adaptation. However, our findings reveal that the Projectors, rather than the SSM, play a crucial role in adapting knowledge for downstream tasks. To the best of our knowledge, this is the first study to conduct an in-depth exploration of the Projectors in the Mamba architecture from a transfer learning perspective.

To efficiently and effectively fine-tune the Projectors for downstream tasks, we propose a novel PEFT method targeting the Projectors in the Mamba architecture, called **Pro**jector-targeted **Dia**gonal-centric **L**inear Transformation (**ProDiaL**). Our ProDiaL builds upon our two key observations: (1) the pretrained Projector weight W can be indirectly fine-tuned by training the linear transformation matrix T, where WT approximates a fine-tuned Projector weight W', and (2) T is a nearly an identity matrix with strong diagonal values and minimal off-diagonal components. Inspired by these insights, ProDiaL adapts pretrained Projectors to downstream tasks by decomposing the transformation matrix T into diagonal and off-diagonal matrices: $W' = WD + \epsilon$, where D is a diagonal matrix, and ϵ represents off-diagonal matrix. Furthermore, to enable more expressive transformations, we use a block-diagonal structure for D, and adopt LoRA [20] for ϵ, which matches W in dimension. As a result, our ProDiaL trains only a block-diagonal matrix for D and two small matrices for ϵ, with the number of parameters controlled by low-rank factors. This design achieves parameter efficiency while preserving flexibility.

Our proposed ProDiaL is specifically designed based on the analysis of Mamba Projectors, making it compatible with all models built upon Mamba architecture such as Mamba-based Large Language Models (LLMs) [13] and Vision models [51]. Our experiment results demonstrate that fine-tuning the Projectors is crucial for adapting the pretrained model to downstream tasks, and our proposed ProDiaL yields superior performance in downstream tasks while training less than 1% of the total model parameters, highlighting its efficiency and effectiveness on both Mamba-based vision and language models.

Our Contributions

- By investigating the impact of each component in Mamba for transfer learning, we discover that Mamba's PEFT primarily benefits from fine-tuning the Projectors.
- With our analysis that fine-tuned Projectors in Mamba can be approximated as a near-diagonal linear transformation of pretrained Projectors, we propose ProDiaL.
- Experiments on both Vision Mamba and Mamba LLM demonstrate that applying not only our proposed ProDiaL method but also existing PEFT methods to the Projectors significantly outperforms targeting other components.

2. Related work

2.1. State Space Models

State Space Models (SSMs) [23] are dynamic systems that represent the relationships between inputs, hidden states, and outputs over time. A notable example, **Structured State Space Model (S4)** [14–16, 21, 34, 41] is designed to handle long-range sequences with Linear Time Invariance (LTI) system, which ensures consistent outputs for identical inputs regardless of their temporal positions in a sequence. A key advantage of S4 is that its computational cost scales linearly with sequence length. However, the fixed internal state transition matrix over time restricts the model's flexibility in adjusting to changing content, limiting its effectiveness for tasks that require context-based reasoning. As a result, despite of the quadratic computational cost with respect to the sequence length, Transformers are preferred for processing sequential data [2, 38, 43, 44].

To address these limitations of SSMs, **Mamba** [5, 13] introduces a selective mechanism and hardware-aware operation to overcome the quadratic computational costs of Transformers. This enables Mamba to support context-aware reasoning with linear computational cost, extending its applications to various sequential data tasks including language and speech tasks [12, 13, 39]. Similar to Transformers, Mamba is also being applied in the vision domain. **Vision Mamba** models [9, 29, 33, 36, 42, 51] adopt bidirectional scanning methods to effectively represent 2-D spatial information of images as 1-D sequences. This feature extracting strategy has proven effective, being adopted not only in simple vision tasks such as classification [29, 33, 36, 51] but also in more complex vision tasks like image generation tasks [9, 42].

Despite the success of these Mamba models and the growing scale of Mamba models, PEFT of Mamba architectures for downstream tasks remains largely unexplored. Therefore, in this work, we uncover the core component in Mamba architecture relevant to PEFT and propose a novel PEFT method tailored to Mamba, called ProDiaL, based on new insights from our analysis.

2.2. Parameter Efficient Fine-Tuning

Fine-tuning large models often demands significant computational and memory resources, especially when working with limited data, which might increase the risk of overfitting. To address these challenges, *parameter-efficient fine-tuning* (PEFT) methods have emerged, particularly targeting Transformer-based architectures. Key PEFT techniques

include: **(1) Adapters** integrate learnable modules within pre-trained models, allowing them to train separately from the model's primary frozen weights. Adapter variations include serial configurations (*e.g.*, Serial Adapter [19]) and parallel structures (*e.g.*, AdaptFormer [3], ControlNet [49]). **(2) Prompt Tuning** uses trainable embeddings that guide downstream learning in visual tasks, optimizing model adaptation for vision applications (*e.g.*, Visual Prompt Tuning [22], LLaMA-Adapter [50], DMP [18]). **(3) Subset Fine-tuning** modifies specific model parameters, such as bias terms (*e.g.*, BitFit [46], DiffFit [45]) or selected key-value weights (*e.g.*, Custom Diffusion [26]) within Transformer attention layers, thereby reducing training overhead. **(4) Low-Rank Adaptation (LoRA)** [20] incorporates low-rank matrices for fine-tuning, preserving the original model weights while facilitating effective downstream learning. This method is particularly useful for conserving model integrity within Transformer architectures (*e.g.*, VeRA [24], DoRA [32], QLoRA [7], MTLoRA [1]).

