LayerVerse: Finding the Sweet Spot for KV-Injection in Training-Free Image Editing

Mikhail Zhirnov[1,2,3]✉, Arsen Kuzhamuratov[1,2], Andrey Kuznetsov[1,2], Ivan Oseledets[2,3], and Konstantin Sobolev[1,2,4]

[1] FusionBrain Lab, Russia
[2] AXXX, Russia [3] Applied AI Institute, Russia
[4] Lomonosov Moscow State University, Russia
✉: zhirnov.m.d@gmail.com

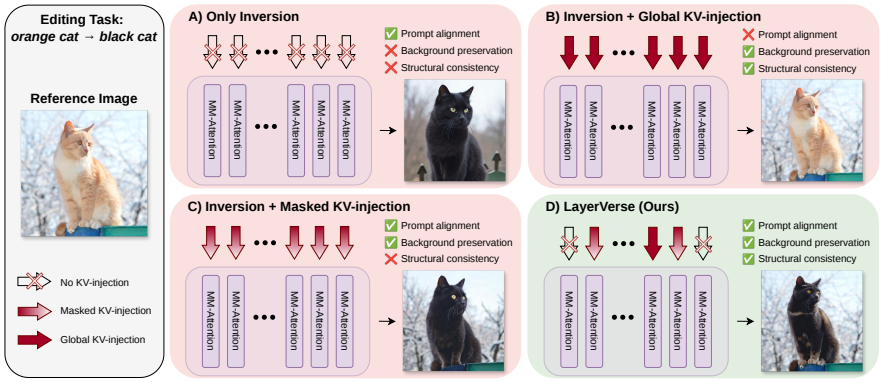

Fig. 1: *LayerVerse* resolves the fidelity-editability trade-off. Decoupling KV-injection into masked background protection and sparse global structural anchors enables precise editing without rigid source reconstruction or unconstrained structural collapse.

Abstract. For image editing with Multimodal Diffusion Transformers (MM-DiT), training-free methods face a critical trade-off: precise background preservation limits editability, while high prompt fidelity degrades the original scene and object structure. We argue that this tension stems from suboptimal Key-Value (KV) injection strategies. Global KV-injection rigidly over-preserves the source, whereas strictly masked injection acts as localized inpainting, destroying the edited object's structural identity. We introduce *LayerVerse*, an optimization framework that resolves this fundamental dilemma of layer selection for KV-injection by assigning specific layers to two distinct roles: Masked Injection to precisely protect the background, and Global Injection to anchor the edited object's structure. Formulating optimal layer allocation as a combinatorial problem, we model the editing error based on pairwise layer interactions and solve it globally via Mixed-Integer Linear Programming (MILP). Supported by lightweight, single-step automatic masking, *LayerVerse* seamlessly integrates into training-free, inversion-based editors, achieving highly competitive results on the PIE-Bench benchmark.

Keywords: Training-free · Image Editing · MM-DiT · Rectified Flow

1 Introduction

Recent advancements in text-to-image (T2I) generation have marked a significant transition from traditional U-Net [25] architectures to Diffusion Transformers (DiT) [22], and from standard Diffusion Models [7] to Flow Matching [16,17]. State-of-the-art flow models, such as Stable Diffusion 3 [6] and FLUX [2], construct a direct probability flow from noise to image, enabling faster generation with superior synthesis quality. Leveraging multimodal architectures with joint attention over text and images (MM-DiT), these models have naturally emerged as a powerful foundation for high-fidelity image editing tasks [5, 26, 30].

Text-guided image editing is broadly categorized into training-required and training-free approaches. While models trained specifically for instruction-based editing (e.g., InstructPix2Pix [3], FLUX.1 Kontext [14], Qwen-Image-Edit [33]) show impressive capabilities, they demand extensive computational resources, rely on highly curated datasets, and often struggle with exact background reconstruction. This work focuses exclusively on training-free editing, which utilizes pre-trained models out-of-the-box to execute targeted semantic changes while perfectly preserving the source image context.

A persistent challenge in training-free editing is resolving the fundamental trade-off between background preservation and target editability. Typically, these methods operate by first inverting the source image into noise and then generating the edited image along a new synthesis trajectory. Recent flow-based editors tackle this trade-off by refining the inversion trajectory (e.g., FireFlow [5], RF-Solver [30]), bypassing inversion entirely (e.g., FlowEdit [13]), identifying vital transformer layers for attention-feature injection (e.g., StableFlow [1]), or manipulating local guidance and attention primitives (e.g., UniEdit-Flow [10], QK-Edit [27]). To preserve source features, some of these methods employ Key-Value (KV) injection. *However, existing techniques largely rely on fixed or heuristic injection policies across all transformer blocks, leaving the critical layer selection problem unsolved.* As illustrated in Fig. 1, this exposes a fundamental dilemma: relying solely on inversion loses the source structure entirely (Fig. 1A), whereas applying KV-injection globally acts as a massive reconstructive attractor that prevents meaningful text-driven edits (Fig. 1B). Conversely, restricting features strictly to the background (e.g., KV-Edit [39]) turns the task into unconstrained localized inpainting, destroying the target object's structural identity (Fig. 1C).

To address this limitation, we propose an approach that is complementary and orthogonal to existing training-free editing methods. Rather than redesigning the underlying ODE solvers or attention mechanisms, we directly optimize the exact block-wise allocation of the injected features. We introduce *LayerVerse* (Fig. 1D), an optimization framework that explicitly separates KV-injection into two distinct roles: *Masked Injection* to preserve the background, and a limited budget of *Global Injections* to retain the essential shape and layout of the edited object. Because identifying the exact optimal subset of transformer blocks for these distinct roles is a vast combinatorial problem, we model the expected editing error based on layer interactions and solve it globally via Mixed-Integer Linear Programming (MILP).

To separate the background from the editing region without relying on user-provided ground-truth masks or computationally heavy external segmentation models, we utilize a straightforward automatic masking technique. Extracting cross-attention maps in one forward pass establishes a robust spatial guide with minimal computational overhead, fully enabling our KV-injection strategy.

Comprehensive evaluation on the PIE-Bench benchmark [11] demonstrates the strong practical impact of our approach. *LayerVerse* consistently yields significant empirical improvements across multiple metrics, effectively balancing prompt adherence with high-fidelity background and structural preservation.

