FSI-Edit: Frequency and Stochasticity Injection for Flexible Diffusion-Based Image Editing

Kaixiang Yang[†], **Xin Li**[†], **Yuxi Li**[†], **Qiang Li, Zhiwei Wang**[*]
Wuhan National Laboratory for Optoelectronics, Huazhong University of Science and Technology
[†]: Co-first authors, [*]: Corresponding author.
{kxyang, lixin2023, liyuxi9, liqiang8, zwwang}@hust.edu.cn

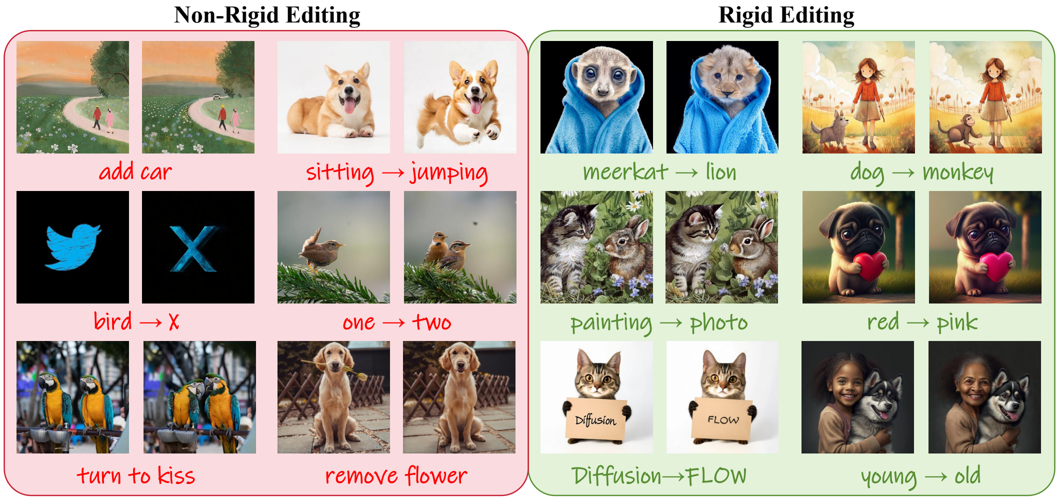
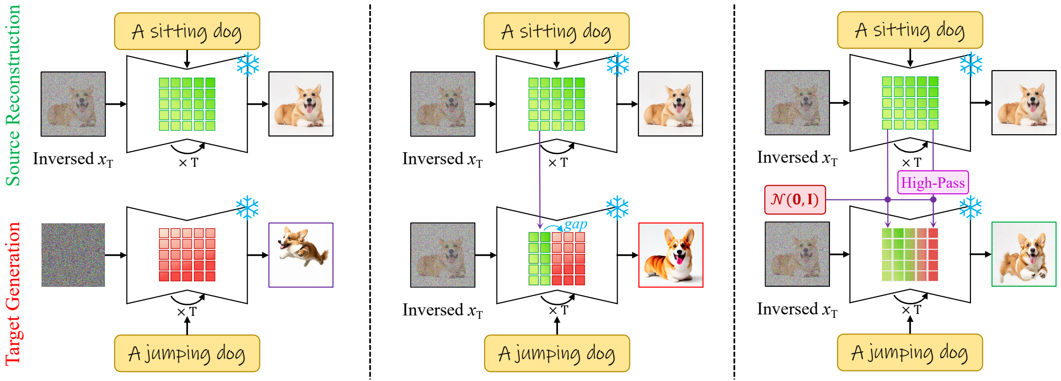


Figure 1: Our method is capable of handling non-rigid editing, including pose changes, object addition, and removal, as well as rigid editing.

Abstract

Latent Diffusion-based Text-to-Image (T2I) is a free image editing tool that typically reverses an image into noise, reconstructs it using its original text prompt, and then generates an edited version under a new target prompt. To preserve unaltered image content, features from the reconstruction are directly injected to replace selected features in the generation. However, this direct replacement often leads to feature incompatibility, compromising editing fidelity and limiting creative flexibility, particularly for non-rigid edits (*e.g.*, structural or pose changes). In this paper, we aim to address these limitations by proposing **FSI-Edit**, a novel framework using frequency- and stochasticity-based feature injection for flexible image editing. First, FSI-Edit enhances feature consistency by injecting *high-frequency* components of reconstruction features into generation features, mitigating incompatibility while preserving the editing ability for major structures encoded in low-frequency information. Second, it introduces controlled *noise* into the replaced reconstruction features, expanding the generative space to enable diverse non-rigid edits beyond the original image's constraints. Experiments on non-rigid edits, *e.g.*, addition, deletion, and pose manipulation, demonstrate that FSI-Edit outperforms existing baselines in target alignment, semantic fidelity and visual quality. Our work highlights the critical roles of frequency-aware design and stochasticity in overcoming rigidity in diffusion-based editing.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).



(a) Pure T2I (b) Existing Image Editing (c) Ours

Figure 2: Editing Paradigm Comparison. Top: source reconstruction; Bottom: target generation. (a) Pure T2I: there is no interaction between source and target images, results are random and unrelated to the source. (b) Existing Image Editing: typical editing methods directly inject features, causing semantic gaps and low flexibility, especially for non-rigid edits. (c) Ours: injects only high-frequency residuals and adds stochasticity, reducing semantic gap and improving edit quality.

1 Introduction

Diffusion models [1] have achieved remarkable success in the domain of Text-to-Image (T2I) generation in recent advances [2–7]. In particular, Latent Diffusion Models (LDMs) demonstrate exceptional ability to translate textual descriptions (*i.e.,* prompts) into high-quality images, leading to their widespread adoption in downstream applications such as image [8, 9] and video editing [10, 11]. Image editing refers to the task of transforming a source image into a desired target image guided by user-provided prompts. This technology has become an integral part of daily life, with broad applications across domains such as social media and visual effects.

The typical workflow of LDM-based image editing follows a common paradigm. First, the source image is inverted back into a latent noise representation, typically through DDIM inversion [12] or more advanced strategies [13–15]. From this latent noise, two parallel denoising processes are initiated: one reconstructs the original image conditioned on the source prompt, while the other generates a modified version guided by a new target prompt. A key challenge is balancing the preservation of the original content with the incorporation of desired semantic changes, which is typically achieved via information replacement from the reconstruction stream to the generation stream.

A simple way is partially blending the latent noise of the source image with random noise at the beginning of the generation process, often in the frequency domain [16]. Recent methods [8, 9, 17–20] have further explored intermediate feature-level replacement, as shown in Figure 2b. They focus primarily on manipulating query, key, and value features within the self-attention or cross-attention layers, enabling more precise and controllable editing. Specifically, during target generation, selected attention layers are identified, and a subset of their features (*e.g.,* query, key, or value) is directly replaced with corresponding features from the reconstruction stream. This provides visual guidance for preserving image regions that are not intended to be edited.

While existing methods perform well in rigid editing tasks, non-rigid editing such as object addition, removal, or significant pose alteration often requires substantial modifications not only to the target object but also to its surrounding context. In such cases, the editing region naturally extends beyond the object itself, and the direct feature replacement widely-employed in most methods faces two major limitations. **First**, a *semantic gap* exists between the reconstruction and generation processes. Injecting attention features from the reconstruction branch often does not faithfully preserve the original content and can introduce artifacts or structural distortions, especially when editing involves significant pose or viewpoint changes. **Second**, non-rigid edits require fully leveraging the *generative capacity* of the base model. Semantic guidance alone is often insufficient to overcome the strong inductive bias of the original image, which restricts the model's ability to perform flexible and meaningful content transformations. As a result, existing approaches struggle to achieve both content fidelity and effective structural reshaping in non-rigid editing tasks.

To address these issues, we propose **FSI-Edit**, a novel tuning-free framework that enables more flexible and effective image editing through frequency- and stochasticity-based feature injection, as shown in Figure 2c. To bridge the semantic gap between the reconstruction and generation features, we introduce a *frequency residual fusion* mechanism. It allows FSI-Edit to selectively inject only the high-frequency components from reconstruction features into the generation stream. This preserves fine-grained textures from the original image while avoiding interference from low-frequency structural information, thereby supporting more expressive non-rigid edits. Furthermore, to fully activate the generative capacity of the base diffusion model, we incorporate *stochastic noise injection* during generation. This controlled perturbation enriches the latent space and empowers the model to perform substantial structural changes while maintaining semantic alignment.

We summarize our contributions as follows:

- 1) We identify and address two core limitations of most current image editing methods for non-rigid edits, *e.g.,* semantic inconsistency between reconstruction and generation features, and insufficient generative flexibility due to strong image priors.
- 2) We propose FSI-Edit, featuring a new frequency residual fusion module that selectively transfers high-frequency details for more accurate feature alignment, and a stochastic noise injection strategy that expands the generation space to enable more precise and flexible structural transformations.
- 3) We conduct extensive experiments on PIE-Bench benchmark, and the comparison results demonstrate that FSI-Edit significantly outperforms existing methods on both rigid and non-rigid editing tasks, confirming its effectiveness and generalizability across diverse editing scenarios.

2 Related Works

2.1 Image Editing with Feature Replacement

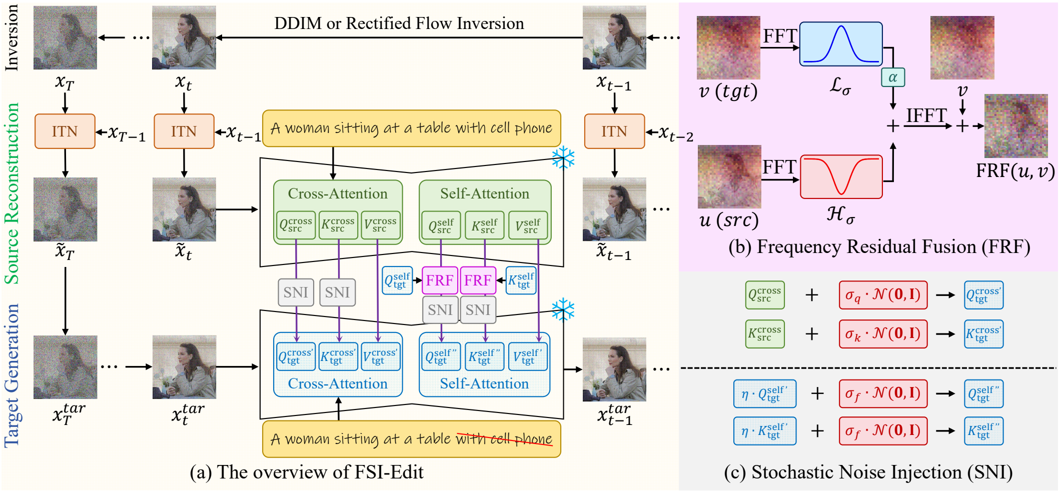
The field of text-to-image (T2I) generation has advanced rapidly in recent years, driven by powerful models such as DALL-E [21, 3, 22], Imagen [4], Stable Diffusion [5], and Diffusion Transformer (DiT) [7]. These models have enabled a wide range of applications, among which image editing has emerged as a prominent research direction. Early editing approaches [17, 23–26] typically rely on fine-tuning the generative model on the source image using editing instructions. However, such methods often suffer from limited generalization and require additional training overhead.

To improve flexibility and generalization, a line of tuning-free methods has been proposed. These approaches commonly exploit feature interaction between the source image reconstruction and the target image generation. For example, Prompt-to-Prompt (P2P) [8] introduces cross-attention feature replacement to modify image content according to textual prompts while preserving background fidelity. Plug-and-Play (PnP) [9] further explores the effects of feature replacement at various layers, including residual and attention blocks. While effective for rigid edits, these methods struggle with non-rigid transformations such as pose or structure changes. MasaCtrl [19] improves on this by using attention-based masks to localize and control editing regions, enabling more flexible manipulation within the self-attention mechanism. In parallel, the Transformer-based DiT architecture [7, 27] has opened new opportunities for editing [28, 29]. Methods like FT-Edit [30], RF-Edit [31], and Fireflow [32] focus on improving inversion and editing precision. DCEdit [33] utilizes attention-based localization to guide prompt-consistent modifications.

Despite their progress, most of these approaches rely on direct feature replacement from the source branch, which can introduce semantic mismatches, artifacts, or structural distortions, especially in non-rigid edits involving significant content change. Furthermore, the reused features often carry a strong prior from the source image, limiting the model's generative flexibility and editing expressiveness. To overcome these limitations, we draw inspiration from frequency-domain analysis. Rather than injecting full features, we selectively transfer only high-frequency residuals from the source, preserving fine-grained textures while minimizing semantic conflicts. Additionally, we introduce stochastic noise injection during generation to expand the latent space, improving the model's adaptability to diverse and complex edits.

