# DECOUPLING ROTATION AND DEFORMATION VIA FROZEN RESERVOIRS FOR EFFICIENT FINE-TUNING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Parameter-efficient fine-tuning (PEFT) is widely used for low-cost adaptation of pretrained models. From a geometric perspective, full fine-tuning permits flexible, high-rank updates that alter the orientation and scale of weight mappings, whereas representative PEFT methods restrict this flexibility through low-rank or orthogonality constraints. To enable high-rank weight updates and independently control rotation and deformation in weight transformations under a limited parameter budget, we propose Geometrically Decoupled Reservoir Adaptation (GeDRA). GeDRA decomposes a frozen operator reservoir into rotation-generator and deformation-generator templates and learns one modulation vector per template, enabling separate control over local rotation and deformation updates in the tangent space. Theoretical analysis reveals an expressivity advantage of geometric decoupling in terms of expected optimal approximation error under matched trainable parameter budgets. We further observe broad directional support for both rotation and deformation in GeDRA, qualitatively similar to full fine-tuning. Experiments on commonsense reasoning, image classification, and natural language understanding demonstrate favorable performance–parameter trade-offs against strong baselines. Qualitative comparisons with LoRA further illustrate GeDRA's effectiveness in subject-driven generation.

## 1 INTRODUCTION

Pretrained models provide a strong foundation for a wide range of language and vision tasks (Devlin et al., 2019; Dosovitskiy et al., 2020). Fine-tuning on downstream data is a common way to adapt these models to specific tasks or domains. However, full fine-tuning updates all model parameters, placing substantial demands on training memory and requiring a separate set of weights for each task (Hu et al., 2021). These costs become increasingly burdensome as both model size and the number of downstream tasks grow. Parameter-efficient fine-tuning (PEFT) addresses this challenge by learning a small set of task-specific parameters while keeping most pretrained weights frozen (Houlsby et al., 2019; Li & Liang, 2021). This allows a shared pretrained backbone to support multiple tasks with only a small amount of additional storage per task. A central challenge is therefore to retain the flexibility needed for downstream adaptation under a limited trainable parameter budget.

Low-rank adaptation (LoRA) addresses this challenge by representing additive weight updates with two low-rank factors (Hu et al., 2021). Subsequent work improves the use of the parameter budget through adaptive rank allocation and refined initialization and optimization (Zhang et al., 2023; Meng et al., 2024; Hayou et al., 2024). Complementary designs separate weight magnitude from direction (Liu et al., 2024a) or decouple angular learning from adaptation strength (Bini et al., 2025). For conventional additive low-rank updates, however, the rank constraint limits the range of representable weight changes (Zeng & Lee, 2024), and increasing the rank requires larger trainable factors. Orthogonal fine-tuning adopts a different structural prior, using orthogonal transformations to reorient pretrained weight vectors while preserving their norms and pairwise angles (Qiu et al., 2023). Block-diagonal and butterfly parameterizations reduce the number of trainable parameters needed for these transformations (Liu et al., 2024b). Strict orthogonality, however, excludes non-isometric deformation, restricting the flexibility of weight adaptation. qGOFT (Ma et al., 2024) and PSOFT (Wu et al., 2026) relax this constraint through soft orthogonality regularization and trainable scaling, respectively. These developments highlight the importance of the geometric freedom retained by an update parameterization, beyond its trainable parameter count.



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

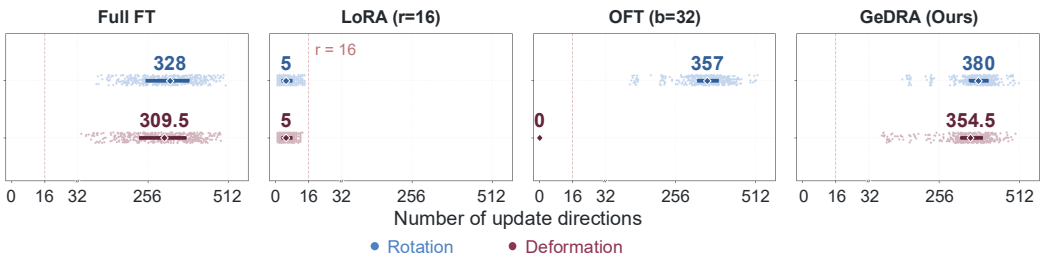



Figure 1: **Directional support of rotation and deformation.** Each point corresponds to a query or value projection weight update for a given VTAB-1K task and layer. The horizontal axis shows the minimum number of singular modes of the weight update needed to account for 99% of each component's mode-wise energy. Thick bars and diamonds indicate interquartile ranges and medians, respectively. All panels use the same piecewise-linear horizontal scale, with $[0, 32]$ expanded; the red dashed line marks the LoRA rank, $r = 16$.

To examine the geometric implications of these constraints, we use a spectral diagnostic to compare the directional support of rotation and deformation in the query and value projection updates of ViT-B/16 across the 19 VTAB-1K tasks (Zhai et al., 2019). For each component, we measure directional support by the minimum number of weight-update singular modes accounting for 99% of its assigned energy. The spectral decomposition and numerical criteria are detailed in Appendix D. Figure 1 shows that full fine-tuning exhibits broad directional support for both rotation and deformation. For LoRA, the energy associated with both components is concentrated in relatively few update directions. OFT, in contrast, exhibits broad rotation support but no detectable change in the weight singular-value spectrum. These observations raise a central question: how can we retain more of full fine-tuning's flexibility in rotation and deformation under a limited trainable parameter budget?

Reservoir computing offers a useful perspective on this question. Classical reservoir models generate high-dimensional nonlinear representations through fixed recurrent connections, with learning confined to a readout layer (Jaeger, 2007; Lukoševičius, 2012; Dong et al., 2020). A related principle appears in fine-tuning with frozen random matrix factors and learned scaling vectors (Kopiczko et al., 2024). Extensions using combinations of frozen low-rank matrices or random tensor networks further support high-rank updates with few trainable parameters (Albert et al., 2025; Gu et al., 2026). These approaches demonstrate that reducing the trainable parameter budget need not entail low-rank weight updates. In these formulations, frozen matrices provide the transformation structure, while learned coefficients adjust the resulting weight mapping. Such formulations allow changes in both orientation and scale without learning the underlying matrix factors, offering a promising direction for retaining geometric flexibility under a limited parameter budget.

In this paper, we propose Geometrically Decoupled Reservoir Adaptation (GeDRA), a parameter-efficient fine-tuning method that independently modulates rotation and deformation in weight transformations. GeDRA constructs a frozen operator reservoir from signed permutation matrices and the identity, then decomposes it into rotation-generator and deformation-generator templates. It learns one modulation vector for each template, yielding task-specific generators while preserving their respective structures. The resulting generators are combined in the tangent space to define a local transformation of the pretrained weights. This parameterization supports high-rank weight updates and accommodates non-isometric deformation, while requiring only two trainable vectors per adapted weight matrix. Theoretical analysis reveals an expressivity advantage of geometric decoupling in terms of expected optimal approximation error under matched trainable parameter budgets. Experiments on commonsense reasoning, image classification, and natural language understanding demonstrate favorable performance–parameter trade-offs against strong baselines. Qualitative comparisons with LoRA further illustrate GeDRA's effectiveness in subject-driven generation.

## 2 Related Work

**Low-rank Adaptation.** Low-rank adaptation controls the trainable parameter budget through a factorized representation of weight updates. LoRA (Hu et al., 2021) learns two matrix factors whose intermediate dimension bounds the update rank. Subsequent methods refine how this limited capacity

is used, through adaptive rank allocation in AdaLoRA (Zhang et al., 2023) and initialization from pretrained principal singular components in PiSSA (Meng et al., 2024). Weight decomposition provides a complementary form of control: DoRA (Liu et al., 2024a) separates magnitude from direction, combining a trainable magnitude vector with low-rank directional adaptation. DeLoRA (Bini et al., 2025) introduces factor normalization and update scaling to decouple angular learning from adaptation strength. Related vector-based approaches adapt activations or singular values without learning matrix factors: $(\mathrm{IA})^3$ (Liu et al., 2022) rescales attention and feed-forward activations, whereas SVDiff (Han et al., 2023) adjusts singular values while retaining the pretrained singular vectors. For conventional additive updates parameterized as a single low-rank product, however, the factorization rank still bounds the number of independent update directions, restricting the weight changes that can be represented (Zeng & Lee, 2024).