Our main contributions are summarized as follows:

- **LayerVerse:** We propose a novel layer selection framework for KV-injections that is entirely orthogonal to existing training-free editing methods and is capable of significantly improving their performance.
- **Pairwise Interaction Modeling:** We propose modeling the editing error as a system of multi-level (pairwise) interactions between transformer layers. By formulating a dual-objective MILP, we efficiently identify the exact optimal combination of masked background protections and limited global structural anchors without exhaustive search.
- **Solver-Agnostic Integration:** Our method acts as a solver-agnostic prior. We calibrate the optimal layer combination using a standard Euler sampler and demonstrate on the PIE-Bench benchmark [11] that this optimized set seamlessly transfers to and improves other advanced solvers (FireFlow [5], RF-Solver [30]). This is supported by a lightweight, single-step attention mask that serves strictly as an auxiliary spatial guide.

2 Related Work

The field of training-free image editing has advanced significantly with the rise of flow and diffusion transformers. Early flow-based editors [5, 26, 30] primarily focused on inversion and sampling trajectories: [26] introduced semantic inversion for rectified flow, while [5,30] improved reconstruction fidelity with stronger ODE solvers. FlowEdit [13] removed the inversion step entirely by constructing a direct source-to-target ODE. Other recent methods, such as UniEdit-Flow [10] and DNAEdit [35], further refine flow-based editing. For instance, UniEdit-Flow combines predictor-corrector inversion with delayed feature injection, locally increasing edited-region guidance for new content generation. While effective for object insertion and prompt adherence, stronger local guidance can still yield less realistic textures and partial background degradation during challenging edits.

Recent studies on MM-DiT architectures have also revisited the core control mechanisms. StableFlow [1] identifies vital transformer layers for attention-feature injection and improves inversion for real-image editing. [36] improves inversion and invariance control within flow transformers, while QK-Edit [27] argues that manipulating queries and keys serves as a more effective attention primitive in MM-DiTs. KV-Edit [39] preserves masked background regions via KV-caching, effectively confining generation to the edited region.

Several works explicitly exploit regional cues derived from attention maps or trajectory signals [8, 18, 28]. Follow-Your-Shape [18] computes a Trajectory Divergence Map from token-wise velocity differences and uses it to guide scheduled KV-injection, while [28] analyzes MM-DiT attention blocks to extract cleaner attention-derived masks for editing. Meanwhile, DCEdit [8] extracts a refined attention cue at the beginning of inversion and reuses it for both feature- and latent-level control. Our masking module is conceptually closer to this latter approach, but functions strictly as a lightweight spatial guide rather than the primary methodological contribution.

Prior work mainly improves inversion, layer selection, attention manipulation, or localization under a single method-specific intervention policy. Our method operates orthogonally: it optimizes the block-wise placement of masked KV-injection together with a small number of global structural anchors.

3 Motivation

In Rectified Flow models, standard text-guided inversion often fails to preserve the fine structural details of the source image. A widely adopted solution is Key-Value (KV) injection. Adapted from Large Language Models [23, 34], this mechanism caches attention features during the inversion of the source image and reuses them during generation to enforce spatial consistency and anchor the process to the source structure [4].

Historically, this technique has introduced a fundamental trade-off between "free generation" (relying only on inverted noise, Fig. 1A) and "rigid reconstruction" (enforcing source features globally, Fig. 1B). However, we observe that this trade-off behaves very differently depending on the spatial scope of the injected features, exposing two critical failure modes in existing methods:

The Global Injection Collapse. Applying KV-injection globally across transformer layers achieves flawless reconstruction but sacrifices editability [5, 30]. More importantly, without spatial constraints, global KV-injection acts as a remarkably strong reconstructive attractor. *Even when applied to a very small subset of layers, global injection rapidly overpowers the text prompt, collapsing the generation into the original image and preventing meaningful edits (Fig. 1B).*

The Masked Inpainting Pitfall. A logical countermeasure is to restrict KV-injection strictly to the unedited background using a spatial mask (e.g., KV-Edit [39]). While this successfully preserves the unedited background pixels, it introduces a new problem: *the editing task devolves into blind localized inpainting (Fig. 1C).* By completely withholding source features from the edited region, the target object loses its original structural identity, shape, and pose, relying entirely on unconstrained inverted noise to synthesize new content.

These observations highlight that neither purely global nor purely masked injections are sufficient on their own. *To achieve high-quality editing (Fig. 1D), we must simultaneously utilize Masked Injection to accurately preserve the background, and a limited budget of Global Injections to act as structural anchors for the edited object.*

4 Method

In this section, we introduce *LayerVerse*, detailing our intuitive approach to modeling how layer selection impacts editing quality and how we efficiently find the optimal combination of injected layers. Finally, we describe a single-step automatic masking technique that serves as an essential auxiliary spatial guide for our method.

4.1 LayerVerse: Optimized Layer Selection

The Core Intuition. At its core, *LayerVerse* seeks to answer a simple question: during the generation process, which transformer layers should receive KV-injection (active layers), and which should not (inactive layers)? Applying KV-injection uniformly to all layers perfectly reconstructs the source but acts as a massive reconstructive attractor, preventing text-driven edits (Fig. 1B). Conversely, turning all layers off gives the model freedom to synthesize new content (Fig. 1A), but completely degrades the background and object structure. To formalize this, for a model with P transformer blocks (e.g., $P = 57$ in FLUX.1 [2]), we define a simple binary indicator vector $\mathbf{x} \in \{0, 1\}^P$, where $x_i = 1$ if block i is active, and $x_i = 0$ otherwise.

Modeling the Editing Error. How does activating specific layers affect the final editing quality? The impact of an active layer is not independent; layers intrinsically interact with each other. For example, activating layer i might yield a high-quality edit only if layer j is also active. To capture this relationship perfectly, one would need to evaluate every possible combination. However, selecting an optimal subset of $N = 20$ active layers from $P = 57$ blocks involves $\binom{57}{20} \approx 1.21 \times 10^{15}$ configurations, making an exhaustive search intractable.

Instead, we approximate the abstract editing error $\mathcal{E}(\mathbf{x})$ mathematically. Crucially, we restrict our formulation to account only for pairwise interactions. While higher-order synergies exist, explicitly modeling third-order interactions (e.g., triplets of layers) or beyond leads to a combinatorial explosion. For instance, in a 57-layer architecture, evaluating pairwise interactions requires measuring $\sim 1,596$ combinations, while third-order interactions jump to over $29,000$. This makes the physical evaluation of the interaction tensor computationally intractable. Thus, we safely truncate the approximation to a robust second-order (pairwise) quadratic model:

$$\mathcal{E}(\mathbf{x}) \approx C + \sum_{i=1}^{P} w_i x_i + \sum_{1 \le i < j \le P} W_{i,j} x_i x_j, \tag{1}$$

where C is a baseline constant, w_i represents the individual contribution of activating layer i, and $W_{i,j}$ captures the pairwise interaction synergy when layers i and j are active together.