Despite the success of these PEFT methods with Transformer models [44], their application to Mamba architectures [5, 13, 33, 51] remains largely unexplored. Recently, Halloran *et al*. [17] examined the use of PEFT techniques, specifically LoRA, within Mamba SSMs. Unlike previous works, we reveal that PEFT in Mamba architectures is more effective when applied to Projectors rather than SSMs. Building on this insight, we introduce ProDiaL, a novel PEFT method focused on efficient adaptation of Projectors, specialized to Mamba architecture.

3. Preliminary

To address the limitations inherent in transformer architectures—specifically, the quadratic increase in computational cost with respect to the number of input tokens—the Mamba architecture is introduced as a novel solution centered around the SSM.

Initially, the SSM is a LTI system that maps an input sequence $x(t)$ to a hidden state $h(t)$ and predicts the output $y(t)$ by leveraging both $x(t)$ and $h(t)$. The process is defined by the following system of Ordinary Differential Equations (ODEs):

$$\begin{aligned} h'(t) &= Ah(t) + Bx(t), \\ y(t) &= Ch(t) + Dx(t), \end{aligned} \tag{1}$$

where $x(t) \in \mathbb{R}$ is the continuous input sequence, $h(t) \in \mathbb{R}^N$ is the hidden state, and $y(t) \in \mathbb{R}$ is the output sequence. While this equation addresses continuous signals, discretization is required for processing discrete signals. In Mamba, the Zero-Order Hold technique is used to discretize parameters A and B, utilizing a step size parameter Δ to derive the discrete equivalents:

$$\begin{aligned} \bar{A} &= \exp(\Delta A), \\ \bar{B} &= (\exp(\Delta A) - I)(\Delta A)^{-1} B, \\ \bar{C} &= C. \end{aligned} \tag{2}$$

With these discretized parameters $\bar{A}$, $\bar{B}$, and $\bar{C}$, the SSM is reformulated to address discrete signals as follows:

$$\begin{aligned} h_n &= \bar{A} h_{n-1} + \bar{B} x_n, \\ y_n &= \bar{C} h_n, \end{aligned} \tag{3}$$

where n indexes the input sequence for the discrete signal. In contrast to the continuous formulation in Eq. (1), this discretized ODE explicitly separates current and previous states, with the hidden state h_n computed from the current input x_n and the previous state h_{n-1}.

The architecture of a Mamba block is a combination of the SSM-based model [10] and a gated MLP [31]. Therefore, in addition to the SSM, Mamba blocks also include two Projectors and a 1D Convolution layer, as illustrated in Fig. 1, representing the original Mamba block.

4. Methodology

Current PEFT methods are primarily designed for Transformer-based architectures. However, despite Mamba architecture is widely used in large-scale models such as LLMs or Diffusion models, it remains underexplored in terms of PEFT techniques. In this section, we analyze the impact of each component in Mamba blocks on downstream task performance by fine-tuning various combinations of these components and propose a novel PEFT method specialized to Mamba architecture.

4.1. Targeting Projectors for PEFT

In Transformer-based models, PEFT methods typically target the attention module as the primary component for adapting to downstream tasks [1, 7, 20, 24, 32]. Following this prior knowledge, a natural approach for PEFT in Mamba might be to fine-tune the SSM, which plays a core role similar to the attention mechanism in Transformers. However, the effectiveness of fine-tuning the SSM in the Mamba architecture has not actually been explored yet.

To address this, we first identify key components for downstream task learning in the Mamba architecture. Based on [17], which fine-tuned parameters including W_x (responsible for determining B, C, and Δ in the SSM), two Projectors, and Embeddings, we selectively fine-tune or exclude these components to assess their individual contributions to downstream performance. The results, as shown in Tabs. 1 and 2, highlight the following points:

(1) **Projector Dominance**: Projectors play a more significant role in learning downstream task knowledge than

Target Components	# of parameters	Accuracy
Full-FT	6.975M	71.11
W_x, Both-Proj [17]	6.196M	71.39
Both-Proj	5.328M	71.74
W_x, In-Proj	4.427M	70.89
In-Proj	3.558M	71.81
W_x, Out-Proj	2.657M	73.00
Out-Proj	1.789M	72.77
SSM	1.441M	70.04

Table 1. **Comparison of downstream task performance (%) by fine-tuning various components in Vision Mamba model.** Notably, fine-tuning only projectors yield a high accuracy compared to targeting other components.

Target Components	# of parameters	Accuracy
Full-FT	130M	38.23
W_x, Both-Proj, Embed [17]	126M	38.68
W_x, Both-Proj	87.9M	39.53
Both-Proj, Embed	124M	39.06
Both-Proj	84.9M	40.18
In-Proj	56.6M	40.36
Out-Proj	28.3M	40.89
W_x, In-Proj	59.6M	39.76
W_x, Out-Proj	31.3M	40.79
SSM	5.38M	37.52
Embed	38.6M	34.39

Table 2. **Downstream task performance(%) results for Mamba LLM**, showing similar trends to the Vision Mamba model.

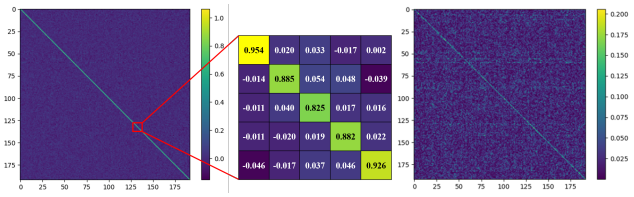
W_x, even though W_x directly affects the most of elements (B, C, and Δ) in the SSM module.

