LoCO: Low-Rank Compositional Rotation Fine-Tuning

An Nguyen[1] , **Jaesik Choi**[2,3] , **Anh Tong**[1]
[1]Korea University,
[2]KAIST,
[3]INEEJI
nguyenan@korea.ac.kr, jaesik.choi@kaist.ac.kr, anhtong12@korea.ac.kr

Abstract

Parameter-efficient fine-tuning (PEFT) has emerged as an critical technique for adapting large-scale foundation models across natural language processing and computer vision. While existing methods such as low-rank adaptations achieve parameter efficiency via low-rank weight updates, they are limited in their ability to preserve the geometric structure of pretrained representations. We introduce Low-rank Compositional Orthogonal fine-tuning (LoCO), a novel PEFT method that constructs orthogonal transformations through low-rank skew-symmetric matrices and compositional rotation chains. We propose an approximation scheme that enables fully parallel computation of compositional rotations, making the approach practical for high-dimensional feature spaces. Our method maintains low computational complexity while maintaining orthogonality with controlled approximation error. We validate LoCO across diverse domains, including diffusion transformer fine-tuning, vision transformer adaptation, and language model adaptation. Our approach demonstrates superior or competitive performance compared to both existing orthogonal and non-orthogonal baselines.

1 Introduction

Large-scale pretrained foundation models [Vaswani *et al.*, 2017] have revolutionized computer vision and natural language processing, achieving impressive performance across diverse downstream tasks. Models such as large language models [Brown *et al.*, 2020], vision transformers [Dosovitskiy, 2020] and diffusion transformers [Rombach *et al.*, 2022; Peebles and Xie, 2023] contain billions of parameters, encoding rich representations learned from massive datasets. Adapting these models to specific applications through full fine-tuning is often impractical: it demands substantial computational resources, risks catastrophic forgetting of pretrained knowledge, and requires storing separate model copies for each task. Parameter-efficient fine-tuning (PEFT) methods [Houlsby *et al.*, 2019; Hu *et al.*, 2022] have emerged as an effective solution for task adaptation while training only a small fraction of parameters.

Orthogonal fine-tuning methods [Qiu *et al.*, 2023; Liu *et al.*, 2024b; Yuan *et al.*, 2024] have attracted attention for their geometric properties. Unlike additive methods such as LoRA [Hu *et al.*, 2022] that injects low-rank adaptation to weight matrices, orthogonal methods apply multiplicative transformations, which are constrained to the special orthogonal group. This constraint offers a theoretical advantage in that orthogonal transformations preserve the pairwise angular relationships among neuron representations and maintain the hyperspherical energy structure of pretrained features [Qiu *et al.*, 2023]. Such properties are particularly valuable when semantic information is encoded in angular relationships rather than magnitudes, as observed in contrastive learning [Chen *et al.*, 2020] and face recognition [Liu *et al.*, 2017].

Despite these properties, existing orthogonal fine-tuning methods face a fundamental computational bottleneck. Constructing and applying a general orthogonal matrix requires cubic time complexity for matrix inversion or matrix exponential computation, which becomes prohibitively expensive for high-dimensional feature spaces typical in modern architectures. Prior work addresses this challenge through structural constraints. For example, OFT [Qiu *et al.*, 2023] uses block-diagonal matrices that restrict rotations to independent subspaces, while BOFT [Liu *et al.*, 2024b] uses butterfly factorizations for hierarchical decomposition. However, these approaches impose specific sparsity patterns that may limit expressivity, and butterfly structures may suffer from non-contiguous memory access patterns that underutilize hardware resources.

In this paper,[1] we propose **Lo**w-rank **C**ompositional **O**rthogonal Fine-tuning (**LoCO**), a novel approach that constructs orthogonal transformations through *low-rank* skew-symmetric matrices. Our key observation is that the tangent space to the special orthogonal group, which is the space of skew-symmetric matrices, admits a low-rank parameterization. By exploiting this structure with the Sherman–Morrison–Woodbury identity [Max, 1950], we reduce the computational complexity of orthogonal fine-tuning, allowing adaptation in high-dimensional spaces. Additionally, we

[1]The extended version, including the full appendix, is available on arXiv: https://arxiv.org/abs/2605.15916

leverage chains of multiple low-rank rotations to enhance the expressiveness of orthogonal transformations. Rather than computing these rotations sequentially, we employ a first-order approximation, enabling parallel computation across all rotation components. Our analysis confirms that this approximation introduces negligible error given the small-magnitude adaptations characteristic of fine-tuning. A practical advantage of LoCO is a test-time temperature scaling mechanism for controllable adaptation strength. By introducing a scalar temperature parameter, we can smoothly interpolate between pretrained and fully-adapted behaviors at inference time.

We test LoCO across three domains: (1) fine-tuning language models, (2) adapting diffusion transformers for conditional generation, and (3) fine-tuning vision transformers. Across all settings, LoCO achieves competitive or superior performance compared to established PEFT methods including LoRA [Hu *et al.*, 2022], OFT [Qiu *et al.*, 2023], BOFT [Liu *et al.*, 2024b], and HRA [Yuan *et al.*, 2024], while maintaining computational efficiency and a low memory footprint.

Contributions. Our main contributions are threefold:

- We propose LoCO, a parameter-efficient fine-tuning method that constructs orthogonal transformations via low-rank skew-symmetric matrices. We further introduce an approximation for compositional rotation chains that enables parallel computation while preserving orthogonality for fine-tuning.
- We demonstrate a test-time temperature scaling mechanism that enables flexible control over adaptation strength without retraining, providing an alternative to costly regularization schemes used in prior orthogonal methods.
- We achieve state-of-the-art or competitive results across diverse benchmarks, including GLUE, mathematical reasoning, and controllable generation.

2 Related Work

Low-rank Adaptation. Building on the low intrinsic rank hypothesis [Aghajanyan *et al.*, 2020], LoRA [Hu *et al.*, 2022] parameterizes weight updates as a product of two low-rank matrices. This avoids the inference overhead of adapters [Houlsby *et al.*, 2019] and the optimization challenges in prompt tuning [Li and Liang, 2021; Lester *et al.*, 2021]. Subsequent works extend this framework through adaptive rank allocation [Zhang *et al.*, 2023; Kalajdzievski, 2023; Jiang *et al.*, 2024], alternative initializations [Meng *et al.*, 2024; Wang *et al.*, 2024], and parameter sharing [Bałazy *et al.*, 2024; Dong *et al.*, 2023]. DoRA [Liu *et al.*, 2024a] further decomposes updates into magnitude and direction components. Unlike this line of research, our approach applies multiplicative transformations, constructing rotation matrices from low-rank skew-symmetric parameterizations.

