LoRA Scope: Semantic Change Probing for LoRA Search and Evaluation

Anonymous authors
Paper under double-blind review

Abstract

The rapid growth of LoRA adapters for text-to-image diffusion models has made them difficult to understand, evaluate, and search beyond thumbnails and noisy metadata. We introduce LoRA Scope, a framework that represents each LoRA by what it changes: the CLIP-space difference between matched base and LoRA-adapted generations. The framework connects three complementary components through diagnostic prompts and generated images. LoUPE uses VLM-guided prompt exploration to construct adapter-specific diagnostic prompts; the LoUPE Score measures the stability, precision, and strength of the induced changes and aligns with human judgments; and LoCAL aligns LoRA weight representations with CLIP embeddings of these prompts and LoRA-generated images through learned query adapters, enabling text- and image-to-LoRA retrieval without query-time generation. On a large corpus of real-world LoRAs, LoRA Scope provides interpretable adapter characterization, human-aligned quality assessment, and scalable metadata-free retrieval directly from LoRA weights.

1 Introduction

Low-rank adaptation (LoRA) makes it easy to personalize large generative models without retraining all of their parameters (Hu et al., 2022). In text-to-image diffusion, this has created a rapidly growing ecosystem of community LoRAs for characters, artistic styles, objects, and visual concepts. Yet the infrastructure for using this ecosystem has not kept pace. Users seeking a particular visual effect often browse thumbnails, titles, tags, and short creator descriptions, even though these artifacts are incomplete proxies for adapter behavior. The core question remains under-specified: *what does this LoRA actually change about generation, and under which prompts does that change reliably appear?*

Existing personalization and retrieval tools do not fully answer this question. Adapter-selection systems such as Stylus retrieve or compose diffusion adapters from textual descriptions and embeddings (Luo et al., 2024), and more recent work such as CARLoS characterizes LoRAs through generated-image behavior for retrieval (Sarfaty et al., 2025). These directions are highly relevant, but they leave three gaps for open-world LoRA libraries: metadata can be noisy and may reflect creator intent rather than realized behavior; image-level metrics such as CLIPScore (Hessel et al., 2021) and human-preference scores (Ma et al., 2025) evaluate generated images rather than adapters; and retrieval systems often depend on metadata, example images, or query-time generation rather than learning a searchable representation directly from adapter weights.

Our starting point is intuitive: *a LoRA is best characterized by its intervention*. Given the same prompt and sampling conditions, we compare the image generated by the base model with the image generated by the LoRA-adapted model, and measure their displacement in CLIP image-embedding space (Radford et al., 2021). While a single displacement captures the adapter's effect under one prompt, a distribution of such *difference vectors* over diagnostic prompts provides a behavior-grounded fingerprint. This representation focuses on the semantic shifts the adapter consistently induces relative to the base model as opposed to the generated images.

We introduce LoRA Scope (Fig. 1), a framework for LoRA characterization, evaluation, and retrieval built around this intervention signal. First, LoUPE constructs adapter-specific diagnostic prompt sets: it populates prompts with language-model guidance, generates matched base and LoRA images, and stops when the induced difference-vector distribution becomes stable. A VLM-guided refinement stage then replaces weak prompts with visually grounded prompts that better activate the adapter.

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.

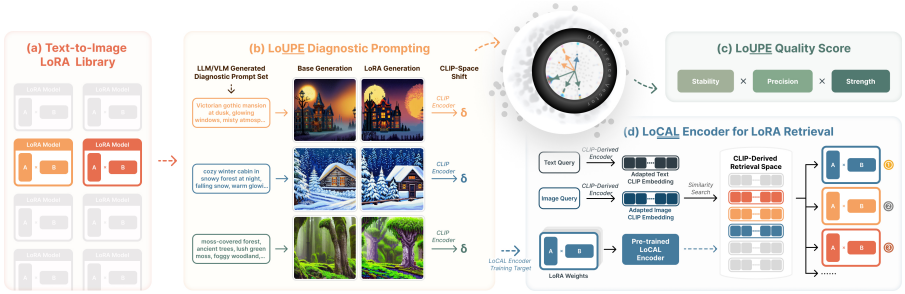

Figure 1: Overview of LoRA Scope. LoUPE characterizes LoRAs through CLIP-space semantic changes and yields the LoUPE Score. LoCAL aligns LoRA weight representations with CLIP embeddings of diagnostic prompts and LoRA-generated images through learned query adapters for retrieval.

Second, the LoUPE Score evaluates adapter quality through the strength, stability, and precision of the induced changes, targeting whether the LoRA has learned a coherent intervention rather than whether individual images score well under generic metrics. Third, LoCAL learns a weight-to-CLIP encoder, adapting recent weight-space learning ideas for LoRA weights (Putterman et al., 2025; Han et al., 2026), so text and image queries can retrieve LoRAs from their weights without query-time diffusion generation.

We evaluate LoRA Scope on a large real-world LoRA corpus drawn from ImageGem (Guo et al., 2025). Experiments show that LoUPE constructs stable diagnostic prompt sets, that VLM-guided refinement strengthens weak prompts, and that the LoUPE Score aligns with human adapter-quality judgments better than generic image-level metrics such as CLIPScore (Hessel et al., 2021), aesthetic score (Schuhmann et al., 2022), and HPSv3 (Ma et al., 2025). For retrieval, LoCAL indexes LoRAs directly from weights and remains competitive with metadata-based Stylus variants, especially for image-to-LoRA search, despite using no metadata at inference time.

In summary, our contributions are fourfold: (i) a difference-vector distribution formalism for characterizing LoRA behavior in CLIP space; (ii) a two-stage LLM/VLM pipeline for automated, convergence-guided diagnostic prompt set construction; (iii) a LoRA quality score measuring the effectiveness and consistency of adapter behavior, validated against human ratings; and (iv) a weight-to-CLIP representation learning framework with contrastive alignment that enables scalable text/image-to-LoRA retrieval from real-world LoRA weights.

2 Related Work

Adapter Selection and Retrieval. Language-side adapter-bank work studies supervised reuse, routing, and retrieve-then-compose pipelines for task adapters or LoRA modules (Pfeiffer et al., 2021; Huang et al., 2024; Zhao et al., 2024). Recent visual systems move closer to open-world diffusion adapters: Stylus retrieves and composes adapters from refined descriptions and prompt keywords (Luo et al., 2024), SemLA indexes LoRA libraries with CLIP for test-time domain adaptation (Qorbani et al., 2025), LoRAverse selects diverse adapter sets with a submodular objective (Sonmezer et al., 2025), and AutoLoRA learns a weight-text retrieval space with gated fusion (Li et al., 2025). Closest to our setting, CARLoS builds a metadata-free LoRA descriptor from CLIP differences between base and LoRA generations for retrieval (Sarfaty et al., 2025). Our problem is similarly retrieval-first but targets scalable text- and image-LoRA search over noisy community libraries, using behavior-grounded diagnostic prompts and a weight-to-CLIP representation learning framework before any downstream composition.

Learning Representation of Model Weights. Recent work has shown that fine-tuned checkpoints contain structure beyond their end-task behavior. Model soups average compatible checkpoints (Wortsman et al., 2022), task arithmetic treats updates as transferable editing vectors (Ilharco et al., 2023), and Git Re-Basin aligns independently trained networks for merging (Ainsworth et al., 2023). For personalizing diffusion models, weights2weights and ImageGem learn latent spaces over LoRA weights for generation or editing (Dravid et al., 2024; Guo et al., 2025). Meta-learning methods generate LoRA or compact adapter weights from image or task conditions (Ruiz et al., 2024; Wu et al., 2024;

Smith et al., 2024; Jin et al., 2024), while Learning on LoRAs introduces GL-equivariant architectures for low-rank weight spaces (Putterman et al., 2025). W2T canonicalizes LoRA factors via QR decomposition and SVD, then processes the resulting components with a Transformer for attribute classification, performance prediction, and adapter retrieval (Han et al., 2026). Our focus is text- and image-to-LoRA search over community diffusion adapters, using diagnostic prompts and generated images to align weight representations with CLIP semantics.

Cross-Modal Alignment to CLIP Space. Shared CLIP-style image-text spaces are a standard substrate for cross-modal retrieval and for many scoring functions in generative modeling (Radford et al., 2021; Zhai et al., 2022). A related line treats this semantic space as an anchor for additional modalities: AudioCLIP adds an audio encoder (Guzhov et al., 2022), MotionCLIP aligns human-motion latents to CLIP text and rendered-image embeddings (Tevet et al., 2022), ULIP aligns point-cloud encoders with image-language representations for 3D understanding (Xue et al., 2023), and ImageBind learns a shared embedding across image, text, audio, depth, thermal, and IMU data (Girdhar et al., 2023). Our target is different from these sensory modalities: we align representations of trained LoRA weights with CLIP embeddings of diagnostic prompts and LoRA-generated images through learned text and image query adapters, so text and image queries can retrieve reusable adaptation modules without test-time generation.

3 METHOD

We characterize each LoRA through the semantic change it induces over the base model under diagnostic prompts. Starting from a CLIP-space difference vector, this section builds toward LoUPE prompt construction, LoUPE Score quality estimation, and LoCAL weight-based retrieval.

3.1 DIFFERENCE VECTOR

Let $\mathcal{G}_{\text{base}}$ and $\mathcal{G}_\theta$ denote the base diffusion model and a LoRA-adapted model with weights θ. For a prompt p, we sample matched images $x_{\text{base}} = \mathcal{G}_{\text{base}}(p)$ and $x_\theta = \mathcal{G}_\theta(p)$, and encode them with the CLIP image encoder $\phi_I(\cdot) \in \mathbb{R}^d$. We measure the LoRA intervention with the image–image difference vector, also used in CARLoS (Sarfaty et al., 2025):

$$\delta_{\text{II}}(p;\theta) = \phi_I(x_\theta) - \phi_I(x_{\text{base}}). \tag{1}$$

δ_{II} isolates the adapter's semantic effect by subtracting content attributable to the prompt and the base-model prior; staying within a single modality further ensures this isolation by avoiding the cross-modal offset in CLIP's joint space (Liang et al., 2022). Over a prompt set $\mathcal{P}$, the empirical distribution $\{\delta_{\text{II}}(p;\theta)\}_{p\in\mathcal{P}}$ supports prompt-set convergence and quality scoring. While CARLoS summarizes CLIP-difference vectors from a fixed prompt-and-seed set for retrieval, we use this distribution to construct adapter-specific diagnostic prompts and estimate quality through convergence.