**Orthogonal Adaptation.** Orthogonal adaptation uses the geometry of pretrained weights as a structural prior. Under strict orthogonality, weight vectors can be reoriented without changing their norms or pairwise angles. Efficient parameterizations differ in how they couple weight coordinates: standard OFT (Qiu et al., 2023) uses block-diagonal transformations, whereas BOFT (Liu et al., 2024b) composes sparse orthogonal butterfly factors to introduce cross-block interactions. Strict geometric preservation also excludes non-isometric deformation, motivating extensions that relax this constraint. qGOFT (Ma et al., 2024) introduces soft orthogonality regularization in a Givens-based parameterization, allowing norms and relative angles to change. PSOFT (Wu et al., 2026) combines orthogonal adaptation within a principal subspace with trainable scaling on both sides of the orthogonal core, reducing the dimension of the learned transformation while permitting non-isometric adjustments. These developments extend adaptation beyond strictly geometry-preserving transformations, highlighting the value of allowing non-isometric changes alongside rotation.

**Random-matrix Adaptation.** Random-matrix adaptation uses frozen random factors and projections to reduce the parameters learned during fine-tuning. VeRA (Kopiczko et al., 2024) shares frozen random factors across layers and learns layer-specific scaling vectors. Frozen projections also enable input-dependent adaptation: FlyLoRA (Zou et al., 2025) combines a frozen sparse random down-projection with top-$k$ expert selection, using projected activations to route inputs to rank-wise experts in a trainable up-projection. High-rank updates can likewise be constructed from frozen random structures, using combinations of low-rank matrices in RandLoRA (Albert et al., 2025) or random tensor networks in TeRA (Gu et al., 2026). Both methods keep the underlying random factors frozen and optimize only scaling vectors. These constructions demonstrate that a small trainable parameter budget need not restrict weight updates to low rank.

## 3 METHODOLOGY

GeDRA separates fixed transformation structures from task-specific parameter learning through independent geometric modulation. As illustrated in Figure 2, it constructs a frozen operator reservoir, decomposes it into rotation- and deformation-generator templates, and learns a modulation vector for each template. For a frozen pretrained weight matrix $W_0 \in \mathbb{R}^{n\times m}$, the adapted weights take the form

$$W_\theta = (I_n + A_\theta)W_0, \qquad \Delta W = A_\theta W_0, \tag{1}$$

where $A_\theta \in \mathbb{R}^{n\times n}$ is the geometric adaptation generator. We first describe its construction and then analyze its approximation capacity under a matched trainable parameter budget.

### 3.1 FROZEN OPERATOR RESERVOIR

The reservoir supplies the fixed operator structure on which adaptation is learned. We construct it from signed permutation matrices,

$$U_\ell = \mathrm{Diag}(\xi_\ell)P_\ell, \qquad Q = I_n + \frac{1}{\sqrt{2M}}\sum_{\ell=1}^{M} U_\ell, \tag{2}$$

where $P_\ell$ is sampled uniformly from the set of $n \times n$ permutation matrices and $\xi_\ell \in \{-1, +1\}^n$ has independent Rademacher entries. The permutations and sign vectors are mutually independent. All sampled operators remain frozen during fine-tuning.

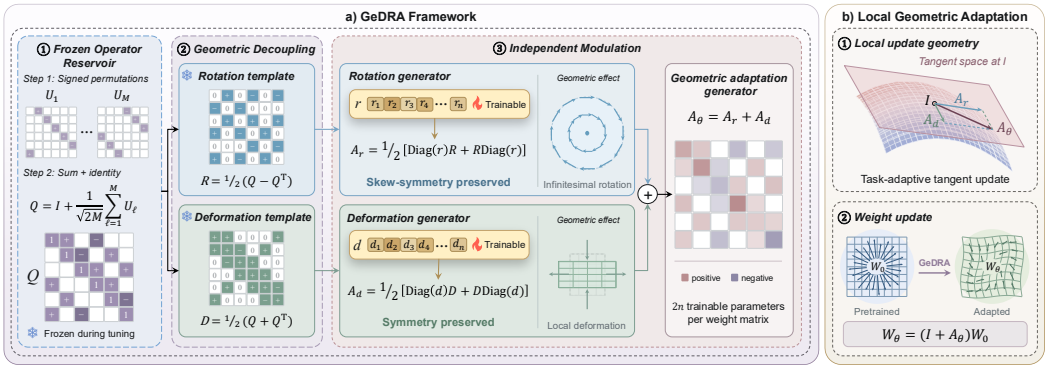



Figure 2: **The framework of GeDRA. a)** GeDRA constructs a frozen operator reservoir and decomposes it into rotation- and deformation-generator templates. Two trainable vectors, $\mathbf{r}$ and $\mathbf{d}$, independently modulate the templates to form generators $A_r$ and $A_d$. **b)** The generators are combined into $A_\theta$, which parameterizes a local transformation of the pretrained weights.

Each $U_\ell$ provides a signed rearrangement of coordinates, and their superposition forms the reservoir's interaction structure. The identity term adds a fixed diagonal component. The normalization keeps the expected squared Frobenius norm of the random component independent of $M$: $\mathbb{E}\|Q - I_n\|_F^2 = n/2$. Thus, $M$ controls the number of frozen operators contributing to the reservoir without changing its expected perturbation energy. Rather than learning the entries of $Q$, GeDRA learns how its geometric components contribute to the adaptation.

### 3.2 Independent Geometric Modulation

**Geometric decomposition.** We decompose the reservoir into its skew-symmetric and symmetric components,

$$R = \frac{Q - Q^\top}{2}, \qquad D = \frac{Q + Q^\top}{2}. \tag{3}$$

These satisfy $R^\top = -R$, $D^\top = D$, and $Q = R + D$. We use $R$ as the rotation-generator template and $D$ as the deformation-generator template. Their structures distinguish infinitesimal rotation from local stretching and compression, providing separate components for geometric adaptation.

**Structure-preserving modulation.** For each adapted weight matrix, GeDRA learns two vectors, $\mathbf{r}, \mathbf{d} \in \mathbb{R}^n$, and constructs

$$A_r = \frac{\mathrm{Diag}(\mathbf{r})R + R\,\mathrm{Diag}(\mathbf{r})}{2}, \qquad A_d = \frac{\mathrm{Diag}(\mathbf{d})D + D\,\mathrm{Diag}(\mathbf{d})}{2}. \tag{4}$$

The paired left and right modulation preserves the template structures: $A_r^\top = -A_r$ and $A_d^\top = A_d$. In contrast, one-sided diagonal modulation generally does not preserve either property. Consequently, $\mathbf{r}$ adjusts only the skew-symmetric component of the generator, while $\mathbf{d}$ adjusts only its symmetric component.

The two generators are combined as

$$A_\theta = A_{\mathrm{GeDRA}}(\mathbf{r}, \mathbf{d}) = A_r + A_d, \tag{5}$$

and the resulting first-order transformation $I_n + A_\theta$ adapts the pretrained weights through Equation (1). The geometric interpretation follows by considering $F(t) = I_n + tA_\theta$:

$$F(t)^\top F(t) = I_n + 2tA_d + t^2 A_\theta^\top A_\theta. \tag{6}$$

To first order, the skew-symmetric generator represents rotation without changing inner products, whereas the symmetric generator determines changes in lengths and angles. GeDRA therefore separates the two types of local geometric adjustment at the generator level. The combined finite transformation is not constrained to be orthogonal.