Orthogonal Fine-Tuning. OFT [Qiu *et al.*, 2023; Qiu *et al.*, 2025] parameterizes orthogonal matrices through block-diagonal, butterfly structures [Liu *et al.*, 2024b], structured matrices [Gorbunov *et al.*, 2024], exploiting sparsity for efficiency. While qGOFT [Ma *et al.*, 2024] decomposes rotations into Givens rotations, its sequential composition limits parallelization. Householder-based methods such as [Dong *et al.*, 2024; Yuan *et al.*, 2024; Arcas *et al.*, 2025] use reflection parameterizations. Recent work [Liao and Monz, 2024] applies 2D rotations directly to representation subspaces. Our method differs by constructing rotations from low-rank skew-symmetric matrices, enabling efficient computation through first-order approximation while operating on higher-dimensional subspaces.

Spectral-based Methods. In another line of research, PiSSA [Meng *et al.*, 2024] and MiLoRA [Wang *et al.*, 2024] decompose weights via singular value decomposition. These approaches adapt residual or minor singular components selectively. Compared to these spectral-based methods, our approach preserves angular structure through orthogonal transformations rather than modifying spectral components.

Geometric Approaches. Several works exploit geometric structures: hyperbolic networks [Ganea *et al.*, 2018; Chen *et al.*, 2022] utilize exponential maps for hierarchical representations, and Lie group methods [Mironenco and Forré, 2024; Finzi *et al.*, 2020] construct equivariant architectures. Skew orthogonal convolutions [Singla and Feizi, 2021] use matrix exponentials for training stability. While building from similar mathematical tools, these works focus on designing new architectures rather than proposing new PEFT methods.

3 Low-rank Compositional Orthogonal Fine-tuning (LoCO)

This section first reviews rotation groups and their connection to skew-symmetric matrices, and then presents the LoCO adaptation method.

3.1 Rotation Groups and Skew-symmetric Matrices

The special orthogonal group. A rotation in $\mathbb{R}^d$ is defined as an element of the special orthogonal group:

$$\mathrm{SO}(d) = \left\{ \mathbf{R} \in \mathbb{R}^{d\times d} : \mathbf{R}^\top \mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = 1 \right\}.$$

Here, the condition $\mathbf{R}^\top\mathbf{R} = \mathbf{I}$ indicates that rotations preserve distance, while $\det(\mathbf{R}) = 1$ distinguishes proper rotations from reflections. [2]

Skew-symmetric matrices as tangent space. The tangent space to $\mathrm{SO}(d)$ at the identity is given by:

$$\mathfrak{so}(d) = \left\{ \mathbf{A} \in \mathbb{R}^{d\times d} : \mathbf{A}^\top = -\mathbf{A} \right\}. \tag{1}$$

These matrices represent infinitesimal rotations or rotation velocities. The dimension of $\mathfrak{so}(d)$ is $\frac{d(d-1)}{2}$, corresponding to the number of independent rotation axes in d dimensions.

Matrix exponential map. The exponential map provides a canonical way to generate rotations from skew-symmetric matrices. Every rotation can be obtained from some skew-symmetric matrix. That is, given $\mathbf{A} \in \mathfrak{so}(d)$, we have

$$\exp(\mathbf{A}) = \sum_{n=0}^{\infty} \frac{\mathbf{A}^n}{n!} \in \mathrm{SO}(d). \tag{2}$$

[2] Reflection transformations have $\det(\mathbf{R}) = -1$.

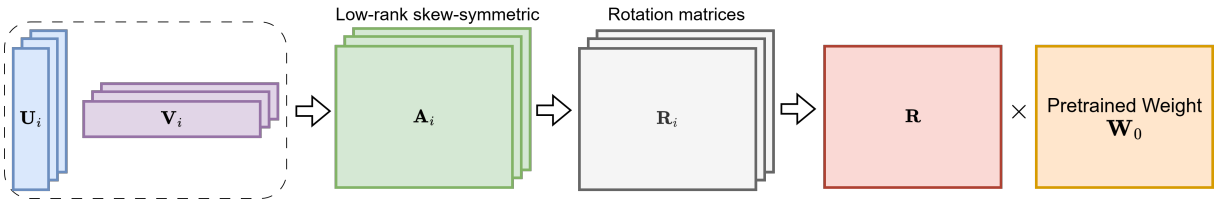

Figure 1: An overview of LoCO. We parameterize each rotation component $\mathbf{R}_i$ using low-rank skew-symmetric matrices constructed from learnable factors $\mathbf{U}_i, \mathbf{V}_i$. We then compose these rotations to form the final orthogonal transformation to adapt pretrained weights.

This provides a *smooth* parameterization of rotation manifold. However, computing the matrix exponential directly is computationally expensive and would be prohibitive for the task of fine-tuning foundation models.

Cayley transform. The Cayley transform offers an alternative approach which is more tractable for generating rotations from skew-symmetric matrices. Given $\mathbf{A} \in \mathfrak{so}(d)$, we have:

$$\mathbf{R} = (\mathbf{I} - \mathbf{A})^{-1}(\mathbf{I} + \mathbf{A}) \in \mathrm{SO}(d). \tag{3}$$

The Cayley transform is particularly well-suited in fine-tuning settings. During fine-tuning, we expect model adaptations to be relatively small, corresponding to rotations with limited angular magnitude. The Cayley transform provides an accurate and smooth parameterization of the rotation manifold near the identity. This approach has been successfully adopted in recent orthogonal fine-tuning methods [Qiu *et al.*, 2023; Liu *et al.*, 2024b].

3.2 LoCO Adaptations

This section presents the main specification of **Lo**w-rank **C**ompositional **O**rthogonal Fine-tuning (LoCO).