In our quadratic model, the coefficients $(C, w_i, W_{i,j})$ are initially unknown, but we can efficiently determine them using a small hold-out calibration dataset.

To avoid a combinatorial explosion, we construct a *bottom-up model* by evaluating the editing error strictly at the boundaries of the network. Specifically, for each configuration – no active layers $L(\emptyset)$, single active layers $L(\{i\})$, and pairs $L(\{i,j\})$ – we compute the empirical error and average it across the calibration image dataset. These averaged values serve as robust estimates that completely define our quadratic landscape parameters without exhaustive search.

Background and Structure Objectives. A high-quality edit actually requires minimizing two distinct, often conflicting types of errors, which we model independently using Equation 1:

- **Background Error ($\mathcal{E}_{\mathbf{bg}}$):** To maintain the original scene context, we must preserve the unedited pixels. We evaluate this error using a single chosen standard fidelity metric (e.g., PSNR [9] or LPIPS [38]).
- **Structure Error ($\mathcal{E}_{\mathbf{str}}$):** To ensure the newly generated object looks coherent with the original layout, we must protect its shape and identity. We measure this structural deviation explicitly using a Structure Distance loss [29].

To simultaneously address both objectives, we expand our binary choice into two distinct active roles. We introduce *Masked Injection* (delivering source features strictly to the background via a spatial mask) and *Global Injection* (delivering features to the entire image to act as a structural anchor). For each block i, we define decision variables $m_i \in \{0,1\}$ and $s_i \in \{0,1\}$. Since a block cannot serve both roles simultaneously, $m_i + s_i \leq 1$. The general active indicator becomes $x_i = m_i + s_i$. The background error depends on *any* injection ($x_i = m_i + s_i$), while the structure error depends *solely* on global anchors (s_i).

Finding the Optimal Layer Combination. Our final goal is to allocate a fixed budget of exactly N active layers, out of which a predefined subset of N_{str} layers strictly serves as global structural anchors ($N_{\text{str}} \leq N$). We seek the precise binary combination that minimizes the combined error:

$$\min_{\mathbf{m},\mathbf{s}} \quad \mathcal{E}_{\text{bg}}(\mathbf{x}) + \lambda \cdot \mathcal{E}_{\text{str}}(\mathbf{s}) \quad \text{subject to} \quad \sum_{i=1}^{P} x_i = N, \quad \sum_{i=1}^{P} s_i = N_{\text{str}}, \tag{2}$$

where λ balances the relative scales of the two objectives.

Because we need to minimize a quadratic function over discrete binary choices subject to strict budget constraints, we seamlessly cast this as a Mixed-Integer Linear Programming (MILP) problem [32]. For reference, MILP is a highly robust mathematical optimization framework. Once the error functions are computed, the optimization problem is solved globally in just a few seconds, easily avoiding the need to brute-force millions of combinations.

Ultimately, this framework provides us with the optimal set of masked and global layers. We emphasize that this configuration acts as a solver-agnostic prior: once calibrated using a standard sampling method, the identical layer configuration can be applied directly to enhance various inversion-based solvers out-of-the-box.

Algorithm 1 Single-Step Automatic Masking

Input: Source image Z_0^{src}, prompts $p_{\text{src}}, p_{\text{tgt}}$
Parameters: Optimal timestep t^*, optimal block subset B^*
Output: Binary editing mask M

$Z_1 \leftarrow \text{Invert}(Z_0^{\text{src}})$ ▷ Obtain exact source-consistent noise
$Z_{t^*} \leftarrow (1 - t^*)Z_0^{\text{src}} + t^* Z_1$ ▷ Interpolate along ODE trajectory
procedure ExtractMask(p)
    $\mathcal{A}_p \leftarrow \text{ForwardPass}(Z_{t^*}, p)$
    $\bar{\mathcal{A}}_p \leftarrow \frac{1}{|B^*|} \sum_{l \in B^*} \mathcal{A}_p^{(l)}$ ▷ Extract and average cross-attention from B^*
    return OtsuBinarize(Smooth($\bar{\mathcal{A}}_p$)) ▷ [21]
end procedure
$M_{\text{src}} \leftarrow \text{ExtractMask}(p_{\text{src}})$
$M_{\text{tgt}} \leftarrow \text{ExtractMask}(p_{\text{tgt}})$
$M \leftarrow \text{MorphologicalClose}(M_{\text{src}} \vee M_{\text{tgt}})$ ▷ Merge and refine boundaries
return M

4.2 Single-Step Automatic Masking

Effective masking acts strictly as an auxiliary spatial guide to localize changes and protect the background. In our framework, relying solely on unconstrained generation easily degrades the background, making a localized masking branch vital to guide the injections. We aim for a maximally simple, lightweight tool that operates within a single forward pass, completely avoiding the computational burden of heavy external models like SAM [12]. Crucially, deriving this mask directly from the model's internal representations frees our inference pipeline from an unfair dependency on user-provided ground-truth (GT) masks.

To generate the mask efficiently, we first invert the source image to obtain a structure-aligned noise trajectory Z_1 and interpolate it to an optimized timestep t^*. During a single forward pass, cross-attention maps from a calibrated subset of blocks B^* are extracted, aggregated, and morphologically refined (Algorithm 1). This low-overhead operation directly yields the precise spatial boundaries required to unlock our optimized layer selection strategy.

5 Experiments

5.1 Implementation Details

Experimental Setup. All experiments are conducted using the open-source FLUX.1-[dev] [2] and SD3.5-medium [6] models. For FLUX.1-[dev], we employ 15 steps with CFG scales of 1.0 for inversion and 2.0 for generation. For SD3.5-medium, we use 28 steps with CFG scales of 1.0 and 7.5, respectively.

We calibrate our quadratic error models from Equation 1 (i.e., compute the baseline constant and pairwise interaction weights) using a hold-out subset of 115 images from the MS-COCO dataset [15]. This entire calibration process is performed using a standard, basic Euler solver. To robustly navigate the combinatorial search space, we ran our MILP optimization independently targeting

two different background preservation metrics: PSNR and LPIPS. Remarkably, the solver converged to the *exact same optimal layer combination* for both metrics, underscoring the physical consistency of the discovered layer interactions.