**Parameter efficiency and update rank.** Only $\mathbf{r}$ and $\mathbf{d}$ are trainable, giving $2n$ scalar parameters per adapted weight matrix, independently of the number $M$ of frozen reservoir operators. This parameter count does not impose a low-rank factorization on $\Delta W$. For example, setting $\mathbf{r} = \mathbf{d} = c\mathbf{1}$ for any nonzero scalar $c$ gives $A_\theta = c(R + D) = cQ$. Whenever $Q$ is nonsingular,

$$\operatorname{rank}(\Delta W) = \operatorname{rank}(cQW_0) = \operatorname{rank}(W_0). \tag{7}$$

Since $c$ can be arbitrarily small, high-rank updates are compatible with both vector-level parameter learning and the local transformation formulation.

We initialize both modulation vectors to zero, so fine-tuning starts from the pretrained weights. After training, the learned transformation can be merged into $W_\theta = (I_n + A_\theta)W_0$, so inference can use an ordinary weight matrix without an additional adaptation branch.

### 3.3 Expressivity Analysis

We examine what independent geometric modulation offers beyond directly scaling the same frozen reservoir. As a reference, consider the two-sided scaling (TSS) parameterization

$$A_{\mathrm{TSS}}(\mathbf{a}, \mathbf{b}) = \operatorname{Diag}(\mathbf{a}) Q \operatorname{Diag}(\mathbf{b}), \tag{8}$$

where $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$. Both GeDRA and TSS use two length-$n$ trainable vectors. This matches their nominal parameter counts, rather than their intrinsic degrees of freedom: TSS is invariant under $(\mathbf{a}, \mathbf{b}) \mapsto (c\mathbf{a}, \mathbf{b}/c)$ for $c \neq 0$.

To assess whether the two geometric components can be reweighted independently, we consider the target family

$$T_{\alpha,\beta}(Q) = \alpha R + \beta D, \qquad \alpha, \beta \in \mathbb{R}. \tag{9}$$

Here, $\alpha$ and $\beta$ prescribe the respective contributions of the rotation and deformation templates. Using Equation (3), the target can also be written as

$$T_{\alpha,\beta}(Q) = \frac{\alpha + \beta}{2} Q + \frac{\beta - \alpha}{2} Q^\top. \tag{10}$$

This expression reveals a structural distinction. TSS preserves every zero entry of $Q$, whereas unequal weighting of the two geometric components can introduce nonzero target entries through $Q^\top$. We quantify this distinction by the one-way support energy $\Gamma(Q) = \sum_{\substack{i \neq j \\ Q_{ji}=0}} Q_{ij}^2$. It measures the squared magnitude of off-diagonal entries whose transposed positions are zero.

**Theorem 1** (Deterministic support separation)**.** *For every fixed $Q \in \mathbb{R}^{n \times n}$ and every $\alpha, \beta \in \mathbb{R}$,*

$$\min_{\mathbf{r},\mathbf{d} \in \mathbb{R}^n} \|T_{\alpha,\beta}(Q) - A_{\mathrm{GeDRA}}(\mathbf{r}, \mathbf{d})\|_F^2 = 0, \tag{11}$$

$$\inf_{\mathbf{a},\mathbf{b} \in \mathbb{R}^n} \|T_{\alpha,\beta}(Q) - A_{\mathrm{TSS}}(\mathbf{a}, \mathbf{b})\|_F^2 \geq \frac{(\alpha - \beta)^2}{4} \Gamma(Q). \tag{12}$$

*The GeDRA minimum is attained by $\mathbf{r} = \alpha\mathbf{1}$ and $\mathbf{d} = \beta\mathbf{1}$. The separation is strict whenever $\alpha \neq \beta$ and $\Gamma(Q) > 0$.*

The lower bound follows from entries that TSS cannot change: when $Q_{ji} = 0$, the target entry is $[T_{\alpha,\beta}(Q)]_{ji} = (\beta - \alpha)Q_{ij}/2$, while the corresponding TSS entry remains zero. The same bound therefore applies to any approximating matrix whose nonzero entries are restricted to those of $Q$. GeDRA represents the target exactly by independently weighting the two templates.

We next quantify this separation for the random reservoir construction in Equation (2).

**Theorem 2** (Random-reservoir separation)**.** *Let $Q$ be sampled according to Equation* (2)*, with integers $n \geq 2$ and $M \geq 1$. For any fixed $\alpha, \beta \in \mathbb{R}$,*

$$\mathbb{E}_Q \left[ \inf_{\mathbf{a},\mathbf{b}} \|T_{\alpha,\beta}(Q) - A_{\mathrm{TSS}}(\mathbf{a}, \mathbf{b})\|_F^2 \right] \geq \frac{(\alpha - \beta)^2}{4} \left( \frac{n}{2} - M \right)_+, \tag{13}$$

*where $(x)_+ = \max\{x, 0\}$. Moreover, for any $\varepsilon, \delta \in (0, 1)$ satisfying $n \geq 2M/(\varepsilon\delta)$, with probability at least $1 - \delta$,*

$$\inf_{\mathbf{a},\mathbf{b}} \|T_{\alpha,\beta}(Q) - A_{\mathrm{TSS}}(\mathbf{a}, \mathbf{b})\|_F^2 \geq \frac{(1 - \varepsilon)n}{8} (\alpha - \beta)^2 \tag{14}$$

*simultaneously for all $\alpha, \beta \in \mathbb{R}$.*

By Theorem 1, GeDRA represents $T_{\alpha,\beta}(Q)$ exactly for every $Q$. The expected approximation gap is therefore strictly positive when $\alpha \neq \beta$ and $n > 2M$. Proofs of both theorems are provided in Appendix C.

# 4 EXPERIMENTS

## 4.1 BASELINES

We compare GeDRA with full fine-tuning (Full FT) and PEFT baselines spanning low-rank adaptation (LoRA, DoRA, PiSSA, and DeLoRA), orthogonal adaptation (OFT, BOFT, qGOFT, and PSOFT), and random-matrix adaptation (VeRA, FlyLoRA, RandLoRA, and TeRA). The NLU comparison additionally includes $(\mathrm{IA})^3$ and SVDiff. Task-specific baselines and trainable parameter counts are reported in the corresponding results tables. For subject-driven generation, we qualitatively compare GeDRA with LoRA.

## 4.2 DATASETS

We evaluate GeDRA across four domains. Experimental settings and dataset details are provided in Appendices A and B.

**1. Commonsense Reasoning (CR).** Following LLM-Adapters (Hu et al., 2023), we fine-tune Meta-LLaMA3-8B (Grattafiori et al., 2024) on Commonsense-170K and evaluate on eight reasoning benchmarks, reporting per-task and mean accuracy.

**2. Image Classification (CLS).** We fine-tune an ImageNet-21K-pretrained ViT-B/16 (Dosovitskiy et al., 2020) on the 19 VTAB-1K tasks (Zhai et al., 2019), using 1,000 labeled examples per task, and report top-1 test accuracy.

**3. Natural Language Understanding (NLU).** We compare the performance of PEFT methods with small trainable parameter budgets by fine-tuning RoBERTa-base (Liu et al., 2019) on eight GLUE tasks (Wang et al., 2018). We report the mean of the standard task-specific validation scores.

**4. Subject-Driven Generation (SdG).** We fine-tune SDXL Base 1.0 (Podell et al., 2024) on the DreamBooth dataset (Ruiz et al., 2023) and qualitatively compare images generated using GeDRA and LoRA.