(2) **Input vs. Output Projectors**: Fine-tuning either the Input Projector (a Projector before SSM) or the Output Projector (a Projector after SSM) yields performance comparable to fine-tuning both Projectors together.

(3) **Embedding Limitations**: Embedding parameters, as used in [17], hinder the learning of downstream task knowledge, ultimately limiting performance.

These findings are consistent across both the Mamba LLMs [13] and Vision Mamba models [51]. Given these observations, one might question whether the higher downstream task performance of Projectors than other components is merely due to their larger number of parameters. However, as demonstrated in Tab. 3, high downstream task performance preserves even with a reduced number of learnable parameters, suggesting that the effectiveness of Projectors is not solely attributable to parameter size. However, comparing the performance of SSM with 5.38M parameters (37.52%) to that of Both-Projectors using LoRA with 2.36M parameters (38.33%) in Tab. 3, high downstream task performance preserves even with a reduced number of learnable parameters. This suggests that the effectiveness of Projectors is not solely due to parameter size.

In summary, our analysis reveals that Projectors—rather than the SSM—are crucial for learning downstream tasks in Mamba architecture. However, Projectors in Mamba represent a large fraction of total parameters, accounting for approximately 65.33% of the model's parameters. Although direct fine-tuning of Projectors is effective, training such a large number of parameters is challenging. This observation motivates the need for PEFT methods that indirectly optimizes pretrained Projectors, preserving their capabilities while requiring fewer learnable parameters.



(a) Visualization of the matrix T_{det}. (b) Changes after training.

Figure 2. **Analysis of a linear transformation matrix T.** (a) The matrix closely resembles an identity matrix, with strong diagonal values and minimal off-diagonal values. (b) The accumulated gradient is concentrated along the diagonal, emphasizing the importance of training these elements for effective adaptation.

4.2. ProDiaL: Projector-targeted Diagonal-centric Linear Transformation

4.2.1. Primarily Training Diagonal Entries in T.

Building on our observation that Projectors are essential for transfer learning in Mamba architecture, we analyze the relationship between the pretrained Projector weight $W \in \mathbb{R}^{d_{out} \times d_{in}}$ and fine-tuned Projector weight $W' \in \mathbb{R}^{d_{out} \times d_{in}}$ to gain deeper insights. Firstly, rather than respresenting a relationship between W and W' as $W' = W + \Delta W$ following [20, 24, 32], we approach this from an uncommon perspective by interpreting the relationship as linear and formulate it as follows:

$$W' = WT, \tag{4}$$

where $T \in \mathbb{R}^{d_{in} \times d_{in}}$ represents a linear transformation matrix representing the relationship between the fine-tuned and pretrained Projector weights. Given the fine-tuned and pretrained Projector weights, we can deterministically calculate T_{det} using the pseudo-inverse of W as follows:

$$T_{det} = W^{-1}W', \tag{5}$$

where W^{-1} represents the pseudo-inverse of W.

As shown in Fig. 2a, the matrix T_{det} closely resembles an identity matrix, with high values (close to 1) along the diagonal and near-zero values elsewhere. This pattern suggests that the fine-tuned Projector weight can be approximated as a linear transformation of the pretrained Projector weight, primarily highlighting diagonal entries. This structure is consistently observed across both Projectors, all layers, and both Mamba LLM and Vision models.

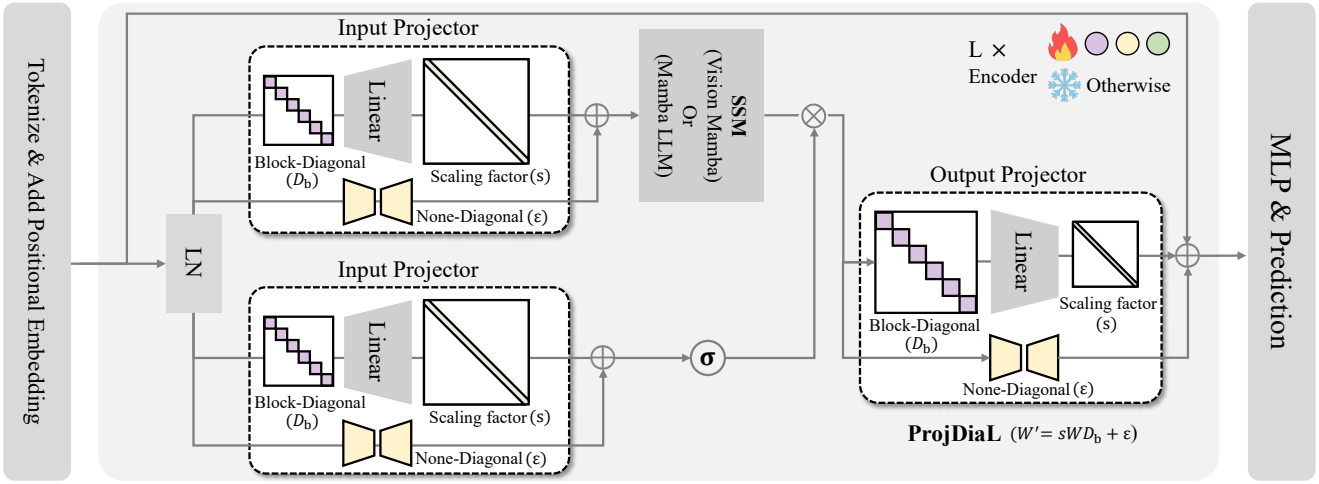


Figure 3. **Overview of ProDiaL Architecture for Efficient Parameter Tuning in Mamba Models**: A detailed structure of ProDiaL's approach to fine-tuning Mamba architecture by focusing on Projector transformations. ProDiaL selectively updates the diagonal(D_b) and non-diagonal(ϵ) matrices in Projectors, enabling efficient learning with minimal parameters.