3.2 LOUPE: CONVERGENCE-GUIDED LORA UNDERSTANDING THROUGH PROMPT EXPLORATION

The difference-vector representation depends on the prompt set used to probe a LoRA. Generic prompts may fail to activate a specialized adapter, while prompts derived only from metadata may be too narrow to expose its capability boundary. We therefore propose *LoUPE* (Fig. 2), an automated pipeline that constructs, for each LoRA θ, a diagnostic prompt set $\mathcal{P}_\theta^\star$ whose induced difference-vector distribution is empirically stable.

At iteration t, the current prompt pool $\mathcal{P}_t$ induces the empirical intervention distribution

$$\mathcal{D}_t(\theta) = \{\delta_{\text{II}}(p;\theta) : p \in \mathcal{P}_t\}. \tag{2}$$

To monitor convergence, we summarize $\mathcal{D}_t(\theta)$ by its mean displacement $\boldsymbol{\mu}_t$, normalized direction $\bar{d}_t$, and cumulative explained-variance curve c_t:

$$\boldsymbol{\mu}_t(\theta) = \frac{1}{|\mathcal{D}_t(\theta)|}\sum_{\delta\in\mathcal{D}_t(\theta)}\delta, \quad \bar{d}_t(\theta) = \frac{\boldsymbol{\mu}_t(\theta)}{\|\boldsymbol{\mu}_t(\theta)\|_2}, \quad c_t(j) = \frac{\sum_{i=1}^{j}\lambda_i}{\sum_{i=1}^{r}\lambda_i},\ j = 1,\dots,r, \tag{3}$$

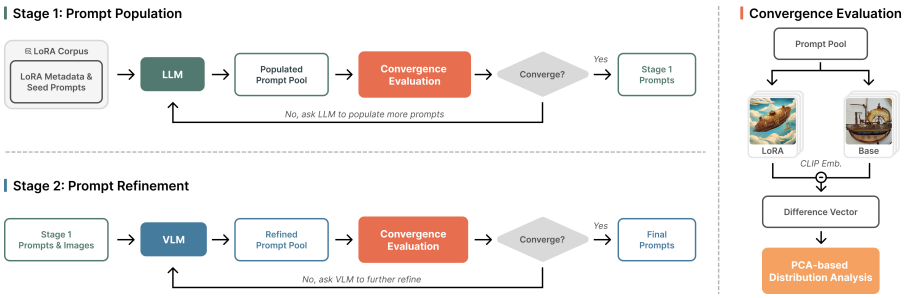

Figure 2: Overview of the LoUPE prompt-construction pipeline. Stage-1 expands metadata and prompts with an LLM until CLIP-space difference statistics stabilize; Stage-2 uses visually grounded VLM feedback to replace weak prompts.

where $\lambda_1 \geq \cdots \geq \lambda_r$ is the PCA spectrum of the delta matrix at iteration t.

We stop prompt expansion when the delta distribution is stable under two criteria:

$$\|c_t - c_{t-1}\|_\infty < \tau_{\text{var}}, \qquad \bar{d}_t^\top \bar{d}_{t-1} > \tau_{\text{dir}}, \tag{4}$$

corresponding to stability of the PCA spectrum and of the mean LoRA-induced direction. Threshold values are given in Appendix B.1.

Stage-1: metadata-guided prompt expansion. We initialize $\mathcal{P}_0$ from user prompts, trigger words, and metadata tags, then use an LLM to propose additional positive and negative prompt pairs. After each expansion round, we recompute $\mathcal{D}_t(\theta)$ and stop when Eq. (4) is satisfied.

Stage-2: VLM-guided weak-prompt replacement. Some Stage-1 prompts weakly perturb the base generation and contribute little signal. We rank prompts by their CLIP-space activation, replace the lowest-activation prompts using VLM suggestions grounded in LoRA-generated reference images, and retain higher-activation prompts unchanged. The final prompt pool $\mathcal{P}_\theta^\star$ is used for downstream characterization, quality scoring, and retrieval training. Full prompt-generation templates, activation details, and threshold choices are provided in Appendix B.1.

3.3 LoUPE Score

A high-quality LoRA should produce a semantic change that is consistent across diagnostic prompts (Stability), concentrated in a compact semantic subspace (Precision), and large enough to be distinguishable from the base model (Strength). We capture these properties with three factors:

$$\text{Stability}(\theta) = s_T(\theta), \quad \text{Precision}(\theta) = \gamma_T, \quad \text{Strength}(\theta) = \min\left(1, \frac{\|\boldsymbol{\mu}_T(\theta)\|_2}{\eta}\right), \tag{5}$$

where $s_T(\theta)$ is the iteration-penalized directional stability, $\gamma_T = c_T(K)$ is the final top-K PCA coverage, and η is the strength saturation threshold. Because Eq. (4) bounds changes in c_t, γ_T is read at a converged value. Full implementation details are given in Appendix C.1.

We combine the three factors multiplicatively:

$$\text{S}_{\text{LoUPE}}(\theta) = \text{Stability}(\theta) \times \text{Precision}(\theta) \times \text{Strength}(\theta). \tag{6}$$

We adopt a multiplicative form so that any single deficient factor drives the overall score toward zero, preventing compensation across components. A high LoUPE Score requires a sufficiently strong intervention that is directionally stable under prompt-set expansion and concentrated in a compact semantic subspace. Conversely, weak adapters, unstable adapters, or adapters with diffuse prompt-specific behavior receive low scores even if they perform well on one individual component.

3.4 LoCAL: Weight-Based Multimodal LoRA Retrieval

LoUPE automatically constructs diagnostic prompt sets that expose the semantic effect of a LoRA adapter. We next ask whether this semantic signal can be recovered directly from the adapter weights, so that new LoRAs can be indexed without rerunning the full diagnostic pipeline.

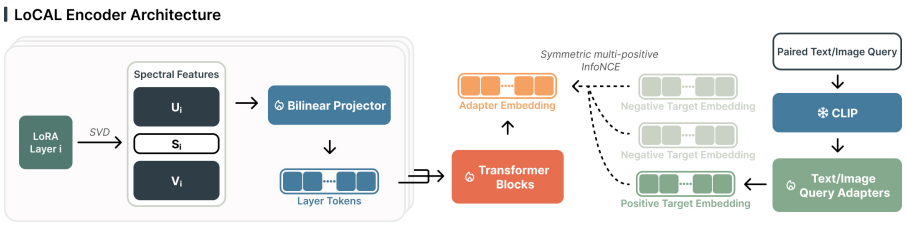

Figure 3: Overview of the LoCAL encoder. Rank-standardized LoRA factors are converted into bilinear spectral features and aggregated by a Transformer into an adapter embedding. Learned image and text query adapters map frozen CLIP features into the shared retrieval space for contrastive alignment.

To this end, we propose LoCAL (LoRA CLIP Aligner), a weight-based model for multimodal LoRA retrieval. LoCAL maps LoRA weights into a shared retrieval space, supervised by frozen CLIP embeddings of LoUPE diagnostic prompts and their generated images. Learned image and text query adapters map these CLIP features into the same space, enabling both text-to-LoRA and image-to-LoRA retrieval without diffusion-model generation at query time.

LoCAL Encoder Design. LoCAL builds on the SVD canonicalization and Transformer design of Han et al. (2026). We replace rank-level Transformer encoding with learned paired bilinear measurements, giving equivalent factorizations of a weight update the same layer representation. A Transformer aggregates these measurements and spectral statistics into a CLIP-dimensional adapter embedding z. Separate learned query adapters g^m map frozen CLIP image and text features into this shared space. Fig. 3 summarizes the architecture, with token construction, aggregation, and query-adapter details in Appendix D.1.

Training objective. For each LoRA a in a minibatch $\mathcal{B}$, we sample paired image and text targets derived from its LoUPE diagnostic prompts and generated images. For modality $m \in \{\text{img}, \text{text}\}$, let $\mathcal{T}_a^m$ index the sampled target occurrences, retaining repetitions as separate entries. Each frozen CLIP feature c_j is mapped to a normalized query embedding $\bar{c}_j = g^m(c_j)/\|g^m(c_j)\|_2$. We denote the normalized LoRA embedding by $\hat{z}_a$, the disjoint union of target indices by $\mathcal{T}_{\mathcal{B}}^m = \bigsqcup_{b\in\mathcal{B}} \mathcal{T}_b^m$, and the owning LoRA of target j by $a(j)$. We train with a symmetric multi-positive InfoNCE loss, averaging the LoRA-to-target and target-to-LoRA directions for each modality. In the LoRA-to-target direction, we maximize the total probability assigned to the LoRA's sampled positives:

$$\mathcal{L}_{z\to c}^m = -\frac{1}{|\mathcal{B}|}\sum_{a\in\mathcal{B}} \log \frac{\sum_{j\in\mathcal{T}_a^m} \exp\left(\hat{z}_a^\top \bar{c}_j/\tau\right)}{\sum_{k\in\mathcal{T}_{\mathcal{B}}^m} \exp(\hat{z}_a^\top \bar{c}_k/\tau)}. \tag{7}$$

In the target-to-LoRA direction, each query embedding predicts its owning LoRA:

$$\mathcal{L}_{c\to z}^m = -\frac{1}{|\mathcal{T}_{\mathcal{B}}^m|}\sum_{j\in\mathcal{T}_{\mathcal{B}}^m} \log \frac{\exp\left(\bar{c}_j^\top \hat{z}_{a(j)}/\tau\right)}{\sum_{b\in\mathcal{B}} \exp\left(\bar{c}_j^\top \hat{z}_b/\tau\right)}, \qquad \mathcal{L} = \frac{1}{4}\sum_{m\in\{\text{img},\text{text}\}} \left(\mathcal{L}_{z\to c}^m + \mathcal{L}_{c\to z}^m\right). \tag{8}$$

The overall objective averages both directions across image and text modalities, with a learned temperature τ.