2.2 Image Editing with Latent Frequency Processing

Recent works have begun exploring frequency-domain operations within the latent space of diffusion models to facilitate more controllable image editing. FlexiEdit [16] suppresses high-frequency components in the DDIM latent corresponding to editable regions, employs a re-inversion mechanism,


Inversion
Source Reconstruction
Target Generation
DDIM or Rectified Flow Inversion
x_T
x_t
x_{t-1}
ITN
x_{T-1}
x_{t-1}
x_{t-2}
A woman sitting at a table with cell phone
Cross-Attention
Self-Attention
Q^{cross}_{src} K^{cross}_{src} V^{cross}_{src}
Q^{self}_{src} K^{self}_{src} V^{self}_{src}
$\tilde{x}_T$
$\tilde{x}_t$
$\tilde{x}_{t-1}$
SNI
FRF
Q^{self}_{tgt}
K^{self}_{tgt}
$Q^{cross'}_{tgt}$ $K^{cross'}_{tgt}$ V^{cross}_{tgt}
$Q^{self''}_{tgt}$ $K^{self''}_{tgt}$ $V^{self'}_{tgt}$
x^{tar}_T
x^{tar}_t
x^{tar}_{t-1}
A woman sitting at a table ~~with cell phone~~
FFT
v (tgt)
$\mathcal{L}_\sigma$
α
IFFT
v
FRF(u, v)
u (src)
$\mathcal{H}_\sigma$
Q^{cross}_{src} + $\sigma_q \cdot \mathcal{N}(\mathbf{0}, \mathbf{I})$ → $Q^{cross'}_{tgt}$
K^{cross}_{src} + $\sigma_k \cdot \mathcal{N}(\mathbf{0}, \mathbf{I})$ → $K^{cross'}_{tgt}$
$\eta \cdot Q^{self'}_{tgt}$ + $\sigma_f \cdot \mathcal{N}(\mathbf{0}, \mathbf{I})$ → $Q^{self''}_{tgt}$
$\eta \cdot K^{self'}_{tgt}$ + $\sigma_f \cdot \mathcal{N}(\mathbf{0}, \mathbf{I})$ → $K^{self''}_{tgt}$

(a) The overview of FSI-Edit

(b) Frequency Residual Fusion (FRF)

(c) Stochastic Noise Injection (SNI)

Figure 3: Overview of FSI-Edit. (a) The full pipeline of FSI-Edit, which consists of three core modules applied in sequence: Frequency Residual Fusion (FRF), Stochastic Noise Injection (SNI), and Inversion Trajectory Navigation (ITN). (b) The structure of FRF, which injects high-frequency source features with low-frequency target features within the self-attention blocks. (c) The SNI module, which injects controlled noise to enhance generative diversity and support non-rigid edits.

and combines attention feature replacement to support non-rigid edits. FreeDiff [34] performs staged frequency-domain filtering during classifier-free guidance (CFG) [35], focusing on attenuating high-frequency noise. However, it requires extensive manual tuning of parameters for each editing scenario. FDS [36] utilizes wavelet decomposition to adaptively select frequency bands based on the editing type, while Gao *et al.* [37] propose a frequency-aware ControlNet module to improve editing controllability.

While these approaches demonstrate the potential of frequency-based manipulation, they operate exclusively in the latent feature space, limiting their ability to address semantic discrepancies between the source and target features. Additionally, their reliance on per-edit parameter tuning hampers practicality and scalability. In contrast, we introduce frequency-domain processing directly at the feature level for the first time. Our method effectively bridges the semantic gap between reconstruction and generation branches without requiring model fine-tuning or laborious hyperparameter adjustment. It generalizes well across both LDM and DiT backbones, offering a robust and efficient solution for high-quality, flexible image editing.

3 Method

As shown in Figure 3, FSI-Edit comprises three core components: (1) Frequency Residual Fusion (FRF), (2) Stochastic Noise Injection (SNI), and (3) Inversion Trajectory Navigation (ITN). It builds upon DDIM inversion, rectified flow, and classifier-free guidance (CFG), which are briefly summarized in the Appendix along with architectural details of both FSI-Edit-LDM and FSI-Edit-DiT.

FRF and SNI are applied during the early denoising steps. FRF operates in *self-attention* layers by selectively fusing high-frequency components from the attention triplet features, *i.e.,* query, key, and value, of the source and target branches. This enhances detail preservation while mitigating semantic inconsistencies. SNI is applied to both *self-attention* and *cross-attention* layers, injecting controlled stochasticity into the attention features to promote diversity and enable more flexible, expressive edits. We denote attention features as $\{Q, K, V\}$. Superscripts `self` and `cross` indicate features from self-attention and cross-attention layers, respectively, while subscripts `src` and `tgt` specify whether the features originate from the source reconstruction or target generation stream. Timestep t and layer l indices are omitted for clarity. Implementation details are provided in the Appendix.

3.1 Frequency Residual Fusion for Bridging Feature Discrepancy

A central challenge in image editing is reconciling the semantic and structural gap between the source and target representations. This gap becomes especially pronounced in non-rigid edits, where the target must adhere to the prompt while preserving visual consistency with the source.

To address this, we propose Frequency Residual Fusion (FRF), a feature-level harmonization strategy that blends source and target attention features in the frequency domain. Our key idea is to preserve the target's low-frequency components, which encode global layout and structure, while injecting only the high-frequency residuals from the source to retain fine-grained texture and identity.

Let $u \in \mathbb{R}^{C\times H\times W}$ denote features from the source branch and $v \in \mathbb{R}^{C\times H\times W}$ from the target. We construct a frequency-domain fusion $\mathcal{F}_{\text{fuse}}$ via:

$$\mathcal{F}_{\text{fuse}} = \mathcal{H}_\sigma \cdot \texttt{FFT}(u) + \alpha \cdot \mathcal{L}_\sigma \cdot \texttt{FFT}(v), \tag{1}$$

where $\texttt{FFT}(\cdot)$ denotes 2D Fast Fourier Transform, $\mathcal{H}_\sigma$, $\mathcal{L}_\sigma$ are Gaussian high-pass and low-pass filters with scaling coefficient $\sigma = 0.3$, and fusion weight α is set to 0.2. The fused feature is then mapped back to the spatial domain using inverse FFT $\texttt{IFFT}(\cdot)$, and combined residually:

$$\texttt{FRF}(u, v) = \texttt{IFFT}(\mathcal{F}_{\text{fuse}}) + v. \tag{2}$$

FRF is applied to the Query and Key projections in the self-attention layers to harmonize attention computation:

$$Q_{\texttt{tgt}}^{\texttt{self}'} = \texttt{FRF}(Q_{\texttt{src}}^{\texttt{self}}, Q_{\texttt{tgt}}^{\texttt{self}}), \quad K_{\texttt{tgt}}^{\texttt{self}'} = \texttt{FRF}(K_{\texttt{src}}^{\texttt{self}}, K_{\texttt{tgt}}^{\texttt{self}}), \quad V_{\texttt{tgt}}^{\texttt{self}'} = V_{\texttt{src}}^{\texttt{self}}. \tag{3}$$

This selective frequency blending helps narrow the source–target feature gap, enabling faithful yet flexible edits across diverse content while avoiding semantic drift or prompt misalignment.

3.2 Stochastic Noise Injection for Generative Flexibility

Non-rigid editing, especially when involving large structural shifts (*e.g.*, transforming a bird into an "X" shape), requires sufficient generative flexibility. However, existing methods often underutilize this flexibility, resulting in limited editability.

Inspired by Stochastic DDIM [1, 12], which shows that added noise improves inversion diversity, we introduce controlled stochasticity into attention features to enhance expressiveness and support structural transformation. Specifically, we inject Gaussian noise into the Query and Key of the cross-attention layers, while preserving semantic consistency via source-based Value injection:

$$Q_{\texttt{tgt}}^{\texttt{cross}'} = Q_{\texttt{src}}^{\texttt{cross}} + \sigma_q \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad K_{\texttt{tgt}}^{\texttt{cross}'} = K_{\texttt{src}}^{\texttt{cross}} + \sigma_k \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad V_{\texttt{tgt}}^{\texttt{cross}'} = V_{\texttt{src}}^{\texttt{cross}}. \tag{4}$$

We further inject noise into frequency-fused Query and Key in self-attention:

$$Q_{\texttt{tgt}}^{\texttt{self}''} = \eta \cdot Q_{\texttt{tgt}}^{\texttt{self}'} + \sigma_f \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad K_{\texttt{tgt}}^{\texttt{self}''} = \eta \cdot K_{\texttt{tgt}}^{\texttt{self}'} + \sigma_f \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{5}$$

where $\eta = 0.2$, $\sigma_q = \sigma_k = 0.1$, and $\sigma_f = 0.8$ by default.

Since attention scores depend on relative dot-product magnitudes such as $QK^\top$, injecting bounded noise into frequency-fused features does not affect numerical stability.

This stochasticity allows the model to escape source constraints and adapt to diverse, complex edit patterns. For rigid edits, we anneal σ_q, σ_k, σ_f to 0 and set $\eta \to 1$ to revert to deterministic attention for precise structure preservation.

3.3 Inversion Trajectory Navigation for Source Trajectory Refinement

Prior works often directly replace latent features during inversion, which can disrupt structural integrity and cause inconsistencies in the reconstructed source. In Sections 3.1 and 3.2, we introduced frequency residual fusion and stochastic noise injection at the feature level to reduce semantic gaps and enrich generative diversity. Here, we extend these ideas to the latent space by applying a similar fusion strategy during inversion.

During source image inversion (via DDIM [12] or Rectified Flow [28, 29, 38]), we obtain a sequence of latent states $\{x_T, x_{T-1}, \dots, x_1\}$. Then, we construct a refined sequence $\{\tilde{x}_T, \tilde{x}_{T-1}, \dots, \tilde{x}_1\}$ for source image reconstruction, where each latent $\tilde{x}_t$ is obtained by blending the high-frequency components of x_t with the low-frequency components of x_{t-1}:

$$\tilde{x}_t = \texttt{IFFT}\big(\mathcal{H}_\sigma \cdot \texttt{FFT}(x_t) + \mathcal{L}_\sigma \cdot \texttt{FFT}(x_{t-1})\big) + \sigma_x \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{6}$$

where $\sigma = 0.3$, and $\sigma_x = 1e{-}3$ controls the level of injected noise.

The resulting $\{\tilde{x}_t\}$ is used solely to extract source-side features for cross-branch interactions. We initialize both source and target branches with $\tilde{x}_T$. This operation stabilizes the global structure of the source while retaining fine-grained details and introducing slight stochasticity for flexibility.

4 Experiment

4.1 Experiment Design

Dataset and Baselines. To comprehensively evaluate our method on both rigid and non-rigid editing tasks, we conduct experiments on the PIE-Bench [15] benchmark, which contains 700 image-prompt pairs across 10 diverse editing categories. For assessing non-rigid editing performance specifically, such as object addition, deletion, and pose modification, we additionally curate a subset of 300 samples from PIE-Bench that emphasize such editing. Our comparisons include LDM-based methods (*P2P* [8], *PnP* [9], *MasaCtrl* [19], *FlexiEdit* [16], *FreeDiff* [34]) and DiT-based approaches (*RF-Inv* [39], *StableFlow* [40], *RF-Edit* [31], *DCEdit* [33]). All models are tested using their publicly available implementations and default configurations for fair comparison.

Metrics. To holistically assess editing performance and background preservation of different methods, we employ six complementary metrics. Structure Distance [41] measure the structural similarity between edited images and original images, while PSNR, LPIPS [42], MSE and SSIM [43] collectively evaluate content preservation in unedited regions. For text-image consistency, we compute CLIP similarity [44] over both the entire image and the edited region. The dataset-provided masks are used to identify the edited regions, but only during evaluation.

Implementation Details. All experiments for FSI-Edit-LDM were conducted using Latent Diffusion Model (LDM) [5] v1.5, while FSI-Edit-DiT was built upon DiT [7] v3.5-Medium. We use 50 DDIM steps for inversion, with a CFG scale of 1. During generation, the target branch uses a CFG scale of 7.5. The same settings are applied across both backbones. All experiments were run on a single NVIDIA RTX 4090 GPU with 17 GB memory usage. The full editing pipeline, including inversion and generation, takes 20 seconds per image. Our code is available at `https://github.com/kk42yy/FSI-Edit`.

4.2 Comparisons on Diverse Editing Types

Experimental results on the PIE-Bench are summarized in Table 1. Visual comparisons are shown in Figure 4 and Figure 5. These results are generated using our DiT-based version of FSI-Edit, additional examples and results for the LDM-based variant can be found in the Appendix. Across all editing categories, our method achieves a better balance between semantic alignment and content preservation, effectively maintaining the appearance of unedited regions while accurately applying edits. In contrast, existing methods often lean toward one end of this trade-off. For example, P2P tends to preserve background well but struggles with semantic modification. Our approach handles both aspects simultaneously, demonstrating stronger generalization, especially in non-rigid tasks. Among the DiT-based baselines, FSI-Edit-DiT achieves competitive performance using only the lightweight DiT v3.5-Medium model, running on a single RTX 4090 GPU with clear visual improvements over other approaches.

Table 1: Quantitative comparisons on PIE-Bench [15]. P2P [8] achieves the best performance in preserving background and structure, ours attains the highest CLIP similarity score while maintaining competitive background preservation.