Given a pretrained model with weight matrix $\mathbf{W}_0 \in \mathbb{R}^{k\times d}$, the standard linear transformation can be adapted through orthogonal fine-tuning as follows.

$$f_{\text{pretrained}}(\mathbf{x}) = \mathbf{W}_0\mathbf{x} \quad \text{becomes} \quad f_{\text{finetune}}(\mathbf{x}) = \mathbf{W}_0\mathbf{R}\mathbf{x},$$

where $\mathbf{R} \in \mathrm{SO}(d)$ is a special orthogonal matrix that performs rotation transformations on the input features $\mathbf{x} \in \mathbb{R}^d$. To enhance model expressivity, we utilize a compositional representation of $\mathbf{R}$:

$$\mathbf{R} = \mathbf{R}_1\mathbf{R}_2\cdots\mathbf{R}_n,$$

where each component $\mathbf{R}_i \in \mathrm{SO}(d)$ represents an elementary rotation. The composition of orthogonal matrices preserves orthogonality and ensures that $\mathbf{R} \in \mathrm{SO}(d)$. Orthogonal fine-tuning takes inspiration from the idea that angular features capture the semantic gap well [Chen *et al.*, 2020; Liu *et al.*, 2017; Liu *et al.*, 2018; Qiu *et al.*, 2023].

In the following subsections, we detail how to parameterize each component $\mathbf{R}_i$ using low-rank skew-symmetric matrices and describe the approximation scheme for the compositional rotation $\mathbf{R}$. Figure 1 provides an overview of the LoCO framework.

Skew-symmetric construction

We parameterize skew-symmetric matrices through a low-rank outer product form. Given learnable matrices $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{d\times r}$ where $r \ll d$ is the rank dimension, we construct

$$\mathbf{A} = \mathbf{U}\mathbf{V}^\top - \mathbf{V}\mathbf{U}^\top.$$

This formula guarantees skew-symmetry since we can verify that $\mathbf{A}^\top = \left(\mathbf{U}\mathbf{V}^\top\right)^\top - (\mathbf{V}\mathbf{U}^\top)^\top = \mathbf{V}\mathbf{U}^\top - \mathbf{U}\mathbf{V}^\top = -\mathbf{A}$. For notation convenience, we rewrite this construction using auxiliary matrices:

$$\mathbf{A} = \mathbf{X}\mathbf{Y}^\top, \text{where} \quad \mathbf{X} = [\mathbf{U} \mid -\mathbf{V}], \quad \mathbf{Y} = [\mathbf{V} \mid \mathbf{U}],$$

with $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{d\times 2r}$.

Efficient computation The naive computation of the Cayley transform $\mathbf{R} = (\mathbf{I} - \mathbf{A})^{-1}(\mathbf{I} + \mathbf{A})$ involves inverting a $d \times d$ matrix, which takes $\mathcal{O}(d^3)$ time complexity. This is prohibitively expensive for high-dimensional features.

We now show how the low-rank structure of $\mathbf{A}$ enables a reduction in computational cost. We apply the Woodbury matrix identity [Max, 1950]. For our setting that $\mathbf{A} = \mathbf{X}\mathbf{Y}^\top$, this identity leads us to

$$(\mathbf{I}_d - \mathbf{X}\mathbf{Y}^\top)^{-1} = \mathbf{I}_d + \mathbf{X}(\mathbf{I}_{2r} - \mathbf{Y}^\top\mathbf{X})^{-1}\mathbf{Y}^\top.$$

In this case, we add subscripts for the identity matrix $\mathbf{I}$ to indicate the dimension of the vector space. The inversion of a $d \times d$ matrix is now converted to the inversion of a $2r \times 2r$ matrix $(\mathbf{I}_{2r} - \mathbf{Y}^\top\mathbf{X})$, which requires only $\mathcal{O}(r^3)$ operations.

With a further algebraic manipulation, we arrive at:

$$\mathbf{R} = (\mathbf{I} - \mathbf{A})^{-1}(\mathbf{I} + \mathbf{A}) = \mathbf{I} + 2\mathbf{X}(\mathbf{I} - \mathbf{Y}^\top\mathbf{X})^{-1}\mathbf{Y}^\top. \tag{4}$$

The main computational cost consists of (1) computing $\mathbf{Y}^\top\mathbf{X} \in \mathbb{R}^{2r\times 2r}$ in $\mathcal{O}(dr^2)$ time, (2) inverting the $2r\times 2r$ matrix in $\mathcal{O}(r^3)$ time, and (3) matrix multiplications in $\mathcal{O}(dr^2)$ time. The total complexity is $\mathcal{O}(dr^2 + r^3)$. Since $r \ll d$, typically $r \in \{2, 4, 8, \ldots\}$, this simplifies to $\mathcal{O}(dr^2)$. In practice, we adopt an input-centric execution by multiplying the input vectors with these low-rank matrices, completely avoiding the materialization of any large intermediate matrices.

Comparison with other orthogonal approaches. Existing orthogonal fine-tuning methods employ alternative structural constraints to manage computational costs: block-diagonal parameterization [Qiu *et al.*, 2023] partitions transformations into independent subspaces, while butterfly structure [Liu *et al.*, 2024b] exploits hierarchical factorizations. Our low-rank approach instead enables explicit control over the rotation subspace dimension through the rank parameter r.

Compositional Chain-of-Rotations

From a theoretical perspective, any rotation in $\mathbb{R}^d$ can be decomposed into a sequence of Givens rotations in 2-dimensional planes. Specifically, any element of $\mathrm{SO}(d)$ can be written as a product of at most $\frac{d(d-1)}{2}$ Givens rotations [Press, 2007]. Our chain-of-rotations framework can be viewed as a learnable, parameter-efficient approximation to this decomposition, where each component rotation $\mathbf{R}_i$ operates on a distinct low-dimensional subspace determined by $\mathbf{U}_i, \mathbf{V}_i$.

We define a chain of n rotations:

$$\mathbf{R} = \mathbf{R}_1\mathbf{R}_2\cdots\mathbf{R}_n = \prod_{i=1}^{n}\left(\mathbf{I} + 2\mathbf{X}_i(\mathbf{I} - \mathbf{Y}_i^\top\mathbf{X}_i)^{-1}\mathbf{Y}_i^\top\right),$$

where each $\mathbf{R}_i$ is parameterized by its own pair $\mathbf{U}_i \in \mathbb{R}^{d\times r}, \mathbf{V}_i \in \mathbb{R}^{d\times r}$. Since the composition of orthogonal matrices remains orthogonal, $\mathbf{R} \in \mathrm{SO}(d)$ is guaranteed.