Retrieval at Query Time. Before retrieval, we encode each LoRA in the corpus once and store its normalized embedding. At query time, we encode the input with the frozen CLIP text or image encoder, apply the corresponding learned query adapter g^{text} or g^{img}, and normalize the resulting query embedding. We rank LoRAs by cosine similarity to this query embedding in the shared retrieval space. Retrieval therefore requires no diffusion-model generation.

4 Experiments

We evaluate LoUPE prompt convergence and refinement, validate LoUPE Score against image-level metrics and human ratings, and test LoCAL text and image retrieval from encoded adapter weights on real-world SD1.5 LoRAs.

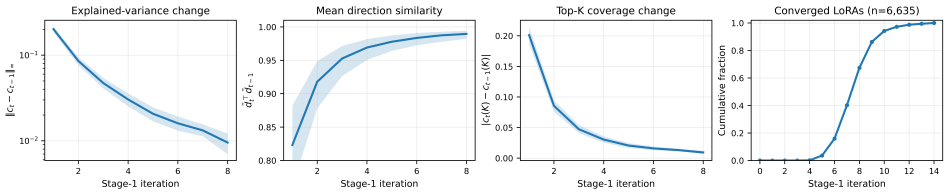

Figure 4: Stage-1 prompt expansion convergence. The left panels show medians and interquartile ranges for 3,965 LoRAs remaining in expansion through iteration 8. They track both stopping statistics in Eq. (4) and top-K coverage change $c_t(K)$, bounded by curve change. The right panel shows the cumulative fraction of all 6,635 LoRAs satisfying Eq. (4).

4.1 Experiment Setup

LoRA Corpus. We used ImageGem (Guo et al., 2025), which contains community LoRAs, descriptions, user prompts, and generated images. We focus on SD1.5 for tractable preprocessing and representation learning, sampling 6,635 adapters, half by download count for popularity and half uniformly from the remainder for long-tail diversity. The pipeline can extend to other backbones.

Rank Standardization. We standardize heterogeneous LoRAs to rank 4 via randomized SVD for a common LoCAL parameterization. Appendix A details reconstruction, with negligible approximation error for most adapters.

Data Subsets. We draw subsets from the 6,635-LoRA corpus for each experiment. LoUPE uses all 6,635 adapters for stopping statistics and a fixed cohort of 3,965 for trajectories. Human evaluation uses 140 adapters. The API budget limited LoCAL preprocessing to 6,273 adapters selected by metadata download counts. We reserve 1,254 for testing and use 5,019 for training and validation. The reported fold of a fivefold split contains 4,015 training and 1,004 validation adapters. LoCAL splits use seed 42 and are LoRA-disjoint. Selection details accompany each experiment.

4.2 LoUPE Pipeline Experiments

We test whether LoUPE in Sec. 3.2 constructs diagnostic prompts $\mathcal{P}_\theta^\star$ with rapidly converging Stage-1 difference-vector distributions and stronger, attribute-focused Stage-2 replacements.

4.2.1 Pipeline Convergence

We report population-level stopping statistics and fixed-cohort trajectories separately. Across 6,635 LoRAs, Eq. (4) gives 67.4% convergence by iteration 8, 94.2% by 10, and 100% by 14, with a median stop of 8 (Fig. 4, right). We track explained-variance curve change, mean direction similarity, and top-K coverage change $c_t(K)$, which is bounded by curve change and underlies Precision. Because early-stopping adapters leave the pool, trajectories use a fixed cohort of 3,965 adapters that remained in expansion through iteration 8. Thresholds shift where expansion stops, but score rankings remain largely stable: doubling each tolerance moves stopping earlier by up to 2.4 iterations on average, while LoUPE Scores recomputed at the new stopping points remain highly correlated with the default (Spearman $\rho \geq 0.92$; Appendix B.4).

4.2.2 Effect of VLM-Guided Weak-Prompt Replacement

Stage-1 convergence indicates a stable difference-vector estimate but does not ensure every prompt is informative. Stage-2 replaces weak prompts, identified by activation ($1 - \cos$ between base and LoRA image embeddings), with VLM-generated prompts grounded in visible adapter behavior.

Across 6,635 LoRAs and 53,817 matched weak-prompt/replacement pairs, Stage-2 raises mean activation from 0.0849 to 0.1971, a gain of 0.1122 (2.32×). Per-LoRA mean gains are positive for 99.98% of adapters, with one-sided Wilcoxon signed-rank $p < 10^{-300}$. Replacing only weak prompts gives smaller full-pool gains, from 0.1702 to 0.1833, with the weakest quartile rising from 0.1010 to 0.1226. The open-source Qwen3-VL-30B-A3B-Instruct also yields gains on this step:

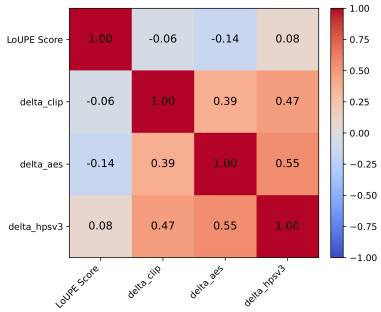

Figure 5: Spearman correlations with image-level metric gains. LoUPE Score has only small correlation with ΔCLIP, ΔAES, and ΔHPSv3, indicating an orthogonal adapter-level signal.

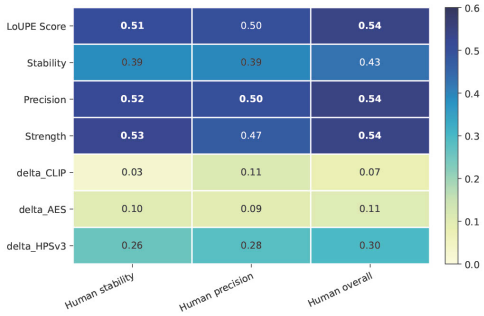

Figure 6: Spearman correlations with human ratings. LoUPE Score aligns more strongly with human judgments than generic metric gains.

repeating the first replacement round on 500 LoRAs, all other inputs frozen, gives $+0.137$ versus $+0.118$ for `gpt-5.2` (Appendix B.5).

4.3 Validating the LoUPE Score

The proposed LoUPE Score is designed to measure the quality of a LoRA adapter as an intervention on the base model, rather than the generic quality of the final generated image. This distinction is important because commonly used metrics such as CLIP alignment (Hessel et al., 2021), aesthetic score (Schuhmann et al., 2022), and HPSv3 (Ma et al., 2025) are image-level metrics: they evaluate whether a generated image is aligned, aesthetically pleasing, or preferred by a generic reward model. In contrast, LoUPE Score measures whether the LoRA induces a clear, stable, sufficiently strong, and concentrated change relative to the base model.

We therefore validate LoUPE Score in two steps. First, we compare it with existing image-level metrics to test whether it is redundant with them. Second, we evaluate LoUPE Score using adapter-centric criteria that directly reflect the intended notion of LoRA quality. We additionally include an ablation study of the LoUPE Score components in Appendix C.2.

4.3.1 Comparison with Existing Image-Level Metrics

We compare LoUPE Score with commonly used image-level metrics, including CLIP alignment, aesthetic score, and HPSv3. To reduce the confounding effect of prompt difficulty and base-model image quality, we use paired metric improvements rather than raw scores. For each prompt, we compare the image generated with the LoRA adapter against the corresponding reference image generated without the LoRA. For an image-level metric m, we define $\Delta m = m(x_{\text{LoRA}}) - m(x_{\text{base}})$, where a positive value indicates that applying the LoRA improves that metric.

Correlation with Image-Level Metrics. Fig. 5 shows small LoUPE Score correlations with ΔCLIP, ΔAES, and ΔHPSv3 of $\rho = -0.06$, -0.14, and 0.08. Bootstrap 95% CIs are $[-0.09, -0.03]$, $[-0.17, -0.11]$, and $[0.05, 0.11]$ over $n = 4{,}519$ adapters. Image-level metrics correlate with one another at 0.39 to 0.55, suggesting they are unsuitable proxies for LoRA training quality.

Qualitative Results. To further illustrate this discrepancy, we present qualitative examples in Fig. 7. These examples are selected from LoRAs where LoUPE Score and existing image-level metric improvements disagree. The left example shows that high image-level scores do not necessarily imply a high-quality adapter: the generated image may remain aesthetically pleasing or well aligned with the prompt, while the LoRA-specific effect is weak. The right example shows the opposite behavior: the LoRA intervention is stable and recognizable, but the induced style or subject change is not rewarded by generic image-level metrics. These cases further support the quantitative observation that LoUPE Score measures a distinct adapter-level property rather than generic image quality.



Figure 7: Qualitative examples where existing image-level metrics disagree with adapter-level quality. On the left, high image scores accompany weak or inconsistent interventions. On the right, stable, precise effects receive high LoUPE and human scores despite small image-metric gains.

4.3.2 Correlation with Human Ratings

We use 140 adapters spanning stability and precision, including score extremes, with evaluation-only ratings and instructions in Appendix C.3. The overall human score averages aggregated ratings as $Q_{\text{human}} = (Q^{\text{human}}_{\text{stability}} + Q^{\text{human}}_{\text{precision}})/2$. We report Spearman correlations.

As shown in Fig. 6, LoUPE Score correlates positively with both human-rated stability and precision, achieving a correlation of $\rho = 0.54$ with the human overall score. The correlation is balanced across the two human dimensions: $\rho = 0.51$ for stability and $\rho = 0.50$ for precision. This suggests that LoUPE Score captures perceptually meaningful LoRA quality rather than optimizing for only one aspect of the human evaluation.