Method	Model	Structure	Background Preservation				CLIP Similarity	
		$Distance_{\times 10^3} \downarrow$	$PSNR \uparrow$	$LPIPS_{\times 10^3} \downarrow$	$MSE_{\times 10^4} \downarrow$	$SSIM_{\times 10^2} \uparrow$	$Whole \uparrow$	$Edited \uparrow$
P2P [8]	UNet	**11.65**	**27.22**	**54.55**	**32.86**	**84.76**	25.02	22.10
PnP [9]		24.29	22.46	106.06	80.45	79.68	25.41	22.62
MasaCtrl [19]		24.70	22.64	87.94	81.09	81.33	24.38	21.35
FlexiEdit [16]		22.13	25.74	80.45	58.45	82.62	25.15	**22.87**
FreeDiff [34]		18.70	24.73	89.76	55.32	81.68	25.03	22.12
Ours-LDM		15.84	24.69	88.42	52.21	81.93	**25.46**	22.30
RF-Inv [39]	Transformer	48.76	19.51	195.85	155.74	68.95	25.11	22.50
StableFlow [40]		19.24	23.04	**76.94**	84.85	**87.22**	24.30	21.28
RF-Edit [31]		24.45	24.41	113.44	56.46	83.84	25.03	22.28
DCEdit [33]		22.36	25.41	94.17	48.09	85.60	25.47	**22.71**
Ours-DiT*		**13.71**	**26.61**	85.44	**36.50**	86.25	**25.69**	22.50

* Ours-DiT is built on DiT v3.5-Medium and runs fully on a single RTX 4090 GPU. In contrast, RF-Inv, StableFlow, and RF-Edit rely on FLUX [45] or its variants, which require over 40GB of memory.

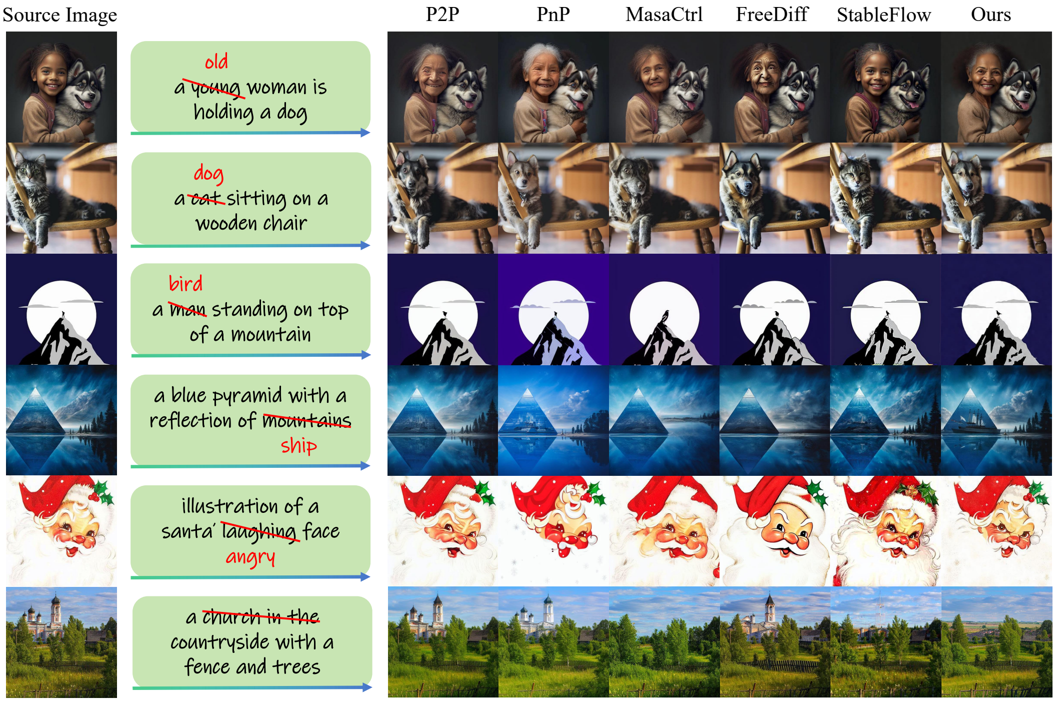

Source Image
P2P
PnP
MasaCtrl
FreeDiff
StableFlow
Ours
old
a ~~young~~ woman is holding a dog
dog
a ~~cat~~ sitting on a wooden chair
bird
a ~~man~~ standing on top of a mountain
a blue pyramid with a reflection of ~~mountains~~
ship
illustration of a santa' ~~laughing~~ face
angry
a ~~church in the~~ countryside with a fence and trees

Figure 4: Editing Results on PIE-Bench. Compared to previous methods, our approach better preserves unedited regions, maintaining high consistency with the original image while accurately reflecting the target semantics.

4.3 Comparisons on Non-Rigid Editing.

To evaluate non-rigid image editing, which involves structural changes such as addition, deletion, or pose modification, we selected nearly 300 images from the PIE-Bench dataset and constructed corresponding editing prompts. P2P [8] was excluded from the comparison as it is not well-suited for non-rigid editing tasks, whereas DCEdit [33] was excluded, since its official implementation had not been released when our experiments were conducted. Quantitative results are reported in Table 2, while qualitative results are visualized in Figure 5. Due to space constraints, the qualitative results of other methods are provided in Appendix.

Our method consistently delivers high-quality non-rigid edits by accurately aligning with the target prompts while preserving the integrity of unedited regions. For example, in Figure 5 (row 3), our model successfully transforms the bird's shape into an "X" form. In the fifth row, when removing the phone, our method preserves surrounding details such as the vase on the table and the watch on the left wrist. Overall, the effectiveness of our non-rigid editing benefits from the proposed FRF and SNI, which together enhance both semantic alignment and structural fidelity. This allows for precise yet flexible editing while maintaining high realism in unedited areas.

Table 2: Quantitative evaluation of non-rigid editing performance. Non-rigid editing requires substantial modifications to the original image. Our method achieves the highest CLIP similarity score while also maintaining strong background preservation.

Method	Model	Structure	Background Preservation				CLIP Similarity	
		$Distance_{\times 10^3} \downarrow$	$PSNR \uparrow$	$LPIPS_{\times 10^3} \downarrow$	$MSE_{\times 10^4} \downarrow$	$SSIM_{\times 10^2} \uparrow$	$Whole \uparrow$	$Edited \uparrow$
PnP [9]	UNet	19.23	22.17	115.86	87.83	77.47	25.10	21.32
MasaCtrl [19]		22.97	22.32	**94.62**	87.67	**79.40**	24.86	20.68
FlexiEdit [16]		76.91	16.83	236.02	299.58	65.42	24.36	20.70
FreeDiff [34]		**18.06**	**23.92**	101.49	**64.04**	78.96	25.29	21.25
Ours-LDM		18.75	23.22	108.55	70.31	78.31	**25.93**	**21.78**
RF-Inv [39]	Transformer	54.27	18.65	253.04	188.62	62.41	26.08	22.03
StableFlow [40]		**19.58**	22.40	**85.63**	98.60	**85.19**	24.93	20.84
RF-Edit [31]		25.36	23.65	128.30	64.37	81.43	25.06	21.58
Ours-DiT		19.87	**23.91**	120.82	**59.95**	81.71	**26.58**	**22.16**

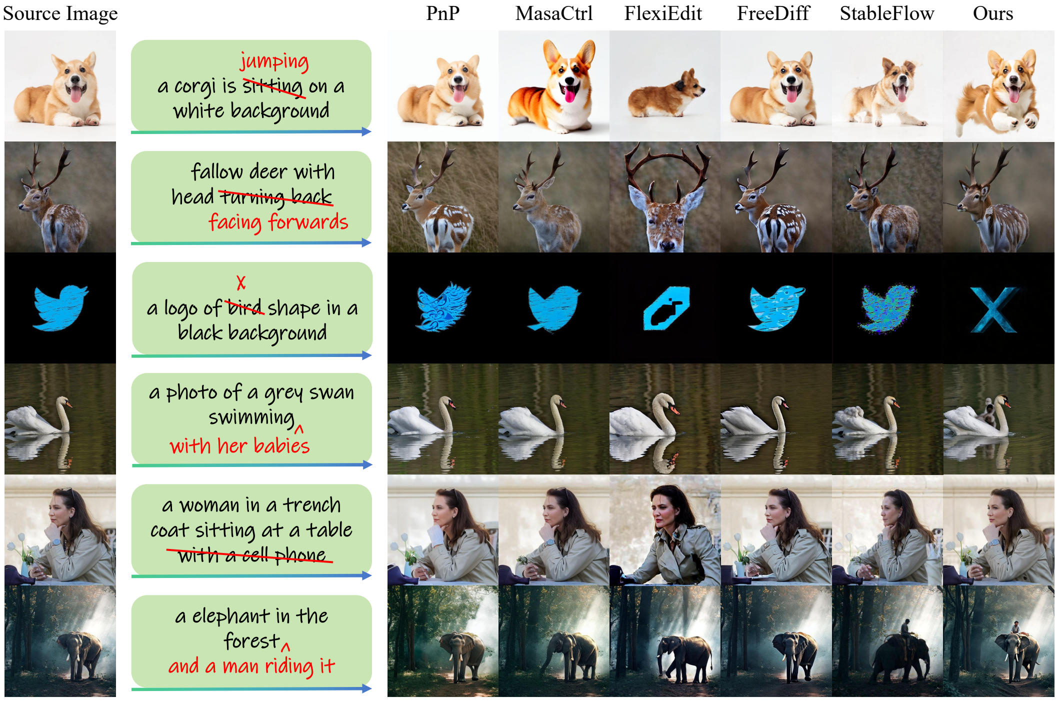


Figure 5: Non-rigid Editing Results. Our method strikes a superior balance between semantic alignment and background preservation, resulting in high-fidelity edits.

4.4 User Study

Setup. To assess perceptual editing quality, we conducted a user study covering both rigid and non-rigid cases. Eight methods were compared: *PnP, MasaCtrl, FlexiEdit, FreeDiff, RF-Inv, StableFlow, RF-Edit*, and *Ours-DiT*. Participants were instructed to rank results based on (1) faithfulness to the editing instruction and (2) preservation of unedited regions. Each user received around 20 image sets, with method order fully randomized to ensure fairness.

Results. We collected **48 valid responses**, including users without prior experience in image editing. Figure 6 reports the ranking distribution, where the *x*-axis represents each method and the *y*-axis shows user ranking (**1 = best**, **8 = worst**). The solid red line denotes the median, while the dashed line indicates the mean.

Methods such as *RF-Inv*, *StableFlow*, and *RF-Edit* show large variance, suggesting unstable performance across different editing types. *PnP*, *MasaCtrl*, *FlexiEdit*, and *FreeDiff* achieve moderate consistency but frequently suffer from semantic drift, as reflected by their higher median.

In contrast, **Ours-DiT achieves the lowest median and mean ranking**, with a compact distribution, indicating strong user agreement. From a perceptual perspective, participants consistently preferred our method for its balance between instruction alignment and source fidelity. These results confirm that FSI-Edit delivers the most reliable and visually convincing edits across diverse scenarios.

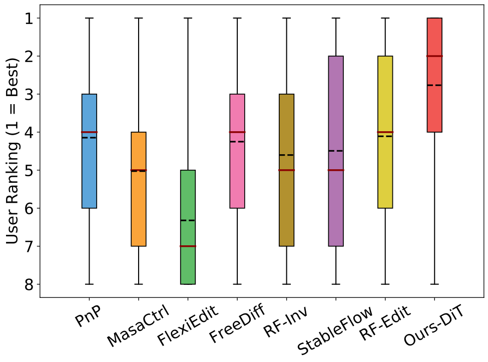


Figure 6: User study ranking distribution across eight editing methods. The *x*-axis lists compared approaches, while the *y*-axis denotes user ranking (1 = best, 8 = worst). The solid red line indicates the median, and the dashed ine represents the mean. Our FSI-Edit achieves the lowest median and most compact variance, showing strong user preference and perceptual robustness.

Table 3: Ablation study on the non-rigid editing dataset. w/o FRF removes the frequency residual fusion module, while w/o SNI denotes the variant without stochastic noise injection, and w/o ITN disables inversion trajectory navigation.

Method	Structure	Background Preservation				CLIP Similarity	
	$Distance_{\times 10^3}\downarrow$	$PSNR\uparrow$	$LPIPS_{\times 10^3}\downarrow$	$MSE_{\times 10^4}\downarrow$	$SSIM_{\times 10^2}\uparrow$	$Whole\uparrow$	$Edited\uparrow$
w/o FRF	10.51	26.97	81.33	30.88	86.38	25.50	21.14
w/o SNI	16.38	26.12	114.22	37.13	83.52	25.85	21.27
w/o ITN	19.90	23.92	120.14	60.47	81.72	26.47	22.05
Ours	19.87	23.91	120.82	59.95	81.71	26.58	22.16

Table 4: Editing performance comparison of ITN with existing reconstruction-refining methods.