To enable efficient parallel computation, we use a first-order approximation by expanding the product and retaining only linear terms:

$$\mathbf{R} \approx \mathbf{I} + 2\sum_{i=1}^{n}\mathbf{X}_i(\mathbf{I} - \mathbf{Y}_i^\top\mathbf{X}_i)^{-1}\mathbf{Y}_i^\top. \tag{5}$$

This approximation replaces sequential matrix multiplication with a sum of low-rank updates, which can be computed in parallel using standard GPU operations.

Approximation error analysis. Consider rotations where $\|\mathbf{X}_i\|_F, \|\mathbf{Y}_i\|_F = \mathcal{O}(\varepsilon)$ for small $\varepsilon > 0$. Each perturbation $\Delta_i = 2\mathbf{X}_i(\mathbf{I} - \mathbf{Y}_i^\top\mathbf{X}_i)^{-1}\mathbf{Y}_i^\top$ has magnitude $\mathcal{O}(\varepsilon^2)$, since it involves the product of $\mathbf{X}_i$ and $\mathbf{Y}_i^\top$. When expanding the product:

$$\mathbf{R} = \prod_{i=1}^{n}(\mathbf{I} + \Delta_i) = \mathbf{I} + \sum_{i=1}^{n}\Delta_i + \sum_{i<j}\Delta_i\Delta_j + \mathcal{O}(\varepsilon^6). \tag{6}$$

Here, the second-order cross terms $\Delta_i\Delta_j$ are $\mathcal{O}(\varepsilon^4)$ and higher-order terms are at least $\mathcal{O}(\varepsilon^6)$. During fine-tuning, where parameters are initialized near zero and remain small throughout training, these higher-order terms contribute negligibly to the transformation.

Orthogonality. We numerically evaluate the first-order approximation by measuring how well it preserves vector norms. For randomly initialized vectors $\mathbf{x}$, we compute the relative error $|\|\mathbf{x}\| - \|\mathbf{R}\mathbf{x}\||/\|\mathbf{x}\|$ across $\|\mathbf{X}_i\|_F = \|\mathbf{Y}_i\|_F = \varepsilon$ for $\varepsilon \in [10^{-6}, 10^{-0.5}]$, a range typical of fine-tuning perturbations. Figure 2 shows that our approximation effectively preserves vector magnitude (ideally $\|\mathbf{R}\mathbf{x}\| = \|\mathbf{x}\|$) across the range of ε values.

Remark. The first-order approximation enables all n rotations to be computed independently and summed, fully utilizing GPU parallelism, while simplifying backpropagation compared to sequential matrix products. While compositional orthogonal transformations appear in BOFT [Liu *et al.*, 2024b] (butterfly factorization) and reflection-based methods [Yuan *et al.*, 2024], neither leverages parallel computation. BOFT uses sequential butterfly stages, and reflection chains require multiple compositions. Our outer-product parameterization with first-order approximation uniquely enables fully parallel execution. This is also reflected by our experimental validation in Section 4.

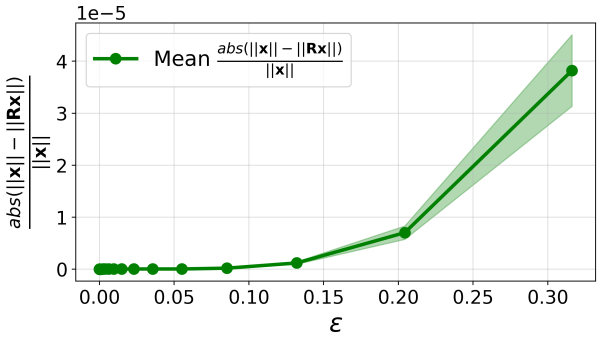

Figure 2: Relative error $|\,\|\mathbf{x}\| - \|\mathbf{R}\mathbf{x}\|\,|\,/\,\|\mathbf{x}\|$ in first-order approximations in Equation (5). This illustrates that our first-order approximation can preserve vector magnitude under linear transformation $\mathbf{R}$. In this experiment, we vary $\|\mathbf{X}\| = \|\mathbf{Y}\|$ across a range of $\varepsilon \in [10^{-6}, 10^{-0.5}]$. The shaded region indicates the standard deviation across multiple random initializations of $\mathbf{X}$ and $\mathbf{Y}$.

4 Experimental Results

We evaluate LoCO on three tasks: fine-tuning large language models, diffusion transformers, and vision transformers.

4.1 Large Language Model Adaptation

We examine LoCO on two tasks: natural language understanding (NLU) and mathematical reasoning.

Baselines. We compare against orthogonal fine-tuning methods: (a) OFT [Qiu *et al.*, 2023], which uses block-diagonal orthogonal matrices; (b) BOFT [Liu *et al.*, 2024b], which employs butterfly factorizations to construct dense orthogonal adapters; and (c) HRA [Yuan *et al.*, 2024], which chains Householder reflections [Householder, 1958].

Natural Language Understanding

Following [Qiu *et al.*, 2023; Liu *et al.*, 2024b; Yuan *et al.*, 2024], we fine-tune DeBERTa-V3-base [He *et al.*, 2021] on GLUE benchmark tasks.

Setup. We adopt the experimental setup of BOFT and HRA, applying each method to all linear modules within transformer layers. We increase OFT's block size from b=16 to b=24 to match parameter counts, and use HRA's best configuration (λ=0). Additional details are in Appendix.

Results. Table 1 reports matched accuracy for MNLI, Matthews correlation for CoLA, Pearson correlation for STS-B, and accuracy for other tasks. LoCO achieves the best average score (87.75), outperforming HRA (85.64) while using 30% and 10% fewer parameters than OFT and BOFT, respectively. LoCO ranks first on 5 out of 8 tasks (SST-2, MRPC, QNLI, RTE, STS-B). Notably, HRA shows a substantial drop on CoLA, suggesting Householder reflections may be less stable across diverse linguistic tasks than orthogonal rotations.

Method	Config	#Params	MNLI	SST-2	MRPC	CoLA	QNLI	QQP	RTE	STS-B	Mean
OFT	b=24	0.954M	**90.27**	95.83	87.93	67.63	94.30	89.77	84.27	91.00	87.63
BOFT	r=2,b=4	0.746M	**90.27**	96.00	87.07	**68.20**	94.07	89.73	81.43	91.13	87.24
HRA	r=8	0.663M	90.20	95.70	88.03	51.67	93.57	**90.20**	84.50	91.25	85.64
LoCO (ours)	n=1,r=4	0.663M	90.20	**96.14**	**88.28**	65.56	**94.36**	90.08	**86.10**	**91.28**	**87.75**

Table 1: Experimental results on GLUE benchmark. The best results on each dataset are shown in **bold**. **#Params** refers to the total number of trainable parameters in the backbone side - excluding the classification head layers which are trained in a standard way for each task.