Automatic Masking Parameters. All benchmark metrics reported in our experiments are computed utilizing our single-step automatic masking module. The masking parameters were derived offline on the exact same hold-out image dataset used for calibration. Specifically, the optimal interpolation step t^* and the subset of cross-attention blocks B^* were selected by maximizing the Intersection over Union (IoU) metric against GT masks. For FLUX.1-[dev], the optimal parameters are $t^* = 11$ and $B^* =$ `[14, 17, 18, 23]`. For SD3.5-medium, the optimal spatial localization occurs at $t^* = 7$ using blocks $B^* =$ `[7, 9, 13, 23]`.

Optimal Layer Budgets. For FLUX.1-[dev], the solver identified an optimal budget of $N = 8$ active blocks: Double blocks `[1, 2]` and Single blocks `[32, 33, 34, 35, 36, 37]` (using 0-based indexing), with only Double block `[2]` allocated as a global structural anchor ($N_{\text{str}} = 1$). For SD3.5-medium, the optimal budget shifts to $N = 13$ blocks (indices `[0, 1, 3, 4, 12, 14, 17, 18, 19, 20, 21, 22, 23]`) and the structural anchors to $N_{\text{str}} = 4$ (indices `[20, 21, 22, 23]`). This variance is due to distinct architectural properties: FLUX triggers an overwhelmingly strong reconstructive collapse if allocated more than one anchor block, while SD3.5-medium inherently allows for much stronger textual editability but loses structural consistency more rapidly, necessitating additional global anchors to preserve object identity. Consequently, this analysis demonstrates that the optimal layer configuration is inherently *architecture-specific*. However, once calibrated, *LayerVerse* acts as a robust, *solver-agnostic prior*, transferring seamlessly to the considered editing solvers without recalibration.

Computational Cost. Calibration is a *one-time*, offline procedure for a specific model architecture. It consists of two primary stages: first, inverting images from the calibration hold-out subset to compute editing errors for single and pairwise layer interactions; second, performing the MILP optimization. For FLUX.1-[dev], completing this entire first stage across the full 115-image subset requires ∼348 GPU hours (at 6.59 seconds per image on a single H100). However, as demonstrated in Fig. 5(A3), a calibration size of just 10 images is sufficient to maintain competitive performance, effectively decreasing the first stage's computation time to ∼30 GPU hours. The subsequent MILP optimization is solved almost instantaneously, taking only ∼1.97 seconds.

Baselines. We compare *LayerVerse* against state-of-the-art training-free editing methods, maintaining the default hyperparameters from their official implementations. For FLUX.1-[dev], baselines include methods utilizing KV-injection (RF-Solver [30], FireFlow [5]) and alternative control techniques (DNAEdit [35], UniEdit-Flow [10], RF-inversion [26], FlowEdit [13]). For SD3.5-medium, we evaluate against DNAEdit [35], FTEdit [36], FlowEdit [13], and FSI-Edit [37]. Since standard RF-Solver and FireFlow lack official SD3.5 implementations, we evaluate them on this architecture strictly using our optimized *LayerVerse*.

Table 1: Quantitative evaluation on the PIE-Bench benchmark using the FLUX.1-[dev] architecture. *LayerVerse* (LV) seamlessly integrates with the evaluated editing solvers and achieves the best overall performance.

Method	LV	Structure	Background Preservation				CLIP Similarity		User Alignment		Rank
		Distance ↓	PSNR ↑	LPIPS ↓	MSE ↓	SSIM ↑	Whole ↑	Edited ↑	HPSv3 ↑	EditScore ↑	Avg. ↓
RF-inversion [26]	×	42.00	20.15	177.11	141.22	70.64	24.57	21.75	6.36	4.16	5.88
FlowEdit [13]	×	27.96	21.92	112.99	95.68	83.35	**25.41**	**22.29**	**6.81**	4.99	4.13
DNAEdit [35]	×	17.00	25.18	86.78	48.36	87.19	24.78	21.77	5.80	4.68	4.50
UniEdit-Flow [10]	×	**8.82**	29.74	55.35	18.16	91.12	24.59	21.29	4.27	4.73	3.50
RF-Solver [30]	×	11.87	27.81	73.21	28.17	88.35	23.74	20.53	4.93	3.71	5.88
	✓	9.45	**30.16**	**37.62**	**18.01**	**92.66**	24.30	21.23	5.16	4.93	3.38
FireFlow [5]	×	9.48	28.33	64.59	24.51	89.23	23.68	20.45	4.88	3.60	5.88
	✓	9.36	30.15	37.82	18.05	92.61	24.39	21.30	5.07	**5.05**	**2.88**

Table 2: Quantitative evaluation on the PIE-Bench benchmark using the SD3.5-medium architecture. *LayerVerse* (LV) seamlessly adapts to this backbone, finding a broader optimal block budget that perfectly balances SD3.5's aggressive editing capabilities with necessary structural protections.

Method	LV	Structure	Background Preservation				CLIP Similarity		User Alignment		Rank
		Distance ↓	PSNR ↑	LPIPS ↓	MSE ↓	SSIM ↑	Whole ↑	Edited ↑	HPSv3 ↑	EditScore ↑	Avg. ↓
FTEdit [36]	×	21.65	23.45	92.85	62.38	85.84	24.47	21.13	4.50	4.33	5.50
FlowEdit [13]	×	23.16	23.24	93.58	69.67	85.10	**26.59**	**23.15**	**6.47**	**5.92**	3.50
FSI-Edit [37]	×	14.02	26.60	84.43	36.79	86.31	25.49	22.01	5.66	4.82	3.63
DNAEdit [35]	×	13.64	**26.65**	74.50	**32.64**	88.91	25.44	22.19	5.41	5.16	2.50
RF-Solver [30]	✓	13.81	26.50	51.46	36.63	89.60	25.14	22.24	4.99	4.91	3.44
FireFlow [5]	✓	**13.52**	26.58	**50.85**	35.95	**89.70**	25.11	22.25	5.01	4.93	**2.44**

5.2 Evaluation and Metrics

We evaluate our method on the established PIE-Bench [11] benchmark, comprising 620 diverse images. We exclude style transfer tasks, as the fidelity-editability trade-off fundamentally assumes the presence of an invariant background region to preserve. We employ a comprehensive set of metrics:

- **Structure Preservation:** Structure Distance ↓ [29] measures how well the generated object retains the original structural layout, shape, and identity.
- **Background Preservation:** LPIPS ↓ [38], MSE ↓, PSNR ↑ [9], and SSIM ↑ [31] measure the pixel-wise and perceptual fidelity of the unedited regions.
- **Editing Quality:** CLIP similarity [24] evaluated on the whole image (Whole ↑) and the edited region (Edited ↑) measures target prompt adherence.
- **Human Preference Modeling:** To complement CLIP, we use advanced reward models: HPSv3 ↑ [20] for aesthetics, and EditScore ↑ [19] for reasoning-based editing assessment.
- **Average Rank ↓:** Methods are ranked per metric (1=best). The Avg. Rank is the mean of four group ranks (Structure, Background, CLIP Similarity, User Alignment), where each group averages its respective metric ranks.