Method	Structure	Background Preservation				CLIP Similarity	
	$Distance_{\times 10^3}\downarrow$	$PSNR\uparrow$	$LPIPS_{\times 10^3}\downarrow$	$MSE_{\times 10^4}\downarrow$	$SSIM_{\times 10^2}\uparrow$	$Whole\uparrow$	$Edited\uparrow$
DDIM	69.43	17.87	208.80	219.88	71.56	25.01	22.44
Direct	11.65	27.22	54.55	32.86	85.10	25.02	22.10
NT	13.44	27.07	59.88	35.47	84.55	24.75	21.87
Friend	11.07	26.17	58.73	38.21	84.26	25.22	22.13
ITN (Ours)	10.31	26.34	57.57	37.20	84.60	25.35	22.13

4.5 Ablation Study

Effects of Key Components. To assess the contribution of each core component in our method, we conduct ablation studies on the curated non-rigid editing subset of PIE-Bench. Specifically, we evaluate the impact of three modules: Frequency Residual Fusion (FRF), Stochastic Noise Injection (SNI), and Inversion Trajectory Navigation (ITN). We consider three ablated variants of FSI-Edit-DiT: (1) w/o **FRF** removes frequency fusion in self-attention; (2) w/o **SNI** disables stochastic noise injection in attention layers; (3) w/o **ITN** excludes frequency fusion during the source inversion trajectory.

Quantitative results are shown in Table 3, which show that removing either FRF or SNI significantly compromises non-rigid editing quality, leading to weaker prompt alignment and limited structural transformations. In contrast, the full model combines stochasticity to unlock the base model's generative flexibility and FRF to bridge the semantic gap between source and target branches, ultimately achieving a better balance between structural preservation and editing fidelity.

ITN vs. Reconstruction-refining Methods. We additionally conduct a dedicated comparison against three widely-used reconstruction-refining methods under the same P2P backbone: Direct Inversion (Direct) [15], Null-text Inversion (NT) [14], and Edit-Friendly Inversion (Friend) [46]. We evaluate both editing fidelity and report the results in Table 4. As shown, ITN achieves the lowest editing distance and the highest whole CLIP alignment, which confirms ITN as an essential component for balancing editability and fidelity in editing tasks.

Case Study. Figure 7 illustrates the role of each component through rigid and non-rigid editing examples. Without **FRF**, the model fails to achieve large-scale structural deformation (*e.g.*, turning a bird into an "X" shape). Without **SNI**, new content such as earrings or additional animals is synthesized by reusing existing textures, leading to semantic distortion and local artifacts. Without **ITN**, the model struggles to introduce entirely new objects (*e.g.*, a car or graffiti), resulting in incomplete or collapsed structures. The red circles highlight key regions where the absence of each module causes failure. Together, FRF, SNI, and ITN enable a balance of structural stability, semantic fidelity, and generative diversity, crucial for high-quality image editing.

5 Limitations and Broader Impact

5.1 Limitation

While our method demonstrates strong performance in non-rigid editing, it may still cause unintended alterations in adjacent regions during fine-grained edits. For instance, in Figure 8a, removing the glasses inadvertently changes the woman's facial expression. Additionally, as shown in Figure 8b, our approach tends to be less responsive to color-specific modifications. In some cases, it may also

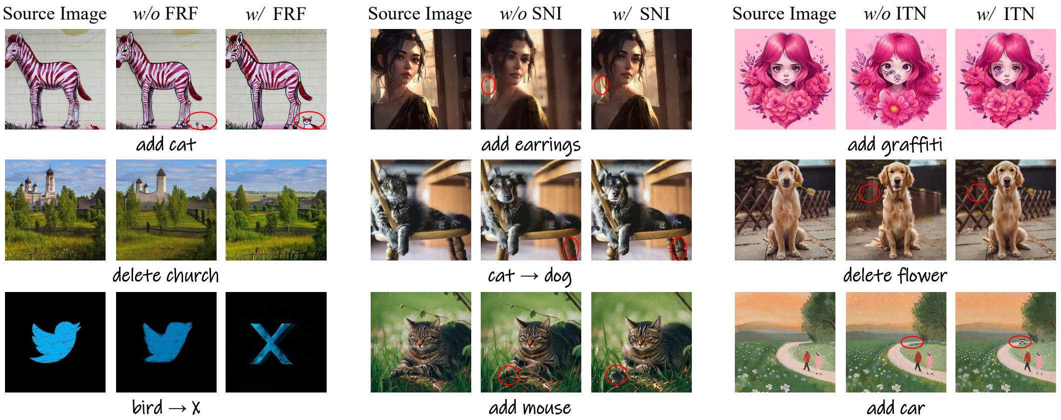


Figure 7: Visual Ablation of Each Module. From left to right, we illustrate the effects of removing FRF, SNI, and ITN, respectively. Without FRF, large-scale non-rigid deformation cannot be achieved, leading to rigid or incomplete edits. Without SNI or ITN, the non-rigid editing quality degrades, and crucial source details may be lost. The red circles highlight the critical differences between using and omitting each module.

fail to fully address all regions implied by the textual prompt, as illustrated in Figure 8c. In future work, we plan to incorporate mask-based spatial guidance, particularly for rigid edits, to enable more localized and accurate modifications. Furthermore, we aim to design frequency fusion strategies tailored to better handle color-sensitive edits, enhancing the controllability of our method in both structural and appearance-level transformations.

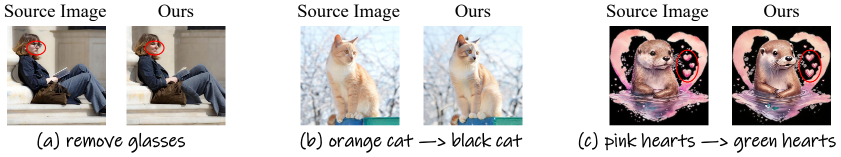


Figure 8: Examples of failed editing cases.

5.2 Broader Impact

This work advances the understanding of attention-based feature manipulation in diffusion models by introducing feature-level frequency fusion and stochasticity. These mechanisms enable a more effective balance between background preservation and content modification, resulting in semantically coherent and visually consistent outcomes for both rigid and non-rigid image editing tasks. Looking forward, flexible non-rigid editing with large structural transformations holds potential for practical applications such as visual effects production, where creative structural changes are common, and in surgical imaging, where it can aid in generating physiologically plausible synthetic data for training and simulation. For instance, our method could facilitate the creation of risk-aware surgical scenarios by simulating realistic anatomical deformations. In summary, this research not only improves image editing techniques, but also paves the way for their broader application in creative and scientific domains that demand both precision and flexibility.

6 Conclusion

In this paper, we introduced FSI-Edit, a novel tuning-free image editing framework that is effective across both LDM and DiT backbones. FSI-Edit enables flexible and high-fidelity non-rigid editing by incorporating two key mechanisms: (1) frequency residual fusion, which injects high-frequency details from the reconstruction branch to mitigate semantic inconsistency while preserving essential textures, and (2) stochastic noise injection, which expands the generative space and facilitates diverse structural transformations. In addition, FSI-Edit applies frequency-domain fusion not only at the feature level but also along the temporal inversion trajectory, helping preserve unedited content more faithfully while enhancing edit controllability. Extensive experiments on both non-rigid and rigid editing tasks demonstrate that FSI-Edit achieves superior performance, validating its effectiveness, generality, and practical value in a wide range of editing scenarios.

Acknowledgments and Disclosure of Funding

This work was supported in part by National Natural Science Foundation of China (Grant No.62202189), National Key R&D Program of China (Grant No. 2023YFC2414900), and research grants from Wuhan United Imaging Healthcare Surgical Technology Co., Ltd.

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[2] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[3] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

[4] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[5] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[6] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[7] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[8] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

[9] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023.

[10] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arXiv:2307.10373*, 2023.

[11] Sunjae Yoon, Gwanhyeong Koo, Ji Woo Hong, and Chang D Yoo. Dni: Dilutional noise initialization for diffusion video editing. In *European Conference on Computer Vision*, pages 180–195. Springer, 2024.

[12] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[13] Daiki Miyake, Akihiro Iohara, Yu Saito, and Toshiyuki Tanaka. Negative-prompt inversion: Fast image inversion for editing with text-guided diffusion models. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 2063–2072. IEEE, 2025.

[14] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6038–6047, 2023.

[15] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Direct inversion: Boosting diffusion-based editing with 3 lines of code. *arXiv preprint arXiv:2310.01506*, 2023.

[16] Gwanhyeong Koo, Sunjae Yoon, Ji Woo Hong, and Chang D Yoo. Flexiedit: Frequency-aware latent refinement for enhanced non-rigid editing. In *European Conference on Computer Vision*, pages 363–379. Springer, 2024.

[17] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[18] Sihan Xu, Yidong Huang, Jiayi Pan, Ziqiao Ma, and Joyce Chai. Inversion-free image editing with natural language. *arXiv preprint arXiv:2312.04965*, 2023.

[19] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22560–22570, 2023.

[20] Qi Mao, Lan Chen, Yuchao Gu, Mike Zheng Shou, and Ming-Hsuan Yang. Tuning-free image editing with fidelity and editability via unified latent diffusion model. *arXiv preprint arXiv:2504.05594*, 2025.

[21] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

[22] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

[23] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6007–6017, 2023.

[24] Shiwen Zhang, Shuai Xiao, and Weilin Huang. Forgedit: Text guided image editing via learning and forgetting. *arXiv preprint arXiv:2309.10556*, 2023.

[25] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7323–7334, 2023.

[26] Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris N Metaxas, and Jian Ren. Sine: Single image editing with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6027–6037, 2023.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[28] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[29] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

[30] Pengcheng Xu, Boyuan Jiang, Xiaobin Hu, Donghao Luo, Qingdong He, Jiangning Zhang, Chengjie Wang, Yunsheng Wu, Charles Ling, and Boyu Wang. Unveil inversion and invariance in flow transformer for versatile image editing. *arXiv preprint arXiv:2411.15843*, 2024.

[31] Jiangshan Wang, Junfu Pu, Zhongang Qi, Jiayi Guo, Yue Ma, Nisha Huang, Yuxin Chen, Xiu Li, and Ying Shan. Taming rectified flow for inversion and editing. *arXiv preprint arXiv:2411.04746*, 2024.

[32] Yingying Deng, Xiangyu He, Changwang Mei, Peisong Wang, and Fan Tang. Fireflow: Fast inversion of rectified flow for image semantic editing. *arXiv preprint arXiv:2412.07517*, 2024.

[33] Yihan Hu, Jianing Peng, Yiheng Lin, Ting Liu, Xiaochao Qu, Luoqi Liu, Yao Zhao, and Yunchao Wei. Dcedit: Dual-level controlled image editing via precisely localized semantics. *arXiv preprint arXiv:2503.16795*, 2025.

[34] Wei Wu, Qingnan Fan, Shuai Qin, Hong Gu, Ruoyu Zhao, and Antoni B Chan. Freediff: Progressive frequency truncation for image editing with diffusion models. In *European Conference on Computer Vision*, pages 194–209. Springer, 2024.

[35] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

[36] Yufan Ren, Zicong Jiang, Tong Zhang, Søren Forchhammer, and Sabine Süsstrunk. Fds: Frequency-aware denoising score for text-guided latent diffusion image editing. *arXiv preprint arXiv:2503.19191*, 2025.

[37] Xiang Gao, Zhengbo Xu, Junhan Zhao, and Jiaying Liu. Frequency-controlled diffusion model for versatile text-guided image-to-image translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1824–1832, 2024.

[38] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[39] Litu Rout, Yujia Chen, Nataniel Ruiz, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Semantic image inversion and editing using rectified stochastic differential equations. *arXiv preprint arXiv:2410.10792*, 2024.

[40] Omri Avrahami, Or Patashnik, Ohad Fried, Egor Nemchinov, Kfir Aberman, Dani Lischinski, and Daniel Cohen-Or. Stable flow: Vital layers for training-free image editing. *arXiv preprint arXiv:2411.14430*, 2024.

[41] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10748–10757, 2022.

[42] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[43] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

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

[46] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12469–12478, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims in the abstract and introduction are well-aligned with the paper's actual contributions, focusing on non-rigid editing via stochasticity and frequency-domain fusion.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations in Sec. 5.1, highlighting challenges in handling fine-grained edits, color changes, and precise region localization.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: We do not include the theoretical assumption and experients.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides all necessary details for reproducing the main experimental results in Sec.4.1, including dataset, baselines, metrics, and implementation details. The code will also be released to facilitate full reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The paper will provide open access to the code and data along with detailed instructions to ensure faithful reproduction of the main experimental results.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please see Sec.4.1 for details of the experiment design.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to resource limitations, we do not report error bars. As detailed in Sec. 4, our experiments involve training multiple models under different configurations, making repeated trials prohibitively expensive. Moreover, our evaluation setup, especially the random seed, keep the same as previous works, which also does not include error bars for similar tasks.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The required computer resources, including the GPU type, memory usage, and execution time, are described in Sec. 4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We followed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The discussion societal impacts is provided in Sec. 5.2.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We properly cite all external assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: Our work proposes a new algorithm evaluated on existing datasets and models. No new assets are released.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: the core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Preliminary

In this section, we provide a brief overview of the three key preliminaries underlying our tuning-free FSI-Edit framework: DDIM Inversion [12], Rectified Flow [29, 38], and Classifier-Free Guidance [35].

