MATCHEDIT: MATCHED-STATE PROMPT DIRECTIONS FOR INVERSION-FREE FLOW-BASED IMAGE EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Inversion-free flow-based image editing typically constructs an editing direction by computing the difference between source and target velocity predictions. However, these predictions are evaluated at different latent states and under different prompts. Consequently, this difference captures not only the desired prompt change, but also the change in latent state. Through a controlled state-prompt decomposition, we show that this state effect strengthens target semantics, but also increases source deviation and unintended changes outside the edit region. To address this issue, we introduce MatchEdit, an inversion-free image editing method based on matched-state prompt comparisons. By comparing the source and target prompts at the same latent state, MatchEdit removes the additive state effect from the base editing direction and yields a more spatially selective edit. We then adaptively enhance target semantics using a strictly bounded correction guided by the spatial response of the edit. MatchEdit achieves state-of-the-art source preservation on PIE-Bench while maintaining competitive semantic alignment, and further generalizes to long-prompt editing on DNA-Bench and sequential multi-turn editing on MSE-Bench.

1 INTRODUCTION

Text-guided image editing aims to modify a real image according to a target prompt while preserving source content unrelated to the desired change. Recent diffusion and flow-based generative models have substantially improved the quality and flexibility of image editing (Ho et al., 2020; Rombach et al., 2022; Esser et al., 2024). However, achieving significant semantic transformations while preserving the original image remains challenging. Applying stronger edits often degrades source structure and appearance, whereas stronger preservation can suppress the intended semantic change.

Existing real-image editing methods are broadly divided into inversion-based and inversion-free approaches. Inversion-based methods first recover a latent, noise, or generation trajectory corresponding to the source image, and then perform editing based on the recovered representation (Hertz et al., 2023; Tumanyan et al., 2023; Cao et al., 2023; Koo et al., 2024; Wu et al., 2024). Although effective, inversion introduces additional computation and often requires further test-time guidance to preserve source content. Recent inversion-free methods instead operate directly within the velocity field of a pretrained flow model. FlowEdit (Kulikov et al., 2025), for example, constructs the editing dynamics from the difference between source and target velocity predictions. Subsequent methods improve this formulation by regularizing the editing trajectory (Kim et al., 2026) or incorporating source-aware feature interactions (Yang et al., 2026). These methods avoid explicit source reconstruction and provide a simple and efficient alternative for real-image editing.

However, the composition of this velocity difference has received considerably less attention. In the commonly used two-branch formulation, the source velocity is evaluated at a source latent state under the source prompt, whereas the target velocity is evaluated at a different latent state under the target prompt. The resulting difference therefore conflates two varying factors: the latent state and the text condition. Consequently, the editing direction reflects not only the desired prompt change, but also state-induced structural shifts. Improving the trajectory or manipulating intermediate features does not remove this variation from the editing signal itself.

To understand its effect, we construct a controlled state-prompt comparison by evaluating both source and target prompts at both latent states, as illustrated in Fig. 1. This construction sepa-

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

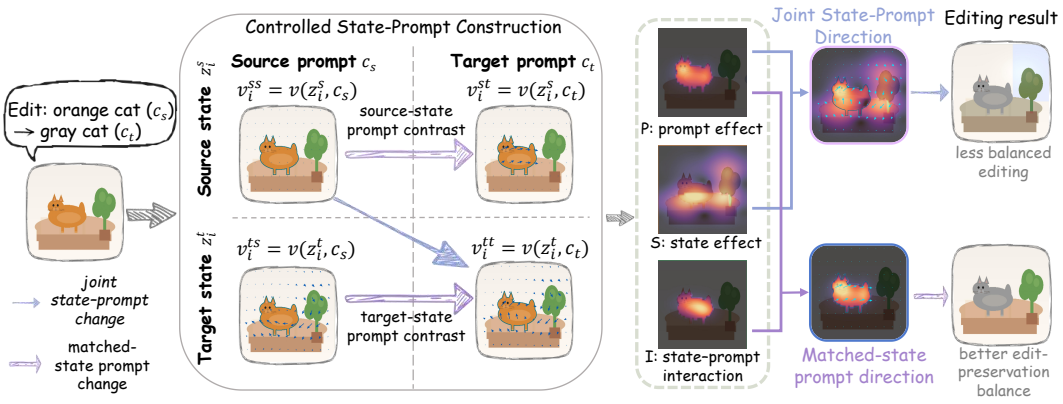


Figure 1: JSPD versus matched-state prompt directions. JSPD changes both latent state and prompt, whereas matched-state contrasts change only the prompt at fixed state. Brighter yellow indicates stronger directional response, orange indicates moderate response, and dark purple indicates weak response. The matched-state direction concentrates its response within the intended edit region.

rates prompt-induced variation from latent-state variation in the conventional source-target velocity difference. Our analysis shows that the latent-state variation introduces an additive state effect into the editing direction. This effect strengthens target-semantic responses, but it also increases source deviation and produces stronger responses outside the intended edit region. In contrast, comparing the source and target prompts at the same latent state removes this additive state effect and yields a more spatially localized editing direction.

These observations suggest that the state effect should not be retained globally in the editing direction. Instead, its useful semantic contribution should be recovered only where additional target guidance is needed. Based on this idea, we introduce MatchEdit, an inversion-free image editing method built on matched-state prompt comparisons. MatchEdit uses the matched-state prompt direction as its base editing signal, thereby removing the additive state effect from the global update. To recover target-semantic strength, it further applies a bounded target-conditioned correction according to the spatial response of the editing direction. This correction is applied selectively and constrained relative to the local target-conditioned velocity, preventing additional guidance from dominating the base direction. MatchEdit requires no image inversion, test-time optimization, external segmentation masks, or manipulation of intermediate Transformer features.

The main contributions are summarized as follows:

- We propose MatchEdit, an inversion-free flow-based image editing method that constructs the editing direction from matched-state prompt comparisons and selectively reinforces target semantics, improving source preservation while maintaining editability.
- We provide a controlled state-prompt analysis of conventional source-target velocity differences and show that latent-state variation introduces an additive state effect that strengthens semantic responses but also increases source deviation and off-target changes.
- MatchEdit requires no image inversion, test-time optimization, external segmentation masks, or intermediate feature manipulation, while achieving state-of-the-art source preservation on PIE-Bench and generalizing to long-prompt and sequential multi-turn editing.

2 Related Work

Inversion-based Image Editing. Many real-image editing methods first recover a latent, noise, or generation trajectory corresponding to the source image and then perform editing based on the recovered representation. Diffusion-based approaches improve inversion or source fidelity through embedding optimization, negative prompting, source-aware inversion, or the reuse of intermediate attention maps and features (Mokady et al., 2023; Miyake et al., 2025; Ju et al., 2024; Hertz et al., 2023; Tumanyan et al., 2023; Cao et al., 2023). Inversion has also been extended to rectified-flow

models, where recent methods improve numerical accuracy and efficiency (Wang et al., 2025; Deng et al., 2024) or further refine the recovered representation and trajectory (Xu et al., 2025; Xie et al., 2025; Wang et al., 2026). While these methods improve reconstruction and editing fidelity, they still require an explicit recovery stage before editing and may additionally depend on model-specific features or sampling procedures.

Inversion-free Image Editing with Rectified Flows. Recent methods avoid explicit inversion and instead operate directly within the velocity field of pretrained rectified-flow models. FlowEdit (Kulikov et al., 2025) constructs an editing ODE from the difference between source and target velocity predictions. Subsequent methods improve this paradigm by regularizing the editing trajectory (Kim et al., 2026; Beaudouin et al., 2026), introducing source-aware feature interactions (Yang et al., 2026), or strengthening semantic guidance along the flow trajectory (Tang et al., 2026).

Despite these differences, the source-target velocity difference remains the basic editing signal in many inversion-free formulations. The two velocities, however, are typically evaluated at different latent states and under different prompts, so their difference reflects changes in both state and text condition. Our work focuses on this editing signal itself. By comparing source and target prompts at matched latent states, MatchEdit separates prompt-induced variation from the additive effect of latent-state change and uses the resulting matched-state direction as the basis for editing.