Mathematical Reasoning

Setup. We fine-tune LLaMA2-7B [Touvron *et al.*, 2023] on MetaMathQA-40K [Yu *et al.*, 2023], adapting only the query and value projections (W_q, W_v). All baselines are reproduced under identical settings [Yuan *et al.*, 2024; Liu *et al.*, 2024b] and evaluated on the GSM8K [Cobbe *et al.*, 2021] and MATH [Hendrycks *et al.*, 2021] validation sets.

Method	Config	%Param	GSM8K	MATH
Llama-2-7B	–	–	14.6	2.5
OFT	b=64	0.12	49.35	8.39
BOFT	m=2, b=8	0.13	45.58	6.85
HRA	r=32	0.12	50.04	7.45
LoCO (ours)	n=1, r=16	0.12	**50.19**	**8.40**

Table 2: Experimental results on mathematical reasoning tasks using Llama-2-7B. We report the accuracy (%) on GSM8K and MATH benchmarks. The best results are highlighted in **bold**.

Results. As shown in Table 2, LoCO achieves the highest GSM8K score (50.19), outperforming OFT (49.35), HRA (49.65), and BOFT (45.58) with only 0.12% trainable parameters. On MATH, LoCO matches OFT while surpassing BOFT and HRA. These results suggest that low-rank skew-symmetric parameterization captures mathematical reasoning patterns more effectively than block-diagonal (OFT/BOFT) or Householder (HRA) structures.

Computational Efficiency

We compare training efficiency by measuring time per step and peak memory after 100 warm-up steps, averaged over 300 iterations across 3 trials. Experiments are conducted on DeBERTa-V3 (184M) and LLaMA2-7B with matched parameter counts. Details are provided in Appendix.

As shown in Figure 3, LoCO matches the efficiency of sparse OFT while outperforming dense methods (HRA, BOFT). Under heavy configurations, HRA incurs 160% longer step time and 20% higher memory usage, whereas LoCO maintains stable throughput. This efficiency stems from two factors: (1) the Sherman–Morrison–Woodbury formula reduces matrix inversion to $O(r^3)$ complexity, and (2) all rotation components compute in parallel, unlike HRA's sequential backward pass or BOFT's non-contiguous memory access patterns [Dao *et al.*, 2022].

Summary. Across NLU and mathematical reasoning benchmarks, LoCO achieves competitive or superior performance while maintaining both parameter efficiency and computational scalability.

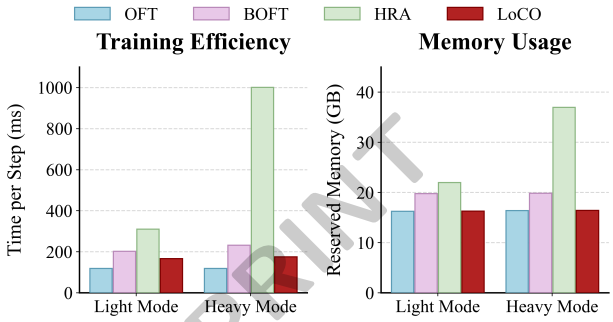

Figure 3: Training efficiency comparison on DeBERTA-V3.

4.2 Fine-tuning Diffusion Transformers

Setup. We follow the setup of [Tan *et al.*, 2025] focusing on conditional generation tasks. Similar to the OminiControl setup, we use FLUX.1 [Labs, 2024] as the pre-trained model. We conduct all experiments on 4 NVIDIA H200 GPUs. To match the total samples of OminiControl [Tan *et al.*, 2025], we train our models for 25,000 iterations (half of the setting of [Tan *et al.*, 2025] when running on 2 GPUs).

Baselines. We compare our approach against two representative baseline methods: LoRA [Hu *et al.*, 2022]—the default adapter of OminiControl [Tan *et al.*, 2025], serving as a non-orthogonal adapter—and butterfly orthogonal fine-tuning (BOFT) [Liu *et al.*, 2024b] as an orthogonal adapter. All baseline experiments utilize the official OminiControl implementation. We extend this framework to incorporate BOFT [Liu *et al.*, 2024b] into the FLUX pipeline. Our LoCO adapter is implemented within the HuggingFace PEFT library to ensure seamless integration with the existing codebase.

We evaluate performance using established metrics: Fréchet Inception Distance (FID) [Heusel *et al.*, 2017], Structural Similarity Index (SSIM), CLIP-IQA [Wang *et al.*, 2023], MANIQA [Yang *et al.*, 2022], MUSIQ [Ke *et al.*, 2021], and Peak Signal-to-Noise Ratio (PSNR) [contributors, 2025]. For controllability assessment, we adopt the OminiControl approach, measuring the F1 score between extracted and input edge maps for edge-conditioned generation, and Mean Squared Error (MSE) between extracted and original condition maps for other tasks. To evaluate consistency between prompts and generated images, we use CLIP Text-Image similarity. Following OminiControl [Tan *et al.*, 2025], we evaluate on 5,000 images from the COCO dataset, with prompts

extracted from corresponding image captions.

Results. Table 3 presents quantitative results across spatially-aligned controllable generation tasks. Our method demonstrates competitive performance across the evaluation landscape. Specifically, on image quality metrics (CLIP-IQA, MAN-IQA, MUSIQ), our approach consistently achieves performance comparable to or exceeding the baselines. More significantly, our method exhibits superior alignment fidelity as measured by CLIP Image scores across the majority of tasks. For instance, on the Mask task, our method achieves a CLIP Image score of 0.824, outperforming BOFT (0.801) and LoRA (0.812). Similarly, on the Deblur task, we obtain 0.832 compared to 0.803 for BOFT. These results indicate that our low-rank compositional rotation approach effectively preserves semantic alignment. Qualitative examples of these methods are provided in Appendix.