A.1 DDIM Inversion

DDIM extends DDPM into a non-Markovian diffusion process. In the LDMs, DDIM uses the model's noise estimator ϵ_θ to sample the latent x_{t-1} from x_t by:

$$x_{t-1} = \frac{\sqrt{\alpha_{t-1}}}{\sqrt{\alpha_t}}\, x_t - \frac{\sqrt{\alpha_{t-1}(1-\alpha_t)} - \sqrt{(1-\alpha_{t-1})\,\alpha_t}}{\sqrt{\alpha_t}}\, \epsilon_\theta(x_t, t), \tag{7}$$

where x_t denotes the latent noisy features at timestep t. By reformulating this discrete update as an ordinary differential equation (ODE), we can apply Euler Integration to solve the reverse process:

$$x_t = \frac{\sqrt{\alpha_t}}{\sqrt{\alpha_{t-1}}}\, x_{t-1} + \frac{\sqrt{\alpha_{t-1}(1-\alpha_t)} - \sqrt{(1-\alpha_{t-1})\,\alpha_t}}{\sqrt{\alpha_{t-1}}}\, \epsilon_\theta(x_{t-1}, t-1). \tag{8}$$

Consequently, when inverting a given source image, we obtain the DDIM Inversion trajectory $\{x_t\}_{t=0}^{T}$.

A.2 Rectified Flow

[29, 38] learns straight-line transport between two distributions by combining linear interpolation with an ODE-based sampler. Given two observed distributions $x_0 \sim p_0$ and $x_1 \sim p_1$, it defines the continuous trajectory:

$$x_t \;=\; t\,x_1 \;+\; (1-t)\,x_0, \quad t \in [0,1], \tag{9}$$

and models its time-aware velocity via a neural network $v_\theta(x_t, t)$. In practice, one discretizes time with steps $\{\sigma_t\}$ and applies Euler integration:

$$x_t \;=\; x_{t-1} \;+\; (\sigma_t - \sigma_{t-1})\, v_\theta(x_{t-1}, t-1). \tag{10}$$

To invert a given endpoint x_1 back toward x_0, we simply reverse the Euler step:

$$x_{t-1} \;=\; x_t \;+\; (\sigma_{t-1} - \sigma_t)\, v_\theta(x_t, t-1). \tag{11}$$

By iterating this from $t = 1$ down to $t = 0$, we recover the Rectified Flow inversion trajectory $\{x_t\}_{t=0}^{1}$, which maps an observed source image latent back to its original distribution.

A.3 Classifier-Free Guidance

To better integrate conditional control into generative diffusion models, Ho *et al.* [35] proposed Classifier-Free Guidance (CFG). CFG replaces the need for an external classifier by interpolating between conditional and unconditional noise predictions. Let c denote the conditioning, so the noise prediction under condition c is $\epsilon_\theta(x_t, c)$, while its unconditional counterpart is $\epsilon_\theta(x_t, \varnothing)$. CFG then computes the guided noise estimate as:

$$\tilde{\epsilon}_\theta(x_t, c) \;=\; \epsilon_\theta(x_t, \varnothing) \;+\; \lambda\big(\epsilon_\theta(x_t, c) \;-\; \epsilon_\theta(x_t, \varnothing)\big), \tag{12}$$

where $\lambda > 1$ is the guidance scale.

B Implementation Details of FSI-Edit-LDM and FSI-Edit-DiT

We implement our method on top of two popular T2I backbones: Latent Diffusion Models (LDM) [5] v1.5 and Diffusion Transformers (DiT) [7] v3.5-Medium, leading to two variants: FSI-Edit-LDM and FSI-Edit-DiT. The core components, Frequency Residual Fusion (FRF), Stochastic Noise Injection (SNI), and Inversion Trajectory Navigation (ITN), remain consistent across both variants. However, their integration is adapted to accommodate the architectural differences between backbones.

B.1 FSI-Edit-LDM

Shown in Algorithm 1, for the LDM-based version of our method, we follow the feature interaction strategy of PnP [9] and apply our proposed modules, Frequency Residual Fusion (FRF), Stochastic Noise Injection (SNI), and Inversion Trajectory Navigation (ITN), at specific locations within the UNet architecture:

- FRF: embedded in the feature map of the 4th decoder *residual* block. This is active for the first 80% of the denoising process (*i.e.*, the first 40 timesteps).
- SNI: applied to the *self-attention* blocks from the 4th to the 11th decoder layers of the UNet. We inject noise into the attention queries and keys ($Q^{\texttt{self}}_{\texttt{tgt}}$, $K^{\texttt{self}}_{\texttt{tgt}}$), while directly injecting the values ($V^{\texttt{self}}_{\texttt{tgt}}$). This operation is performed during the first 50% of the denoising steps (*i.e.*, the first 25 timesteps).
- ITN: operated over the entire denoising trajectory, performing frequency-based blending between successive latent states to extract source-consistent features for cross-branch editing.

Algorithm 1 FSI-Edit-LDM

```
Input: origin image $x_0$, inversion steps $T$, denoising model $\epsilon_\theta$, source target prompts $\mathcal{P}_{src}$, $\mathcal{P}_{tgt}$,
  res-block and self-attention thresholds $\tau_{res}$ and $\tau_{self}$
Stage I: DDIM Inversion
for $t = 1, \cdots, T$ do
    $x_t = \frac{\sqrt{\alpha_t}}{\sqrt{\alpha_{t-1}}} x_{t-1} + \frac{\sqrt{\alpha_{t-1}(1-\alpha_t)} - \sqrt{(1-\alpha_{t-1})\,\alpha_t}}{\sqrt{\alpha_{t-1}}} \epsilon_\theta(x_{t-1}, t-1, \mathcal{P}_{src})$
end for
Get the inversion trajectory $\{x_t\}_{t=1}^{T}$
Stage II: FSI Editing
$x_T^{tar} = \texttt{ITN}(x_T, x_{T-1})$
for $t = T, \cdots, 1$ do
    $\tilde{x}_t = \mathrm{ITN}(x_t, x_{t-1})$
    $f^{\texttt{res}}_{t,\texttt{src}}, Q^{\texttt{self}}_{t,\texttt{src}}, K^{\texttt{self}}_{t,\texttt{src}}, V^{\texttt{self}}_{t,\texttt{src}} \leftarrow \epsilon_\theta(\tilde{x}_t, t, \mathcal{P}_{tgt})$
    if $t > \tau_{res}$ then
        $f^{\texttt{res}''}_{t,\texttt{tgt}} = \mathrm{SNI}\big(\mathrm{FRF}(f^{\texttt{res}}_{t,\texttt{src}}, f^{\texttt{res}}_{t,\texttt{tgt}})\big)$
    else
        $f^{\texttt{res}''}_{t,\texttt{tgt}} = f^{\texttt{res}}_{t,\texttt{tgt}}$
    end if
    if $t > \tau_{self}$ then
        $Q^{\texttt{self}''}_{t,\texttt{tgt}}, K^{\texttt{self}''}_{t,\texttt{tgt}} = \mathrm{SNI}\big(\mathrm{FRF}(Q^{\texttt{self}}_{t,\texttt{src}}, K^{\texttt{self}}_{t,\texttt{src}})\big)\ ;\ V^{\texttt{self}'}_{t,\texttt{tgt}} = V^{\texttt{self}}_{t,\texttt{src}}$
    else
        $Q^{\texttt{self}''}_{t,\texttt{tgt}}, K^{\texttt{self}''}_{t,\texttt{tgt}}, V^{\texttt{self}'}_{t,\texttt{tgt}} = Q^{\texttt{self}}_{t,\texttt{tgt}}, K^{\texttt{self}}_{t,\texttt{tgt}}, V^{\texttt{self}}_{t,\texttt{tgt}}$
    end if
    $x^{tar'}_{t-1} = \epsilon_\theta(x^{tar}_t, t, \mathcal{P}_{tgt};\ f^{\texttt{res}''}_{t,\texttt{tgt}}, Q^{\texttt{self}''}_{t,\texttt{tgt}}, K^{\texttt{self}''}_{t,\texttt{tgt}}, V^{\texttt{self}'}_{t,\texttt{tgt}})$
    $x^{tar}_{t-1} = \text{DDIM-Samp}(x^{tar}_t, x^{tar'}_{t-1})$
end for
Output: Editing image $x_0^{tar}$
```

B.2 FSI-Edit-DiT

As illustrated in Algorithm 2, we adopt DiT v3.5-Medium (bfloat16) as the backbone for our transformer-based implementation. FSI-Edit-DiT incorporates all three key modules with the following configurations:

- FRF: embedded to the *self-attention* layers of the 0th to 12th Transformer blocks, and is active during the first 50% of the denoising timesteps.
- SNI: applied to all *cross-attention* layers throughout the Transformer blocks, and is active for the first 50% of the denoising steps.
- ITN: performed across all timesteps to refine source latent representations

Algorithm 2 FSI-Edit-DiT

```
Input: origin image $x_0$, inversion steps $T$, velocity field $v_\theta$, source target prompts $\mathcal{P}_{src}$, $\mathcal{P}_{tgt}$,
    cross-block and self-attention thresholds $\tau_{cross}$ and $\tau_{self}$
Stage I: Rectified Flow Inversion
for $t = 1, \cdots, T$ do
    $x_t = x_{t-1} + (\sigma_t - \sigma_{t-1}) v_\theta(x_{t-1}, t-1, \mathcal{P}_{src})$
end for
Get the inversion trajectory $\{x_t\}_{t=1}^{T}$
Stage II: FSI Editing
$x_T^{tar} = \texttt{ITN}(x_T, x_{T-1})$
for $t = T, \cdots, 1$ do
    $\tilde{x}_t = \mathrm{ITN}(x_t, x_{t-1})$
    $(Q_{t,\texttt{src}}^{\texttt{cross}}, K_{t,\texttt{src}}^{\texttt{cross}}, V_{t,\texttt{src}}^{\texttt{cross}}), (Q_{t,\texttt{src}}^{\texttt{self}}, K_{t,\texttt{src}}^{\texttt{self}}, V_{t,\texttt{src}}^{\texttt{self}}) \leftarrow v_\theta(\tilde{x}_t, t, \mathcal{P}_{tgt})$
    if $t > \tau_{cross}$ then
        $Q_{t,\texttt{tgt}}^{\texttt{cross}'}, K_{t,\texttt{tgt}}^{\texttt{cross}'}, V_{t,\texttt{tgt}}^{\texttt{cross}'} = \mathrm{SNI}\big(Q_{t,\texttt{src}}^{\texttt{cross}}, K_{t,\texttt{src}}^{\texttt{cross}}, V_{t,\texttt{src}}^{\texttt{cross}}\big)$
    else
        $Q_{t,\texttt{tgt}}^{\texttt{cross}'}, K_{t,\texttt{tgt}}^{\texttt{cross}'}, V_{t,\texttt{tgt}}^{\texttt{cross}'} = Q_{t,\texttt{tgt}}^{\texttt{cross}}, K_{t,\texttt{tgt}}^{\texttt{cross}}, V_{t,\texttt{tgt}}^{\texttt{cross}}$
    end if
    if $t > \tau_{self}$ then
        $Q_{t,\texttt{tgt}}^{\texttt{self}''}, K_{t,\texttt{tgt}}^{\texttt{self}''} = \mathrm{SNI}\big(\mathrm{FRF}(Q_{t,\texttt{src}}^{\texttt{self}}, K_{t,\texttt{src}}^{\texttt{self}})\big)\ ; V_{t,\texttt{tgt}}^{\texttt{self}'} = V_{t,\texttt{src}}^{\texttt{self}}$
    else
        $Q_{t,\texttt{tgt}}^{\texttt{self}''}, K_{t,\texttt{tgt}}^{\texttt{self}''}, V_{t,\texttt{tgt}}^{\texttt{self}'} = Q_{t,\texttt{tgt}}^{\texttt{self}}, K_{t,\texttt{tgt}}^{\texttt{self}}, V_{t,\texttt{tgt}}^{\texttt{self}}$
    end if
    $x_{t-1}^{tar'} = v_\theta(x_t^{tar}, t, \mathcal{P}_{tgt};\ Q_{t,\texttt{tgt}}^{\texttt{cross}'}, K_{t,\texttt{tgt}}^{\texttt{cross}'}, V_{t,\texttt{tgt}}^{\texttt{cross}'}, Q_{t,\texttt{tgt}}^{\texttt{self}''}, K_{t,\texttt{tgt}}^{\texttt{self}''}, V_{t,\texttt{tgt}}^{\texttt{self}'})$
    $x_{t-1}^{tar} = \mathrm{RectifiedFlow\text{-}Samp}(x_t^{tar}, x_{t-1}^{tar'})$
end for
Output: Editing image $x_0^{tar}$
```

C Comparison Methods and Experimental Setup

In this section, we present the experimental setup and parameter configurations for the baseline methods used in our comparisons.

C.1 LDM-Based

For P2P [8], PnP [9], and MasaCtrl [19], we adopt DDIM Direct Inversion [15][1] as the inversion backbone. All image editing experiments are conducted using their default parameter settings.