Table 1: Experimental results of fine-tuned LLaMA3-8B on commonsense reasoning benchmarks. Results are averaged over 5 random seeds.

| Methods | #Params | BoolQ | PIQA | SIQA | HS | WG | ARC-C | ARC-E | OBQA | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|
| Full FT | 8.03B | 73.90 | 86.20 | 79.10 | 93.10 | 85.80 | 78.20 | 88.10 | 84.00 | 83.55 |
| **Low-rank Adaptation** | | | | | | | | | | |
| LoRA | 28.3M | 69.68 | 83.38 | 78.15 | 89.04 | 83.51 | 72.19 | 84.60 | 81.00 | 80.19 |
| DoRA | 29.1M | 71.42 | 83.92 | 79.25 | 86.50 | 82.56 | 73.42 | 84.68 | 80.70 | 80.31 |
| PiSSA | 28.3M | 69.70 | 87.20 | 79.50 | 90.30 | 81.00 | 82.90 | 75.50 | 77.80 | 80.49 |
| DeLoRA | 28.3M | 72.19 | 87.35 | 78.00 | 92.88 | 83.58 | 78.97 | 90.72 | 82.40 | 83.26 |
| **Orthogonal Adaptation** | | | | | | | | | | |
| OFT | 15.2M | 72.54 | 85.81 | 79.97 | 91.90 | 87.17 | 76.42 | 84.78 | 82.80 | 82.67 |
| BOFT | 16.5M | 73.24 | 85.87 | 80.59 | 93.86 | 86.19 | 78.68 | 91.41 | 84.60 | 84.30 |
| PSOFT | 1.34M | 68.01 | 83.41 | 79.12 | 92.74 | 84.37 | 74.32 | 87.08 | 80.80 | 81.23 |
| **Random-matrix Adaptation** | | | | | | | | | | |
| RandLoRA | 112.7M | 67.52 | 85.20 | 78.10 | 91.89 | 84.29 | 77.90 | 88.09 | 81.80 | 81.85 |
| FlyLoRA | 25.2M | 70.43 | 83.57 | 80.91 | 92.28 | 86.11 | 75.60 | 86.66 | 82.22 | 82.22 |
| VeRA | 0.79M | 69.33 | 86.51 | 75.44 | 91.69 | 79.01 | 77.47 | 90.32 | 77.00 | 80.85 |
| TeRA | 0.99M | 70.46 | 86.56 | 76.46 | 92.26 | 81.37 | 78.84 | 90.32 | 79.20 | 81.93 |
| **GeDRA** | 1.57M | 72.60 | 89.01 | 81.37 | 95.50 | 86.82 | 81.57 | 92.97 | 85.60 | **85.68** |

### 4.3 Comparative Results

We evaluate whether independent modulation of local rotation and deformation supports effective adaptation across language, image classification, and subject-driven generation tasks.

#### 4.3.1 Commonsense Reasoning

Table 1 shows that GeDRA achieves the highest average accuracy of 85.68% with 1.57M trainable parameters. It exceeds BOFT, the strongest baseline, by 1.38 percentage points while using approximately 90.5% fewer trainable parameters, and improves over LoRA by 5.49 points. GeDRA also obtains the best results on five of the eight benchmarks. These results show that GeDRA achieves strong commonsense reasoning performance while learning only modulation vectors rather than matrix factors.

#### 4.3.2 Image Classification

On VTAB-1K, GeDRA achieves the best average accuracy among the compared PEFT methods, reaching 62.7% with 0.04M trainable parameters (Table 2). It outperforms DeLoRA by 2.1 percentage points and exceeds TeRA, which has the same reported parameter budget, by 8.9 points. GeDRA ranks first among PEFT methods on eight of the 19 tasks, although its average remains below full fine-tuning (66.1%). These results demonstrate a favorable accuracy–parameter trade-off through geometric modulation of frozen operators.

Table 2: Experimental results of fine-tuned ViT-B/16 on the VTAB-1K benchmark. Reported values (top-1 accuracy %) are the mean of 5 runs with different random seeds.

| | | **Natural** | | | | | | | **Specialized** | | | | **Structured** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Methods** | **#Params** | CIFAR100 | Caltech101 | DTD | Flowers102 | Pets | SVHN | SUN397 | Camelyon | EuroSAT | Resisc45 | Retinopathy | Clevr-Count | Clevr-Dist | DMLab | KITTI-Dist | dSpr-Loc | dSpr-Ori | sNORB-Azim | sNORB-Ele | **Avg.** |
| Full FT | 85.8M | 39.0 | 88.0 | 66.2 | 95.3 | 83.2 | 80.8 | 30.4 | 83.2 | 94.6 | 81.7 | 72.0 | 76.5 | 64.3 | 44.5 | 76.4 | 73.6 | 48.3 | 23.6 | 33.8 | 66.1 |
| **Low-rank Adaptation** | | | | | | | | | | | | | | | | | | | | | |
| LoRA | 0.59M | 35.5 | 79.5 | 50.8 | 95.3 | 82.4 | 37.4 | 25.2 | 80.6 | 83.5 | 51.2 | 73.6 | 50.0 | 52.1 | 38.9 | 70.5 | 44.0 | 23.6 | 10.1 | 24.9 | 53.1 |
| DoRA | 0.61M | 42.7 | 85.9 | 57.6 | 96.3 | 84.9 | 50.1 | 43.8 | 79.8 | 82.1 | 62.4 | 72.3 | 52.7 | 43.5 | 37.4 | 62.9 | 43.3 | 26.5 | 15.2 | 26.9 | 56.1 |
| PiSSA | 0.59M | 34.3 | 79.9 | 52.6 | 94.9 | 82.3 | 43.0 | 31.0 | 80.2 | 88.9 | 56.8 | 72.5 | 55.6 | 59.4 | 39.7 | 70.0 | 58.9 | 25.9 | 10.7 | 23.3 | 55.8 |
| DeLoRA | 0.59M | 34.4 | 76.9 | 56.2 | 86.4 | 75.7 | 76.1 | 26.5 | 81.2 | 93.3 | 74.3 | 72.6 | 63.3 | 55.6 | 42.0 | 75.4 | 67.7 | 43.8 | 20.7 | 29.7 | 60.6 |
| **Orthogonal Adaptation** | | | | | | | | | | | | | | | | | | | | | |
| OFT | 0.29M | 34.6 | 79.6 | 53.6 | 95.0 | 83.7 | 39.0 | 31.0 | 80.2 | 88.6 | 57.5 | 73.9 | 53.0 | 55.2 | 37.5 | 68.9 | 53.5 | 25.0 | 10.9 | 22.3 | 54.9 |
| BOFT | 0.31M | 34.4 | 80.6 | 56.2 | 94.7 | 83.6 | 51.3 | 31.5 | 79.6 | 87.3 | 61.7 | 72.7 | 55.3 | 53.6 | 37.8 | 69.3 | 51.2 | 29.1 | 12.4 | 24.7 | 56.1 |
| PSOFT | 0.03M | 43.0 | 85.1 | 58.6 | 95.7 | 84.4 | 51.5 | 43.2 | 79.2 | 84.8 | 63.4 | 72.2 | 47.0 | 48.4 | 37.3 | 69.1 | 48.6 | 28.5 | 14.2 | 23.3 | 56.7 |
| **Random-matrix Adaptation** | | | | | | | | | | | | | | | | | | | | | |
| RandLoRA | 2.38M | 42.7 | 86.2 | 59.6 | 96.2 | 84.4 | 49.1 | 45.9 | 80.2 | 82.0 | 62.9 | 72.3 | 48.3 | 41.8 | 36.8 | 57.9 | 39.9 | 27.6 | 16.0 | 25.4 | 55.5 |
| FlyLoRA | 0.59M | 45.7 | 83.0 | 61.3 | 96.9 | 87.4 | 37.2 | 41.8 | 77.9 | 85.8 | 65.1 | 74.2 | 43.8 | 44.9 | 36.9 | 63.6 | 33.0 | 22.9 | 11.4 | 22.5 | 54.5 |
| VeRA | 0.02M | 46.6 | 86.5 | 64.0 | 97.2 | 87.2 | 35.4 | 50.4 | 79.1 | 86.3 | 67.5 | 73.9 | 33.2 | 30.9 | 35.6 | 59.1 | 18.1 | 19.4 | 12.9 | 21.9 | 52.9 |
| TeRA | 0.04M | 48.2 | 86.7 | 64.3 | 97.3 | 87.4 | 36.3 | 50.2 | 79.3 | 86.8 | 69.1 | 74.0 | 34.6 | 32.0 | 36.0 | 60.5 | 22.5 | 21.4 | 13.2 | 22.3 | 53.8 |
| **GeDRA** | 0.04M | 54.4 | 89.7 | 67.3 | 98.3 | 89.6 | 59.0 | 52.2 | 82.6 | 92.9 | 78.1 | 72.2 | 51.9 | 48.1 | 40.0 | 71.6 | 57.2 | 41.6 | 18.0 | 25.9 | **62.7** |