3 Preliminaries

We first introduce the rectified-flow formulation and the inversion-free editing states used throughout our analysis. Rectified flow defines a continuous transport between a data distribution and a simple noise distribution through the ordinary differential equation

$$dz_\tau = v_\theta(z_\tau, \tau, c)\, d\tau, \qquad \tau \in [0, 1], \tag{1}$$

where v_θ denotes the pretrained text-conditioned velocity field, τ is the flow time, and c is the text condition. Its marginal states follow the standard linear probability path between data and Gaussian noise (Liu et al., 2022).

Given a source image I_s, let c_s and c_t denote the source and target text prompts, respectively. We denote the encoded source latent by $x_s = E_{\mathrm{VAE}}(I_s)$ and initialize the editing latent as $y_0 = x_s$. Following inversion-free flow editing (Kulikov et al., 2025), at editing step i we construct a noisy source state as

$$z_i^s = (1 - \tau_i)x_s + \tau_i \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, I), \tag{2}$$

where τ_i denotes the current flow timestep. The corresponding target state is

$$z_i^t = y_i + z_i^s - x_s. \tag{3}$$

The conventional inversion-free editing direction compares the source-conditioned velocity at z_i^s with the target-conditioned velocity at z_i^t. We next examine what variations are contained in this source-target difference.

4 Analysis of Editing Direction Composition

To build intuition, consider changing an orange cat to a gray cat, as shown in Fig. 1. The conventional source-target velocity difference changes both the prompt and latent state, mixing the desired prompt-induced edit with variation caused by the evolving image state. To disentangle them, we evaluate both prompts at both latent states (additional analysis in Appendix A.1).

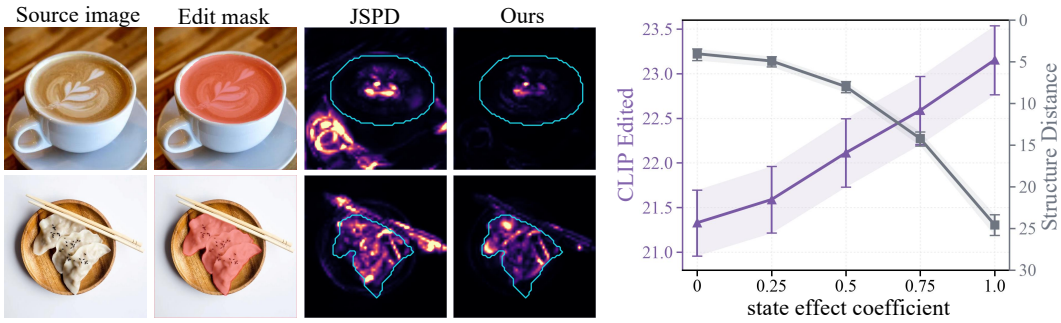
4.1 Controlled State-Prompt Decomposition

For each paired state and prompt, we define

$$v_i^{ab} = v_\theta(z_i^a, \tau_i, c_b), \qquad a, b \in \{s, t\}, \tag{4}$$

where the first superscript indexes the latent state and the second indexes the prompt condition. Under this notation, the conventional source-target direction becomes

$$D_i^{\mathrm{JSPD}} = v_i^{tt} - v_i^{ss}, \tag{5}$$



(a) Spatial localization of editing directions

(b) Effect of the direct state contribution

Figure 2: State effect and spatial selectivity of editing directions. (a) The matched-state prompt direction concentrates a larger fraction of response magnitude inside the intended edit region than JSPD; cyan contours denote the edit regions. (b) Increasing the state-effect coefficient improves target-semantic alignment but also increases structural deviation. Shaded regions and error bars indicate 95% CIs.

which we refer to as the Joint State-Prompt Direction (JSPD). To characterize the contributions of state and prompt changes, we define the prompt effect:

$$P_i = \frac{1}{2}\left[\left(v_i^{st} - v_i^{ss}\right) + \left(v_i^{tt} - v_i^{ts}\right)\right], \tag{6}$$

the state effect:

$$S_i = \frac{1}{2}\left[\left(v_i^{ts} - v_i^{ss}\right) + \left(v_i^{tt} - v_i^{st}\right)\right], \tag{7}$$

and the state-prompt interaction:

$$I_i = \left(v_i^{tt} - v_i^{ts}\right) - \left(v_i^{st} - v_i^{ss}\right). \tag{8}$$

Here, P_i captures the average prompt-induced semantic response, S_i the direct effect of changing the latent state, and I_i the state-dependent variation of the prompt response.

We then compare the source and target prompts while holding the latent state fixed:

$$d_i^s = v_i^{st} - v_i^{ss}, \qquad d_i^t = v_i^{tt} - v_i^{ts}, \tag{9}$$

and combine the two matched-state prompt contrasts as

$$D_i^{\text{match}} = \alpha d_i^s + (1-\alpha) d_i^t, \qquad \alpha \in [0,1]. \tag{10}$$

By direct substitution,

$$D_i^{\text{JSPD}} = P_i + S_i, \qquad D_i^{\text{match}} = P_i + \left(\frac{1}{2} - \alpha\right) I_i. \tag{11}$$

Thus, JSPD contains the full additive state effect S_i, whereas the matched-state direction removes S_i and retains only an α-controlled component of the state-prompt interaction I_i. Fig. 1 directly visualizes this decomposition. The broader response associated with S_i appears as off-target activation in JSPD, while the matched-state direction is more concentrated within the intended edit region.

4.2 Spatial Selectivity

We next examine where the two editing directions act. For $D_i \in \mathbb{R}^{C\times H\times W}$, we define its squared spatial response as

$$E_{D_i}(h,w) = \sum_{c=1}^{C} |D_i(c,h,w)|^2. \tag{12}$$

Using the edit-region mask M_e, we measure the fraction of the response inside the intended edit region with the Edit-Region Magnitude Ratio (EMR):

$$\mathrm{EMR}(D_i) = \frac{\sum_{h,w} M_e(h,w) E_{D_i}(h,w)}{\sum_{h,w} E_{D_i}(h,w) + \epsilon}. \tag{13}$$

The PIE-Bench edit mask M_e is used only for analysis and evaluation and is not used during editing. As shown in Fig. 2(a), the average EMR increases from 55.8% for JSPD to 62.0% for the matched-state direction. Removing the additive state effect therefore concentrates more of the editing response inside the intended region.

4.3 Impact of the State Effect

The preceding analysis shows where the additive state effect acts. We next examine what it contributes to editing. To isolate its role, we vary only the coefficient of S_i while excluding the interaction term:

$$D_i(\lambda_S) = P_i + \lambda_S S_i, \qquad \lambda_S \in \{0, 0.25, 0.50, 0.75, 1.00\}. \tag{14}$$

Fig. 2(b) shows a clear trade-off. As λ_S increases, CLIP Edited rises from 21.33 to 23.15, whereas Structure Distance increases from 4.04×10^{-3} to 24.57×10^{-3}. The state effect therefore provides additional semantic strength, but retaining it globally also causes substantially larger source deviation. Taken together, these results suggest a simple design principle: use the matched-state direction as the editing base, and recover additional target semantics selectively rather than retaining the state effect throughout the image.

5 MatchEdit

Our analysis suggests a simple design principle: *remove the state effect globally and recover target semantics selectively*. Based on this principle, MatchEdit consists of two complementary components. First, we use the matched-state prompt direction D_i^{match} in Eq. 10 as a preservation-oriented base direction. It removes the additive state effect S_i while retaining state-dependent prompt variation through the matched-state contrasts. Second, because removing S_i also weakens part of the target-semantic response, we introduce **direction-aware bounded modulation** to selectively restore target guidance according to the spatial response of the editing direction. The modulation strengthens target semantics in edit-responsive regions while locally bounding the added guidance to avoid excessive source deviation.

5.1 Directional Response

The matched-state direction naturally provides a soft spatial response map: locations involved in the requested edit tend to exhibit stronger directional responses, while unrelated regions remain weaker. We therefore use this directional response to estimate where additional target guidance is needed.