In contrast, image-level metrics correlate weakly with human LoRA ratings, with ΔCLIP and ΔAES reaching only $\rho = 0.07$ and $\rho = 0.11$ on the overall score, and ΔHPSv3 reaching $\rho = 0.30$. This suggests that generic image-level improvements are poor proxies for adapter-level quality, whereas the LoUPE Score aligns better with human judgments by evaluating the LoRA as an intervention relative to the reference generation.

4.4 LoRA Retrieval

We test metadata-free LoRA retrieval from text or image queries using LoCAL's weight embeddings, comparing against Stylus-style baselines indexing creator-provided or generated descriptions.

Stylus-style retrieval baseline. We adapt Stylus (Luo et al., 2024) Refine-Retrieve with three document variants. *Stylus* uses Civitai titles and descriptions, and *Stylus+* adds tags. *Stylus++* replaces descriptions with `gpt-5.2` (OpenAI, 2025) descriptions generated from example images and prompts, falling back to original descriptions when unavailable. Stylus++ thus provides an example-informed reference beyond metadata alone. We encode documents with the CLIP text encoder (Radford et al., 2021) and rank LoRAs by cosine similarity in the shared CLIP space.

LoCAL training setup. LoCAL uses rank-4 unscaled updates $B_i A_i$, excluding the α/r multiplier and text-encoder weights, with 128 bilinear measurements per layer. We train on the partition in Sec. 4.1. Architecture dimensions and training settings appear in Appendices D.1 and D.2.

Main Retrieval Results. We compare LoCAL with replicated metadata-based RAG baselines on $1,254$ unseen LoRAs using $89,852$ unique text and $269,616$ individual image queries. Tab. 1 reports Recall@1, Recall@5, Recall@10, and MRR averaged across LoRAs. Scaling uses nested pools of 125, 627, and $1,254$ candidates, with queries restricted to sampled adapters. We use one validation-selected checkpoint throughout, reporting means and standard deviations over five subset seeds for smaller pools. Oracle represents each LoRA by its mean generated-prompt or image CLIP embedding for text and image retrieval, respectively. We omit CARLoS (Sarfaty et al., 2025), which supports only short, fixed-template text queries rather than full prompts and requires approximately $4,480$ generated images per LoRA in its original setup, impractical at our scale.

The key advantage of LoCAL is that it retrieves from weights without inference-time titles, descriptions, tags, or generated metadata. Despite stricter inputs, it outperforms Stylus and Stylus+ on every

Table 1: Image- and text-to-LoRA retrieval performance on ImageGem test sets under increasing candidate-pool sizes. We compare Stylus and Stylus+ with LoCAL (Ours). Rows below each separator report Stylus++ as an example-informed reference and Oracle as a centroid reference. Bold and italics mark the highest and second-highest scores within each pool, including Stylus++ and excluding Oracle.

		Image-to-LoRA				Text-to-LoRA			
# LoRAs	**Method**	Recall@1↑	Recall@5↑	Recall@10↑	MRR↑	Recall@1↑	Recall@5↑	Recall@10↑	MRR↑
125	Stylus	0.1094 ± 0.0192	0.2115 ± 0.0163	0.2763 ± 0.0210	0.1548 ± 0.0158	0.1436 ± 0.0138	0.2517 ± 0.0277	0.3232 ± 0.0297	0.1920 ± 0.0188
	Stylus+	0.1365 ± 0.0322	0.2645 ± 0.0169	0.3427 ± 0.0107	0.1932 ± 0.0252	$\mathit{0.1705 \pm 0.0228}$	0.3076 ± 0.0342	0.3889 ± 0.0330	0.2305 ± 0.0266
	LoCAL	$\mathit{0.1961 \pm 0.0171}$	$\mathit{0.4292 \pm 0.0204}$	$\mathbf{0.5632 \pm 0.0171}$	$\mathit{0.3142 \pm 0.0171}$	0.1535 ± 0.0143	$\mathit{0.3757 \pm 0.0146}$	$\mathit{0.5156 \pm 0.0148}$	$\mathit{0.2693 \pm 0.0136}$
	Stylus++	$\mathbf{0.2609 \pm 0.0358}$	$\mathbf{0.4536 \pm 0.0442}$	$\mathit{0.5512 \pm 0.0419}$	$\mathbf{0.3437 \pm 0.0390}$	$\mathbf{0.2981 \pm 0.0359}$	$\mathbf{0.5025 \pm 0.0448}$	$\mathbf{0.5964 \pm 0.0483}$	$\mathbf{0.3863 \pm 0.0401}$
	Oracle	0.5495 ± 0.0240	0.7456 ± 0.0178	0.8261 ± 0.0133	0.6419 ± 0.0212	0.7291 ± 0.0098	0.8719 ± 0.0077	0.9172 ± 0.0061	0.7941 ± 0.0086
627	Stylus	0.0586 ± 0.0030	0.1191 ± 0.0028	0.1542 ± 0.0033	0.0850 ± 0.0030	0.0853 ± 0.0047	0.1578 ± 0.0061	0.1984 ± 0.0078	0.1168 ± 0.0050
	Stylus+	0.0714 ± 0.0038	0.1497 ± 0.0074	0.1932 ± 0.0074	0.1054 ± 0.0050	$\mathit{0.1023 \pm 0.0072}$	$\mathit{0.1909 \pm 0.0107}$	0.2392 ± 0.0111	$\mathit{0.1407 \pm 0.0086}$
	LoCAL	$\mathit{0.0884 \pm 0.0028}$	$\mathit{0.2222 \pm 0.0060}$	$\mathit{0.3046 \pm 0.0072}$	$\mathit{0.1623 \pm 0.0041}$	0.0609 ± 0.0016	0.1690 ± 0.0031	$\mathit{0.2459 \pm 0.0031}$	0.1250 ± 0.0023
	Stylus++	$\mathbf{0.1516 \pm 0.0051}$	$\mathbf{0.2925 \pm 0.0091}$	$\mathbf{0.3610 \pm 0.0104}$	$\mathbf{0.2124 \pm 0.0060}$	$\mathbf{0.1871 \pm 0.0116}$	$\mathbf{0.3394 \pm 0.0065}$	$\mathbf{0.4135 \pm 0.0078}$	$\mathbf{0.2526 \pm 0.0095}$
	Oracle	0.4155 ± 0.0050	0.5806 ± 0.0061	0.6505 ± 0.0063	0.4969 ± 0.0053	0.6065 ± 0.0042	0.7607 ± 0.0060	0.8135 ± 0.0051	0.6789 ± 0.0046
1,254	Stylus	0.0476	0.0957	0.1232	0.0686	0.0721	0.1313	0.1646	0.0981
	Stylus+	0.0563	0.1186	0.1530	0.0832	*0.0849*	*0.1578*	*0.1992*	*0.1166*
	LoCAL	*0.0603*	*0.1621*	*0.2292*	*0.1184*	0.0407	0.1190	0.1749	0.0885
	Stylus++	**0.1167**	**0.2380**	**0.2971**	**0.1691**	**0.1487**	**0.2822**	**0.3503**	**0.2062**
	Oracle	0.3713	0.5266	0.5917	0.4479	0.5565	0.7119	0.7657	0.6295

image-to-LoRA metric and pool size, demonstrating useful weight-derived semantics on unseen adapters without external annotations.

Performance decreases with pool size, with text-to-LoRA Recall@1 below Stylus+ throughout. Stylus++ leads non-Oracle methods at $1,254$ candidates, but uses richer information from adapter-associated images and prompts. Greater encoder capacity and training-set size may improve LoCAL.

5 Discussion

Findings and implications. Informative prompts are essential to adapter evaluation, as specialized LoRAs may appear ineffective when prompts miss their learned behavior. LoUPE's results in Sec. 4.2 support adaptive probing, although statistical convergence does not establish exhaustive capability coverage. Agreement with human-rated stability and precision in Sec. 4.3 supports evaluating reusable modifications alongside image quality. LoCAL separates behavioral training supervision from candidate indexing, retrieving held-out adapters from weights alone in Tab. 1. This enables search without candidate metadata or example images, complemented by behavioral evaluation.

Limitations. We focus on SD1.5 for its large, diverse public adapter corpus, but SDXL and DiT validation remains necessary. Neither component assumes SD1.5-specific structure. LoUPE uses paired base/LoRA generations in a shared embedding space, and LoCAL uses rank-standardized weights. Limited CLIP and VLM semantic coverage may obscure visually salient changes, motivating richer vision encoders. LoUPE evaluates consistency over natural usage prompts rather than a controlled factorial design. Free-form prompts reduce assumptions about intended use but cannot isolate modified factors, motivating complementary controlled probing. LoCAL still trails metadata-rich baselines in Recall@1 and pool-size scaling. We hypothesize that encoder capacity and training-set scale largely explain this gap. Combining weight, metadata, and behavioral signals may help.

Future Work. LoRA Scope could extend to adapter composition, conflict detection, automatic library curation, and intervention alignment beyond CLIP. This view may support side-effect auditing, cross-backbone comparisons, and explanations of retrieval relevance and expected changes.

6 Conclusion

We introduced LoRA Scope, a framework for understanding, evaluating, and retrieving text-to-image LoRA adapters through their semantic interventions. To answer what a LoRA changes in generation and when that change reliably appears, we characterize adapters with CLIP-space difference vectors from matched base and LoRA outputs. LoUPE constructs adapter-specific diagnostic prompts, the LoUPE Score evaluates LoRA-level quality from the induced intervention, and LoCAL recovers a searchable semantic representation directly from LoRA weights. Together, these components show that LoRAs can be indexed and assessed by the visual changes they consistently produce.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data sets, design or provide feedback on research methodology or experiments, implement methods, clean and reformat dataset. To be specific, we used generative AI tools to construct synthetic data for LoRA characterization, evaluation, and retrieval training. In the LoUPE pipeline, OpenAI `o4-mini` expands adapter-specific positive and negative prompts from user prompts, trigger words, and metadata, while `gpt-5.2` proposes replacements for weak prompts using LoRA-generated reference images. We generate matched base-model and LoRA-adapted images from these prompts and use their CLIP-space differences for downstream characterization, quality scoring, and retrieval training. We also use `gpt-5.2` to generate adapter descriptions from example images and prompts for the Stylus++ baseline. Model configurations, and processing procedures are described in the paper and appendix.