For FlexiEdit [16], we use the official implementation[2]. To support large-scale, consistent batch processing across methods, we follow the configuration most commonly used in the official examples by fixing the reinversion steps to $t_R = 30$. Since FlexiEdit requires a 'blended word' to localize the editing region, we extract this information from corresponding prompts in PIE-Bench [15]. In cases where the model fails to locate the semantic region associated with the 'blended word', we default to editing the entire image. All other parameters remain at their default settings.

For FreeDiff [34], we use the official implementation[3] and adopt the most representative configuration from the official examples, setting the time steps for filter scheduling to $\tau_i = (801, 781, 581)$ and the high-pass filter sizes to $r_t^H = (32, 32, 10, 10)$. All other parameters are kept at their default values.

All LDM-based methods above are implemented using the v1.4 or v1.5 Stable Diffusion backbone and are executed on a single NVIDIA RTX 4090 GPU with 24GB of memory.

[1] https://github.com/cure-lab/PnPInversion

[2] https://github.com/kookie12/FlexiEdit

[3] https://github.com/Thermal-Dynamics/FreeDiff

Table 5: User study ranking results (Mean ± Standard Deviation; lower indicates better perceptual quality). Supplement to Fig. 6 in the main paper.

Method	PnP	MasaCtrl	FlexiEdit	FreeDiff	RF-Inv	StableFlow	RF-Edit	Ours-DiT
Ranking	3.94 ± 2.22	4.80 ± 2.30	5.99 ± 2.59	4.09 ± 2.22	4.39 ± 2.42	4.29 ± 2.51	3.92 ± 2.29	**2.66 ± 1.96**

C.2 DiT-Based

We use the official implementation of StableFlow[4] [40], with the inversion steps set to 50. For RF-Inv [39], we adopt the official code[5] with the default settings. For RF-Edit [31], we use the official repository[6], with the guidance scale set to 2 and the injection step set to 5. All other parameters remain at their default values. The above three methods are executed on a single NVIDIA A100-PCIE-80GB GPU.

D User Study Statistics

To complement Fig. 6 in the main paper, we report the detailed statistical results of the user study, including the mean and standard deviation of ranking scores for each method. As described in the main text, each participant was asked to rank eight editing methods based on overall quality (semantic alignment and preservation). Lower scores indicate better perceived performance. Table. 5 provides a clearer view of user preference consistency and further confirm the advantage of FSI-Edit-DiT, which shows both the lowest mean rank and the smallest variance.

E Extended Ablation Studies

In this section, we investigate how different parameter choices in each module of our method affect editing quality and controllability. We conduct a series of controlled experiments to analyze the sensitivity and contribution of key hyperparameters within the FRF, SNI, and ITN components. Specifically, we set the default values as follows: pairing distance $d = 1$ for ITN, fusion weight $\alpha = 0.2$ and Gaussian scaling coefficient $\sigma = 0.3$ for FRF, noise ratio $\eta = 0.2$ (corresponding to $\sigma_f = 0.8$) for SNI, and FSI-Edit intervention durations of 50% and 65% for non-rigid and rigid editing tasks, respectively.

To ensure fairness, all other parameters are kept fixed at these default values when exploring the effect of any single variable. To directly assess the influence of each parameter on editing behavior, all ablation studies are conducted on the non-rigid editing dataset using the FSI-Edit-DiT.

E.1 Effect of ITN Pairing Distance

Table 6: Ablation study on the pairing distance d in ITN.

d	Structure	Background Preservation				CLIP Similarity	
	$Distance_{\times 10^3} \downarrow$	$PSNR \uparrow$	$LPIPS_{\times 10^3} \downarrow$	$MSE_{\times 10^4} \downarrow$	$SSIM_{\times 10^2} \uparrow$	$Whole \uparrow$	$Edited \uparrow$
w/o ITN	19.90	23.92	120.14	60.47	81.72	26.47	22.05
1 (Ours)	19.87	23.91	120.82	59.95	81.71	26.58	22.16
2	19.80	23.94	119.08	60.04	81.80	26.57	22.06
5	19.80	23.94	119.44	60.28	81.74	26.59	22.15
10	19.88	23.99	119.22	59.87	81.83	26.48	22.07

In our default ITN design, each latent x_t is fused with its immediate predecessor x_{t-1} to stabilize inversion while preserving detail. Here, we investigate how increasing the pairing distance d affects the quality of source reconstruction and downstream editing.

[4] https://github.com/snap-research/stable-flow

[5] https://github.com/LituRout/RF-Inversion

[6] https://github.com/wangjiangshan0725/RF-Solver-Edit

Table 7: Ablation study on the pairing distance d in ITN for **source image reconstruction**.

d	Structure	Background Preservation				CLIP Similarity
	$Distance_{\times 10^3} \downarrow$	$PSNR \uparrow$	$LPIPS_{\times 10^3} \downarrow$	$MSE_{\times 10^4} \downarrow$	$SSIM_{\times 10^2} \uparrow$	$Whole \uparrow$
w/o ITN	3.06	32.30	25.35	9.06	91.40	25.97
1 (Ours)	3.08	32.29	25.37	9.07	91.40	25.97
2	3.06	32.30	25.35	9.06	91.40	25.97
5	3.07	32.30	25.35	9.06	91.40	25.97
10	3.06	32.30	25.35	9.06	91.40	25.97

Specifically, we vary the reference latent from x_{t-1} to x_{t-d}, where $d \in \{1, 2, 5, 10\}$, and apply the same frequency-domain fusion strategy:

$$\tilde{x}_t = \texttt{IFFT}\left(\mathcal{H}_\sigma \cdot \texttt{FFT}(x_t) + \mathcal{L}_\sigma \cdot \texttt{FFT}(x_{t-d})\right) + \sigma_x \cdot \mathcal{N}(0, I). \tag{13}$$

We keep $\sigma = 0.3$, and $\sigma_x = 1e{-}3$ here.

The results are summarized in Table 6. Additionally, we evaluate the impact of different pairing distances d in ITN on source reconstruction quality, as shown in Table 7. From Table 7, we observe that varying d has negligible effect on source image reconstruction. Combined with Table 6, we find that applying ITN consistently improves the CLIP similarity in the edited regions compared to w/o ITN.

In addition, as a complementary result to Table 4 in the main paper, we further provide a quantitative comparison of source image reconstruction among ITN and other reconstruction-refining methods in Table 8. These results further demonstrate that ITN provides superior reconstruction fidelity, preserving both structural details and semantic consistency.

Table 8: Ablation study on the ITN with other reconstruction-refining methods for **source image reconstruction**.

d	Structure	Background Preservation				CLIP Similarity
	$Distance_{\times 10^3} \downarrow$	$PSNR \uparrow$	$LPIPS_{\times 10^3} \downarrow$	$MSE_{\times 10^4} \downarrow$	$SSIM_{\times 10^2} \uparrow$	$Whole \uparrow$
DDIM	70.23	17.76	210.84	224.43	71.39	27.07
Direct	2.95	30.57	31.41	17.60	87.50	25.45
NT	3.30	30.17	33.50	18.94	87.13	25.50
Friend	3.12	30.37	32.66	18.21	87.19	25.55
ITN (Ours)	2.33	30.49	31.88	17.85	87.47	25.64

E.2 Ablation Study on Filter Strength and Fusion Weight in FRF

To evaluate the effect of key parameters in Frequency Residual Fusion (FRF), we conduct systematic ablation studies on the Gaussian filter strength. Specifically, the low-pass and high-pass filters $\mathcal{L}_\sigma$ and $\mathcal{H}_\sigma$ are defined as:

$$\mathcal{L}_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}} \in \mathbb{R}^{W \times H}, \quad \mathcal{H}_\sigma = 1 - \mathcal{L}_\sigma \in \mathbb{R}^{W \times H}, \tag{14}$$

where r denotes the frequency distance from the center and σ controls degree of Gaussian curve.

We experiment with various values of $\sigma \in \{0.1, 0.3, 0.4, 0.5, 0.6, 0.8, 0.9\}$ to control the frequency selectivity of the filters, and investigate the influence of the fusion weight $\alpha \in \{0.0, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9, 1.0\}$, which modulates the relative contribution of the target's low-frequency components. The frequency-domain fusion is then computed as:

$$\mathcal{F}_{\text{fuse}} = \mathcal{H}_\sigma \cdot \texttt{FFT}(u) + \alpha \cdot \mathcal{L}_\sigma \cdot \texttt{FFT}(v), \tag{15}$$

$$\texttt{FRF}(u, v) = \texttt{IFFT}(\mathcal{F}_{\text{fuse}}) + v. \tag{16}$$

As shown in Table 9 and Table 10, the value of σ and α have little impact on the results. However, setting $\alpha = 0$ leads to a noticeable drop in editing performance, indicating that the lack of low-frequency information from the target feature impairs the quality of non-rigid editing. This also highlights the limitations of directly performing feature injection.

Table 9: Ablation study on the Gaussian filter scaling coefficient σ in FRF.

σ	Structure	Background Preservation				CLIP Similarity	
	$Distance_{\times 10^3}\downarrow$	$PSNR\uparrow$	$LPIPS_{\times 10^3}\downarrow$	$MSE_{\times 10^4}\downarrow$	$SSIM_{\times 10^2}\uparrow$	$Whole\uparrow$	$Edited\uparrow$
w/o FRF	10.51	26.97	81.33	30.88	86.38	25.50	21.14
0.1	19.84	23.92	120.65	59.88	81.72	26.59	22.15
0.3 (Ours)	19.87	23.91	120.82	59.95	81.71	26.58	22.16
0.4	19.87	23.92	120.64	59.89	81.73	26.60	22.25
0.5	19.73	23.93	120.19	59.74	81.76	26.60	22.13
0.6	19.87	23.94	119.95	59.92	81.78	26.54	22.19
0.8	19.57	23.96	119.42	59.67	81.84	26.60	22.14
0.9	19.62	23.97	119.10	59.59	81.59	26.59	22.18

Table 10: Ablation study on the Fusion Weight α in FRF.

α	Structure	Background Preservation				CLIP Similarity	
	$Distance_{\times 10^3}\downarrow$	$PSNR\uparrow$	$LPIPS_{\times 10^3}\downarrow$	$MSE_{\times 10^4}\downarrow$	$SSIM_{\times 10^2}\uparrow$	$Whole\uparrow$	$Edited\uparrow$
w/o FRF	10.51	26.97	81.33	30.88	86.38	25.50	21.14
0.0	19.08	23.98	116.44	59.27	82.03	23.39	22.04
0.1	19.81	23.92	120.56	59.88	81.72	26.62	22.27
0.2 (Ours)	19.87	23.91	120.82	59.95	81.71	26.58	22.16
0.3	19.81	23.92	120.71	59.88	81.72	26.54	22.21
0.5	19.90	23.92	120.56	59.85	81.72	26.61	22.22
0.7	19.92	23.92	120.74	59.94	81.72	26.57	22.17
0.9	19.90	23.92	120.61	59.87	81.72	26.61	22.25
1.0	19.85	23.92	120.64	59.92	81.71	26.58	22.19

E.3 Ablation Study on the Noise Mixing Coefficient in SNI

To investigate the effect of the noise-content trade-off in our Stochastic Noise Injection (SNI) module, we conduct ablation experiments on the mixing coefficient η in the query and key injection formulation:

$$Q_{\texttt{tgt}}^{\texttt{self}''} = \eta \cdot Q_{\texttt{tgt}}^{\texttt{self}'} + \sigma_f \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad K_{\texttt{tgt}}^{\texttt{self}''} = \eta \cdot K_{\texttt{tgt}}^{\texttt{self}'} + \sigma_f \cdot \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{17}$$

To ensure a balanced contribution between deterministic content and injected noise, we fix $\eta + \sigma_f = 1$ and vary η across $\{0.0, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9, 1.0\}$. This allows us to systematically evaluate how different levels of stochasticity affect the flexibility and consistency of non-rigid edits.

Table 11: Ablation study on the Noise Mixing Coefficient η in SNI. w/o SNI corresponds to the setting where $\eta = 1.0$, and no Gaussian noise is injected into the query and key of the cross-attention layers in the target branch.

η	Structure	Background Preservation				CLIP Similarity	
	$Distance_{\times 10^3}\downarrow$	$PSNR\uparrow$	$LPIPS_{\times 10^3}\downarrow$	$MSE_{\times 10^4}\downarrow$	$SSIM_{\times 10^2}\uparrow$	$Whole\uparrow$	$Edited\uparrow$
w/o SNI	16.38	26.12	114.22	37.13	83.52	25.85	21.27
0.0	19.27	23.92	118.06	59.71	81.90	26.64	22.12
0.1	19.42	23.93	118.41	59.56	81.86	26.60	22.15
0.2 (Ours)	19.87	23.91	120.82	59.95	81.71	26.58	22.16
0.3	20.46	23.88	123.56	60.37	81.48	26.47	22.19
0.5	21.73	23.77	130.38	61.52	80.88	26.50	22.23
0.7	23.04	23.71	137.19	62.57	80.37	26.50	22.14
0.9	24.45	23.64	144.15	63.02	80.18	26.58	22.25
1.0	24.88	23.60	147.03	63.04	80.14	26.53	21.99

As shown in Table 11, varying η has limited influence on the CLIP similarity of the edited regions, indicating stable editing performance. However, setting $\eta = 1.0$, which corresponds to the absence of randomness, leads to suboptimal editing quality and background preservation. To better leverage the refined source and target branch features for editing, we ultimately choose $\eta = 0.2$ and $\sigma_f = 0.8$ as our default settings.