Let $d_i^{\mathrm{final}} \in \mathbb{R}^{C\times H\times W}$ denote the final editing direction. We compute its channel-wise magnitude as $A_i(h,w) = \frac{1}{C}\sum_{c=1}^{C} |d_i^{\mathrm{final}}(c,h,w)|$ and normalize it as $Q_i(h,w) = (A_i(h,w) - m_i)/(\mathrm{MAD}_i + \varepsilon)$, where $m_i = \mathrm{Median}_{h,w}\, A_i(h,w)$ and $\mathrm{MAD}_i = \mathrm{Median}_{h,w}\, |A_i(h,w) - m_i|$. The normalized response is mapped to $[0,1]$ using a fixed monotone squashing function ψ followed by local smoothing:

$$G_i = \mathrm{clip}\left(\mathrm{AvgPool}\left(\psi(Q_i)\right), 0, 1\right) = \mathcal{R}\left(d_i^{\mathrm{final}}\right). \tag{15}$$

To avoid a circular dependency during computation, we use the spatial response from the preceding editing step, $G_{i-1} = \mathcal{R}(d_{i-1}^{\mathrm{final}})$. We initialize $G_{-1} = 0$ such that the first effective editing step relies solely on the matched-state base direction. The response is computed directly from the editing direction and requires neither external segmentation masks nor access to intermediate Transformer features.

5.2 Bounded Semantic Correction

To apply stronger semantic corrections to high-response regions, for each state $a \in \{s, t\}$, we map G_{i-1} to a nonnegative local weight $g : [0, 1] \to \mathbb{R}_{\geq 0}$, with $g(0) = 0$, and define:

$$r_i^a = s_{\text{cfg}}\, g(G_{i-1}) \odot [v_c(z_i^a, \tau_i, c_t) - v_u(z_i^a, \tau_i)], \tag{16}$$

where v_c and v_u denote the conditional and unconditional velocity predictions, respectively, s_{cfg} is the guidance scale, and $\odot$ denotes spatial element-wise multiplication. Consequently, regions with a strong spatial response naturally receive larger semantic corrections.

To ensure that these localized additions do not overwhelm the global velocity field and destroy image coherence, we introduce a pixel-wise safety budget $B_i^a(p) = \rho(G_{i-1}(p))\|v_i^{at}(p)\|_2$, where p indexes spatial locations and $\rho : [0, 1] \to \mathbb{R}_{\geq 0}$ is a fixed nondecreasing mapping. We then project the unconstrained correction $r_i^a(p)$ onto the ℓ_2-ball of radius $B_i^a(p)$. This Euclidean projection admits a closed-form solution:

$$\Delta_i^a(p) = \min\left(1, \frac{B_i^a(p)}{\|r_i^a(p)\|_2 + \varepsilon}\right) r_i^a(p). \tag{17}$$

By definition of this projection, $\|\Delta_i^a(p)\|_2 \leq \rho(G_{i-1}(p))\|v_i^{at}(p)\|_2$, so the added target guidance remains locally bounded relative to the target-conditioned velocity.

We use the same mixing coefficient α for the semantic corrections to preserve the source-target weighting used in the matched-state base direction. The source- and target-state corrections are then combined as

$$d_i^{\text{final}} = D_i^{\text{match}} + \alpha\Delta_i^s + (1 - \alpha)\Delta_i^t. \tag{18}$$

Finally, we advance the editing trajectory using a standard first-order Euler update:

$$y_{i+1} = y_i + (\tau_{i+1} - \tau_i)\left(d_i^{\text{final}} + \tau_i\epsilon_i\right), \qquad \epsilon_i \sim \mathcal{N}(0, I). \tag{19}$$

After the final editing step, the edited image is obtained by decoding y_K with the pretrained VAE decoder. MatchEdit requires no image inversion, test-time optimization, external segmentation masks, or intermediate Transformer feature modification (additional details in Appendix A.2).

6 Experiments

Benchmarks and Metrics. We conduct our main evaluation on PIE-Bench (Ju et al., 2024), which contains 700 image-prompt pairs across 10 editing categories. We further evaluate long-prompt editing on DNA-Bench (Xie et al., 2025) and sequential multi-turn editing on MSE-Bench (Qu et al., 2026). Following the PIE-Bench protocol, we evaluate source preservation using Structure Distance, PSNR, LPIPS (Zhang et al., 2018), MSE, and SSIM (Wang et al., 2004). CLIP (Radford et al., 2021) Whole and Edited measure target-text alignment over the full image and the intended edit region, respectively.

Implementation Details. We implement MatchEdit on Stable Diffusion 3.5 using the official pretrained weights. All experiments use 50 scheduler steps with a first-order Euler solver, skipping the first 17 steps and performing 33 effective editing steps with $\alpha = 0.3$. Unless otherwise specified, the same hyperparameters are used across all images and editing categories (additional experimental details, efficiency analysis, extended results, and ablation studies are provided in Appendix A.3-A.7).

Competing Methods. We compare MatchEdit with representative diffusion- and flow-based image editing methods, including Prompt-to-Prompt (Hertz et al., 2023), PnP (Tumanyan et al., 2023), MasaCtrl (Cao et al., 2023), LEDITS++ (Brack et al., 2024), FreeDiff (Wu et al., 2024), RF-Inversion (Rout et al., 2025), StableFlow (Avrahami et al., 2025), RF-Edit (Wang et al., 2025), DCEdit (Hu et al., 2025), FTEdit (Xu et al., 2025), FlowEdit (Kulikov et al., 2025), DNAEdit (Xie et al., 2025), FlowAlign (Kim et al., 2026), FSI-Edit (Yang et al., 2025), and FIA-Edit (Yang et al., 2026). For FLUX-based methods, we report PIE-Bench results provided by FIA-Edit (Yang et al., 2026).

Table 1: Quantitative comparison on PIE-Bench. The best/second/third results in each numeric column are highlighted with yellow / orange / blue backgrounds, respectively.

Method	Model	Structure	Background Preservation				CLIP Similarity		Rank
		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 $\uparrow$	Avg. $\downarrow$
P2P + DDIM (ICLR, 2023)	SD1.4	70.59	17.87	208.85	219.61	71.64	25.28	22.57	15.57
P2P + Direct Inv. (ICLR, 2024)	SD1.4	10.96	27.28	53.81	32.50	85.34	25.34	22.17	6.29
PnP + DDIM (CVPR, 2023)	SD1.5	27.32	22.31	112.78	82.99	79.58	25.60	22.58	12.29
PnP + Direct Inv. (ICLR, 2024)	SD1.5	23.49	22.49	105.49	79.91	80.25	25.79	22.64	10.43
MasaCtrl (ICCV, 2023)	SD1.4	23.67	22.66	87.25	80.67	81.92	24.70	21.45	13.00
LEDITS++ (CVPR, 2024)	SD1.5	15.46	24.49	71.68	52.89	83.11	23.87	21.00	10.57
FreeDiff (ECCV, 2024)	SD1.4	33.00	22.36	121.20	89.63	79.30	25.46	22.47	13.86
RF-Inversion (ICLR, 2025)	FLUX	48.76	19.51	195.85	155.74	68.95	25.11	22.50	15.71
StableFlow (CVPR, 2025)	FLUX	19.24	23.04	76.94	84.85	87.22	24.30	21.28	11.43
RF-Edit (ICML, 2025)	FLUX	27.70	23.22	131.18	75.00	81.44	25.22	22.40	12.86
DCEdit (arXiv, 2025)	FLUX	22.36	25.41	94.17	48.09	85.60	25.47	22.71	7.43
FTEdit (CVPR, 2025)	SD3.5	21.67	23.93	94.28	56.48	89.10	25.25	21.93	9.86
FlowEdit (ICCV, 2025)	SD3.5	23.62	23.22	93.74	69.95	85.10	**26.76**[1]	**23.70**[1]	7.57
DNAEdit (NeurIPS, 2025)	SD3.5	14.87	26.64	74.89	32.75	88.90	25.63	22.71	4.71
FlowAlign (ICLR, 2026)	SD3.5	31.00	24.59	64.38	48.55	86.78	25.72	22.47	7.71
FSI-Edit (NeurIPS, 2025)	SD3.5	13.77	26.61	84.83	36.53	86.26	25.31	22.40	7.43
FIA-Edit (AAAI, 2026)	SD3.5	10.34[2]	27.32[2]	55.01	28.67[2]	89.20[2]	25.89	22.81[2]	2.29
MatchEdit	SD3.5	**7.45**[1]	**29.02**[1]	**46.15**[1]	**19.14**[1]	**90.32**[1]	26.03[2]	22.71[3]	**1.43**[1]
Δ *vs. best baseline*		−2.89	+1.70	−7.66	−9.53	+1.12	−0.73	−0.99	−0.86

6.1 Comparison with Prior Methods

Quantitative Results. Tab. 1 compares MatchEdit with existing methods on PIE-Bench. MatchEdit achieves the best result on all five source-preservation metrics and the best average rank of 1.43. Compared with the strongest baseline, it reduces Structure Distance, LPIPS, and MSE by 2.89, 7.66, and 9.53, respectively, while improving PSNR by 1.70 dB and SSIM by 1.12 points. Its CLIP scores remain competitive, ranking second on CLIP Whole and third on CLIP Edited. These results show that MatchEdit better preserves source content without preventing the intended semantic edit.