In addition, we measure the L_1 norm of the difference between the identity matrix $\mathbb{I}$ and fully fine-tuned matrix T (with W frozen) to evaluate the relative importance of diagonal versus off-diagonal entries by the extent of accumulated gradients during training. As visualized in Fig. 2b, the diagonal entries exhibit large L_1 norm differences from $\mathbb{I}$, while the non-diagonal entries show only minor differences. This finding indicates that the gradients of diagonal entries are accumulated during training, and then primarily training the diagonal entries is critical to making T an ideal transformation matrix that approximates W to W'.

These analyses demonstrate that indirectly updating the pretrained Projector weight W can be achieved through linear transformation, with focusing on training the diagonal entries to effectively approximate W to W'. This provides a key insight for our novel PEFT method targeting Projectors in the Mamba architecture. Further analysis of the diagonal entries in T is provided in Sec. 5.4.

However, as indicated by the ablation study results in Tab. 7 and the presence of some large values at the off-diagonal entries shown in Fig. 2b, training off-diagonal entries together with diagonal entries further enhance performance. Consequently, our proposed method incorporates both entries in T but trains them separately, reflecting this observation in our approach.

4.2.2. Decomposing Diagonal and Off-diagonal Entries.

Despite our observations highlight the importance of diagonal entries in linear transformation matrix, applying the existing PEFT methods like LoRA [20] and DoRA [32] to Projectors cannot fully consider these insights. Therefore, we propose a novel Projector-targeted PEFT method, which is specialized for Mamba architecture, called **Pro**jector-targeted **Dia**gonal-centric **L**inear Transformation (**ProDiaL**). The overall framework of ProDiaL is illustrated in Fig. 3.

Considering the different scales of the diagonal and non-diagonal components, as shown in Fig. 2b, we decompose the linear transformation matrix T into a diagonal matrix $D \in \mathbb{R}^{d_{in} \times d_{in}}$ and a off-diagonal matrix $b \in \mathbb{R}^{d_{in} \times d_{in}}$. Given that the values of b are small, we simplify the term Wb as ϵ. This decomposition can be expressed as follows:

$$\begin{aligned} W' = WT &= W(D + b) \\ &= WD + Wb \\ &= WD + \epsilon. \end{aligned} \tag{6}$$

This decomposition forms main strategy of ProDiaL: we freeze the pretrained Projector weight W and fine-tune only the diagonal matrix D and off-diagonal matrix ϵ, indirectly updating W towards the fine-tuned Projector weight W'.

Specifically, to enhance flexibility, we replace the diagonal matrix D with a block-diagonal matrix D_b and introduce a learnable scaling parameter s, as inspired by Fig. 4. This allows subtle transformations, such as minor rotations, beyond simple scaling of Projector weights. For the off-diagonal matrix ϵ, which has same dimensions as W and W' but is relatively less important than the diagonal entries, we apply LoRA with a low-rank structure to achieve parameter efficiency. In Fig. 3, the colored parts indicate the learnable parameters in ProDiaL, where only D_b, ϵ, and s are trained while W remains unchanged.

In addition, inspired by Fig. 2b, we indirectly optimize our block-diagonal matrix D_b by learning the difference between an identity matrix $\mathbb{I}$ and an auxiliary block-diagonal matrix $D_a \in \mathbb{R}^{d_{in} \times d_{in}}$. Consequently, our ProDiaL method is formulated as follows:

$$
\begin{aligned}
W' &= sWD_b + \epsilon, \\
D_b &= [\mathbb{I} - relu(\mathbb{I} * D_a)] + (\mathbf{1} - \mathbb{I}) * D_a, \\
D_a &= diag(x_1, x_2, ..., x_n), \\
\epsilon &= B_\epsilon A_\epsilon,
\end{aligned} \tag{7}
$$

where $s \in \mathbb{R}^{d_{out}}$ is an learnable scaling parameter, $\{x\}_{i=1}^{n} \in \mathbb{R}^{(d_{in}/r_b)\times(d_{in}/r_b)}$ are small matrices forming D_a, and $A_\epsilon \in \mathbb{R}^{r_\epsilon \times d_{in}}$ and $B_\epsilon \in \mathbb{R}^{d_{out}\times r_\epsilon}$ are low-rank matrices employed through LoRA to train ϵ. The detailed algorithm of our ProDiaL including explanation of r_b is provided in Section A of the Supplementary Material.

With these approaches, ProDiaL provides flexibility in controlling the number of learnable parameters by adjusting the block size r_b in D_a and the low rank value r_ϵ of ϵ. The experiment with varying values of r_b and r_ϵ are reported in Section B of the Supplementary Material. Furthermore, as Tab. 3 demonstrated, ProDiaL is effective regardless of whether it targets only the Input Projector or Output Projector, achieving comparable performance to fine-tuning both Projectors. After the training, ProDiaL stores only the updated W, transformed with matrices D_b and ϵ. Consequently, there is no need to store additional parameters used during training, allowing ProDiaL to follow the advantage of LoRA.