We have not used generative AI tools to help develop theoretical models or conceptual frameworks, propose or refine hypotheses, assist with translation, support qualitative and thematic data analysis, interpret results. And formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs are not applicable to this work. Additionally, we used generative AI tools for summarize or analyze existing literature, sourcing/searching for information, edit a research paper to improve readability, identify relevant literature. We have reviewed all AI-assisted work. Generative AI tools are considered as part of our proposed method, and went through manually designed and crafted evaluations. We summarize and analyzed existing literature manually as cross validation to prevent hallucination. We manually review and refine the research paper after LLM editing. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

LoRA Scope can make large public LoRA libraries easier to search and evaluate, which may reduce trial-and-error costs for creative use. At the same time, improved adapter discovery could also make harmful, copyrighted, or identity-misusing adapters easier to find, so deployment should be paired with dataset filtering, provenance tracking, and policy-aware moderation.

REFERENCES

Samuel Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=CQsmMYmlP5T.

Amil Dravid, Yossi Gandelsman, Kuan-Chieh Wang, Rameen Abdal, Gordon Wetzstein, Alexei A. Efros, and Kfir Aberman. Interpreting the weight space of customized diffusion models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 137334–137371. Curran Associates, Inc., 2024. doi: 10.52202/079017-4363. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f85364507054c257959c2011c28bfc0d-Paper-Conference.pdf.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. ImageBind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15180–15190, June 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Girdhar_ImageBind_One_Embedding_Space_To_Bind_Them_All_CVPR_2023_paper.html.

Yuanhe Guo, Linxi Xie, Zhuoran Chen, Kangrui Yu, Ryan Po, Guandao Yang, Gordon Wetzstein, and Hongyi Wen. Imagegem: In-the-wild generative image interaction dataset for generative model personalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19577–19586, October 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Guo_ImageGem_In-the-wild_Generative_Image_Interaction_Dataset_for_Generative_Model_Personalization_ICCV_2025_paper.html.

Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. AudioCLIP: Extending CLIP to image, text and audio. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 976–980, 2022. doi: 10.1109/ICASSP43922.2022.9747631. URL https://www.dfki.de/en/web/research/projects-and-publications/publication/12157.

Xiaolong Han, Ferrante Neri, Zijian Jiang, Fang Wu, Yanfang Ye, Lu Yin, and Zehong Wang. W2t: Lora weights already know what they can do, 2026. URL https://arxiv.org/abs/2603.15990.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7514–7528, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.595. URL https://aclanthology.org/2021.emnlp-main.595/.

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. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. Lorahub: Efficient cross-task generalization via dynamic lora composition. In *Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=TrloAXEJ2B.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Xiaolong Jin, Kai Wang, Dongwen Tang, Wangbo Zhao, Yukun Zhou, Junshu Tang, and Yang You. Conditional LoRA parameter generation, 2024. URL https://arxiv.org/abs/2408.01415.

Zhiwen Li, Zhongjie Duan, Die Chen, Cen Chen, Daoyuan Chen, Yaliang Li, and Yingda Chen. AutoLoRA: Automatic LoRA retrieval and fine-grained gated fusion for text-to-image generation, 2025. URL https://arxiv.org/abs/2508.02107.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625, 2022.

Michael Luo, Justin Wong, Brandon Trabucco, Yanping Huang, Joseph E. Gonzalez, Zhifeng Chen, Ruslan Salakhutdinov, and Ion Stoica. Stylus: Automatic adapter selection for diffusion models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 32888–32915. Curran Associates, Inc., 2024. doi: 10.52202/079017-1035. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/39a746f507dc5087bd85cc39ded8c52f-Paper-Conference.pdf.

Yuhang Ma, Yunhao Shui, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score, 2025. URL https://arxiv.org/abs/2508.03789.

OpenAI. Introducing GPT-5.2. https://openai.com/index/introducing-gpt-5-2/, 2025. Product release, December 11, 2025. Accessed: May 5, 2026.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty (eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 487–503, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.39. URL https://aclanthology.org/2021.eacl-main.39/.

Theo Putterman, Derek Lim, Yoav Gelberg, Stefanie Jegelka, and Haggai Maron. Learning on loRAs: GL-equivariant processing of low-rank weight spaces for large finetuned models. In *Workshop on Neural Network Weights as a New Data Modality*, 2025. URL https://openreview.net/forum?id=MZmPapNtih.

Reza Qorbani, Gianluca Villani, Theodoros Panagiotakopoulos, Marc Botet Colomer, Linus Härenstam-Nielsen, Mattia Segu, Pier Luigi Dovesi, Jussi Karlgren, Daniel Cremers, Federico Tombari, and Matteo Poggi. Semantic library adaptation: LoRA retrieval and fusion for open-vocabulary semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9804–9815, June 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Qorbani_Semantic_Library_Adaptation_LoRA_Retrieval_and_Fusion_for_Open-Vocabulary_Semantic_CVPR_2025_paper.html.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6527–6536, June 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Ruiz_HyperDreamBooth_HyperNetworks_for_Fast_Personalization_of_Text-to-Image_Models_CVPR_2024_paper.html.

Shahar Sarfaty, Adi Haviv, Uri Hacohen, Niva Elkin-Koren, Roi Livni, and Amit H. Bermano. CARLoS: Retrieval via concise assessment representation of LoRAs at scale. *arXiv preprint arXiv:2512.08826*, 2025. URL https://arxiv.org/abs/2512.08826.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/a1859debfb3b59d094f3504d5ebb6c25-Abstract-Datasets_and_Benchmarks.html.

Ethan Smith, Rami Seid, Alberto Hojel, Paramita Mishra, and Jianbo Wu. LoRA diffusion: Zero-shot LoRA synthesis for diffusion model personalization, 2024. URL https://arxiv.org/abs/2412.02352.

Mert Sonmezer, Matthew Zheng, and Pinar Yanardag. LoRAverse: A submodular framework to retrieve diverse adapters for diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17879–17888, October 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Sonmezer_LoRAverse_A_Submodular_Framework_to_Retrieve_Diverse_Adapters_for_Diffusion_ICCV_2025_paper.html.

Guy Tevet, Brian Gordon, Amir Hertz, Amit H. Bermano, and Daniel Cohen-Or. MotionCLIP: Exposing human motion generation to CLIP space. In *Computer Vision – ECCV 2022*, volume 13682 of *Lecture Notes in Computer Science*, pp. 358–374. Springer, 2022. doi: 10.1007/978-3-031-20047-2_21. URL https://www.ecva.net/papers/eccv_2022/papers_ECCV/html/4022_ECCV_2022_paper.php.

Ivan Vulić, Adam Grycner, Quentin de Laroussilhe, and Jonas Pfeiffer. LoRA-Squeeze: Simple and effective post-tuning and in-tuning compression of LoRA modules, 2026. URL https://arxiv.org/abs/2602.10993.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23965–23998. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/wortsman22a.html.

Yujia Wu, Yiming Shi, Jiwei Wei, Chengwei Sun, Yang Yang, and Heng Tao Shen. Difflora: Generating personalized low-rank adaptation weights with diffusion, 2024. URL https://arxiv.org/abs/2408.06740.

Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. ULIP: Learning a unified representation of language, images, and point clouds for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1179–1189, June 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Gao_ULIP_Learning_a_Unified_Representation_of_Language_Images_and_Point_CVPR_2023_paper.html.

Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. Lit: Zero-shot transfer with locked-image text tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18123–18133, June 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Zhai_LiT_Zero-Shot_Transfer_With_Locked-Image_Text_Tuning_CVPR_2022_paper.html.

Ziyu Zhao, Leilei Gan, Guoyin Wang, Wangchunshu Zhou, Hongxia Yang, Kun Kuang, and Fei Wu. LoraRetriever: Input-aware LoRA retrieval and composition for mixed tasks in the wild. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 4447–4462, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.263. URL https://aclanthology.org/2024.findings-acl.263/.

APPENDIX OUTLINE

- **Appendix A** (A): LoRA rank standardization procedure and reconstruction error.
- **Appendix B** (B.1, B.2, B.3, B.4, B.5): LoUPE prompt construction, post-hoc representation analysis for LoRA identity clustering, empirical justification for the convergence signal choice, a threshold-sensitivity replay of the stopping rule, and an open-source VLM replication of Stage-2.
- **Appendix C** (C.1, C.2, C.3): Full definitions of the stability, precision, and strength score components, a qualitative ablation on distinct failure mode, and the human rating protocol.
- **Appendix D** (D.1, D.2): LoCAL encoder architecture, learned image/text query adapters, and sampling and optimization settings.

A RANK STANDARDIZATION DETAILS

For each `lora_down` and `lora_up` pair, we reconstruct the update matrix $\Delta W = W_{\text{up}} W_{\text{down}}$ and compute an approximate randomized SVD with query rank set to the original LoRA rank. Let $U_{r'}$, $S_{r'}$, and $V_{r'}^{\top}$ denote the top r' retained components. We split the singular values symmetrically:

$$W'_{\text{up}} = U_{r'} S_{r'}^{1/2}, \qquad W'_{\text{down}} = S_{r'}^{1/2} V_{r'}^{\top}. \tag{9}$$

This yields a fixed-rank adapter with the same layer structure. Fig. 8 shows the average reconstruction error across target ranks. We use $r' = 4$ in all subsequent experiments as a balance between approximation quality and efficiency. This post-hoc compression workflow independently arrives at a similar procedure to the concurrent LoRA-Squeeze method of Vulić et al. (2026), which also reconstructs LoRA updates and applies randomized SVD to obtain lower-rank adapters.