Qualitative Results. Fig. 3 shows representative editing results on PIE-Bench. MatchEdit performs the requested edits while retaining source content outside the intended edit region. For example, it adds the floral pattern to the carpet without changing the surrounding floor tiles. Similar behavior is observed for background, where MatchEdit better preserves foreground structure, facial appearance, pose, and global composition. Competing methods more often either produce weaker target changes or modify unrelated source content. (additional qualitative comparisons in Appendix A.5).

6.2 Ablation Studies

Editing-Direction Construction. As shown in Tab. 2, JSPD achieves the strongest CLIP scores but also causes the largest source deviation. The source-state contrast favors preservation, while

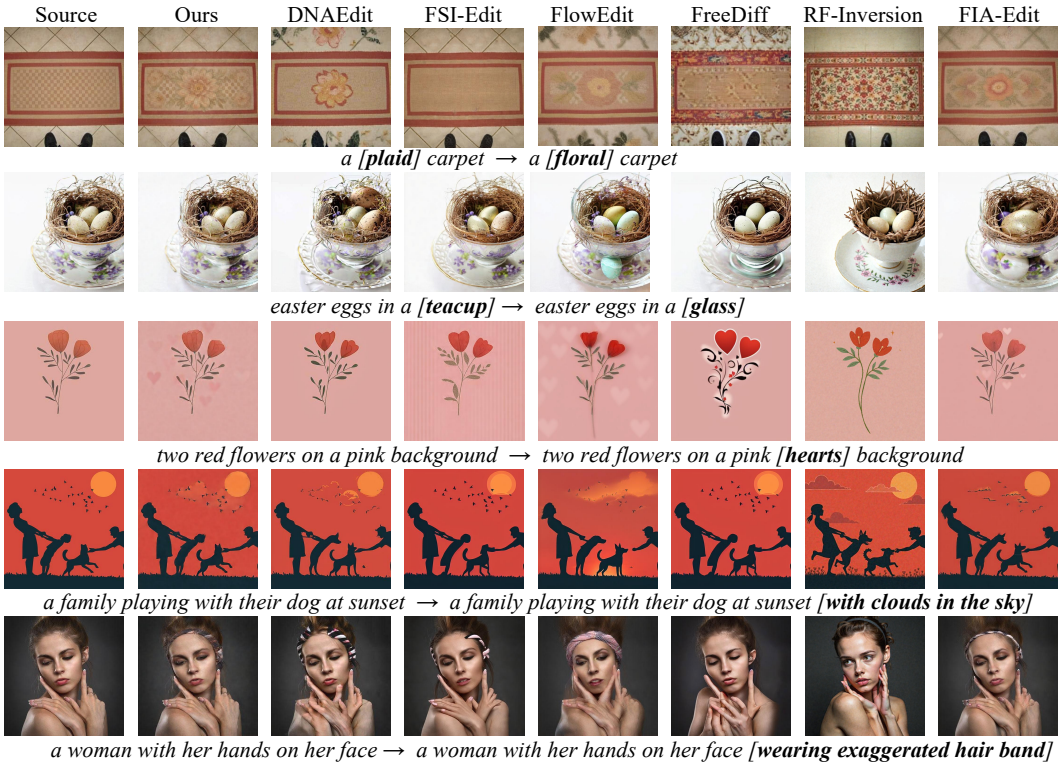


Figure 3: Qualitative comparison on PIE-Bench.

Table 2: Direction construction ablation on PIE-Bench without bounded modulation. Structure, Preservation, and CLIP.

Direction	Distance $\times 10^3 \downarrow$	PSNR ↑	LPIPS $\times 10^3 \downarrow$	MSE $\times 10^4 \downarrow$	SSIM $\times 10^2 \uparrow$	Whole ↑	Edited ↑
JSPD	24.54	23.11	90.38	75.21	85.00	**26.39**	**23.17**
source-state prompt contrast	<u>4.56</u>	**31.47**	**33.12**	<u>11.65</u>	**91.98**	23.67	20.72
target-state prompt contrast	5.24	30.96	36.99	12.88	91.66	<u>25.19</u>	<u>21.86</u>
matched-state prompt direction	**4.40**	**31.47**	<u>34.01</u>	**11.53**	<u>91.96</u>	24.81	21.60

Table 3: Component ablation on PIE-Bench. Structure, Preservation, and CLIP.

Matched-State Prompt Direction	Bounded Modulation	Distance $\times 10^3 \downarrow$	PSNR ↑	LPIPS $\times 10^3 \downarrow$	MSE $\times 10^4 \downarrow$	SSIM $\times 10^2 \uparrow$	Whole ↑	Edited ↑
×	×	24.54	23.11	90.38	75.21	85.00	**26.39**	**23.17**
✓	×	**4.40**	**31.47**	**34.01**	**11.53**	**91.96**	24.81	21.60
✓	✓	<u>7.45</u>	<u>29.02</u>	<u>46.15</u>	<u>19.14</u>	<u>90.32</u>	<u>26.03</u>	<u>22.71</u>

the target-state contrast retains stronger semantics. Combining the two yields the lowest Structure Distance and MSE while preserving more semantic strength than the source-state contrast alone. These results support the analysis in Section 4: the additive state effect strengthens editing but increases source deviation, whereas matched-state comparisons remove this global contribution while retaining state-dependent prompt responses.

Component Contributions. As shown in Tab. 3, replacing JSPD with the matched-state direction reduces Structure Distance and MSE by 20.14 and 63.68, respectively, at the cost of 1.58/1.57 lower CLIP Whole/Edited scores. Adding bounded modulation recovers 1.22/1.11 in the two CLIP

Table 4: Overall results on DNA-Bench. Structure , Preservation , and CLIP .

Method	Distance $\times 10^3 \downarrow$	PSNR ↑	LPIPS $\times 10^3 \downarrow$	MSE $\times 10^4 \downarrow$	SSIM $\times 10^2 \uparrow$	Whole ↑	Edited ↑
FlowEdit	25.12	22.53	101.31	80.68	84.04	29.11	**23.67**
DNAEdit	14.21	26.37	74.74	34.67	88.62	28.25	22.89
FIA-Edit	11.05	26.83	59.21	32.13	88.63	28.39	22.69
MatchEdit	**7.28**	**29.16**	**44.85**	**18.59**	**90.41**	**28.40**	22.45

Table 5: Results on MSE-Bench. Cumulative alignment and Turn-5 outcomes .

Method	Turn 1 CLIP ↑	Turn 5 CLIP ↑	Δ CLIP ↓	Prior Ret. ↑	SSIM ↑	MSE ↓
FlowEdit	30.08	28.15	1.933	22.42	48.15	0.0634
FIA-Edit	29.90	28.27	1.633	22.55	56.10	0.0327
MatchEdit	**30.49**	**29.81**	**0.688**	**22.96**	**57.59**	**0.0215**

scores while retaining substantially better source preservation than JSPD. Thus, the matched-state direction provides the main preservation gain, while bounded modulation recovers much of the semantic strength removed with the state effect.

6.3 Generalization Analysis

Long-Prompt Editing. We evaluate more complex editing instructions on DNA-Bench (Xie et al., 2025). As shown in Tab. 4, MatchEdit again achieves the best result on all five preservation metrics among the compared SD3.5-based methods. It reduces Structure Distance from 11.05 for FIA-Edit to 7.28 and improves PSNR from 26.83 to 29.16, while maintaining comparable target-text alignment. Similar gains are observed for LPIPS, MSE, and SSIM. The preservation advantage of MatchEdit therefore extends beyond the shorter editing instructions used in PIE-Bench. (additional results in Appendix A.7).