5. Experiment

In this section, we evaluate the effectiveness of applying PEFT methods to Projectors within the Mamba architecture using our ProDiaL across various downstream tasks.

5.1. Experiment Setup

To evaluate the effectiveness of fine-tuning projectors using our ProDiaL, we conducted experiments on Mamba-130M [13], which is based on the Mamba 1 architecture for Mamba LLM, and on Vim-tiny [51], which is built upon the Mamba 2 architecture [5] for Vision Mamba. Specifically, we optimize Mamba LLM pre-trained on the PILE dataset [11] to reasoning tasks [4, 40, 47] and Vision Mamba model pre-trained on the ImageNet dataset [6] to other classification tasks [25, 28, 35]. For Mamba LLM, we measure accuracy by sampling N checkpoints of fine-tuned model weights every M iterations, selecting the checkpoint with the highest test accuracy among the samples. The values of N and M vary across datasets. For Vision Mamba, we measure accuracy of the last checkpoint on the test set. For comparison, we established baselines using Full Fine-Tuning (Full-FT) and existing PEFT methods targeting different parameters, such as BitFit [46] and Strong [17]. All experiments details, including the details for analysis experiments, are provided in Section D of the Supplementary Material.

5.2. Experiment Results

As the first study to report downstream task performance of Mamba LLM and Vision Mamba within Mamba architecture research, we provide extensive experimental results across various datasets, as shown in Tab. 3. Given our experiments yield consistent results across both Mamba LLM and Vision Mamba, we explain the findings comprehensively.

Our results reveal that fine-tuning only the Projectors—or even a single Projector (either input or output)—achieves competitive performance to full fine-tuning, but with a significantly fewer learnable parameters. Even, fully fine-tuning the entire Mamba model or Projectors often leads to overfitting due to the large number of parameters relative to the small size of downstream task data, highlighting the necessity of PEFT methods.

Among existing PEFT methods, LoRA [20] and DoRA [32] applied to Mamba's Projectors shows impressive downstream performance, demonstrating that high accuracy can be achieved with fewer parameters by targeting Projectors in Mamba. For example, compared to Strong [17]—which uses LoRA to train W_x, both Projectors, and Embeddings—applying LoRA solely to Projectors achieves significant downstream task performance while utilizing only 63.6% of the parameter counts used by Strong for Mamba LLM and 69.8% for Vision Mamba.

Beyond the existing PEFT methods, our ProDiaL, specifically designed based on the analysis of Projectors in Mamba architecture, consistently achieves superior downstream performance with few learnable parameters. This results stem not only from targeting Projectors in Mamba architecture but also from highlighting the importance of diagonal entries observed in the linear transformation relationship between pretrained and fine-tuned Projectors. In Tab. 3, we align the number of parameters of ProDiaL with those of LoRA and DoRA for comparison. More detailed settings, including hyperparameters for r_b and r_ϵ, are provided in Section D of the Supplementary Material. Furthermore, additional experiments on the Mamba LLM based on the Mamba 2 architecture, along with comprehensive experiment details, are also included in Section C of the Supplementary Material.

5.3. Results Across Various Model Sizes

Table 4 highlights performance across different model sizes, including Mamba-370M and Mamba-1.4B for Mamba LLM and Vim-small for Vision Mamba, evaluated on the Winogrande [40] and Caltech [28] datasets, respectively. Consistent with findings from smaller models, fine-tuning only the Projectors——either individually or jointly——demonstrates notable effectiveness in adapting to downstream tasks. Our ProDiaL, tailored to the Mamba architecture, consistently outperforms other PEFT methods, achieving high accuracy with minimal parameters. These