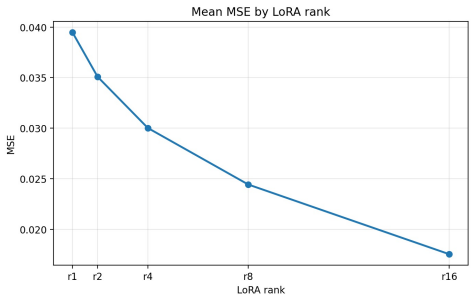

Figure 8: Reconstruction error of LoRA rank standardization. We report the MSE between the original update ΔW and its randomized-SVD approximation at different target ranks.

B LoUPE Details

B.1 Prompt Construction

Metadata-guided expansion. For each LoRA, we initialize $\mathcal{P}_0$ from real user prompts associated with the adapter, available trigger words, and metadata tags. The LLM generates additional positive and negative prompt pairs that preserve the observed subject, style, and composition patterns while remaining distinct from the seed examples. When trigger words are available, each generated positive prompt must include at least one trigger word. After each expansion round, we render matched base and LoRA-adapted images, recompute $\mathcal{D}_t(\theta)$, and apply the convergence test in Eq. (4).

Weak-prompt replacement. Stage-1 can produce prompts that only weakly activate the adapter. We measure prompt activation by the CLIP-space image difference

$$A(p;\theta) = 1 - \cos(\phi_I(x_{\text{base}}), \phi_I(x_\theta)), \tag{10}$$

where higher activation indicates that the prompt exposes a stronger LoRA-specific intervention relative to the base model. We rank Stage-1 prompts by $A(p;\theta)$ and select the lowest-activation prompts as replacement candidates. A VLM receives a small set of LoRA-generated reference images and the weak prompts, then proposes attribute-focused replacements grounded in the visible LoRA behavior. These replacements substitute for the weak prompts in $\mathcal{P}^\star_\theta$, while higher-activation Stage-1 prompts are retained unchanged. Stage-2 runs up to three replacement rounds; after each round the convergence test of Eq. (4) is re-evaluated on the refined pool, and replacement stops early once it is satisfied.

Convergence thresholds. The convergence thresholds τ_{var} and τ_{dir} control the stability criteria for the PCA spectrum and the mean intervention direction. In all pipeline experiments, we use $\tau_{\text{var}} = 0.01$ for the explained-variance curve delta and $\tau_{\text{dir}} = 0.98$ for mean-direction cosine stability; the thresholds are fixed across LoRAs. Because the number of PCA components can differ between iterations, curves of different length are compared after padding the shorter one with its final value. Our implementation additionally evaluates $|c_t(K) - c_{t-1}(K)| < \tau_{\text{gain}}$ with $\tau_{\text{gain}} = \tau_{\text{var}}$; since $|c_t(K) - c_{t-1}(K)| \le \|c_t - c_{t-1}\|_\infty$, this check is implied by the spectrum criterion and never alters a stopping decision. The resulting prompt pool is optimized for a stable estimate of the LoRA-induced difference-vector distribution rather than for linguistic diversity alone.

LLM/VLM configuration. For the pipeline experiments, the Stage-1 LLM prompt-expansion component uses the OpenAI `o4-mini` model, and the Stage-2 VLM weak-prompt replacement component uses `gpt-5.2`. Appendix B.5 repeats Stage-2 replacement with an open-source VLM.

B.2 Post-Hoc Representation Analysis

After LoUPE produces matched prompt–image pairs, we analyze which CLIP-space representations preserve semantic LoRA identity.

We use the converged LoUPE outputs from the pipeline run. From the pipeline manifest, we randomly sample 100 SD1.5 LoRAs with seed 42. For each LoRA, we group the matched outputs by prompt, sample up to 32 unique prompts and keep up to four generated variations per selected prompt. This gives 12,800 matched examples in total, where each example contains the prompt p, the base-model image x_{base}, and the LoRA-adapted image x_θ for the same prompt and variation.

We embed prompts and images with CLIP ViT-L/14 and compare six representations: the LoRA image embedding $\phi_I(x_\theta)$, the base image embedding $\phi_I(x_{\text{base}})$, the prompt embedding $\phi_T(p)$, the prompt–image residual $\delta_{\text{PI}} = \phi_I(x_\theta) - \phi_T(p)$, the base–prompt residual $\phi_I(x_{\text{base}}) - \phi_T(p)$, and the intervention residual $\delta_{\text{II}} = \phi_I(x_\theta) - \phi_I(x_{\text{base}})$. All residual features are L2-normalized before clustering. We fit KMeans with $k = 100$ clusters and evaluate agreement with LoRA identity using NMI and ARI. We also report silhouette score using the ground-truth LoRA identities.

Tab. 2 shows that δ_{PI} gives the strongest cluster-label agreement among the tested representations, with NMI 0.634 and ARI 0.337. Prompt embeddings are also competitive, which is expected because the LoUPE prompt set is adapter-specific and often contains LoRA-relevant trigger concepts. In contrast, δ_{II} is much weaker for post-hoc semantic clustering, despite being the intervention-grounded

signal used by the main method. Fig. 9 visualizes this contrast with t-SNE projections of a 20-LoRA subset.

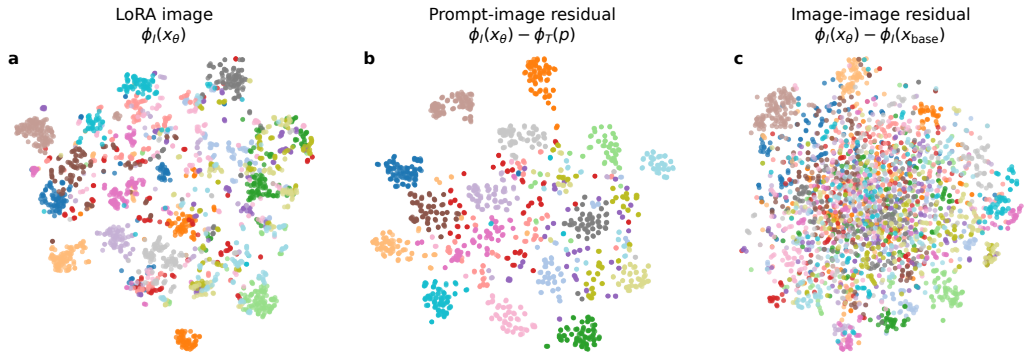

Figure 9: t-SNE projections for a readable 20-LoRA subset from the 100-LoRA balanced run. **(a)** LoRA image embedding $\phi_I(x_\theta)$. **(b)** Prompt–image residual $\delta_{\text{PI}} = \phi_I(x_\theta) - \phi_T(p)$. **(c)** Image–image residual $\delta_{\text{II}} = \phi_I(x_\theta) - \phi_I(x_{\text{base}})$. Each color represents one LoRA identity. The quantitative clustering metrics in Tab. 2 are computed on the full 100-LoRA set.

Table 2: Representation comparison for LoRA identity clustering on the balanced 100-LoRA LoUPE output set. Silhouette is computed using ground-truth LoRA identities; NMI and ARI compare KMeans assignments with LoRA identity. δ_{PI} gives the best cluster-label agreement.

Representation	**Sil.** ↑	**NMI** ↑	**ARI** ↑
$\phi_I(x_\theta)$	0.010	0.527	0.243
$\phi_I(x_{\text{base}})$	-0.018	0.427	0.134
$\phi_T(p)$	**0.076**	0.613	0.320
$\delta_{\text{PI}} = \phi_I(x_\theta) - \phi_T(p)$	0.041	**0.634**	**0.337**
$\phi_I(x_{\text{base}}) - \phi_T(p)$	0.033	0.589	0.287
$\delta_{\text{II}} = \phi_I(x_\theta) - \phi_I(x_{\text{base}})$	-0.008	0.267	0.061

These results support using the prompt–image residual as a descriptive post-hoc representation of LoRA identity. Subtracting the prompt embedding emphasizes visual content present in the LoRA output beyond the text prompt. By contrast, δ_{II} isolates the adapter's intervention relative to the base model, making it appropriate for LoUPE's behavior-grounded convergence signal but less effective as a standalone semantic clustering embedding.

B.3 Choice of Difference Vector for Prompt-Set Convergence

LoUPE uses the image–image intervention residual $\delta_{\text{II}} = \phi_I(x_\theta) - \phi_I(x_{\text{base}})$ as the prompt-set convergence signal. We compare this choice against the prompt–image residual $\delta_{\text{PI}} = \phi_I(x_\theta) - \phi_T(p)$ on a random sample of 100 LoRAs from the 6,635 LoRA corpus (seed 42). The experiment uses the same LLM-only Stage-1 prompt-expansion phase, CLIP ViT-L/14 embeddings, and top-K PCA coverage statistic with $K = 8$. We evaluate both residuals under the LoUPE stopping thresholds specified above ($\tau_{\text{var}} = 0.01$, $\tau_{\text{dir}} = 0.98$).

At the final paired iteration for each adapter, δ_{II} gives a more compact and stable intervention distribution than δ_{PI}. Its final top-K PCA coverage is higher (0.3868 vs. 0.3442), and its explained-variance curve change is lower (0.0073 vs. 0.0120). The paired mean difference is significant for both coverage ($\delta_{\text{PI}} - \delta_{\text{II}} = -0.0426$, 95% CI $[-0.0490, -0.0363]$) and variance-curve change (0.0048, 95% CI $[0.0037, 0.0059]$).

δ_{PI} also has slightly higher final mean-direction cosine similarity (0.9990 vs. 0.9909), but this directional stability is partly expected because δ_{PI} retains prompt-semantic structure shared across expansion rounds. Overall, these results support using δ_{II} as LoUPE's convergence control signal: it more directly isolates the LoRA intervention relative to the base model and converges reliably

across the sampled adapters. In contrast, δ_{PI} remains useful for post-hoc semantic clustering, where retaining prompt-related semantic identity can be beneficial.