E.4 Ablation Study on the Duration of FRF and SNI Interventions

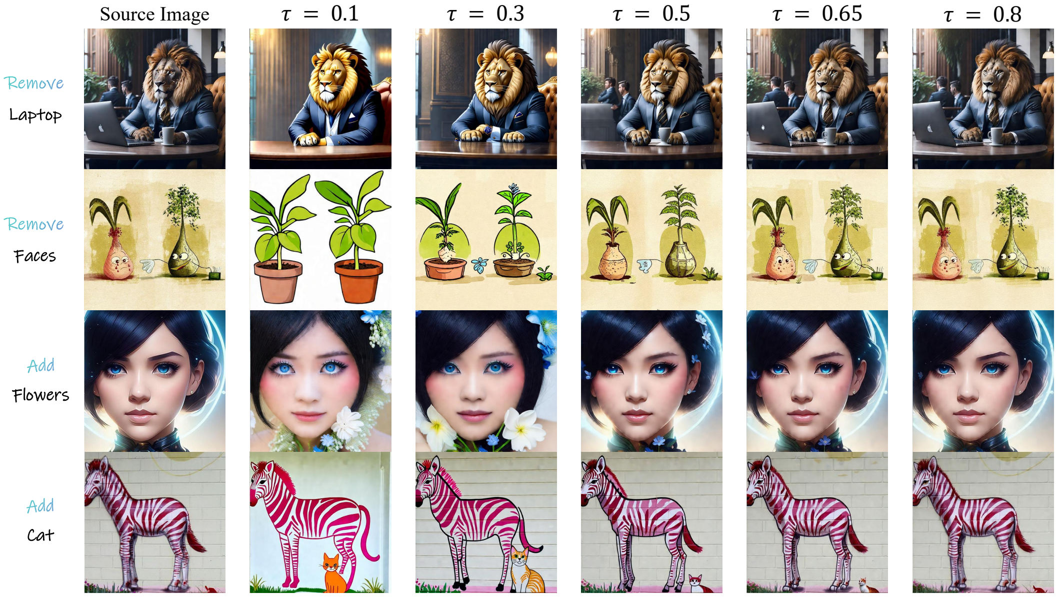


Figure 9: Effect of the duration τ of the intervention for FRF and SNI. As τ increases, the target images better preserve source background structures and maintain semantically meaningful edits. However, excessively large τ may overly constrain generation, limiting the extent of semantic transformations.

To understand the impact of the duration of the intervention for FRF and SNI, we conduct ablation studies by varying the proportion of timesteps τ during which these modules are applied. Specifically, we experiment with 10%, 30%, 50%, 65%, and 80% of the total denoising steps. Note that $\tau = 0\%$ corresponds to random generation, which tends to strictly adhere to the semantics of the target prompt. As the proportion of guided timesteps increases, background preservation improves, but this often comes at the cost of reduced editability.

Table 12 and Figure 9 present the quantitative and qualitative trade-offs between content preservation and editability. Based on these results, we select a duration of 50% for non-rigid editing, and 65% for rigid editing to achieve a balanced performance.

Table 12: Ablation study on the Duration of FRF and SNI Interventions.

τ	Structure	Background Preservation				CLIP Similarity	
	$Distance_{\times 10^3} \downarrow$	$PSNR \uparrow$	$LPIPS_{\times 10^3} \downarrow$	$MSE_{\times 10^4} \downarrow$	$SSIM_{\times 10^2} \uparrow$	$Whole \uparrow$	$Edited \uparrow$
0.1	87.01	14.96	292.32	402.72	64.17	27.63	23.40
0.3	41.97	19.75	187.95	144.51	74.50	27.13	23.29
0.5 (Ours)	19.87	23.91	120.82	59.95	81.71	26.58	22.16
0.65	11.17	26.69	82.69	32.40	85.76	25.69	21.29
0.8	5.48	29.94	49.90	16.25	89.35	24.39	20.22

F Additional Qualitative Comparisons

F.1 Supplementary Visual Comparisons

In this section, we present additional comparison results that could not be included in the main paper due to space limitations. As shown in the Figure 10, our DiT-based method demonstrates superior editing performance across both non-rigid and rigid tasks. However, the LDM-based version is more constrained by the limitations of its underlying generative backbone, resulting in suboptimal outputs in certain cases. For example, in the first row, the model incorrectly alters the dog from the original image; in rows 8 and 9, the results fail to reflect the intended semantic edits, indicating the difficulty LDM has with large-scale non-rigid transformations. In the deletion task of row 11, the model introduces unintended changes, such as adding a glove to the woman's hand. Overall, while our method is partially influenced by the choice of backbone, it achieves consistently strong results when built upon DiT v3.5-Medium.

F.2 Extended Visual Results and Analysis

Figure 11 and Figure 12 provide additional visual comparisons against all baseline methods on both non-rigid and rigid editing tasks. Our approach, in both DiT-based and LDM-based versions, consistently demonstrates superior editing performance across various scenarios.

In particular, Figure 12 includes failure cases from our main paper (rightmost three columns). As illustrated, most competing methods struggle to preserve fine-grained details and maintain color consistency. In more challenging cases, they fail to execute the intended semantic transformations, such as removing glasses or altering the color of a cat.

F.3 Diverse Editing Capabilities of FSI-Edit-DiT

To further demonstrate the versatility of our approach, we present qualitative results across a broad spectrum of editing categories. These include rigid edits (*e.g.*, changes in color, material, or style) as well as non-rigid edits (*e.g.*, object addition, removal and pose changes). As shown in Fig. 13 and Fig. 14, our method consistently produces high-fidelity results across all scenarios, underscoring its robustness and generalizability.

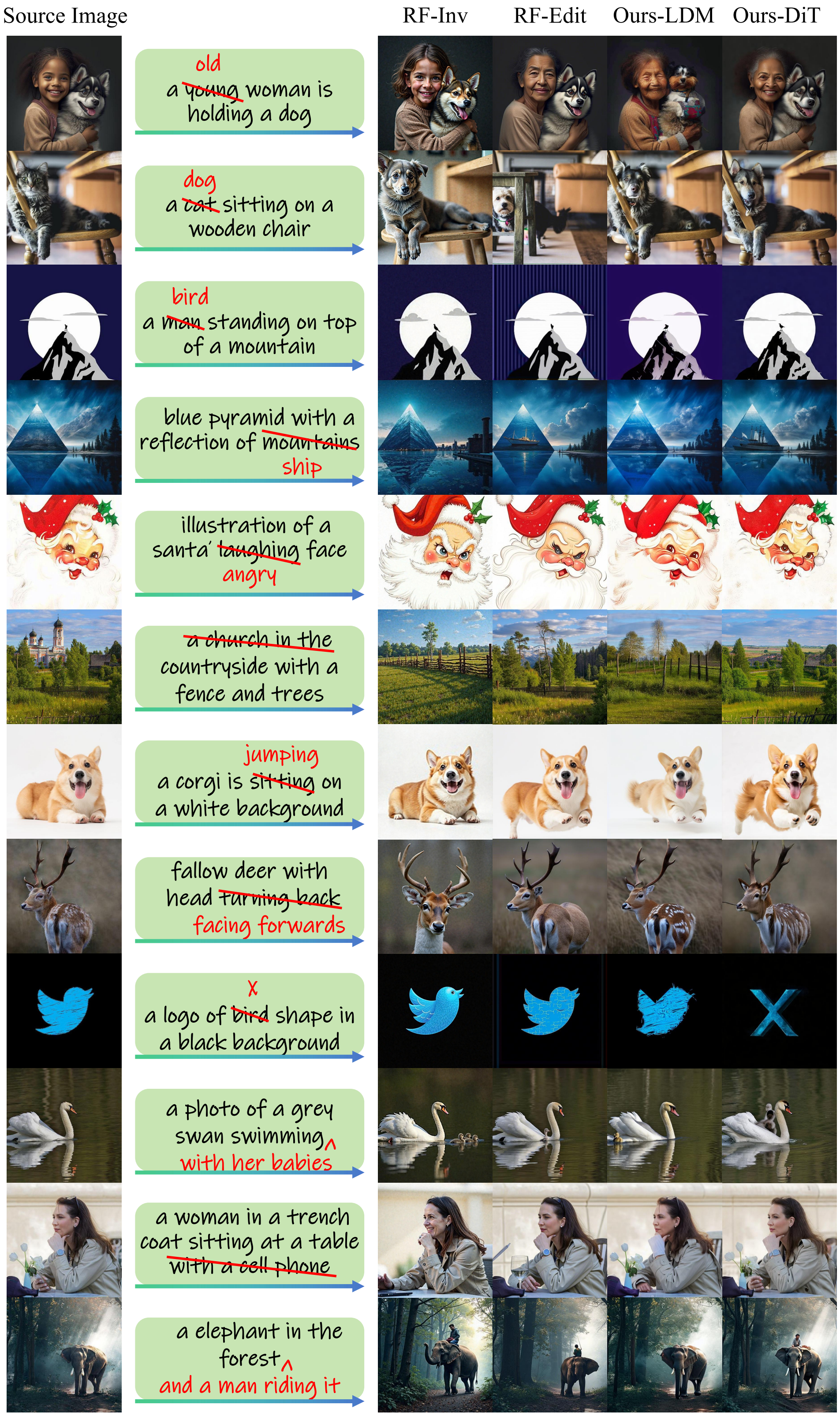

Source Image
RF-Inv
RF-Edit
Ours-LDM
Ours-DiT
old
a ~~young~~ woman is holding a dog
dog
a ~~cat~~ sitting on a wooden chair
bird
a ~~man~~ standing on top of a mountain
blue pyramid with a reflection of ~~mountains~~
ship
illustration of a santa' ~~laughing~~ face
angry
~~a church in the~~ countryside with a fence and trees
jumping
a corgi is ~~sitting~~ on a white background
fallow deer with head ~~turning back~~
facing forwards
X
a logo of ~~bird~~ shape in a black background
a photo of a grey swan swimming^
with her babies
a woman in a trench coat sitting at a table ~~with a cell phone~~
a elephant in the forest^
and a man riding it

Figure 10: Visual comparison of different methods. This figure supplements the main paper with additional results, including comparisons with RF-Inv, RF-Edit, and our LDM-based approach on both non-rigid and rigid editing tasks.

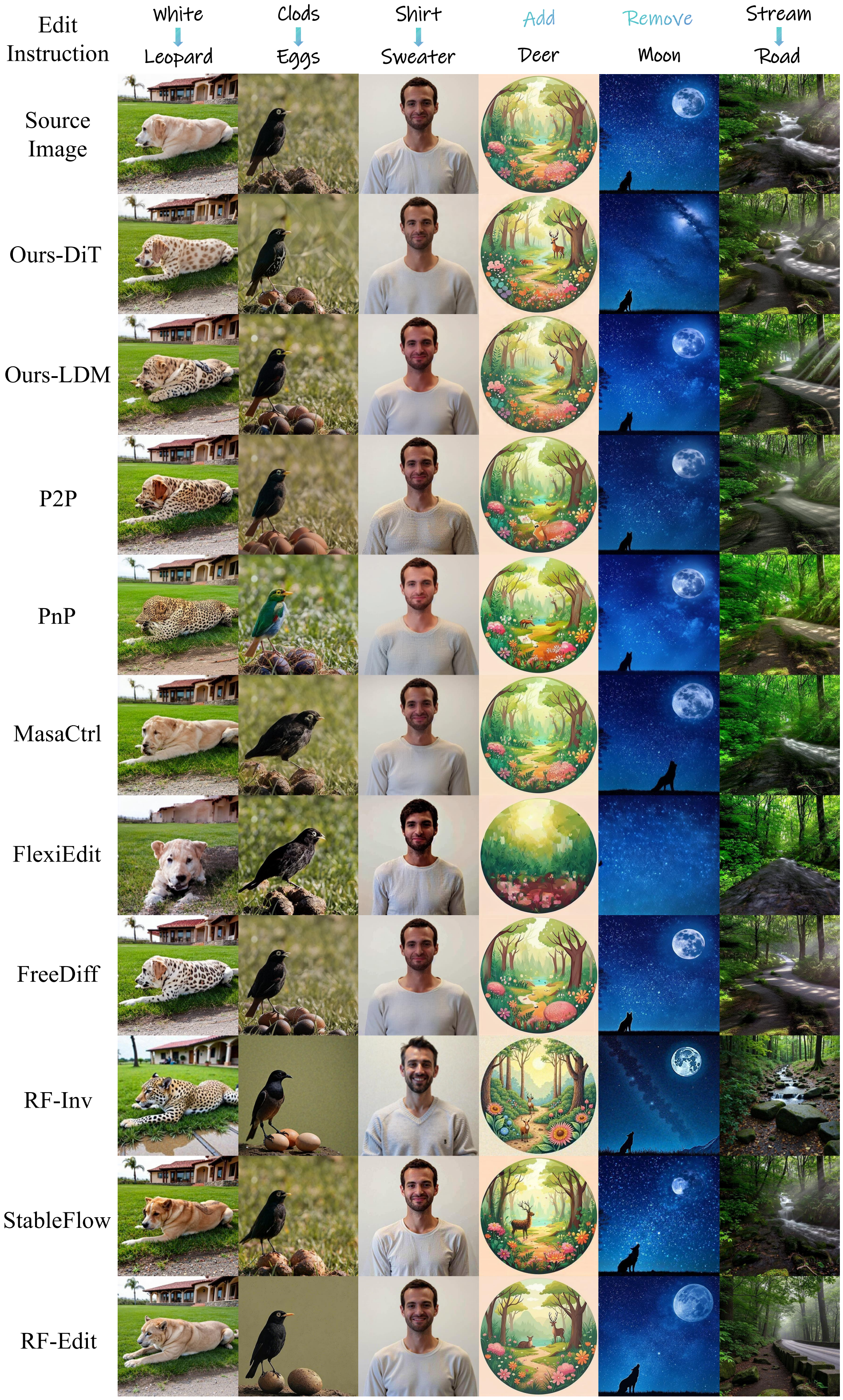


Figure 11: Additional qualitative editing results on PIE-Bench Part I.