	Method	Mamba LLM					Vision Mamba			
		HellaSwag	Winogrande	ARC-E	ARC-C	Avg	StanfordCars	Caltech	Flowers	Avg.
Baselines	Full-FT	38.23 (130.00M)	53.12 (130.00M)	53.54 (130.00M)	28.84 (130.00M)	43.43	90.06 (7.00M)	92.86 (7.00M)	92.05 (7.00M)	91.66
	Linear Probing	-	-	-	-	-	57.46 (0.04M)	91.10 (0.02M)	59.90 (0.02M)	69.49
	BitFit [46]	35.69 (0.07M)	53.12 (0.07M)	52.86 (0.07M)	26.88 (0.07M)	42.14	65.51 (0.08M)	93.71 (0.06M)	78.84 (0.06M)	79.35
	Strong [17]	38.66 (3.80M)	53.04 (3.80M)	54.17 (3.80M)	28.67 (3.80M)	43.64	84.78 (0.96M)	95.70 (0.94M)	86.76 (0.94M)	89.08
Both-Proj	FT	40.18 (84.94M)	52.57 (84.94M)	54.38 (84.94M)	29.52 (84.94M)	44.16	89.67 (5.35M)	95.01 (5.33M)	92.00 (5.33M)	92.22
	LoRA	38.33 (2.36M)	53.12 (2.36M)	53.87 (2.36M)	**29.52** (2.36M)	43.71	85.06 (0.63M)	96.01 (0.61M)	87.32 (0.61M)	89.46
	DoRA	38.13 (2.45M)	52.88 (2.45M)	54.12 (2.45M)	28.75 (2.45M)	43.47	85.18 (0.69M)	96.09 (0.65M)	86.60 (0.65M)	89.29
	ProDiaL	**38.92** (2.42M)	**53.28** (2.42M)	**55.18** (2.38M)	28.84 (2.38M)	**44.06**	**85.38** (0.67M)	**96.24** (0.65M)	**88.00** (0.65M)	**89.87**
In-Proj	FT	40.36 (56.62M)	53.20 (56.62M)	54.59 (56.62M)	29.61 (56.62M)	44.44	89.62 (3.58M)	95.24 (3.56M)	91.02 (3.56M)	91.96
	LoRA	**38.46** (1.48M)	52.80 (1.48M)	53.87 (1.48M)	28.41 (1.48M)	43.39	82.12 (0.41M)	95.78 (0.39M)	85.71 (0.39M)	87.87
	DoRA	38.08 (1.55M)	52.64 (1.55M)	54.04 (1.55M)	28.50 (1.55M)	43.32	82.17 (0.43M)	95.55 (0.41M)	85.95 (0.41M)	87.89
	ProDiaL	38.41 (1.49M)	**52.96** (1.49M)	**54.50** (1.25M)	**29.61** (1.25M)	**43.87**	**82.45** (0.42M)	**95.93** (0.41M)	**85.97** (0.33M)	**88.12**
Out-Proj	FT	40.89 (28.31M)	52.80 (28.31M)	55.09 (28.31M)	29.27 (28.31M)	44.51	88.86 (1.81M)	95.63 (1.77M)	91.45 (1.77M)	91.98
	LoRA	37.30 (0.89M)	53.12 (0.89M)	53.66 (0.89M)	28.54 (0.89M)	43.08	77.81 (0.26M)	95.40 (0.24M)	80.60 (0.24M)	84.60
	DoRA	37.19 (0.90M)	52.88 (0.90M)	53.66 (0.90M)	28.67 (0.90M)	43.10	77.70 (0.28M)	95.47 (0.26M)	80.97 (0.26M)	84.71
	ProDiaL	**38.19** (0.92M)	**53.75** (0.92M)	**54.84** (0.90M)	**30.80** (0.90M)	**44.40**	**78.00** (0.27M)	**95.55** (0.25M)	**81.90** (0.25M)	**85.15**

Table 3. **Comparison of accuracy(%) as a downstream task performance in Mamba LLM and Vision Mamba across various datasets.** ProDiaL methods (in Both-Proj, In-Proj, and Out-Proj) consistently demonstrate superior accuracy across both language and vision tasks, achieving similar parameter efficiency compared to other methods.

	Method	Mamba-370M	Mamba-1.4B	Vim-small
Base	Full-FT	56.99 (370M)	61.17 (1.40B)	94.09 (25.45M)
	BitFit [46]	56.99 (0.20M)	61.25 (0.39M)	96.62 (0.11M)
	Strong [17]	57.14 (8.19M)	61.80 (0.06B)	96.47 (1.85M)
Both-Proj	FT	57.22 (302M)	61.72 (1.21B)	94.94 (21.27M)
	LoRA	56.75 (6.29M)	61.72 (0.05B)	96.85 (1.22M)
	DoRA	56.99 (6.54M)	61.72 (0.05B)	96.85 (1.27M)
	ProDiaL	**57.06** (5.75M)	**61.96** (0.05B)	**97.16** (1.32M)
In-Proj	FT	57.06 (201M)	61.56 (0.81B)	95.17 (14.20M)
	LoRA	56.75 (3.93M)	**61.56** (0.03B)	**97.16** (0.88M)
	DoRA	**57.22** (4.13M)	61.48 (0.03B)	**97.16** (0.91M)
	ProDiaL	**57.22** (3.74M)	**61.56** (0.03B)	97.09 (0.82M)
Out-Proj	FT	57.22 (101M)	61.72 (0.40B)	95.86 (7.12M)
	LoRA	56.91 (2.36M)	61.56 (0.02B)	96.70 (0.48M)
	DoRA	57.22 (2.41M)	61.48 (0.02B)	96.78 (0.59M)
	ProDiaL	**57.30** (2.02M)	**61.80** (0.02B)	**96.85** (0.51M)

Table 4. **Comparison of downstream task performance (%) across various model sizes with different fine-tuning methods.**

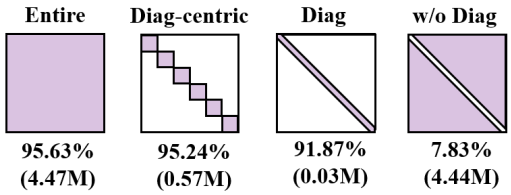


Figure 4. **Performance comparison across four T configurations in Mamba model.** The diagonal-centric approach achieves near-optimal performance with a significantly reduced parameter number ($0.57M$), supporting the validity of our ProDiaL.

results emphasize the adaptability and efficiency of our ProDiaL method across model scales, suggesting its potential applicability to other Mamba-based architectures.

5.4. Analysis

Q: Training Only Diagonal Components is Sufficient?

To empirically verify that training primarily the diagonal elements of a linear transformation matrix T is sufficient, we conduct a experiment, comparing four approaches: (1) directly fine-tuning T, (2) fine-tuning only the diagonal-centric matrix, (3) fine-tuning only the diagonal matrix, (4) fine-tuning only the off-diagonal matrix. The four approaches and their performances are illustrated in Fig. 4. The results show that fine-tuning only the diagonal matrix achieves significant performance despite of its small number of parameters. Expanding the diagonal matrix to a diagonal-centric matrix further enhances performance, achieving results comparable to directly fine-tuning T. In contrast, fine-tuning only the off-diagonal matrix significantly degrades performance. Consequently, this analysis emphasizes the importance of primarily training the diagonal elements in the linear transformation matrix T, supporting our strategy in ProDiaL.