B.4 Convergence Threshold Sensitivity

The convergence thresholds determine only where Stage-1 expansion stops; they do not alter the prompts proposed before that point. We therefore replay the two-criterion stopping rule of Eq. (4) on the saved per-iteration statistics of all $6{,}635$ LoRAs under perturbed thresholds. For τ_{dir} we scale the tolerance $1 - \tau_{\mathrm{dir}}$, since scaling the cosine itself is meaningless. Because the saved histories end at the default stopping iteration, looser variants are fully observable (they can only stop earlier), whereas tighter variants are right-censored: for these we report only the fraction of LoRAs whose default stopping iteration already satisfies the tighter rule. The replay perturbs only the Stage-1 stopping decision; in the pipeline the same criteria also terminate Stage-2 replacement rounds, which the replay does not reproduce.

Stage-1 stopping directly determines the statistics that enter the LoUPE Score, so at each variant's stopping iteration T' we recompute Stability and Precision from the Stage-1 statistics at T' following Appendix C.1, with the iteration penalty applied to T'; Strength is held at its stored converged value because per-iteration mean norms are not logged. The scores reported in Sec. 4.3 are computed after Stage-2 replacement, which we do not re-run here. Stage-2 changes these inputs little: over the $4{,}519$ scored adapters in the default runs, replacement shifts Precision by a median of 0.0035 and the raw stability cosine by 0.0027. The score recomputed from the Stage-1 end statistics correlates with the reported score at Spearman $\rho = 0.958$ with the penalty applied at the Stage-1 stop and at $\rho = 0.995$ when the penalty is evaluated at the same final iteration, so the remaining difference stems from the penalty's iteration count rather than from Stage-2 changes to the statistics. Variant and default are always compared under identical computation.

Table 3: Replay of the Stage-1 stopping rule under perturbed thresholds ($n = 6{,}635$). Stop unchanged: fraction of LoRAs with the same stopping iteration as the default. Shift: mean number of iterations by which stopping moves earlier. Jaccard: mean prompt-pool overlap with the default pool. ρ and top-10%: Spearman correlation and top-decile overlap of LoUPE Score rankings, recomputed from Stage-1 statistics, against the default.

Variant	τ_{var}	τ_{dir}	Stop unchanged	Shift	Median stop	Jaccard	ρ / top-10%
Default	0.01	0.98	–	–	8	–	–
$\tau_{\mathrm{var}} \times 2$	0.02	0.98	23.5%	1.68	6	0.817	0.940 / 78.6%
$(1 - \tau_{\mathrm{dir}}) \times 2$	0.01	0.96	83.7%	0.35	8	0.968	0.985 / 98.6%
Both $\times 2$	0.02	0.96	8.5%	2.40	5	0.746	0.925 / 77.7%
$\tau_{\mathrm{var}}/2$	0.005	0.98	17.5%		right-censored		
$(1 - \tau_{\mathrm{dir}})/2$	0.01	0.99	56.7%		right-censored		
Both $/2$	0.005	0.99	9.5%		right-censored		

Tab. 3 shows that loosening τ_{var} moves most stopping points earlier and reduces prompt-pool overlap, so the stopping iteration itself is sensitive to this threshold. The score ranking is not: its Spearman correlation with the default stays at or above 0.92 under every loosened variant, and the top decile is at least 77% preserved. We do not extrapolate beyond the observed trajectories for the tighter variants. All replays use SD1.5 rank-4 adapters; other backbones or ranks may stop at different iterations and could require retuned tolerances, which would test the portability of the stopping parameters rather than the robustness of the rankings reported here.

B.5 Stage-2 Replacement with an Open-Source VLM

To test whether Stage-2 depends on the closed-source `gpt-5.2`, we repeat the first replacement round with the open-source Qwen3-VL-30B-A3B-Instruct on a random sample of 500 LoRAs (seed 42) from the corpus. Everything except the VLM is frozen to the original run: the Stage-1 prompt pool and its activation ranking, the four context images shown to the VLM, the four weak prompts selected for replacement, the decoding temperature (0.4), and the rendering configuration (SD1.5, 20 denoising steps, guidance 7.5, four seeds per prompt, CLIP ViT-L/14 embeddings). This yields $2{,}000$ matched weak-prompt/replacement pairs per VLM. The VLM only proposes candidates; whether a candidate is retained is decided by its measured activation after rendering, and the pool is re-ranked

after each round, so a weak replacement can itself be replaced later. Because this experiment evaluates first-round proposals before any later correction, it measures proposal failures directly.

Table 4: Stage-2 weak-prompt replacement with two VLMs on the same 500 LoRAs (2,000 pairs each). Gains are paired differences in activation $A(p;\theta)$ relative to the replaced weak prompt; intervals are paired LoRA-level bootstrap 95% CIs (10,000 resamples).

	Weak prompts	`gpt-5.2`	Qwen3-VL-30B-A3B
Mean activation	0.075	0.193	0.212
Mean gain over weak prompts	–	$+0.118$ $[0.114, 0.122]$	$+0.137$ $[0.132, 0.142]$
Pairs with positive gain	–	97.5%	98.6%
LoRAs with positive mean gain	–	100%	99.4%

Both VLMs reproduce the Stage-2 effect (Tab. 4); the open-source model's gain is slightly larger (Qwen minus `gpt-5.2`: $+0.019$, 95% CI $[0.014, 0.024]$). Absolute levels differ from Sec. 4.2 because this is a 500-LoRA subsample restricted to the first round. Proposal failures are rare for both models: 51 of 2,000 `gpt-5.2` replacements and 29 of 2,000 Qwen replacements decrease activation, with mean changes of -0.015 and -0.043 against $+0.122$ and $+0.139$ for improving replacements. These failing pairs start from higher activation than the weak-prompt average (0.100 and 0.165 versus 0.075), consistent with less room for improvement.

C LoUPE Score Details

C.1 Score Definition

Stability. A high-quality LoRA should induce a directionally stable effect: after the diagnostic prompt set has been expanded, adding more prompts should not substantially change the mean intervention direction. We use the cosine similarity between the final mean direction and the previous iteration's mean direction:

$$\mathrm{Stab}_{\mathrm{raw}}(\theta) = \mathrm{clip}_{[0,1]}\left(\bar{d}_T(\theta)^\top \bar{d}_{T-1}(\theta)\right), \tag{11}$$

where $\mathrm{clip}_{[0,1]}(z) = \min(1, \max(0, z))$. However, an adapter that requires many refinement iterations to stabilize is less reliable as a training artifact than one whose behavior is identified quickly. We therefore apply an iteration penalty:

$$s_T(\theta) = \mathrm{Stab}_{\mathrm{raw}}(\theta) \cdot 2^{-\frac{\max(0, T - T_{\mathrm{ref}})}{\tau_{\mathrm{iter}}}}. \tag{12}$$

In our experiments, we set $T_{\mathrm{ref}} = 10$ and $\tau_{\mathrm{iter}} = 8$. Thus, no penalty is applied when the final iteration is at or below 10, and every additional 8 iterations halves the stability multiplier.

Precision. We define precision as the final top-K PCA coverage:

$$\mathrm{Precision}(\theta) = \gamma_T = c_T(K). \tag{13}$$

This quantity is the final value of the top-K PCA coverage $c_t(K)$ monitored during convergence; its change is bounded by the spectrum criterion in Eq. (4), so it has saturated by iteration T. We refer to it as precision because it measures how concentrated the adapter's effect is: a precise LoRA produces variations that are largely explained by a small number of dominant directions, whereas a noisy or prompt-dependent LoRA spreads its effect across many unrelated directions. We use $K = 8$ throughout our experiments.

Strength. Stability and precision alone are insufficient: an adapter may appear stable and low-dimensional simply because it has little measurable effect. We measure the magnitude of the final mean intervention as

$$\mathrm{strength_norm}(\theta) = \left\|\boldsymbol{\mu}_T(\theta)\right\|_2. \tag{14}$$

To prevent very large magnitudes from dominating the final score, we use a saturating strength factor

$$\mathrm{Strength}(\theta) = \min\left(1, \frac{\mathrm{strength_norm}(\theta)}{0.02}\right), \tag{15}$$

so $\eta = 0.02$ in Eq. (5). This term penalizes adapters whose average effect is too weak to be reliably observed in the embedding space, while assigning full strength credit once the effect reaches the threshold.

C.2 Score Component Ablation

Fig. 10 shows representative LoRAs selected by their component scores. The high-quality example receives consistently high scores across stability, precision, and strength, and visually exhibits a clear and localized LoRA-induced effect. In contrast, the low-stability example produces the target effect only in some generations, indicating that measuring precision or strength alone is insufficient. The low-precision example introduces a visible change, but the modification is spatially inaccurate or affects irrelevant regions. Finally, the weak-intervention example preserves image quality but induces only a subtle change relative to the base generation.

These cases demonstrate that the three components capture complementary failure modes. Stability measures whether the LoRA effect is consistent, precision measures whether the effect is localized to the intended target, and strength prevents trivial adapters that barely alter the base generation from receiving high scores. Their combination provides a more faithful adapter-level quality measure than any single component alone.