#### 4.3.3 Natural Language Understanding

To assess adaptation performance under very small parameter budgets, we compare GeDRA with PEFT baselines using only 0.02M–0.04M trainable parameters on GLUE. As shown in Table 3, GeDRA achieves an average score of 83.96 with 0.04M trainable parameters, outperforming the strongest PEFT baseline, PSOFT, by 1.12 points. It obtains the best or tied-best PEFT scores across all

Table 3: Experimental results of fine-tuned RoBERTa-base on the GLUE benchmark. Results are averaged over 5 random seeds.

| Methods | #Params | CoLA | SST-2 | MRPC | QQP | STS-B | MNLI | QNLI | RTE | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|
| Full FT | 124.65M | 61.35 | 94.27 | 91.83 | 90.07 | 90.91 | 87.52 | 93.04 | 76.17 | 85.65 |
| $(IA)^3$ | 0.02M | 53.19 | 94.04 | 85.96 | 84.16 | 88.94 | 82.75 | 87.83 | 62.09 | 79.87 |
| SVDiff | 0.02M | 56.85 | 93.12 | 86.98 | 83.75 | 89.02 | 82.66 | 87.77 | 59.57 | 79.97 |
| TeRA | 0.02M | 47.49 | 93.00 | 83.34 | 83.17 | 86.89 | 81.39 | 85.14 | 58.48 | 77.36 |
| PSOFT | 0.03M | 59.78 | 93.69 | 88.13 | 85.24 | 89.55 | 84.47 | 90.02 | 71.84 | 82.84 |
| qGOFT | 0.04M | 60.09 | 93.46 | 89.24 | 85.51 | 89.75 | 83.85 | 89.68 | 66.79 | 82.30 |
| VeRA | 0.04M | 52.08 | 93.35 | 86.18 | 83.50 | 89.22 | 83.00 | 86.88 | 58.48 | 79.09 |
| **GeDRA** | 0.04M | 62.88 | 94.27 | 89.72 | 85.51 | 89.94 | 84.53 | 90.85 | 74.01 | **83.96** |

eight tasks and exceeds qGOFT and VeRA by 1.66 and 4.87 points, respectively, at the same reported parameter budget. These results support the effectiveness of independent geometric modulation even when very few task-specific parameters are learned.

#### 4.3.4 Subject-Driven Generation

Figure 3 compares GeDRA and LoRA on subject-driven generation. In the displayed examples, GeDRA better balances subject fidelity and prompt adherence: it retains the vase's red appearance, renders the robot and plush bear in the requested purple, and places the duck toy on snow. LoRA misses some of these subject details or prompt constraints. These examples illustrate that GeDRA's compact geometric parameterization can accommodate prompt-specific changes while retaining recognizable subject characteristics, extending its applicability beyond discriminative tasks.

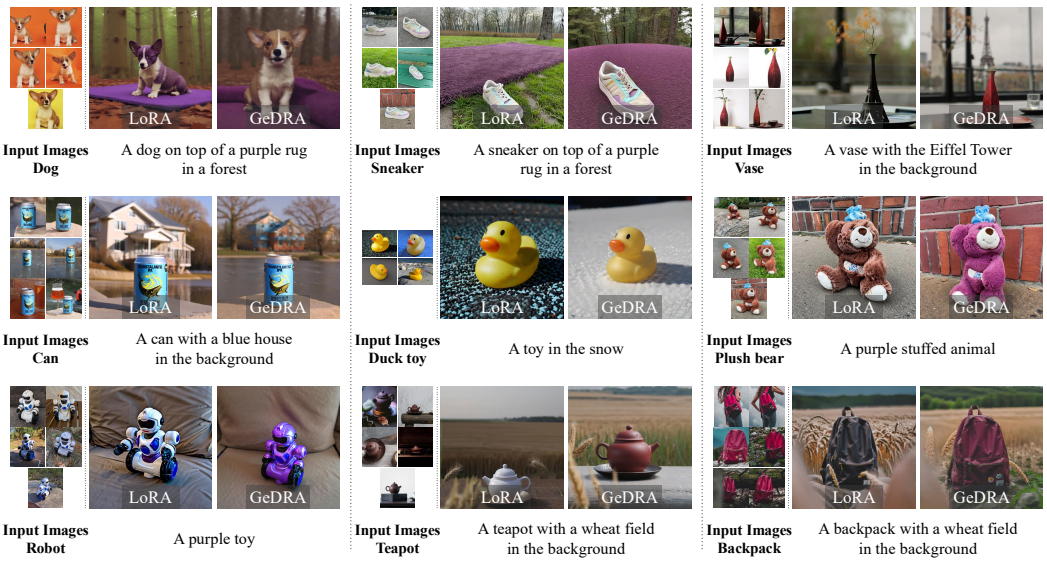



Figure 3: Qualitative comparison of LoRA (left) and GeDRA (right) for subject-driven generation. In the examples shown, GeDRA better preserves subject characteristics while following the text prompts.

### 4.4 Ablation Study

**Geometric components.** We evaluate rotation-only, deformation-only, and joint configurations on VTAB-1K with $M = 2$. As shown in Figure 4(a-i), the two single-component variants achieve

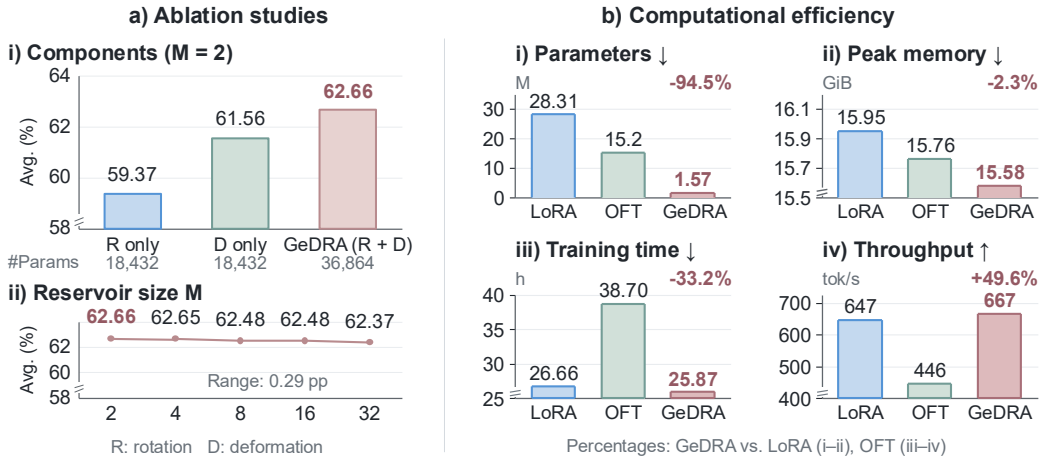



Figure 4: **Ablation studies and computational efficiency. a)** Component and reservoir-size ablations using ViT-B/16 on VTAB-1K. **b)** Trainable parameters, peak memory, training time, and throughput for fine-tuning LLaMA3-8B. Percentage changes are relative to LoRA for parameter count and peak memory, and to OFT for training time and throughput.