Q: Diagonal is Important Only in Projectors?

Our observations highlight that fine-tuning only the diagonal entries in the transformation matrix T effectively approximates pretrained projector weights W to fine-tuned weights W'. However, it is unclear whether the importance of the diagonal entries is unique to Projectors or extends to all linear layers in the Mamba architecture.

To investigate this, we insert learnable diagonal matri-

Target	Accuracy (%)
SSM (FT)	94.01
SSM (Diag)	93.55
Proj (FT)	95.01
Proj (Diag)	95.24
SSM+Proj (FT)	94.55
SSM+Proj (Diag)	95.55

Table 5. **Analysis of Diagonal-only Fine-tuning.** Projectors benefit more than SSM from diagonal-only fine-tuning.

Target Components	# of parameters	Accuracy
Full-FT	85.9M	92.83
Attention	28.4M	93.07
FFN	56.8M	92.89
Attention (LoRA)	0.67M	92.60
FFN (LoRA)	0.81M	92.51
Attention+FFN (LoRA)	1.40M	92.91

Table 6. **Performance(%) of Transformer Components under Fine-Tuning.** Attention module achieves slightly higher performance than FFN, but yields similar performance levels.

Method	Accuracy (%)
Proj-D_b	95.24 (0.57M)
Proj-D_b+ϵ	96.16 (0.62M)
Proj-D_b+ϵ+s (ProDiaL)	96.24 (0.65M)

Table 7. **Ablation Study on ProDiaL Components.**

ces to the SSM's linear layers (W_x and W_Δ) and evaluate their impact. Table 5 shows that in the SSM, fine-tuning only the diagonal matrix degrades performance due to the SSM's already small parameter count, limiting effective learning. In contrast, for Projectors, diagonal-only fine-tuning improves performance by avoiding overfitting, a common issue with large parameter numbers and limited data. Furthermore, applying diagonal matrices to both SSM and Projectors confirms that indirectly fine-tuning with a diagonal matrix works best for components with a large number of parameters, as it reduces overfitting.

Consequently, diagonal fine-tuning benefits any linear layer with large number of parameters, but in Mamba architecture, Projectors are the solely component with large number of parameters, making them the ideal target for this parameter-efficient approach.

Q: How Does Mamba PEFT Differ from Transformers? To further investigate whether non-attention modules in Transformers can effectively capture downstream task knowledge as Projectors do in Mamba, we conduct an experiment by applying LoRA [20] to each component in the ViT model [8]. As shown in Tab. 6, both attention modules and the Feed Forward Network (FFN)—a non-attention module—achieve comparable performance, with a slight advantage for the attention modules. This result aligns with Mamba, where fine-tuning non-SSM components (Projectors) is effective for downstream tasks.

However, in Transformers, jointly fine-tuning both attention and FFN together synergistically enhance the downstream task performance. In contrast, in Mamba architecture, fine-tuning both SSM and Projectors together degrades performance, as shown in Tabs. 1 and 2. In other words, Mamba benefits from fine-tuning only Projectors without the SSM, whereas Transformers achieve optimal performance by fine-tuning both Attention and FFN. This suggests that fine-tuning only Projectors for downstream tasks is uniquely beneficial to Mamba architecture, distinguishing the characteristics of Mamba from Transformers.

5.5. Ablation Study

Our ProDiaL consists of a learnable block-diagonal matrix D_b, a non-diagonal matrix ϵ, and a scaling factor s, which together enable effective learning of downstream tasks in Mamba architecture. To analyze the contribution of each component, we perform an ablation study by adding components incrementally. As shown in Tab. 7, fine-tuning only the diagonal-centric matrices achieves significant downstream task performance. While fine-tuning only diagonal-centric matrices proves effective, fine-tuning both the non-diagonal matrices and the diagonal matrices yields even higher downstream task performance. In addition, inspired by [37], incorporating scaling factors at the output stage further enhances downstream task performance. These results demonstrate that each component of ProDiaL contributes meaningfully to maximizing downstream task performance within the Mamba architecture.

6. Limitations

Our ProDiaL primiarily relies on LoRA [20] to efficiently train off-diagonal matrix ϵ. Developing a method specifically for optimizing ϵ could further enhance performance, and is an interesting direction for future work.

7. Conclusion

In this work, we reveal that Projectors play a critical role in transfer learning within Mamba architecture. Based on our observation that fine-tuned Projectors can be approximated through a near-diagonal linear transformation, we propose ProDiaL, a novel PEFT method specifically designed for Mamba architecture. Rather than fine-tuning SSM, ProDiaL targets Projectors, optimizing them indirectly through a diagonal-centric linear transformation. This enables ProDiaL to achieve superior performance while fine-tuning less than 1% of the model parameters. Our experiments across both vision and language Mamba models demonstrate ProDiaL's effectiveness, establishing a new benchmark for PEFT in Mamba-based architecture.