Sequential Multi-Turn Editing. We further evaluate MatchEdit on MSE-Bench (Qu et al., 2026), where the output of each editing turn becomes the input to the next. To measure the retention of previously introduced semantics, we define $\mathrm{PriorRet}(t) = \frac{1}{t-1}\sum_{k=1}^{t-1}\mathrm{CLIP}(I_t, p_k)$, where p_k denotes the target introduced at turn k. As shown in Tab. 5, MatchEdit retains semantic and structural information more effectively over successive edits. Its CLIP score decreases by only 0.69 from Turn 1 to Turn 5, compared to 1.93 for FlowEdit and 1.63 for FIA-Edit. At Turn 5, MatchEdit also achieves the highest prior-retention score and SSIM and the lowest MSE. These results show that reducing unnecessary source changes at each editing step becomes particularly beneficial when deviations accumulate over multiple turns. (additional results in Appendix A.7).

7 Conclusion and Limitations

We introduced MatchEdit, an inversion-free method for text-guided image editing with pretrained rectified-flow models. Our analysis shows that the conventional source-target velocity difference contains an additive state effect that strengthens target semantics but also increases source deviation. MatchEdit removes this state effect from the base editing direction through matched-state prompt comparisons and selectively restores semantic strength where additional target guidance is needed. Experiments show state-of-the-art source preservation on PIE-Bench, with consistent advantages on long-prompt and sequential multi-turn editing. MatchEdit favors source preservation and may therefore be less suitable for large geometric or global edits that require substantial changes to the source image. Its current semantic correction also relies on the response from the preceding editing step, which can provide limited guidance at the beginning of editing or for spatially extensive changes. Addressing these cases is an interesting direction for future work.

AI USE STATEMENT

LLMs were used to refine the language and improve presentation. They were not used to develop the technical content of this work. All scientific content was produced and verified by the authors.

REPRODUCIBILITY STATEMENT

We provide implementation details and experimental settings in the paper. We plan to release the source code upon acceptance to facilitate reproducibility.

REFERENCES

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. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7877–7888. IEEE, 2025.

Gaspard Beaudouin, Minghan Li, Jaeyeon Kim, Sung-Hoon Yoon, and Mengyu Wang. Delta rectified flow sampling for text-to-image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18662–18672, 2026.

Manuel Brack, Felix Friedrich, Katharia Kornmeier, Linoy Tsaban, Patrick Schramowski, Kristian Kersting, and Apolinário Passos. Ledits++: Limitless image editing using text-to-image models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8861–8870, 2024.

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 *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22503–22513. IEEE, 2023.

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.

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.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross-attention control. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=_CDixzkzeyb.

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

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.

Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=FoMZ4ljhVw.

Jeongsol Kim, Yeobin Hong, Jonghyun Park, and Jong Chul Ye. Flowalign: Trajectory-regularized, inversion-free flow-based image editing. In *International Conference on Learning Representations*, volume 2026, pp. 121814–121846, 2026.

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*, pp. 363–379. Springer, 2024.

Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. Flowedit: Inversion-free text-based editing using pre-trained flow models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19721–19730. IEEE, 2025.

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.

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)*, pp. 2063–2072. IEEE, 2025.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6038–6047. IEEE, 2023.

Leigang Qu, Feng Cheng, Ziyan Yang, Qi Zhao, Shanchuan Lin, Yichun Shi, Yicong Li, Wenjie Wang, Tat-Seng Chua, and Lu Jiang. Vincie: Unlocking in-context image editing from video. In *International Conference on Learning Representations*, volume 2026, pp. 114873–114918, 2026.

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*, pp. 8748–8763. PmLR, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Litu Rout, Yujia Chen, Nataniel Ruiz, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Semantic image inversion and editing using rectified stochastic differential equations. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Hu0FSOSEyS.

Anqi Tang, Wenhao Sun, and Zhaoqiang Liu. Wavelet-guided semantic signal compensation for inversion-free image editing. *arXiv preprint arXiv:2607.02421*, 2026.

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1921–1930. IEEE, 2023.

Chenru Wang, Beier Zhu, and Chi Zhang. Free lunch for stabilizing rectified flow inversion. In *International Conference on Learning Representations*, volume 2026, pp. 43792–43822, 2026.

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. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 64044–64058. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/wang25ce.html.

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.

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*, pp. 194–209. Springer, 2024.

Chenxi Xie, Minghan Li, Shuai Li, Yuhui Wu, Qiaosi Yi, and Lei Zhang. DNAEdit: Direct noise alignment for text-guided rectified flow editing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=JxBA9OJExP.

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. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28479–28489. IEEE, 2025.

Kaixiang Yang, Xin Li, Yuxi Li, Qiang Li, and Zhiwei Wang. Fsi-edit: Frequency and stochasticity injection for flexible diffusion-based image editing. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 133100–133132. Curran Associates, Inc., 2025. doi: 10.52202/085713-4434. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/c0d7b7a75dc49664bfc6ac9cb935792d-Paper-Conference.pdf.

Kaixiang Yang, Boyang Shen, Xin Li, Yuchen Dai, Yuxuan Luo, Yueran Ma, Wei Fang, Qiang Li, and Zhiwei Wang. Fia-edit: Frequency-interactive attention for efficient and high-fidelity inversion-free text-guided image editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 11613–11621, 2026.

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

A Appendix

Contents

A.1 Additional Details of Editing-Direction Composition 12
A.1.1 Spatial Response Visualization 12
A.1.2 State-Effect Coefficient Analysis 13
A.2 Additional Method Details 15
A.2.1 Details of Direction-Aware Bounded Modulation 15
A.2.2 Proof of Local Boundedness 15
A.2.3 Algorithm 16
A.3 Additional Experimental Settings 16
A.4 Efficiency Analysis 16
A.5 Additional PIE-Bench Results and Analysis 17
A.6 Additional Ablation Results 17
A.6.1 Effect of the Fusion Weight α 17
A.6.2 Component Ablation Examples 18
A.7 Additional Generalization Results 22
A.7.1 DNA-Bench Results (Long-Prompt Editing) 22
A.7.2 MSE-Bench Results (Multi-Turn Editing) 22

A.1 Additional Details of Editing-Direction Composition

A.1.1 Spatial Response Visualization

To validate our structural decomposition, we visualize the spatial response magnitudes of the different editing directions on PIE-Bench.

As illustrated in Fig. 4, each row corresponds to a distinct editing instruction, spanning object replacement, attribute modification, color adjustment, and appearance transformation. The highlighted