59.37% and 61.56% accuracy, respectively, each using 18,432 trainable parameters. Deformation therefore provides the stronger single-component configuration at the same parameter budget. Jointly learning both modulation vectors achieves 62.66% with 36,864 trainable parameters, improving over rotation-only and deformation-only adaptation by 3.29 and 1.10 percentage points, respectively. Complementing this comparison, Figure 1 shows that GeDRA exhibits broad directional support for both rotation and deformation, qualitatively resembling full fine-tuning.

**Reservoir size.** We vary $M \in \{2, 4, 8, 16, 32\}$ while keeping the trainable parameter count fixed at 36,864. As shown in Figure 4(a-ii), average accuracy ranges from 62.37% to 62.66%, a difference of 0.29 percentage points. The highest observed accuracy occurs at $M = 2$, with no improvement from larger reservoirs in this sweep. These results indicate limited sensitivity to reservoir size in the tested configuration and support the choice of $M = 2$ for VTAB-1K.

### 4.5 Efficiency Analysis

Figure 4(b) compares LoRA, OFT, and GeDRA under the same profiling setup. Compared with LoRA, GeDRA reduces trainable parameters from 28.31M to 1.57M (94.5%) and peak memory from 15.95 to 15.58 GiB (2.3%). Compared with OFT, it reduces training time from 38.70 to 25.87 hours (33.2%) and increases throughput from 446 to 667 tokens/s (49.6%). Among the three evaluated methods, GeDRA achieves the lowest trainable parameter count, peak memory, and training time, together with the highest throughput. These results demonstrate favorable training efficiency alongside a compact parameterization.

## 5 Conclusion

We propose GeDRA, which independently modulates rotation- and deformation-generator templates from a frozen operator reservoir. This structure-preserving design supports high-rank updates with only two trainable vectors per adapted weight matrix. For independently reweighted geometric templates, our analysis establishes conditions for an approximation advantage over two-sided scaling of the same reservoir under matched nominal parameter budgets. Experiments demonstrate favorable performance–parameter trade-offs in commonsense reasoning, image classification, and natural language understanding, complemented by qualitative comparisons with LoRA for subject-driven generation. These findings show that a limited trainable parameter budget need not impose low-rank updates or strictly geometry-preserving transformations.

### AI USE STATEMENT

No generative AI tools were used in conducting the research or preparing the manuscript. The authors reviewed this statement and take full responsibility for the content of the paper.

## REFERENCES

Paul Albert, Frederic Z. Zhang, Hemanth Saratchandran, Cristian Rodriguez-Opazo, Anton van den Hengel, and Ehsan Abbasnejad. Randlora: Full rank parameter-efficient fine-tuning of large models. In *International Conference on Learning Representations*, volume 2025, pp. 22258–22270, 2025.

Massimo Bini, Leander Girrbach, and Zeynep Akata. Decoupling angles and strength in low-rank adaptation. In *International Conference on Learning Representations*, volume 2025, pp. 20216–20233, 2025.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Jonathan Dong, Ruben Ohana, Mushegh Rafayelyan, and Florent Krzakala. Reservoir computing meets recurrent kernels and structured transforms. *Advances in Neural Information Processing Systems*, 33:16785–16796, 2020.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yuxuan Gu, Wuyang Zhou, Giorgos Iacovides, and Danilo Mandic. Tera: Vector-based random tensor network for high-rank adaptation of large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2314–2329, 2026.

Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7289–7300. IEEE, 2023.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. Lora+: Efficient low rank adaptation of large models. *arXiv preprint arXiv:2402.12354*, 2024.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 5254–5276, 2023.

Herbert Jaeger. Echo state network. *scholarpedia*, 2(9):2330, 2007.

Dawid Kopiczko, Tijmen Blankevoort, and Yuki Asano. Vera: Vector-based random matrix adaptation. In *International Conference on Learning Representations*, volume 2024, pp. 6815–6835, 2024.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: Long papers)*, pp. 4582–4597, 2021.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in neural information processing systems*, 35:1950–1965, 2022.

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*, 2024a.

Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, et al. Parameter-efficient orthogonal finetuning via butterfly factorization. In *International Conference on Learning Representations*, volume 2024, pp. 38317–38350, 2024b.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Mantas Lukoševičius. A practical guide to applying echo state networks. In *Neural Networks: Tricks of the Trade: Second Edition*, pp. 659–686. Springer, 2012.

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.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: Principal singular values and singular vectors adaptation of large language models. *Advances in Neural Information Processing Systems*, 37: 121038–121072, 2024.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations*, volume 2024, pp. 1862–1874, 2024.

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.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22500–22510. IEEE, 2023.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pp. 353–355, 2018.

Fei Wu, Jia Hu, Geyong Min, and Shiqiang Wang. Efficient orthogonal fine-tuning with principal subspace adaptation. In *International Conference on Learning Representations*, volume 2026, pp. 25798–25826, 2026.

Yuchen Zeng and Kangwook Lee. The expressive power of low-rank adaptation. In *International Conference on Learning Representations*, volume 2024, pp. 5078–5123, 2024.

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.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*, 2023.

Heming Zou, Yunliang Zang, Wutong Xu, Yao Zhu, and Xiangyang Ji. Flylora: Boosting task decoupling and parameter efficiency via implicit rank-wise mixture-of-experts. *Advances in Neural Information Processing Systems*, 38:10386–10419, 2025.

# A Experimental Settings

**Implementation details.** All experiments are conducted on a workstation equipped with eight NVIDIA GeForce RTX 4090 GPUs, each with 24 GB of memory. We use 16-bit mixed-precision training throughout.

**Comparison protocol.** Within each task, methods share the pretrained backbone, data splits, and evaluation protocol. Baseline-specific hyperparameters follow the original papers or are selected using validation data under the shared task settings. For CR, configuration selection is separated from the public evaluation splits used by LLM-Adapters. For CLS, configurations are selected on the standard validation split, with the test split reserved for final evaluation. GLUE results are reported on the official validation splits.

**Commonsense reasoning.** We fine-tune LLaMA3-8B on Commonsense-170K for three epochs using AdamW, zero weight decay, and a linear learning-rate schedule with 100 warmup steps. PEFT methods target `q_proj`, `k_proj`, `v_proj`, `up_proj`, and `down_proj` in every transformer layer. We use a maximum sequence length of 256, a micro-batch size of 1, and 16 gradient accumulation steps, with a global effective batch size of 16. BF16 training and gradient checkpointing are enabled. GeDRA uses $M = 4$ and a learning rate of $6 \times 10^{-4}$. Evaluation uses beam search with four beams and at most 32 newly generated tokens.

**Image classification.** We fine-tune ViT-B/16 on VTAB-1K for 100 epochs using AdamW, a batch size of 384, and a cosine learning-rate schedule with five warmup epochs. PEFT methods target the query and value projections in all 12 attention layers, totaling 24 modules. GeDRA uses the identity-augmented reservoir with $M = 2$ and a learning rate of $2 \times 10^{-2}$.

**Natural language understanding.** We fine-tune RoBERTa-base on GLUE with a batch size of 32, targeting the query and value projections in all 12 layers (24 modules). We use AdamW with linear warmup followed by linear learning-rate decay, and clip the gradient norm at 1.0. Bias and LayerNorm parameters are exempt from weight decay. GeDRA uses $M = 2$. Learning rates are shared across methods within each task: $8 \times 10^{-4}$ for CoLA, SST-2, and QQP; $10^{-3}$ for MRPC; $2.2 \times 10^{-3}$ for STS-B; and $5 \times 10^{-4}$ for MNLI, QNLI, and RTE.