Test-time adjustable generation. We introduce a temperature parameter t for the skew-symmetric matrix $\mathbf{A}$, whereby the rotation matrix is computed as $\mathbf{R} = (\mathbf{I} - \mathbf{A}t)^{-1}(\mathbf{I} + \mathbf{A}t)$. During training, we use $t = 1$. At inference time, t can be adjusted to control the degree of adaptation. Since the scaled matrix $\mathbf{A}t$ remains within the tangent space $\mathfrak{so}(d)$ for varying values of t, the resulting transformations continue to yield semantically meaningful outputs in the generated images. This property is preserved because the Cayley transformation provides a good approximation to $\exp(\mathbf{A}t)$ given a small magnitude of $\mathbf{A}$.

This adjustable inference mechanism described here has not been explored by prior orthogonal fine-tuning approaches [Qiu *et al.*, 2023; Liu *et al.*, 2024b]. Furthermore, orthogonal fine-tuning methods based on reflection transformations, such as Householder-based approaches [Yuan *et al.*, 2024], do not possess this capability due to their different geometric structure.

Figure 4 illustrates the visual quality of generated images across a range of temperature values $t \in [0, 2]$. Additional figures demonstrating generated images beyond this range using LoCO are provided in the Appendix. Compared with images generated by varying the scaling parameter α in LoRA for OminiControl (the two bottom rows in Figure 4), our method exhibits robustness and stability across different scaling factors. This is verified quantitatively by image quality metrics in Figure 5. The generation quality when varying t is more robust and less brittle than LoRA.

Effect of the number of rotations n. We ablate the number of rotations (using values of 2, 4, 6, 8) on the Canny edge detection setting. Table 4 shows the results. We observe that the model benefits from a higher number of rotations, which increases the expressiveness of the LoCO adapter. However, since each rotation approximation can be computed in parallel, GPU memory increases linearly with the number of rotations.

4.3 Fine-tuning Vision Transformers (ViTs)

Dataset. We conduct experiments fine-tuning vision models on the Visual Task Adaptation Benchmark (VTAB). Specifically, we employ VTAB-1k [Zhai *et al.*, 2019], which comprises 19 diverse visual classification tasks organized into

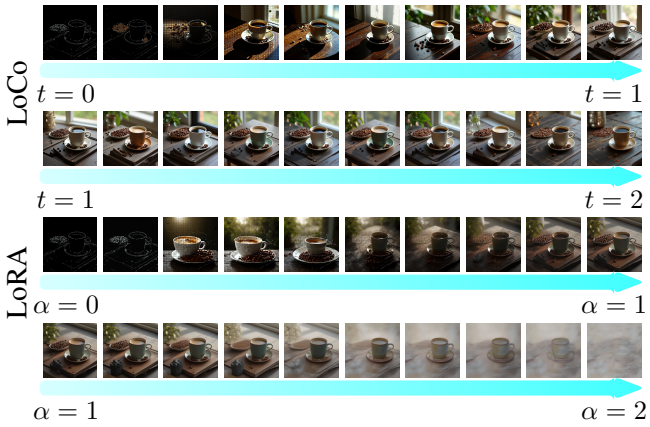

Figure 4: Effect of varying the temperature parameter t for LoCo (top two rows) and scaling parameter α for LoRA (bottom two rows) on image generation quality. LoCo was trained using $\mathbf{R} = (\mathbf{I} - \mathbf{A}t)^{-1}(\mathbf{I} + \mathbf{A}t)$. During inference, we vary t within the range $[0, 1]$ (first row) and $[1, 2]$ (second row). We observe that for $t \in [1-l, 1+u]$ for some l, u, the generated images maintain high visual quality. Zoom for better view quality. Additional high-quality visualizations are available in Appendix.

three distinct groups: Natural (tasks featuring natural images), Specialized (medical and satellite imagery), and Structured (tasks requiring geometric understanding). In the fine-tuning setting, the number of training samples for each task is constrained to 1,000 examples. The test dataset for evaluation remains consistent with the original VTAB test set.

Pretrained model. We use ViT-B/16 [Dosovitskiy, 2020] as our main backbone, which has been pretrained on the large-scale ImageNet-21K dataset.

Baselines. We compare our method against two distinct categories of baselines: non-orthogonal transformation methods and orthogonal adaptation methods. For non-orthogonal baselines, we include LoRA [Hu *et al.*, 2022], which decomposes weight updates using low-rank matrices; Adapter [Houlsby *et al.*, 2019], which inserts bottleneck modules; Scale-and-Shift Feature (SSF) [Lian *et al.*, 2022], which performs lightweight feature transformations; and Visual Prompt Tuning (VPT) [Jia *et al.*, 2022], which learns task-specific prompt embeddings. For orthogonal fine-tuning baselines, we compare with BOFT [Liu *et al.*, 2024b] and HRA [Yuan *et al.*, 2024]. We perform a hyperparameter search for the learning rate of all models from the set $[5e{-}5, 1e{-}4, 5e{-}4, 1e{-}3]$ and report the best test performance for each method.

Results. Figure 6 shows the performance across models. LoCO achieves competitive accuracy across all three VTAB-1k benchmark categories, with the highest average among orthogonal PEFT methods on Structured tasks (63.4%) while remaining competitive on Specialized tasks (86.4%) and Natural tasks (82.3%). On Natural images, LoCO obtains strong results on Caltech101 (95.6%), DTD (71.4%), and Flowers102 (99.4%). Notably, LoCO shows substantial gains on

Task	Methods	Controllability	Image Quality						Alignment	
		F1↑ / MSE↓	FID↓	SSIM↑	CLIP-IQA↑	MAN-IQA↑	MUSIQ↑	PSNR↑	CLIP Text↑	CLIP Image↑
Canny	LoRA	**0.52**	25.12	**0.45**	0.62	0.57	74.22	**11.44**	0.331	0.798
	BOFT	0.4	28.71	0.39	0.65	0.58	73.82	10.89	0.308	**0.799**
	LoCO (ours)	0.49	**25.00**	0.44	**0.66**	**0.61**	**74.24**	11.17	**0.336**	0.788
Depth	LoRA	**624**	**29.98**	0.31	0.62	0.52	71.65	11.33	0.315	0.729
	BOFT	752	31.29	0.32	0.59	0.50	71.99	**11.35**	0.304	0.726
	LoCO (ours)	723	32.01	**0.36**	**0.63**	**0.53**	**72.50**	9.85	**0.319**	**0.741**
Mask	LoRA	**6402**	10.11	0.62	0.58	0.47	68.81	**18.31**	**0.292**	0.812
	BOFT	7482	**10.06**	0.55	0.51	0.49	66.31	18.01	0.283	0.801
	LoCO (ours)	6931	10.14	**0.63**	**0.59**	**0.50**	**69.80**	18.08	0.290	**0.824**
Colorization	LoRA	**103**	10.51	0.91	**0.54**	**0.45**	**67.33**	**22.30**	0.298	0.826
	BOFT	121	**10.01**	**0.92**	0.51	0.42	66.32	22.19	0.297	0.812
	LLoCO (ours)	106	10.33	0.90	0.50	0.44	66.44	21.26	**0.310**	0.831
Deblur	LoRA	**79**	20.27	**0.65**	0.41	0.35	58.75	**23.95**	0.290	0.812
	BOFT	87	22.94	0.60	0.49	0.40	57.95	20.78	0.304	0.803
	LoCO (ours)	83	**20.06**	0.61	**0.54**	**0.48**	**65.85**	21.79	**0.318**	**0.832**