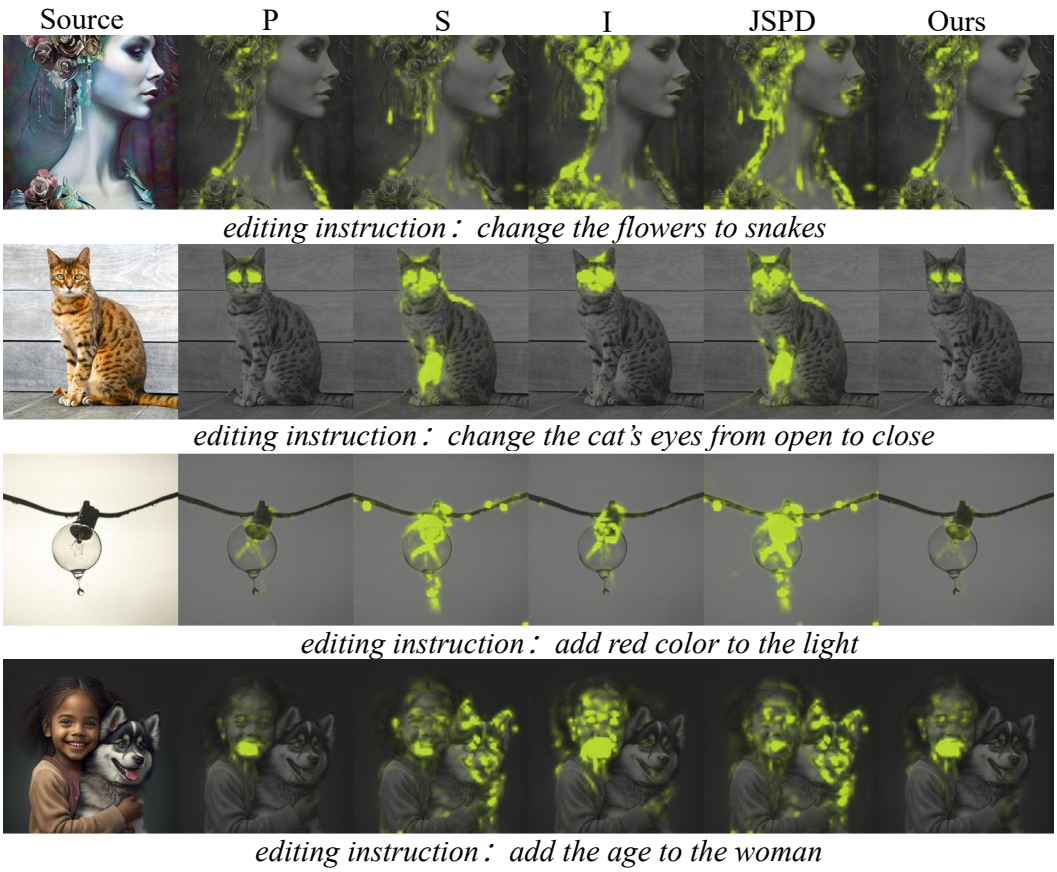


Figure 4: Spatial response visualization. S and JSPD exhibit broader responses, while ours is more concentrated on the intended edit regions.

regions indicate the spatial areas activated by each respective direction. Across all editing tasks, the pure prompt effect (P) and the state-prompt interaction (I) primarily activate target-relevant regions. Conversely, the pure state effect (S) introduces spurious responses that disperse broadly across the background and unedited regions. Because the Joint State-Prompt Direction (JSPD) absorbs this full state effect, it inherits these diffuse background activations. In contrast, our matched-state prompt direction strongly concentrates its response magnitude specifically within the intended edit regions, confirming that our algebraic decomposition yields a highly localized spatial signal.

A.1.2 State-Effect Coefficient Analysis

To systematically examine how the state effect influences the final editing result, we vary its coefficient:

$$\lambda_S \in \{0, 0.25, 0.50, 0.75, 1.00\}.$$

All five settings are evaluated across the full PIE-Bench dataset under identical conditions, with λ_S acting as the sole independent variable.

Tab. 6 reveals a clear preservation-editability trade-off as the coefficient of S increases. Compared to the pure prompt effect ($\lambda_S = 0$), fully injecting the state effect ($\lambda_S = 1.00$) increases CLIP Whole and CLIP Edited by 1.92 and 1.82, respectively. However, this comes at a substantial cost to source preservation: Structure Distance increases by 0.02053, PSNR drops by 8.56 dB, and SSIM decreases by 0.0707.

Fig. 5 visualizes these continuous coefficient trends. As λ_S increases, Structure Distance, LPIPS, and MSE increase, while PSNR and SSIM decrease; simultaneously, both CLIP metrics increase.

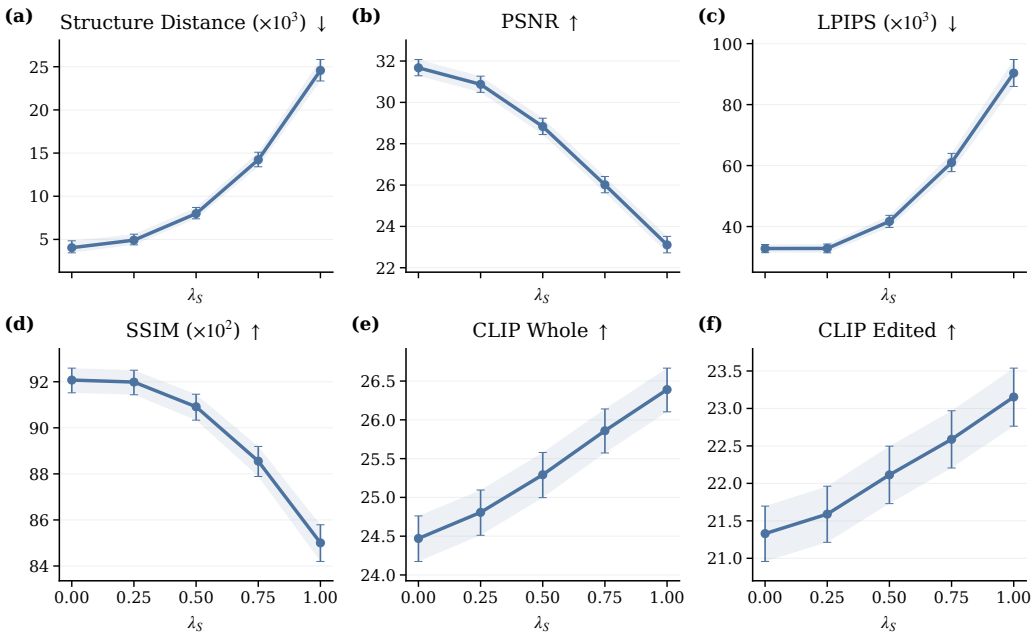


Figure 5: State effect coefficient curves for representative metrics.

To quantify these dynamics, Tab. 7 summarizes the linear trends across the coefficients. All five preservation metrics exhibit statistically significant slopes toward reduced fidelity, while both target CLIP metrics trend toward stronger semantic alignment. The corresponding 95% confidence intervals strictly exclude zero, indicating consistent coefficient-dependent effects across the dataset.

Tab. 8 further reports the paired metric changes relative to the pure prompt effect, confirming that the preservation loss and semantic gain become progressively more pronounced as the contribution of S expands.

Table 6: Final-image metrics under different state effect coefficients.

Direction	Structure ↓	PSNR ↑	LPIPS ↓	MSE ↓
P-only	**0.00404**	**31.67**	**0.03279**	**0.00111**
$+0.25S$	0.00491	30.87	<u>0.03282</u>	0.00130
$+0.50S$	0.00799	28.83	0.04168	0.00204
$+0.75S$	0.01422	26.01	0.06100	0.00388
$+1.00S$	0.02457	23.11	0.09036	0.00752

Direction	SSIM ↑	CLIP Whole ↑	CLIP Edited ↑
P-only	**0.9207**	24.47	21.33
$+0.25S$	<u>0.9199</u>	24.81	21.59
$+0.50S$	0.9091	25.29	22.11
$+0.75S$	0.8855	<u>25.86</u>	<u>22.59</u>
$+1.00S$	0.8500	**26.39**	**23.15**

Table 7: Linear trends across state effect coefficients.

Metric	Slope	95% CI
Distance $\times 10^3$	20.15	[19.14, 21.19]
PSNR	-8.80	[-9.08, -8.52]
LPIPS $\times 10^3$	57.33	[53.60, 61.03]
MSE $\times 10^4$	61.68	[56.94, 66.45]
SSIM $\times 10^2$	-7.03	[-7.47, -6.58]
CLIP Whole	1.96	[1.74, 2.17]
CLIP Edited	1.86	[1.62, 2.09]

Table 8: Metric changes compared to P-only.

Metric	$+0.25S$	$+0.50S$	$+0.75S$	$+1.00S$
Distance $\times 10^3$	0.87 [0.68, 1.02]	3.95 [3.57, 4.28]	10.18 [9.51, 10.83]	20.53 [19.52, 21.57]
PSNR	-0.80 [-0.87, -0.74]	-2.84 [-2.99, -2.69]	-5.66 [-5.88, -5.44]	-8.57 [-8.83, -8.31]
LPIPS $\times 10^3$	0.03 [-0.43, 0.48]	8.89 [7.74, 10.04]	28.21 [25.92, 30.46]	57.57 [53.87, 61.25]
MSE $\times 10^4$	1.90 [1.70, 2.11]	9.32 [8.48, 10.20]	27.75 [25.48, 30.06]	64.17 [59.29, 69.09]
SSIM $\times 10^2$	-0.09 [-0.12, -0.05]	-1.16 [-1.28, -1.04]	-3.52 [-3.78, -3.26]	-7.07 [-7.51, -6.63]
CLIP Whole	0.34 [0.25, 0.43]	0.82 [0.68, 0.97]	1.39 [1.21, 1.56]	1.92 [1.71, 2.13]
CLIP Edited	0.26 [0.17, 0.35]	0.78 [0.64, 0.93]	1.26 [1.07, 1.45]	1.82 [1.59, 2.06]

A.2 Additional Method Details

A.2.1 Details of Direction-Aware Bounded Modulation

Previous-Step Directional Response. The response operator $\mathcal{R}(\cdot)$ first computes the mean absolute magnitude across channels, followed by median-based normalization, a smooth monotone mapping ψ, and local average pooling. Specifically, we use

$$\psi(q) = \sigma(k(q-b)), \qquad k = 2.5, \quad b = 0, \tag{20}$$

where $\sigma(\cdot)$ denotes the sigmoid function. The average-pooling kernel size is 3×3, with stride 1 and padding 1. The response operator uses only the editing direction and requires neither external segmentation masks nor access to intermediate Transformer features.