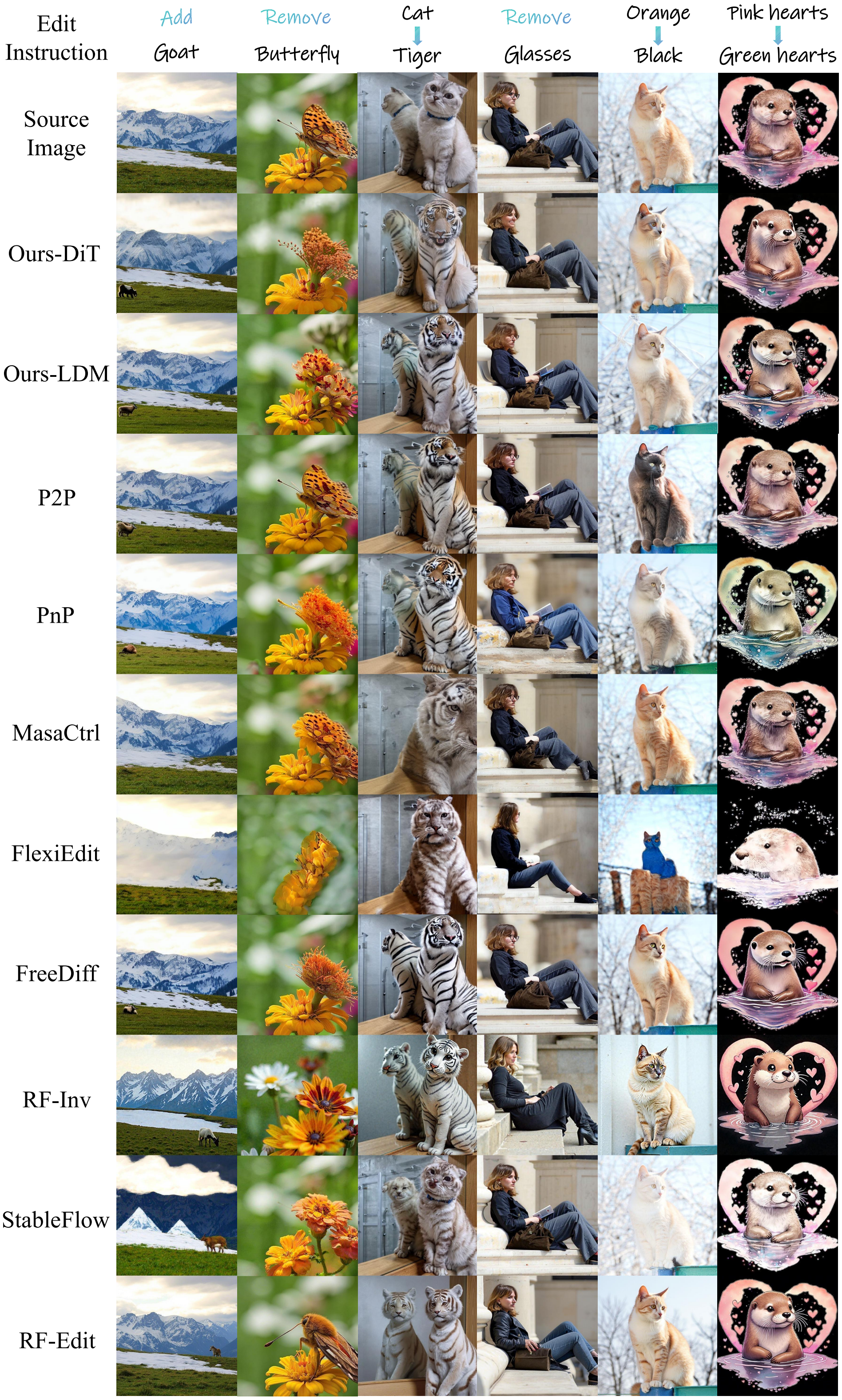


Figure 12: Additional qualitative editing results on PIE-Bench Part II. The rightmost three columns show failure cases from our main paper. Competing methods often fail to preserve fine details or to perform the desired semantic edits (*e.g.*, removing woman's glasses, changing the cat's color).

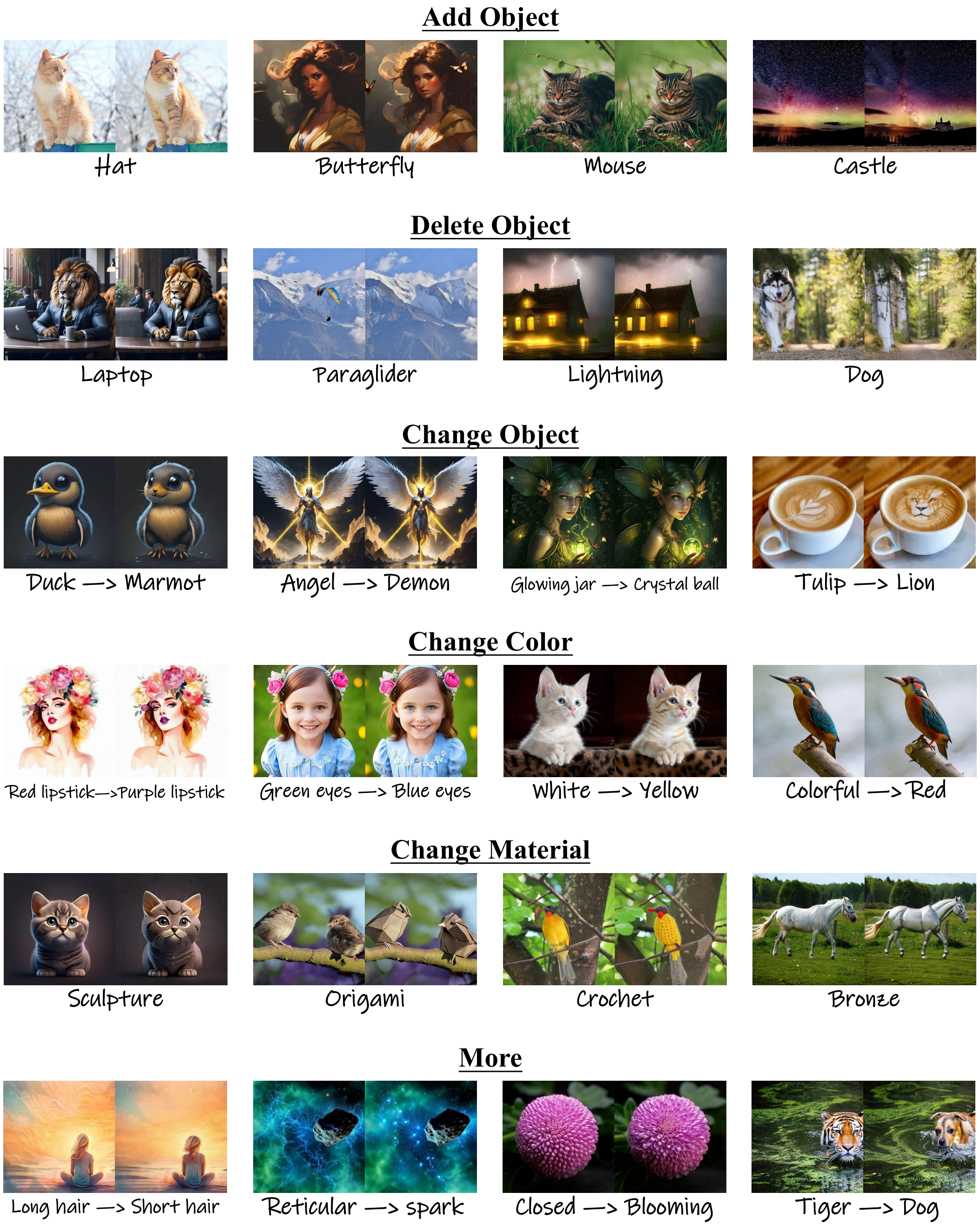


Figure 13: Additional qualitative results on diverse editing types from PIE-Bench Part I.

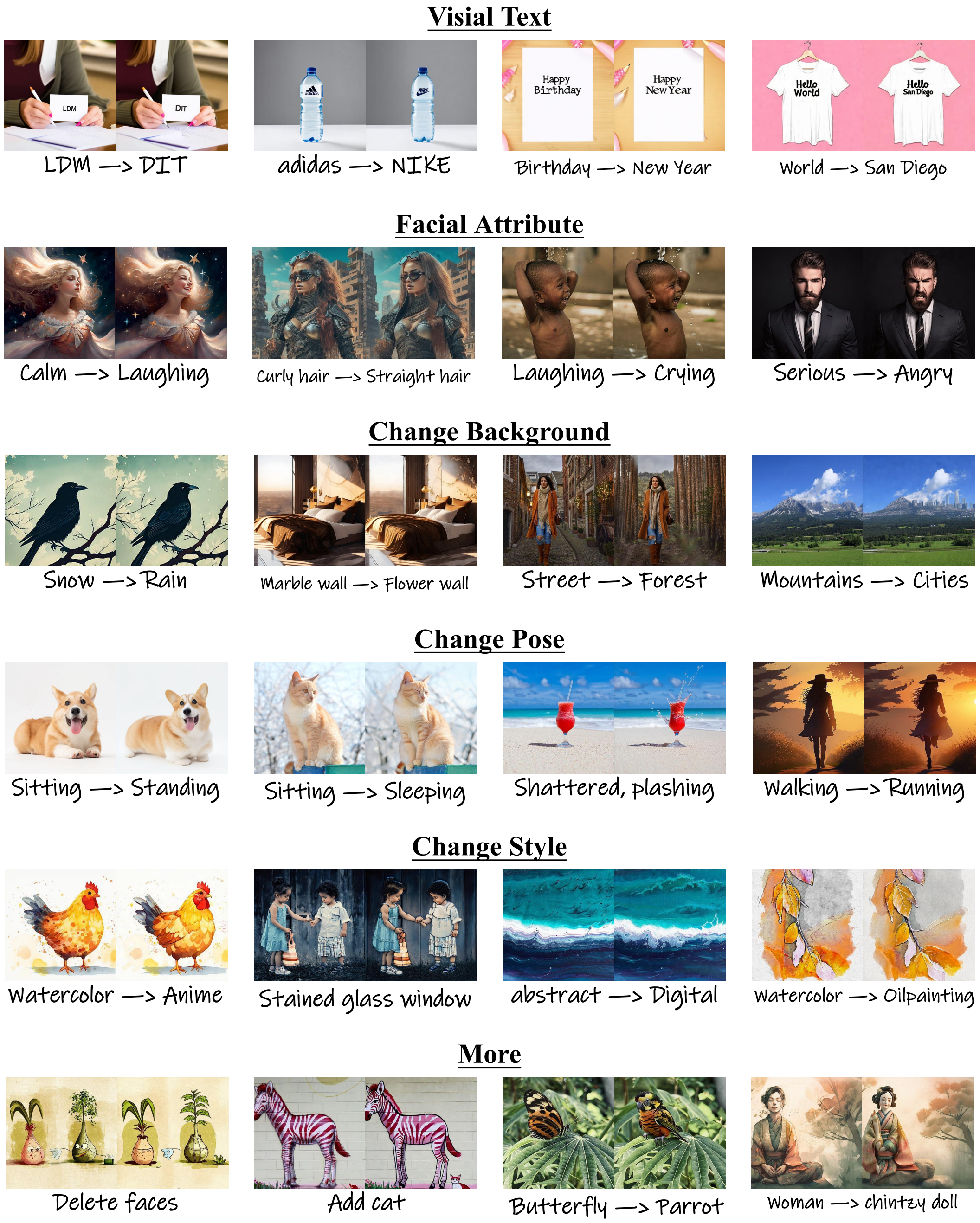


Figure 14: Additional qualitative results on diverse editing types from PIE-Bench Part II.