5.3 Quantitative Results

As shown in Tab. 1, our primary evaluation focuses on how *LayerVerse* (+ LV) enhances existing baselines on the FLUX.1-[dev] model. On their own, standard RF-Solver and FireFlow struggle with the fidelity-editability trade-off, both tying for a suboptimal Average Rank of 5.88. However, integrating the *LayerVerse* optimized layer configuration elevates both methods to the top of the benchmark, yielding simultaneous improvements across *all* evaluated dimensions. For instance, applying LV to FireFlow drastically improves background preservation (LPIPS drops from 64.59 to 37.82), tightens structural consistency (Structure Distance improves from 9.48 to 9.36), and boosts human-aligned reasoning scores (EditScore jumps from 3.60 to 5.05). A similar performance leap is observed when applying *LayerVerse* to RF-Solver.

To demonstrate generalizability, we extend our evaluation to the SD3.5-medium model (Tab. 2). While SD3.5 inherently allows for stronger textual editability than FLUX, it suffers from more severe structural volatility. *LayerVerse* adapts to these architectural shifts by determining a broader optimal block budget to enforce necessary structural protections. Consequently, integrating our method with FireFlow on SD3.5 yields the best Structure Distance (13.52) and excellent background preservation (SSIM of 89.70) while maintaining highly competitive prompt adherence. This combination secures an outstanding Average Rank (2.44) among all evaluated baselines, confirming that our optimization is robust and solver-agnostic.

Comparing our LV-enhanced solvers against state-of-the-art methods on both architectures reveals a clear trade-off. Approaches that aggressively alter or bypass the inversion process (e.g., FlowEdit, RF-inversion) achieve the highest prompt adherence scores (CLIP, HPSv3). However, this comes at the critical cost of losing the source image context, leading to severe background degradation and structural collapse. Conversely, highly constrained methods like UniEdit-Flow preserve structure and background exceptionally well but suffer significant drops in human-aligned editing scores, struggling to produce realistic edits. DNAEdit offers a stable middle ground, yet *LayerVerse* consistently surpasses it.

In summary, *LayerVerse* successfully bridges these extremes across MM-DiT architectures. While specialized baselines may marginally outperform it on individual metrics, our approach consistently finds an optimal "sweet spot." By delivering top-tier background and structure preservation while maintaining competitive prompt adherence and aesthetics, FireFlow + LV achieves the best overall balance among all tested approaches.

5.4 Qualitative Results

Due to space constraints, our visual comparisons are limited to showing *LayerVerse* applied to the FireFlow baseline on the FLUX.1-[dev] model. Fig. 2 compares state-of-the-art training-free editing methods across various tasks, including object addition/modification, background change, and attribute manipulation (color, material, pose). FireFlow alone often produces minimal changes,

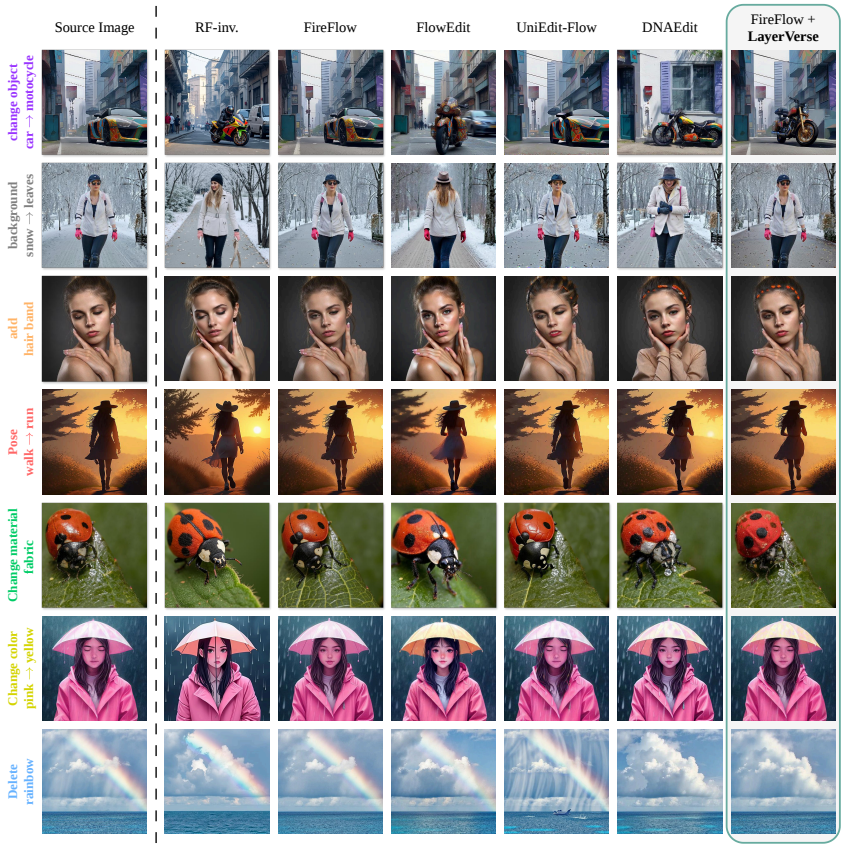

Fig. 2: Qualitative comparison on the PIE-Bench benchmark using the FLUX.1-[dev] model. *LayerVerse* acts as a powerful solver-agnostic prior. When integrated with the FireFlow baseline, it significantly improves editing quality, effectively bridging the gap between rigid source reconstruction and unconstrained structural collapse.

prioritizing rigid reconstruction over editability. In contrast, RF-inversion [26] and FlowEdit [13] exhibit unconstrained editing, leading to severe background degradation (rows 1, 2, 5), loss of critical details (rows 1, 2, 4, 5), and failure to preserve identity (rows 2, 3, 5, 6). UniEdit-Flow [10] attempts to balance new content generation with source preservation but remains overly constrained by the original structure, limiting flexibility. DNAEdit [35] yields stable results but may sacrifice identity or awkwardly alter the object's pose (rows 1–3, 5, 6).