Response-Adaptive Correction. The mapping $g : [0,1] \to \mathbb{R}_{\geq 0}$ converts the spatial response G_{i-1} into a localized enhancement weight. It strictly suppresses weak responses and increases monotonically:

$$u(G) = \mathrm{clip}\left(\frac{G - \tau_g}{1 - \tau_g}, 0, 1\right)^{p_g}, \qquad g(G) = \beta_g u(G), \tag{21}$$

For our experiments, we set the maximum enhancement scale $\beta_g = 1.8$, the activation threshold $\tau_g = 0.40$, and the power factor $p_g = 1.0$.

Relative Local Budget. To allocate the pixel-wise safety budget, we scale the localized enhancement using the mapping $\rho(\cdot)$, which interpolates between a minimum and maximum relative budget:

$$\rho(G) = \rho_{\mathrm{bg}} + (\rho_{\mathrm{edit}} - \rho_{\mathrm{bg}})\, u(G)^{p_\rho}. \tag{22}$$

We set the background budget $\rho_{\mathrm{bg}} = 0.18$, the active edit budget $\rho_{\mathrm{edit}} = 0.50$, and the curvature $p_\rho = 2.0$. If the response falls below the threshold τ_g, $g(G_{i-1}) = 0$, enforcing a zero unconstrained correction ($r_i^a = 0$) and a zero final projection ($\Delta_i^a = 0$).

A.2.2 Proof of Local Boundedness

By construction, $\Delta_i^a(p)$ is obtained by projecting $r_i^a(p)$ onto the closed ℓ_2-ball with radius

$$B_i^a(p) = \rho(G_{i-1}(p)) \left\| v_i^{at}(p) \right\|_2.$$

Therefore,

$$\|\Delta_i^a(p)\|_2 \le \rho(G_{i-1}(p)) \left\|v_i^{at}(p)\right\|_2 . \tag{23}$$

From the definition of the final editing direction,

$$d_i^{\text{final}}(p) - D_i^{\text{match}}(p) = \alpha\Delta_i^s(p) + (1-\alpha)\Delta_i^t(p). \tag{24}$$

Applying the triangle inequality together with Eq. 23 gives

$$\left\|d_i^{\text{final}}(p) - D_i^{\text{match}}(p)\right\|_2 \le \alpha\rho(G_{i-1}(p)) \left\|v_i^{st}(p)\right\|_2 + (1-\alpha)\rho(G_{i-1}(p)) \left\|v_i^{tt}(p)\right\|_2 . \tag{25}$$

Thus, the target-conditioned semantic correction remains strictly bounded relative to the original velocity field.

A.2.3 Algorithm

Algorithm 1 summarizes the overall inference procedure of MatchEdit.

Algorithm 1 Inference Procedure of MatchEdit

Require: Source image I_s; source prompt c_s; target prompt c_t; frozen velocity model v; VAE encoder E_{VAE} and decoder D_{VAE}; time sequence $\tau_0 > \tau_1 > \cdots > \tau_K$; fusion weight α.
Ensure: Edited image I_{edit}.

$x_s \leftarrow E_{\text{VAE}}(I_s)$
$y_0 \leftarrow x_s, \quad G_{-1} \leftarrow 0$
for $i = 0, \dots, K-1$ **do**
  $\epsilon_i \sim \mathcal{N}(0, I)$
  $z_i^s \leftarrow (1-\tau_i)x_s + \tau_i\epsilon_i$
  $z_i^t \leftarrow y_i + z_i^s - x_s$
  $v_i^{ab} \leftarrow v(z_i^a, \tau_i, c_b), \quad a, b \in \{s, t\}$
  $d_i^s \leftarrow v_i^{st} - v_i^{ss}, \qquad d_i^t \leftarrow v_i^{tt} - v_i^{ts}$
  $D_i^{\text{match}} \leftarrow \alpha d_i^s + (1-\alpha)d_i^t$
  for $a \in \{s, t\}$ **do**
    $\Delta_i^a \leftarrow \text{Proj}_{B_i^a}(r_i^a)$
    $\widetilde{v}_i^{at} \leftarrow v_i^{at} + \Delta_i^a$
  end for
  $d_i^{\text{final}} \leftarrow \alpha(\widetilde{v}_i^{st} - v_i^{ss}) + (1-\alpha)(\widetilde{v}_i^{tt} - v_i^{ts})$
  $y_{i+1} \leftarrow y_i + (\tau_{i+1} - \tau_i)\left(d_i^{\text{final}} + \tau_i\epsilon_i\right)$
  $G_i \leftarrow \mathcal{R}(d_i^{\text{final}})$
end for
$I_{\text{edit}} \leftarrow D_{\text{VAE}}(y_K)$
return I_{edit}

A.3 Additional Experimental Settings

The main experiments are conducted on PIE-Bench, which contains 700 image-prompt pairs spanning 10 editing categories. Long-prompt generalization uses 700 samples from DNA-Bench, and sequential multi-turn generalization uses 100 five-turn editing sessions from MSE-Bench. For MSE-Bench, we additionally evaluate a direct Turn-5 anchor that edits once from the original image to the cumulative fifth-turn target; the three compared methods therefore produce 1,500 sequential outputs and 300 direct-anchor outputs. All results generated in this work are obtained on a single NVIDIA RTX 5090D 32 GB GPU using a unified evaluation pipeline. For FLUX-based methods, we use the PIE-Bench and efficiency results reported by FIA-Edit (Yang et al., 2026), which were obtained under an NVIDIA A100 80 GB GPU setting.

A.4 Efficiency Analysis

Tab. 9 reports GPU memory usage and per-image runtime. MatchEdit remains comparable to other SD3.5-based inversion-free methods in both memory and runtime. FLUX-based results are included only as a resource reference because they were measured on different hardware.

Table 9: GPU memory and per-image inference time on PIE-Bench.

Method	Model	Memory (GB)	Time/Image (s)
LDM-based methods			
P2P + DDIM	SD1.4	16.86	15.23
P2P + Direct Inversion	SD1.4	15.54	20.70
PnP + DDIM	SD1.5	11.62	8.31
PnP + Direct Inversion	SD1.5	11.30	10.52
MasaCtrl	SD1.4	13.95	17.86
FLUX-based methods			
RF-Inversion	FLUX	69.22	76.74
StableFlow	FLUX	35.39	26.07
RF-Edit	FLUX	32.91	34.51
DiT-based methods			
FTEdit	SD3.5	27.62	83.66
FlowEdit	SD3.5	20.76	5.28
DNAEdit	SD3.5	20.72	5.29
FSI-Edit	SD3.5	18.98	10.21
FIA-Edit	SD3.5	20.79	5.77
MatchEdit	SD3.5	20.76	5.63

A.5 Additional PIE-Bench Results and Analysis

Extended Quantitative Analysis. Table 1 confirms that MatchEdit consistently achieves state-of-the-art source preservation across both DiT and LDM architectures. Under identical SD3.5 backbones, compared to FIA-Edit, MatchEdit reduces Structure Distance by 2.89, LPIPS by 8.86, and MSE by 9.53, while improving PSNR by 1.70 dB. Notably, these preservation gains are secured with merely a 0.10 decrease in CLIP Edited, indicating a favorable preservation-editability balance.

Extended Qualitative Analysis. Fig. 3 provides granular visual comparisons. For the teacup-to-glass transformation, MatchEdit accurately renders the glass material while preserving the nest structure and surrounding eggs; methods such as FlowEdit and RF-Inversion inadvertently alter the vessel geometry or strip away surrounding details. In background modifications (e.g., adding clouds to a sunset), MatchEdit successfully synthesizes the target atmosphere without disrupting foreground silhouettes, whereas FreeDiff introduces noticeable off-target color shifts.