Table 3: Quantitative comparison across different controllable generation tasks. ↑ indicates higher is better, ↓ indicates lower is better.

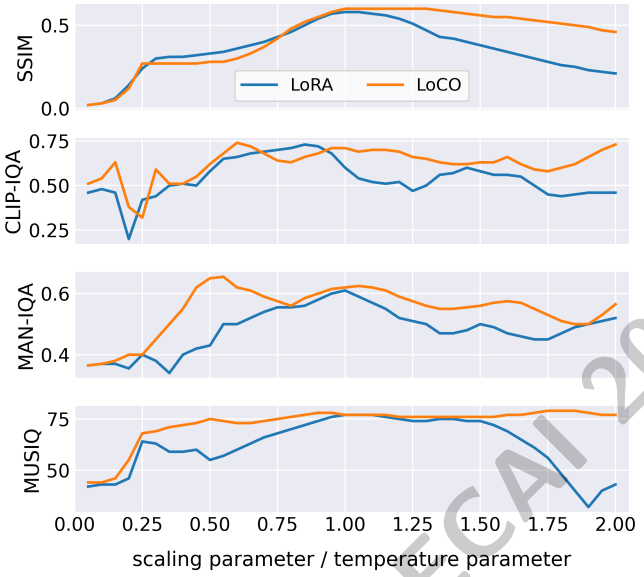

Figure 5: Image quality indicated by SSIM, CLIP-IQA, MAN-IQA, and MUSIQ criteria of the "coffee" sample as the scaling parameter α or temperature parameter t varies. We observe that the generated images by LoCO maintain high quality over a wider range than those by LoRA.

#Rotations	FID ↓	SSIM ↑	CLIP-IQA ↑	MAN-IQA ↑
2	33.66	0.37	0.38	0.51
4	32.91	0.36	0.42	0.55
6	27.12	0.42	0.59	0.59
8	25.00	0.44	0.66	0.61

Table 4: Impact of the number of rotations n on LoCO performance given a fixed rank r.

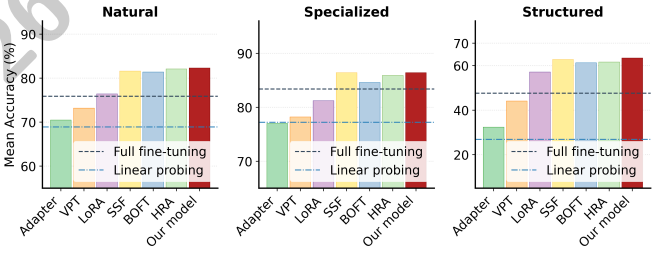

Figure 6: Performance comparison on VTAB-1k benchmark across Natural, Specialized, and Structured task groups. LoCO consistently achieves competitive or superior accuracy compared to both non-orthogonal and orthogonal fine-tuning baselines.

out-of-distribution tasks: PatchCamelyon reaches 88.1% versus BOFT's 83.2% (+4.9%), demonstrating effective transfer to medical imaging. Performance on EuroSAT (95.6%) and RESISC45 (85.4%) further confirms robustness on specialized domains. Detailed per-dataset results are provided in Appendix.

5 Conclusion

We introduce a low-rank compositional orthogonal fine-tuning (LoCO), a parameter-efficient method that adapts foundation models through orthogonal transformations. By constructing orthogonal matrices via low-rank skew-symmetric parameterization and leveraging the Cayley transform with the Sherman-Morrison-Woodbury identity, LoCO reduces computational complexity while allowing fully parallel computation of compositional rotations. Experiments across natural language understanding, mathematical reasoning, diffusion transformers, and vision transformers demonstrate consistent improvements over existing orthogonal fine-tuning methods.

Limitations and future work. While LoCO's low-rank formulation addresses scalability for typical fine-tuning scenarios, high-dimensional settings may benefit from further optimization. Future directions include theoretical analysis of optimal rotation decompositions, extending temperature scaling to advanced applications such as combining multiple adaptations with multiple scaling factors, and application to multimodal models.

Acknowledgements

This work was supported by 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);RS-2019-II190075, Artificial Intelligence Graduate School Program (KAIST); No. RS-2022-II220984, Development of Artificial Intelligence Technology for Personalized Plug-and-Play Explanation and Verification of Explanation; No. RS-2024-00457882, AI Research Hub Project), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. RS-2025-23525509) and the High-Performance Computing Support Project, funded by the Government of the Republic of Korea (Ministry of Science and ICT) (No. RQT-25-070263).

References

[Aghajanyan *et al.*, 2020] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv preprint arXiv:2012.13255*, 2020.

[Arcas *et al.*, 2025] Alejandro Moreno Arcas, Albert Sanchis, Jorge Civera, and Alfons Juan. Hoft: Householder orthogonal fine-tuning. *arXiv preprint arXiv:2505.16531*, 2025.

[Bałazy *et al.*, 2024] Klaudia Bałazy, Mohammadreza Banaei, Karl Aberer, and Jacek Tabor. Lora-xs: Low-rank adaptation with extremely small number of parameters. *arXiv preprint arXiv:2405.17604*, 2024.

[Brown *et al.*, 2020] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 2020.

[Chen *et al.*, 2020] Beidi Chen, Weiyang Liu, Zhiding Yu, Jan Kautz, Anshumali Shrivastava, Animesh Garg, and Animashree Anandkumar. Angular visual hardness. In *ICML*, 2020.

[Chen *et al.*, 2022] Weize Chen, Xu Han, Yankai Lin, Hexu Zhao, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. Fully hyperbolic neural networks. In *ACL*, 2022.