Compared to these methods, *LayerVerse* significantly enhances the editing capabilities of the FireFlow [5] baseline while maintaining exceptional image quality. Notably, our approach successfully modifies the background (row 2) and removes the object without disrupting the remaining image layout (row 7). This demonstrates a superior balance between structural fidelity and editing freedom, visually corroborating our strong quantitative results.

Table 3: Ablation on Injection Roles. On FireFlow, Global KV causes rigid reconstruction, while Masked KV achieves high prompt adherence but suffers severe structural collapse. *LayerVerse* uniquely balances structure preservation and editability.

Injection Type	Structure	CLIP Similarity		User Alignment	
	Distance ↓	Whole ↑	Edited ↑	HPSv3 ↑	EditScore ↑
Standard FireFlow	9.48	23.68	20.45	4.88	3.59
Only Global KV	11.01	24.28	20.99	4.69	4.23
Only Masked KV	19.03	**25.03**	**22.06**	**5.64**	**5.91**
LayerVerse (Combined)	**9.36**	24.39	21.30	5.09	5.05

Table 4: Optimization vs. Random Heuristics. Comparing *LayerVerse* against the average of 14 random block configurations (using an identical layer budget). Random selection destroys structural integrity, proving exact block identity matters.

Solver	Layer Selection	Structure	CLIP Similarity		User Alignment	
		Distance ↓	Whole ↑	Edited ↑	HPSv3 ↑	EditScore ↑
FireFlow	*LayerVerse* (Ours)	**9.36**	24.39	21.30	5.09	**5.05**
	Random Average (14)	60.54	**26.07**	**23.20**	**7.38**	4.64
Euler	*LayerVerse* (Ours)	**12.24**	24.38	21.18	4.84	**5.04**
	Random Average (14)	64.16	**25.89**	**22.84**	**6.36**	4.51

6 Ablation Study

To validate our core design choices, all ablations are conducted on the FLUX.1-[dev] model. We analyze the distinct roles of masked and global injections, the necessity of rigorous optimization, the empirical importance of pairwise block interactions, and the framework's robustness to data perturbations.

Deconstructing the Injection Roles. As hypothesized in Section 3, neither purely global nor purely masked KV-injections alone suffice for high-quality editing. Isolating these components using FireFlow (Tab. 3) reveals their individual flaws. The *Standard* baseline (Fig. 3, col. 2) relies on heuristic injections, causing rigid reconstruction and poor editability (EditScore 3.59). Allocating our optimized layer budget to act *Only as Global KV* injections (col. 3) slightly improves editability, but generation remains overly constrained. Conversely, applying *Only Masked KV* injections (col. 4) protects the background but leaves the object unconstrained, artificially inflating CLIP and HPSv3 scores. This deceptive success illustrates the "Masked Inpainting Pitfall": without global anchors, the model hallucinates the target object from scratch, destroying its original layout and increasing Structure Distance to 19.03. *LayerVerse* (col. 5) elegantly resolves this tension. By combining masked injections for the background and sparse global anchors for the object, it achieves the best Structure Distance (9.36) alongside

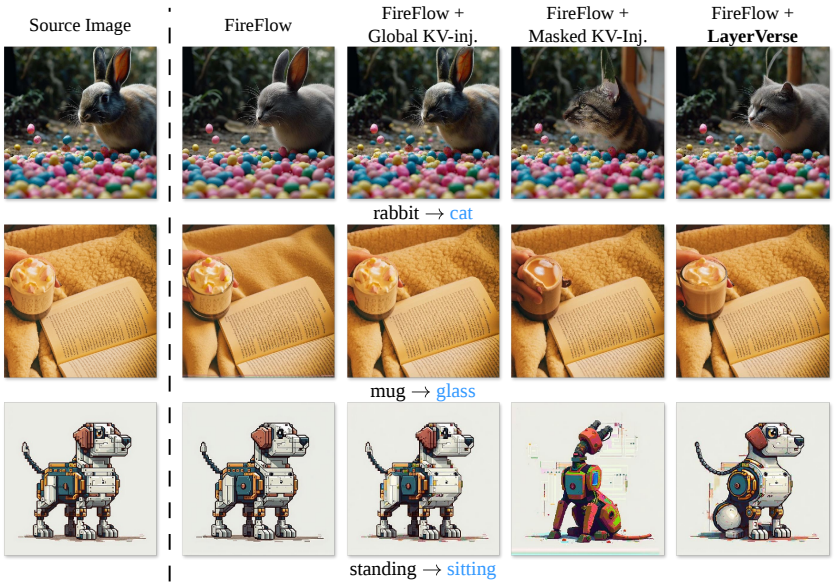

Fig. 3: Visual ablation comparing KV injection strategies. Standard FireFlow and Global KV injection fail to adhere to the text prompt. While Masked KV injection suffers from context-inconsistent inpainting artifacts, *LayerVerse* achieves an optimal balance between background preservation, target editability, and object consistency.

competitive editing scores. This "sweet spot" is visually corroborated, demonstrating that only the combined *LayerVerse* configuration successfully edits the object while seamlessly preserving its identity and the surrounding scene.

The Necessity of Optimized Selection. Could random selection achieve similar results? We evaluated 14 random block combinations on FireFlow and Euler, strictly matching our optimal budget (1 global anchor, 7 masked blocks). As shown in Tab. 4, random layer selection catastrophically fails at structure preservation, with Structure Distance exploding to 60.54 (FireFlow) and 64.16 (Euler). Lacking the ability to anchor image semantics, random blocks cause the model to ignore the source structure entirely. While these unconstrained configurations artificially inflate text-alignment scores (acting as text-to-image generation), they fail to produce coherent edits, decreasing the reasoning-based EditScore. This confirms that the exact topological identity of selected blocks is critical. Random guessing provides virtually zero structural protection, proving that optimization is required to locate the true structural bottlenecks.