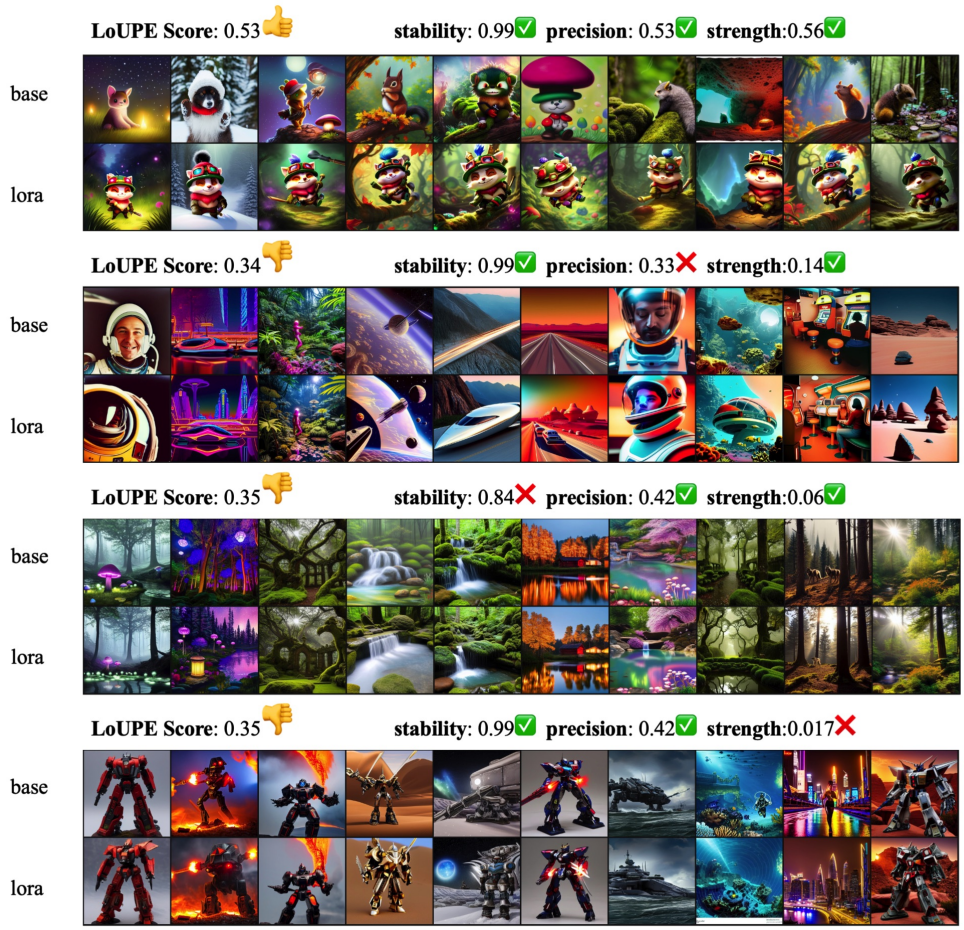

Figure 10: Qualitative case study of LoUPE Score components. High-quality adapters require stable, precise, and sufficiently strong interventions. Removing any component can fail to distinguish important failure modes, such as inconsistent effects, imprecise modifications, or overly weak LoRA-induced changes.

C.3 Human Rating Design

To quantitatively assess the perceptual quality of LoRA adaptations, we conducted a human rating study centered on two dimensions: *stability* and *precision*. Six internal raters participated in the study, with each LoRA independently rated by three raters. *Stability* refers to whether the LoRA produces a reliable and uniform visual effect across independently generated images, while *precision* refers to whether the LoRA modifies only the intended target region, leaving all other areas visually faithful to the unmodified reference.

LoRA Selection. We sample the rated adapters from the 3,965 LoRAs that remained in Stage-1 expansion through iteration 8 (Fig. 4). We first sample 120 LoRAs across low, medium, and high levels of stability and precision, so that the rated set covers combinations of consistent and inconsistent, precise and imprecise interventions. We then add 20 anchors, namely the 10 highest- and 10 lowest-scoring adapters, to ensure that both extremes of the quality range are represented. All sampling uses seed 42, yielding 140 unique LoRAs. The human ratings are used only to evaluate automatic scores; they are not used as LoCAL supervision, nor for model selection or tuning any LoUPE Score parameter. Because this sampling and the anchors widen the spread of adapter quality relative to the full corpus, the reported correlations apply to this sample rather than to a uniform draw from the corpus.

Rating Presentation. For each LoRA, each evaluator was presented with two sets of 20 images: a reference set generated without any LoRA applied, and a target set generated using identical prompts with the LoRA under evaluation. The two sets were displayed in parallel to facilitate direct perceptual comparison. All images were generated under identical sampling conditions.

Rating Procedure. Raters scored each LoRA along both dimensions independently from 0 to 5, where 0 denotes the lowest quality, and 5 denotes the highest. For *stability*, raters judged the extent to which the LoRA effect was coherent and uniform across all 20 target images, penalizing cases where the effect was absent in some images or varied erratically in strength. For *precision*, raters judged whether the non-target regions in the target set were faithfully preserved relative to the reference set, penalizing unintended alterations in areas that should remain unchanged. Ratings were made holistically over the full set of 20 image pairs rather than on a per-image basis, encouraging evaluators to form an overall impression of each LoRA's behavior.

D LoCAL Details

D.1 Encoder Architecture

LoCAL combines a bilinear layer representation with learned image and text query adapters to support multimodal LoRA retrieval. Building on the weight canonicalization and Transformer encoding introduced by Han et al. (2026), we replace rank-level Transformer encoding with paired bilinear measurements that are invariant to equivalent decompositions of the weight update. Canonicalization uses QR decompositions of the LoRA factors followed by a singular value decomposition of their small core matrix, as detailed below. For a LoRA a, we suppress the adapter index on its layer-wise quantities and use rank-standardized LoRA pairs $A_i \in \mathbb{R}^{r' \times d_{\text{in},i}}$ and $B_i \in \mathbb{R}^{d_{\text{out},i} \times r'}$, where $r' = 4$. Here, A_i and B_i are the standardized down and up factors W'_{down} and W'_{up} for pair i. The encoder represents the unscaled update $B_i A_i$, without applying the LoRA α/r multiplier or including text-encoder weights.

Spectral decomposition. We obtain the singular components of each update without materializing the dense matrix. Thin QR decompositions give $B_i = Q_{B,i} R_{B,i}$ and $A_i^\top = Q_{A,i} R_{A,i}$. We then decompose the small $r' \times r'$ core:

$$R_{B,i} R_{A,i}^\top = \widetilde{U}_i \Sigma_i \widetilde{V}_i^\top, \qquad U_i = Q_{B,i} \widetilde{U}_i, \qquad V_i = Q_{A,i} \widetilde{V}_i. \tag{16}$$

Thus, $B_i A_i = U_i \Sigma_i V_i^\top$, with singular values σ_{ik} and paired directions u_{ik}, v_{ik}. The decomposition is computed in float32 without gradients; learning starts at the direction projectors. In practice, the retained SVD components saved during the rank standardization described in Appendix A can be reused, avoiding a separate decomposition in the encoder.

Bilinear layer representation. Instead of a rank-level Transformer, LoCAL combines the singular components through 128 learned bilinear measurements. Let $\Phi_{d_{\text{out},i}}$ and $\Psi_{d_{\text{in},i}}$ be bias-free linear maps to $\mathbb{R}^{128}$, shared across pairs with the same output or input dimension, respectively. With $s_i = \sum_{k=1}^{r'} \sigma_{ik}$ and $p_{ik} = \sigma_{ik}/s_i$, the layer sketch is

$$b_i = \mathrm{LN}\left(\sum_{k=1}^{r'} p_{ik} \left(\Phi_{d_{\text{out},i}} u_{ik} \right) \odot \left(\Psi_{d_{\text{in},i}} v_{ik} \right) \right) \in \mathbb{R}^{128}. \tag{17}$$

Before layer normalization, this is the diagonal of $\Phi_{d_{\text{out},i}} (B_i A_i) \Psi_{d_{\text{in},i}}^\top / s_i$. Consequently, simultaneous sign flips of paired singular vectors cancel, and rotations within repeated-singular-value subspaces leave the sketch unchanged. The representation therefore depends on the weight update rather than its choice of low-rank factors. In the implementation, singular values are clamped below at 10^{-12} for numerical stability.

We retain magnitude and spectral shape through four additional statistics:

$$q_i = \left[\log s_i, \quad \log s_i - \frac{1}{N} \sum_{j=1}^{N} \log s_j, \quad -\frac{\sum_{k=1}^{r'} p_{ik} \log p_{ik}}{\log r'}, \quad p_{i1} \right], \tag{18}$$

where N is the number of LoRA pairs and singular values are ordered from largest to smallest. These terms encode total strength, strength relative to the adapter's other layers, normalized spectral entropy, and the leading singular value's share of total strength. Concatenating b_i and q_i gives a 132-dimensional feature, which is mapped to a layer token:

$$\mathrm{tok}_i = \mathrm{LN}(\mathrm{GELU}(W_{\text{tok}} [b_i; q_i] + b_{\text{tok}})) + e_i^{\text{pos}} + e_{\gamma(i)}^{\text{group}} \in \mathbb{R}^{256}. \tag{19}$$

Here, e_i^{pos} identifies the pair's position in the fixed layer layout, and $e_{\gamma(i)}^{\text{group}}$ identifies its shape group $\gamma(i)$.

Adapter and query embeddings. We prepend a learned CLS token and aggregate the layer tokens with a two-layer, pre-normalized Transformer of width 256, with eight attention heads, feed-forward

width $1{,}024$, GELU activations, and dropout 0.1. After the final layer normalization, a linear head maps the CLS representation to $z_a \in \mathbb{R}^{512}$, the adapter-indexed form of z in the method section.

On the query side, separate image and text adapters map frozen CLIP features into the shared retrieval space. For modality $m \in \{\text{img}, \text{text}\}$ and normalized CLIP feature $\hat{c}_j = c_j / \|c_j\|_2$, we use

$$g^m(c_j) = \text{normalize}(\hat{c}_j + W_2^m \, \text{GELU}(W_1^m \hat{c}_j)) \, . \tag{20}$$

The bias-free down and up projections W_1^m and W_2^m have dimensions 64×512 and 512×64, respectively. The up projection is initialized to zero, so each query adapter initially preserves the normalized CLIP feature. Following the training objective, we use $\hat{z}_a = z_a / \|z_a\|_2$ and $\bar{c}_j = g^m(c_j) / \|g^m(c_j)\|_2$ for contrastive similarities and retrieval scores. The query adapter already returns a unit-normalized vector, so the latter normalization leaves its output unchanged.

D.2 Training Settings

Each minibatch contains 48 LoRAs, with 16 paired image/text targets sampled per LoRA. The learned temperature τ is initialized at 0.07. We use AdamW with peak learning rate 2.5×10^{-4}, weight decay 6.609×10^{-6}, OneCycleLR scheduling, mixed precision, and gradient clipping at 1.0.