[Cobbe *et al.*, 2021] Karl Cobbe, Vineet Kosaraju, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[contributors, 2025] Wikipedia contributors. Peak signal-to-noise ratio. https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio, 2025.

[Dao *et al.*, 2022] Tri Dao, Beidi Chen, et al. Monarch: Expressive structured matrices for efficient and accurate training. In *International Conference on Machine Learning*, 2022.

[Dong *et al.*, 2023] Wei Dong, Dawei Yan, Zhijun Lin, and Peng Wang. Efficient adaptation of large vision transformer via adapter re-composing. In *NeurIPS*, 2023.

[Dong *et al.*, 2024] Wei Dong, Yuan Sun, Yiting Yang, Xing Zhang, Zhijun Lin, Qingsen Yan, Haokui Zhang, Peng Wang, Yang Yang, and Hengtao Shen. Efficient adaptation of pre-trained vision transformer via householder transformation. *NeurIPS*, 2024.

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

[Finzi *et al.*, 2020] Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data. In *ICML*, 2020.

[Ganea *et al.*, 2018] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *ICML*, 2018.

[Gorbunov *et al.*, 2024] Mikhail Gorbunov, Kolya Yudin, Maxim Rakhuba, et al. Group and Shuffle: Efficient Structured Orthogonal Parametrization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[He *et al.*, 2021] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *ICLR*, 2021.

[Hendrycks *et al.*, 2021] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *ICLR*, 2021.

[Heusel *et al.*, 2017] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017.

[Houlsby *et al.*, 2019] 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 *ICML*, 2019.

[Householder, 1958] Alston S. Householder. Unitary triangularization of a nonsymmetric matrix. *J. ACM*, 1958.

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

[Jia *et al.*, 2022] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, 2022.

[Jiang *et al.*, 2024] Ting Jiang, Shaohan Huang, Shengyue Luo, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, Qi Zhang, Deqing Wang, and Fuzhen Zhuang. Mora: High-rank updating for parameter-efficient fine-tuning. arXiv preprint arXiv:2405.12130, 2024.

[Kalajdzievski, 2023] Damjan Kalajdzievski. A rank stabilization scaling factor for fine-tuning with lora. arXiv preprint arXiv:2312.03732, 2023.

[Ke *et al.*, 2021] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *ICCV*, 2021.

[Labs, 2024] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

[Lester *et al.*, 2021] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *EMNLP*, 2021.

[Li and Liang, 2021] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL*, 2021.

[Lian *et al.*, 2022] Dongze Lian, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Scaling & shifting your features: A new baseline for efficient model tuning. *NeurIPS*, 2022.

[Liao and Monz, 2024] Baohao Liao and Christof Monz. 3-in-1: 2d rotary adaptation for efficient finetuning, efficient batching and composability. *NeurIPS*, 2024.

[Liu *et al.*, 2017] Weiyang Liu, Yan-Ming Zhang, Xingguo Li, Zhiding Yu, Bo Dai, Tuo Zhao, and Le Song. Deep hyperspherical learning. *NeurIPS*, 2017.

[Liu *et al.*, 2018] Weiyang Liu, Zhen Liu, Zhiding Yu, Bo Dai, Rongmei Lin, Yisen Wang, James M Rehg, and Le Song. Decoupled networks. In *CVPR*, 2018.

[Liu *et al.*, 2024a] 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. In *ICML*, 2024.

[Liu *et al.*, 2024b] Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, Yandong Wen, Michael J. Black, Adrian Weller, and Bernhard Schölkopf. Parameter-efficient orthogonal finetuning via butterfly factorization. In *ICLR*, 2024.

[Ma *et al.*, 2024] Xinyu Ma, Xu Chu, Zhibang Yang, Yang Lin, Xin Gao, and Junfeng Zhao. Parameter efficient quasi-orthogonal fine-tuning via givens rotation. *arXiv preprint arXiv:2404.04316*, 2024.

[Max, 1950] A Woodbury Max. Inverting modified matrices. In *Memorandum Rept. 42, Statistical Research Group*. Princeton Univ., 1950.

[Meng *et al.*, 2024] Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: Principal singular values and singular vectors adaptation of large language models. *NeurIPS*, 2024.

[Mironenco and Forré, 2024] Mircea Mironenco and Patrick Forré. Lie group decompositions for equivariant neural networks. In *ICLR*, 2024.

[Peebles and Xie, 2023] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *CVPR*, 2023.

[Press, 2007] William H Press. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[Qiu *et al.*, 2023] 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. *NeurIPS*, 2023.

[Qiu *et al.*, 2025] Zeju Qiu, Simon Buchholz, Tim Z Xiao, Maximilian Dax, Bernhard Schölkopf, and Weiyang Liu. Reparameterized llm training via orthogonal equivalence transformation. *arXiv preprint arXiv:2506.08001*, 2025.

[Rombach *et al.*, 2022] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[Singla and Feizi, 2021] Sahil Singla and Soheil Feizi. Skew orthogonal convolutions. In *ICML*, 2021.

[Tan *et al.*, 2025] Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and universal control for diffusion transformer. In *ICCV*, 2025.

[Touvron *et al.*, 2023] Hugo Touvron, Louis Martin, Kevin Stone, et al. Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288, 2023.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017.

[Wang *et al.*, 2023] Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *AAAI*, 2023.

[Wang *et al.*, 2024] Hanqing Wang, Yixia Li, Shuo Wang, Guanhua Chen, and Yun Chen. Milora: Harnessing minor singular components for parameter-efficient llm finetuning. *arXiv preprint arXiv:2406.09044*, 2024.

[Yang *et al.*, 2022] Sidi Yang, Tianhe Wu, Shuwei Shi, Shanshan Lao, Yuan Gong, Mingdeng Cao, Jiahao Wang, and Yujiu Yang. Maniqa: Multi-dimension attention network for no-reference image quality assessment. In *CVPR*, 2022.

[Yu *et al.*, 2023] Longhui Yu, Weisen Jiang, et al. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*, 2023.

[Yuan *et al.*, 2024] Shen Yuan, Haotian Liu, and Hongteng Xu. Bridging the gap between low-rank and orthogonal adaptation via householder reflection adaptation. *NeurIPS*, 2024.

[Zhai *et al.*, 2019] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

[Zhang *et al.*, 2023] Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *ICLR*, 2023.