**Subject-driven generation.** We train only the adaptation parameters in the UNet of SDXL Base 1.0, keeping the text encoders frozen. Adaptation targets `to_q`, `to_k`, `to_v`, and `to_out.0`, totaling 560 projection modules. Both GeDRA and LoRA are trained for 500 steps at $1024 \times 1024$ resolution, with a micro-batch size of 1 and four gradient accumulation steps. We use standard AdamW with weight decay $10^{-4}$, gradient-norm clipping at 1.0, and a constant learning rate without warmup. GeDRA uses $M = 4$ and a learning rate of $3 \times 10^{-2}$.

# B Datasets and Evaluation Protocols

This appendix details the dataset composition, evaluation splits, and score aggregation used in the four evaluation domains. The evaluation covers eight commonsense reasoning benchmarks, 19 image classification tasks, eight natural language understanding tasks, and 30 subjects for personalized image generation.

## B.1 Commonsense Reasoning

We fine-tune Meta-LLaMA3-8B (Grattafiori et al., 2024) on Commonsense-170K following the setup of LLM-Adapters (Hu et al., 2023). Evaluation covers BoolQ, PIQA, Social IQa, HellaSwag,

WinoGrande, ARC-Challenge, ARC-Easy, and OpenBookQA. These benchmarks assess complementary reasoning capabilities: BoolQ concerns binary question answering; PIQA and Social IQa address physical and social commonsense; HellaSwag evaluates plausible continuation selection; WinoGrande tests commonsense coreference resolution; and the ARC subsets and OpenBookQA involve science-oriented multiple-choice reasoning.

We evaluate on the public evaluation splits used by the benchmark protocol and report accuracy for each dataset. The overall CR score is the unweighted mean of the eight accuracies, with ARC-Challenge and ARC-Easy treated as separate benchmarks.

### B.2 Image Classification

We use ViT-B/16 pretrained on ImageNet-21K (Dosovitskiy et al., 2020) and evaluate on all 19 tasks of VTAB-1K (Zhai et al., 2019). The tasks are organized into three groups:

- **Natural (7 tasks):** CIFAR-100, Caltech101, DTD, Flowers102, Oxford-IIIT Pets, SVHN, and SUN397.
- **Specialized (4 tasks):** Patch Camelyon, EuroSAT, RESISC45, and Diabetic Retinopathy.
- **Structured (8 tasks):** CLEVR-Count, CLEVR-Distance, DMLab, KITTI-Distance, dSprites-Location, dSprites-Orientation, SmallNORB-Azimuth, and SmallNORB-Elevation.

Each task provides 1,000 labeled examples. When selecting configurations, we train on the canonical `train800` split and validate on `val200`. After fixing the configuration, we retrain on the combined `train800val200` split and evaluate on the official test split. Test data are not used for configuration selection. We report top-1 accuracy for each task and the unweighted mean across all 19 tasks, rather than averaging the three group means.

### B.3 Natural Language Understanding

We fine-tune RoBERTa-base (Liu et al., 2019) on eight GLUE tasks (Wang et al., 2018): CoLA, SST-2, MRPC, QQP, STS-B, MNLI, QNLI, and RTE. These tasks cover linguistic acceptability, sentiment classification, paraphrase detection, semantic textual similarity, and natural language inference. The PEFT comparison focuses on methods with very small trainable parameter budgets; the parameter count of each method is reported alongside its results.

We use the official training splits for fine-tuning and report results on the official validation splits. Table 4 specifies the score used for each task. For tasks with multiple metrics or validation subsets, we first average the corresponding scores to obtain one task-level score. The overall NLU score is the unweighted mean of the eight task-level scores, with MNLI counted once.

Table 4: Task-level metrics used for GLUE evaluation. MNLI-m and MNLI-mm denote the matched and mismatched validation sets, respectively.

| Task | Task-level score |
|---|---|
| CoLA | Matthews correlation |
| SST-2 | Accuracy |
| MRPC | Mean of accuracy and F1 |
| QQP | Mean of accuracy and F1 |
| STS-B | Mean of Pearson and Spearman correlations |
| MNLI | Mean of MNLI-m and MNLI-mm accuracies |
| QNLI | Accuracy |
| RTE | Accuracy |

### B.4 Subject-Driven Generation

We compare GeDRA with LoRA by fine-tuning SDXL Base 1.0 (Podell et al., 2024) on the DreamBooth dataset (Ruiz et al., 2023). The dataset contains 30 subjects, including nine live subjects and 21 objects, with four to six reference images per subject.

For each subject, we use the 25 official evaluation prompts and generate four images per prompt, yielding 3,000 images per method. Random seeds are paired between GeDRA and LoRA for each subject–prompt pair. We present qualitative comparisons of the generated images alongside the corresponding reference images and text prompts.

# C Proofs of the Theoretical Results

## C.1 Proof of Theorem 1

*Proof.* Set

$$\mathbf{r} = \alpha \mathbf{1}_n, \qquad \mathbf{d} = \beta \mathbf{1}_n. \tag{15}$$

Then $\mathrm{Diag}(\mathbf{r}) = \alpha I_n$ and $\mathrm{Diag}(\mathbf{d}) = \beta I_n$. Therefore,

$$\begin{aligned} A_{\mathrm{GeDRA}}(\mathbf{r}, \mathbf{d}) &= \frac{\alpha I_n R + R \alpha I_n}{2} + \frac{\beta I_n D + D \beta I_n}{2} \\ &= \alpha R + \beta D \\ &= T_{\alpha,\beta}(Q). \end{aligned} \tag{16}$$

Since squared Frobenius errors are nonnegative, this proves Equation (11).

Now fix $i \neq j$ such that $Q_{ji} = 0$. By the definitions of $R$ and $D$,

$$\begin{aligned} [T_{\alpha,\beta}(Q)]_{ji} &= \alpha \frac{Q_{ji} - Q_{ij}}{2} + \beta \frac{Q_{ji} + Q_{ij}}{2} \\ &= \frac{\beta - \alpha}{2} Q_{ij}. \end{aligned} \tag{17}$$

In contrast, for every $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$,

$$[A_{\mathrm{TSS}}(\mathbf{a}, \mathbf{b})]_{ji} = a_j Q_{ji} b_i = 0. \tag{18}$$

Retaining only these entries in the squared Frobenius norm gives

$$\begin{aligned} &\|T_{\alpha,\beta}(Q) - A_{\mathrm{TSS}}(\mathbf{a}, \mathbf{b})\|_F^2 \\ &\geq \sum_{\substack{i \neq j \\ Q_{ji} = 0}} \left| \frac{\beta - \alpha}{2} Q_{ij} \right|^2 \\ &= \frac{(\alpha - \beta)^2}{4} \Gamma(Q). \end{aligned} \tag{19}$$

Taking the infimum over $\mathbf{a}$ and $\mathbf{b}$ proves Equation (12). Together with the exact GeDRA representation, this yields strict separation whenever $\alpha \neq \beta$ and $\Gamma(Q) > 0$.

The same argument applies to any $X \in \mathbb{R}^{n \times n}$ satisfying $X_{ij} = 0$ whenever $Q_{ij} = 0$. Indeed, every entry used in Equation (19) remains zero in $X$. □

## C.2 Proof of Theorem 2

*Proof.* For each $\ell$, let $\pi_\ell$ denote the permutation represented by $P_\ell$, with the convention

$$(P_\ell)_{i,\pi_\ell(i)} = 1. \tag{20}$$

The $M$ permutations generate $Mn$ candidate directed entries

$$\big(i, \pi_\ell(i)\big), \qquad \ell \in \{1, \ldots, M\}, \quad i \in \{1, \ldots, n\}. \tag{21}$$

For a candidate $(\ell, i)$, write $j = \pi_\ell(i)$. We call it clean if $j \neq i$, $\pi_h(i) \neq j$ for every $h \neq \ell$, and $\pi_h(j) \neq i$ for every $h \in \{1, \ldots, M\}$. A clean candidate is the only contribution from the sampled permutations to the directed entry $(i, j)$, and no permutation contributes to the reverse entry $(j, i)$. Since the identity matrix has no off-diagonal entries,

$$Q_{ij} = \frac{\xi_{\ell,i}}{\sqrt{2M}}, \qquad Q_{ji} = 0. \tag{22}$$

Thus, each clean candidate contributes exactly $1/(2M)$ to $\Gamma(Q)$. Distinct clean candidates correspond to distinct matrix entries.