Validating the Pairwise Interaction Model. Next, we justify our core assumption: accurately modeling the editing error requires capturing pairwise interactions between blocks (Eq. 1). In Fig. 4(B), we evaluate reconstruction PSNR on PIE-Bench across varying layer budgets ($N \in [2, 15]$) using a naive linear error model (which assumes independent block contributions) versus our quadratic error model. The quadratic error model consistently outperforms the linear one,

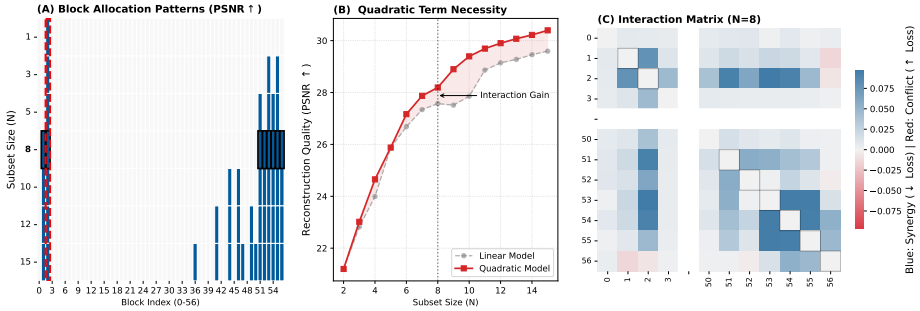

Fig. 4: Quadratic landscape analysis. (A) Evolution of optimal subsets (PSNR ↑). The distinct, scattered patterns confirm that optimal blocks are architecture-specific. (B) Reconstruction PSNR evaluated on PIE-Bench ($N \in [2, 15]$). The quadratic error model (red) outperforms the linear error model (gray), proving pairwise terms are necessary. (C) Interaction matrix of the combined model ($N = 8$) for LPIPS ↓. Dark blue cells indicate strong synergistic coupling, where pairs cooperate to reduce error.

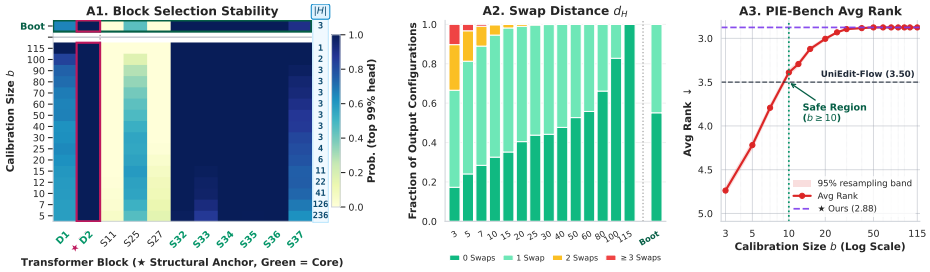

Fig. 5: Data sensitivity analysis. (A1) Block selection probability vs. calibration size b shows that core blocks are highly stable. (A2) Masked-block swap distance d_H (differing blocks) strongly concentrates on ≤ 1 swap across resampling schemes. (A3) Imputed Avg. Rank confirms that a small calibration subset ($b \geq 10$) is sufficient to maintain robust performance compared to baselines.

empirically proving that transformer blocks exhibit synergistic behaviors. This synergy is explicitly visualized in the interaction matrix (Fig. 4(C)). Dark blue cells indicate strong synergistic coupling, where pairs cooperate to reduce error. Because these dynamics are complex, optimal blocks do not stack sequentially; instead, they form distinct, scattered patterns as the budget grows (Fig. 4(A)). This non-linear behavior confirms that optimal layer selection is architecture-specific, making our MILP framework essential to bypass manual guesswork.

Data Sensitivity and Robustness. Since *LayerVerse* calibrates its layer budget on a hold-out dataset, we evaluate its stability against data perturbations under our exact layer budget ($N = 8, N_{str} = 1$) using a non-parametric bootstrap ($T = 20,000$ iterations) and sampling $T' = 3,000$ subsets per calibration size b. The optimization proves highly consistent: selection probabilities for the core structural anchor and primary masked blocks remain stable (Fig. 5 (A1)).

Table 5: Data Sensitivity on PIE-Bench. The ± interval shows the maximum metric deviation across the ≥ 99.9%-mass candidate pool from the non-parametric bootstrap, demonstrating consistent downstream quality. Against Tab. 1 baselines, the optimal Avg. Rank remains stable (±0.00), proving data invariance.

Method	Structure	Background Preservation				CLIP Similarity		User Alignment		Rank
	Distance ↓	PSNR ↑	LPIPS ↓	MSE ↓	SSIM ↑	Whole ↑	Edited ↑	HPSv3 ↑	EditScore ↑	Avg. ↓
UniEdit-Flow	8.82	29.74	55.35	18.16	91.12	24.59	21.29	4.27	4.73	3.50
FireFlow + LV	9.21±0.37	30.29±0.21	36.39±2.08	17.64±0.56	92.77±0.19	24.39±0.01	21.28±0.03	5.10±0.03	5.01±0.06	**2.88**±0.00

With H defined as the set of active masked blocks, the swap distance d_H (Hamming distance to the optimal set) strongly concentrates on ≤ 1 swap across all resampling schemes (Fig. 5 (A2)). Crucially, this internal consistency guarantees robust downstream performance. Evaluating candidate configurations that account for $\geq 99.9\%$ of the empirical mass from the total candidate pool generated during the bootstrap, the optimal PIE-Bench average rank (2.88, against Tab. 1 baselines) is exactly preserved (Tab. 5). Furthermore, imputed average rank analysis confirms that a reduced calibration size ($b \geq 10$) maintains competitive performance compared to baselines (Fig. 5 (A3)). This proves *LayerVerse* provides a robust solver-agnostic prior without overfitting.

7 Conclusion

In this work, we successfully addressed the fundamental trade-off between background fidelity, structural consistency, and text-guided editability in training-free image editing with MM-DiTs. We introduced *LayerVerse*, an optimization framework that resolves this dilemma by explicitly decoupling KV-injection into two distinct roles: Masked Injections to preserve the background, and sparse Global Injections to act as structural anchors. By modeling the expected editing error through pairwise layer interactions and solving the combinatorial problem via Mixed-Integer Linear Programming (MILP), *LayerVerse* identifies the optimal, architecture-specific layer configuration without exhaustive search. Coupled with a lightweight, single-step automatic mask serving as an auxiliary spatial guide, this configuration acts as a robust, solver-agnostic prior. Experiments demonstrate that integrating *LayerVerse* elevates existing inversion-based image editing baselines, achieving highly competitive results on the PIE-Bench benchmark.
Limitations. Our method is not intended for style transfer tasks, which require global semantic modifications rather than localized editing. Since *LayerVerse* explicitly relies on a preserved background to guide masked injections, this aligns with our core focus on the fidelity-editability trade-off in object-centric edits.