References

[1] Ahmed Agiza, Marina Neseem, and Sherief Reda. Mtlora: Low-rank adaptation approach for efficient multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16196–16205, 2024. 3

[2] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 2

[3] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022. 1, 3

[4] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018. 6

[5] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024. 1, 2, 3, 6

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 6

[7] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[8] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 8

[9] Zhengcong Fei, Mingyuan Fan, Changqian Yu, Debang Li, Youqiang Zhang, and Junshi Huang. Dimba: Transformer-mamba diffusion models. *arXiv preprint arXiv:2406.01159*, 2024. 1, 2

[10] Daniel Y Fu, Tri Dao, Khaled K Saab, Armin W Thomas, Atri Rudra, and Christopher Ré. Hungry hungry hippos: Towards language modeling with state space models. *arXiv preprint arXiv:2212.14052*, 2022. 1, 3

[11] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020. 6

[12] Paolo Glorioso, Quentin Anthony, Yury Tokpanov, James Whittington, Jonathan Pilault, Adam Ibrahim, and Beren Millidge. Zamba: A compact 7b ssm hybrid model. *arXiv preprint arXiv:2405.16712*, 2024. 1, 2

[13] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023. 1, 2, 3, 4, 6

[14] Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2021. 1, 2

[15] Albert Gu, Isys Johnson, Karan Goel, Khaled Saab, Tri Dao, Atri Rudra, and Christopher Ré. Combining recurrent, convolutional, and continuous-time models with linear state space layers. *Advances in neural information processing systems*, 34:572–585, 2021.

[16] Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. *Advances in Neural Information Processing Systems*, 35:22982–22994, 2022. 2

[17] John T Halloran, Manbir Gulati, and Paul F Roysdon. Mamba state-space models can be strong downstream learners. *arXiv preprint arXiv:2406.00209*, 2024. 1, 2, 3, 4, 6, 7

[18] Seokil Ham, Sangmin Woo, Jin-Young Kim, Hyojun Go, Byeongjun Park, and Changick Kim. Diffusion model patching via mixture-of-prompts. *arXiv preprint arXiv:2405.17825*, 2024. 3

[19] 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*, pages 2790–2799. PMLR, 2019. 3

[20] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 1, 2, 3, 4, 5, 6, 8

[21] Md Mohaiminul Islam and Gedas Bertasius. Long movie clip classification with state-space video models. In *European Conference on Computer Vision*, pages 87–104. Springer, 2022. 2

[22] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022. 1, 3

[23] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 1960. 2

[24] Dawid J Kopiczko, Tijmen Blankevoort, and Yuki M Asano. Vera: Vector-based random matrix adaptation. *arXiv preprint arXiv:2310.11454*, 2023. 3, 4

[25] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013. 6

[26] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 3

[27] Byung-Kwan Lee, Chae Won Kim, Beomchan Park, and Yong Man Ro. Meteor: Mamba-based traversal of rationale for large language and vision models. *arXiv preprint arXiv:2405.15574*, 2024. 1

[28] Fei-Fei Li, Marco Andreeto, Marc'Aurelio Ranzato, and Pietro Perona. Caltech 101, 2022. 6

[29] Shufan Li, Harkanwar Singh, and Aditya Grover. Mamba-nd: Selective state space modeling for multi-dimensional data. *arXiv preprint arXiv:2402.05892*, 2024. 2

[30] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021. 1

[31] Hanxiao Liu, Zihang Dai, David So, and Quoc V Le. Pay attention to mlps. *Advances in neural information processing systems*, 34:9204–9215, 2021. 3

[32] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*, 2024. 1, 3, 4, 5, 6

[33] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024. 1, 2, 3

[34] Eric Nguyen, Karan Goel, Albert Gu, Gordon Downs, Preey Shah, Tri Dao, Stephen Baccus, and Christopher Ré. S4nd: Modeling images and videos as multidimensional signals with state spaces. *Advances in neural information processing systems*, 35:2846–2861, 2022. 2

[35] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008. 6

[36] Jinyoung Park, Hee-Seon Kim, Kangwook Ko, Minbeom Kim, and Changick Kim. Videomamba: Spatio-temporal selective state space model. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 2

[37] Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. *Advances in Neural Information Processing Systems*, 36:79320–79362, 2023. 8

[38] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR, 2023. 2

[39] Liliang Ren, Yang Liu, Yadong Lu, Yelong Shen, Chen Liang, and Weizhu Chen. Samba: Simple hybrid state space models for efficient unlimited context language modeling. *arXiv preprint arXiv:2406.07522*, 2024. 1, 2

[40] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64 (9):99–106, 2021. 6

[41] Jimmy TH Smith, Andrew Warrington, and Scott W Linderman. Simplified state space layers for sequence modeling. *arXiv preprint arXiv:2208.04933*, 2022. 2

[42] Yao Teng, Yue Wu, Han Shi, Xuefei Ning, Guohao Dai, Yu Wang, Zhenguo Li, and Xihui Liu. Dim: Diffusion mamba for efficient high-resolution image synthesis. *arXiv preprint arXiv:2405.14224*, 2024. 1, 2

[43] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: open and efficient foundation language models. arxiv. *arXiv preprint arXiv:2302.13971*, 2023. 2

[44] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 1, 2, 3

[45] Enze Xie, Lewei Yao, Han Shi, Zhili Liu, Daquan Zhou, Zhaoqiang Liu, Jiawei Li, and Zhenguo Li. Difffit: Unlocking transferability of large diffusion models via simple parameter-efficient fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4230–4239, 2023. 3

[46] Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021. 1, 3, 6, 7

[47] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019. 6

[48] Jeffrey O Zhang, Alexander Sax, Amir Zamir, Leonidas Guibas, and Jitendra Malik. Side-tuning: a baseline for network adaptation via additive side networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 698–714. Springer, 2020. 1

[49] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 3

[50] Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023. 3

[51] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*, 2024. 1, 2, 3, 4, 6