Let $B$ be the number of candidates that are not clean. For a fixed candidate $(\ell, i)$, non-cleanliness can occur through one of the following events:

1. $\pi_\ell(i) = i$;
2. $\pi_h(i) = \pi_\ell(i)$ for some $h \neq \ell$;
3. $\pi_\ell(\pi_\ell(i)) = i$ with $\pi_\ell(i) \neq i$;
4. $\pi_h(\pi_\ell(i)) = i$ for some $h \neq \ell$, with $\pi_\ell(i) \neq i$.

The first event has probability $1/n$. For each $h \neq \ell$, a same-direction collision in the second event has probability $1/n$. The third event means that $i$ belongs to a two-cycle of $\pi_\ell$, which also has probability $1/n$. For each $h \neq \ell$, independence of $\pi_h$ and $\pi_\ell$ gives

$$\mathbb{P}(\pi_\ell(i) \neq i,\ \pi_h(\pi_\ell(i)) = i) = \frac{n-1}{n^2} \leq \frac{1}{n}. \tag{23}$$

A union bound therefore yields

$$\begin{aligned} \mathbb{P}((\ell, i) \text{ is not clean}) &\leq \frac{1}{n} + \frac{M-1}{n} + \frac{1}{n} + \frac{M-1}{n} \\ &= \frac{2M}{n}. \end{aligned} \tag{24}$$

By linearity of expectation,

$$\mathbb{E}[B] \leq Mn\frac{2M}{n} = 2M^2. \tag{25}$$

There are $Mn - B$ clean candidates. Consequently, for every realization of the reservoir construction, Equation (22) implies

$$\Gamma(Q) \geq \frac{Mn - B}{2M} = \frac{n}{2} - \frac{B}{2M}. \tag{26}$$

Taking expectations and using Equation (25) gives

$$\mathbb{E}_Q[\Gamma(Q)] \geq \frac{n}{2} - \frac{\mathbb{E}[B]}{2M} \geq \frac{n}{2} - M. \tag{27}$$

Since $\Gamma(Q) \geq 0$, we obtain

$$\mathbb{E}_Q[\Gamma(Q)] \geq \left(\frac{n}{2} - M\right)_+, \tag{28}$$

where $(x)_+ = \max\{x, 0\}$.

For fixed $\alpha, \beta \in \mathbb{R}$, applying Theorem 1 to each realization of $Q$ and taking expectations yields

$$\begin{aligned} &\mathbb{E}_Q\left[\inf_{\mathbf{a},\mathbf{b}\in\mathbb{R}^n} \|T_{\alpha,\beta}(Q) - A_{\mathrm{TSS}}(\mathbf{a}, \mathbf{b})\|_F^2\right] \\ &\quad\geq \frac{(\alpha-\beta)^2}{4}\mathbb{E}_Q[\Gamma(Q)] \\ &\quad\geq \frac{(\alpha-\beta)^2}{4}\left(\frac{n}{2} - M\right)_+. \end{aligned} \tag{29}$$

This proves Equation (13). By Theorem 1, GeDRA attains zero approximation error for every realization of $Q$.

It remains to establish the high-probability statement. Fix $\varepsilon, \delta \in (0, 1)$ and suppose that $n \geq 2M/(\varepsilon\delta)$. By Equation (26), the event

$$\Gamma(Q) < \frac{(1-\varepsilon)n}{2} \tag{30}$$

implies $B > \varepsilon Mn$. Hence, Markov's inequality gives

$$\begin{aligned}\mathbb{P}\left(\Gamma(Q) < \frac{(1-\varepsilon)n}{2}\right) &\le \mathbb{P}(B > \varepsilon Mn) \\ &\le \frac{\mathbb{E}[B]}{\varepsilon Mn} \le \frac{2M}{\varepsilon n} \le \delta.\end{aligned} \tag{31}$$

Therefore, with probability at least $1-\delta$,

$$\Gamma(Q) \ge \frac{(1-\varepsilon)n}{2}. \tag{32}$$

On this event, Theorem 1 implies

$$\begin{aligned}&\inf_{\mathbf{a},\mathbf{b}\in\mathbb{R}^n} \|T_{\alpha,\beta}(Q) - A_{\mathrm{TSS}}(\mathbf{a},\mathbf{b})\|_F^2 \\ &\quad\ge \frac{(\alpha-\beta)^2}{4}\Gamma(Q) \\ &\quad\ge \frac{(1-\varepsilon)n}{8}(\alpha-\beta)^2.\end{aligned} \tag{33}$$

This proves Equation (14). Because the event in Equation (32) depends only on $Q$, the bound holds simultaneously for all $\alpha, \beta \in \mathbb{R}$. □

## D Geometric Analysis

We analyze the query and value projection weight updates in all 12 layers of ViT-B/16 across the 19 VTAB-1K tasks, yielding 456 task–layer–projection matrices per method.

**Mode-wise geometric attribution.** For pretrained and adapted weights $W_0, W_\theta \in \mathbb{R}^{n\times n}$, let $\Delta W = W_\theta - W_0$. We compute the singular value decompositions

$$W_0 = U\Sigma V^\top, \qquad \Delta W = \sum_{k=1}^{n} \Delta W_k, \qquad \Delta W_k = s_k a_k b_k^\top. \tag{34}$$

In the fixed pretrained singular frame, define

$$P_{\mathrm{def}}(X) = U\,\mathrm{Diag}\big(\mathrm{diag}(U^\top X V)\big)\,V^\top, \tag{35}$$

where diag extracts the diagonal and Diag constructs a diagonal matrix. For simple positive singular values, the extracted entries describe their first-order variations. The complementary component is used as a diagnostic of singular-frame reorientation, termed rotation in Figure 1.

**Spectral-change detection.** We determine whether endpoint spectral change is detectable using

$$\tau_{\mathrm{spec}} = 64\epsilon_{\mathrm{fp32}}\sqrt{n}\,\max\{\|W_0\|_F, \|W_\theta\|_F\}, \tag{36}$$

$$\chi_{\mathrm{spec}} = \mathbf{1}[\|\sigma(W_\theta) - \sigma(W_0)\|_2 > \tau_{\mathrm{spec}}], \tag{37}$$

where $\sigma(W)$ lists singular values in descending order and $\epsilon_{\mathrm{fp32}} = 2^{-23}$. The assigned mode-wise energies are

$$e_k^{\mathrm{def}} = \chi_{\mathrm{spec}}\|P_{\mathrm{def}}(\Delta W_k)\|_F^2, \qquad e_k^{\mathrm{rot}} = \|\Delta W_k\|_F^2 - e_k^{\mathrm{def}}. \tag{38}$$

Thus, when no endpoint spectral change is detected, all update-mode energy is assigned to reorientation. The endpoint test determines detectability, whereas the projection determines the mode-wise spectral attribution.

**Directional support.** For each component $c \in \{\mathrm{rot}, \mathrm{def}\}$, let $E_c = \sum_{k=1}^n e_k^c$ and independently sort its energies as $e_{(1)}^c \ge \cdots \ge e_{(n)}^c$. For $E_c > 0$, define

$$N_c = \min\left\{ q \in \{1,\dots,n\} : \sum_{j=1}^{q} e_{(j)}^c \ge 0.99E_c \right\}. \tag{39}$$

We set $N_c = 0$ when $E_c = 0$. Rotation and deformation may therefore select different subsets of update singular modes. These counts characterize mode-wise energy concentration, rather than the ranks of separate geometric components. For repeated nonzero singular values, mode-wise attribution can depend on the chosen singular basis.