Acknowledgements

The work of Konstantin Sobolev was supported by the Ministry of Economic Development of the Russian Federation in accordance with the subsidy agreement (agreement identifier 000000C313925P4H0002; grant No 139-15-2025-012).

References

1. Avrahami, O., Patashnik, O., Fried, O., Nemchinov, E., Aberman, K., Lischinski, D., Cohen-Or, D.: Stable flow: Vital layers for training-free image editing. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 7877–7888 (2025)
2. Black Forest Labs: Flux. `https://github.com/black-forest-labs/flux` (2024), accessed: 2025-09-15
3. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18392–18402 (2023)
4. Cao, M., Wang, X., Qi, Z., Shan, Y., Qie, X., Zheng, Y.: Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 22560–22570 (2023)
5. Deng, Y., He, X., Mei, C., Wang, P., Tang, F.: Fireflow: Fast inversion of rectified flow for image semantic editing. In: Forty-second International Conference on Machine Learning (2025)
6. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning (2024)
7. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
8. Hu, Y., Peng, J., Lin, Y., Liu, T., Qu, X., Liu, L., Zhao, Y., Wei, Y.: Dcedit: Dual-level controlled image editing via precisely localized semantics. arXiv preprint arXiv:2503.16795 (2025)
9. Huynh-Thu, Q., Ghanbari, M.: Scope of validity of psnr in image/video quality assessment. Electronics letters **44**(13), 800–801 (2008)
10. Jiao, G., Huang, B., Wang, K.C., Liao, R.: Uniedit-flow: Unleashing inversion and editing in the era of flow models. In: The Fourteenth International Conference on Learning Representations (2026)
11. Ju, X., Zeng, A., Bian, Y., Liu, S., Xu, Q.: Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In: The Twelfth International Conference on Learning Representations (2024)
12. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollar, P., Girshick, R.: Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4015–4026 (October 2023)
13. Kulikov, V., Kleiner, M., Huberman-Spiegelglas, I., Michaeli, T.: Flowedit: Inversion-free text-based editing using pre-trained flow models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19721–19730 (2025)
14. Labs, B.F., Batifol, S., Blattmann, A., Boesel, F., Consul, S., Diagne, C., Dockhorn, T., English, J., English, Z., Esser, P., et al.: Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. arXiv preprint arXiv:2506.15742 (2025)
15. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)

16. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: The Eleventh International Conference on Learning Representations (2023)
17. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: The Eleventh International Conference on Learning Representations (2023)
18. Long, Z., Zheng, M., Feng, K., Zhang, X., Liu, H., Yang, H., Zhang, L., Chen, Q., Ma, Y.: Follow-your-shape: Shape-aware image editing via trajectory-guided region control. In: The Fourteenth International Conference on Learning Representations (2026)
19. Luo, X., Wang, J., Wu, C., Xiao, S., Jiang, X., Lian, D., Zhang, J., Liu, D., liu, Z.: Editscore: Unlocking online rl for image editing via high-fidelity reward modeling. In: The Fourteenth International Conference on Learning Representations (2026)
20. Ma, Y., Wu, X., Sun, K., Li, H.: Hpsv3: Towards wide-spectrum human preference score. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15086–15095 (2025)
21. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics **9**(1), 62–66 (1979). https://doi.org/10.1109/TSMC.1979.4310076
22. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4195–4205 (2023)
23. Pope, R., Douglas, S., Chowdhery, A., Devlin, J., Bradbury, J., Heek, J., Xiao, K., Agrawal, S., Dean, J.: Efficiently scaling transformer inference. Proceedings of machine learning and systems **5**, 606–624 (2023)
24. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
25. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
26. Rout, L., Chen, Y., Ruiz, N., Caramanis, C., Shakkottai, S., Chu, W.: Semantic image inversion and editing using rectified stochastic differential equations. In: The Thirteenth International Conference on Learning Representations (2025)
27. Shen, T., Huang, Z., Li, X., Lin, Z., Liu, J., Wang, Y., Feng, J., Yang, M.H., Liew, J.H.: Qk-edit: Revisiting attention-based injection in mm-dit for image and video editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 19043–19053 (October 2025)
28. Shin, J., Hwang, A., Kim, Y., Kim, D., Park, J.: Exploring multimodal diffusion transformers for enhanced prompt-based image editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19492–19502 (2025)
29. Tumanyan, N., Bar-Tal, O., Bagon, S., Dekel, T.: Splicing vit features for semantic appearance transfer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10748–10757 (2022)
30. Wang, J., Pu, J., Qi, Z., Guo, J., Ma, Y., Huang, N., Chen, Y., Li, X., Shan, Y.: Taming rectified flow for inversion and editing. In: Forty-second International Conference on Machine Learning (2025)
31. Wang, Z.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
32. Wolsey, L.A.: Integer programming. John Wiley & Sons (2020)

33. Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., Yin, S.m., Bai, S., Xu, X., Chen, Y., et al.: Qwen-image technical report. arXiv preprint arXiv:2508.02324 (2025)
34. Xiao, G., Tian, Y., Chen, B., Han, S., Lewis, M.: Efficient streaming language models with attention sinks. In: The Twelfth International Conference on Learning Representations (2024)
35. Xie, C., Li, M., Li, S., Wu, Y., Yi, Q., Zhang, L.: Dnaedit: Direct noise alignment for text-guided rectified flow editing. Advances in Neural Information Processing Systems **38**, 124456–124474 (2026)
36. Xu, P., Jiang, B., Hu, X., Luo, D., He, Q., Zhang, J., Wang, C., Wu, Y., Ling, C., Wang, B.: Unveil inversion and invariance in flow transformer for versatile image editing. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 28479–28489 (2025)
37. Yang, K., Li, X., Li, Y., Li, Q., Wang, Z.: Fsi-edit: Frequency and stochasticity injection for flexible diffusion-based image editing. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems (2025)
38. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)
39. Zhu, T., Zhang, S., Shao, J., Tang, Y.: Kv-edit: Training-free image editing for precise background preservation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 16607–16617 (2025)