Fig. 8 illustrates further categorical edits. When applying a watercolor stylization, MatchEdit applies the target artistic style while retaining the spatial layout and identity of the subject, whereas FTEdit leaves structural artifacts. For object transformation (cat-to-tiger), MatchEdit cleanly converts the species while preserving the pink hat and posture, whereas FlowAlign introduces larger structural deviations.

A.6 Additional Ablation Results

A.6.1 Effect of the Fusion Weight α

As shown in Tab. 10 and Fig. 6, the contrast weight α controls the preservation-editability trade-off by balancing the source-state and target-state prompt contrasts. As α increases, CLIP Whole and CLIP Edited decrease, indicating weaker target-semantic editing, while the preservation metrics are generally optimized at intermediate values of α. We therefore use $\alpha = 0.3$ in the main experiments to retain stronger target-semantic alignment while maintaining substantial source preservation.

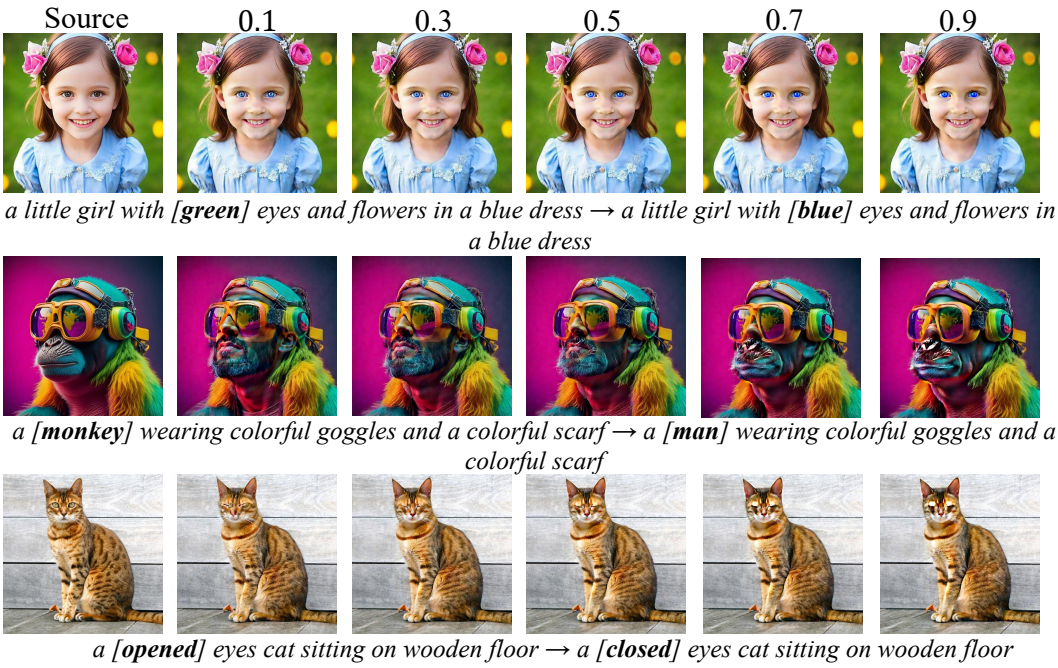


Figure 6: Effect of the fusion weight α.

Table 10: Sensitivity to the fusion weight α.

α	Distance $\times 10^3 \downarrow$	PSNR ↑	LPIPS $\times 10^3 \downarrow$	MSE $\times 10^4 \downarrow$	SSIM $\times 10^2 \uparrow$	CLIP Whole ↑	CLIP Edited ↑
0.1	8.57	28.52	50.66	21.43	89.84	**26.21**	**22.81**
0.3	7.45	29.02	46.15	19.14	90.32	<u>26.03</u>	<u>22.71</u>
0.5	**6.93**	**29.25**	<u>43.67</u>	**18.19**	**90.53**	25.71	22.39
0.7	<u>7.10</u>	<u>29.13</u>	**43.66**	<u>18.70</u>	<u>90.46</u>	25.19	21.88
0.9	8.09	28.66	46.01	20.73	90.10	24.73	21.52

A.6.2 Component Ablation Examples

As shown in Fig. 7, the matched-state prompt direction effectively preserves structural and contextual information from the source image. Without direction-aware bounded modulation, the matched-state prompt direction can under-express the target semantics in challenging edits despite its strong source-preservation capability. For example, in moon replacement and animal-category transformation, the matched-state prompt direction preserves the original scene layout, but the extent of the intended semantic change remains limited. Direction-aware bounded modulation strengthens these localized semantic changes while retaining the structural advantages of the matched-state base direction.

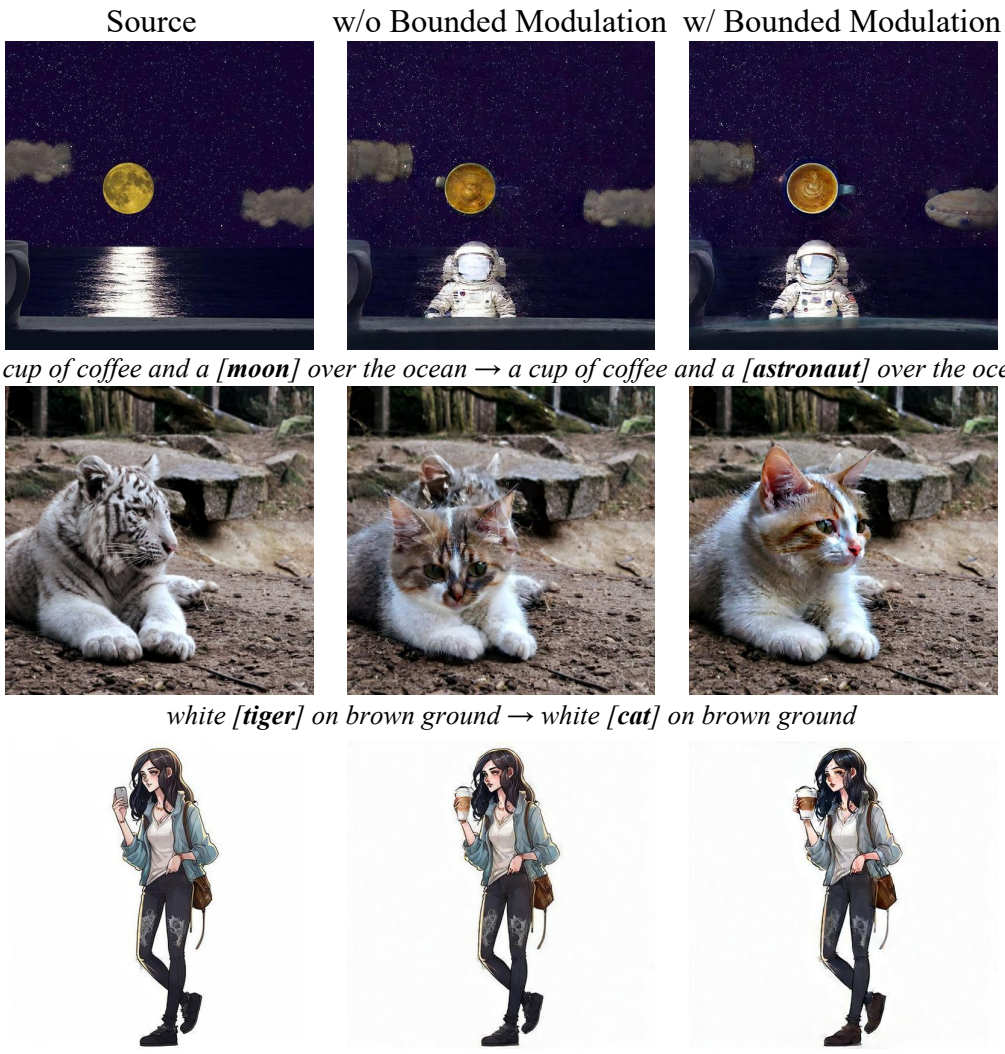


Figure 7: Qualitative ablation of the direction-aware bounded modulation.



Figure 8: Additional qualitative results on PIE-Bench.

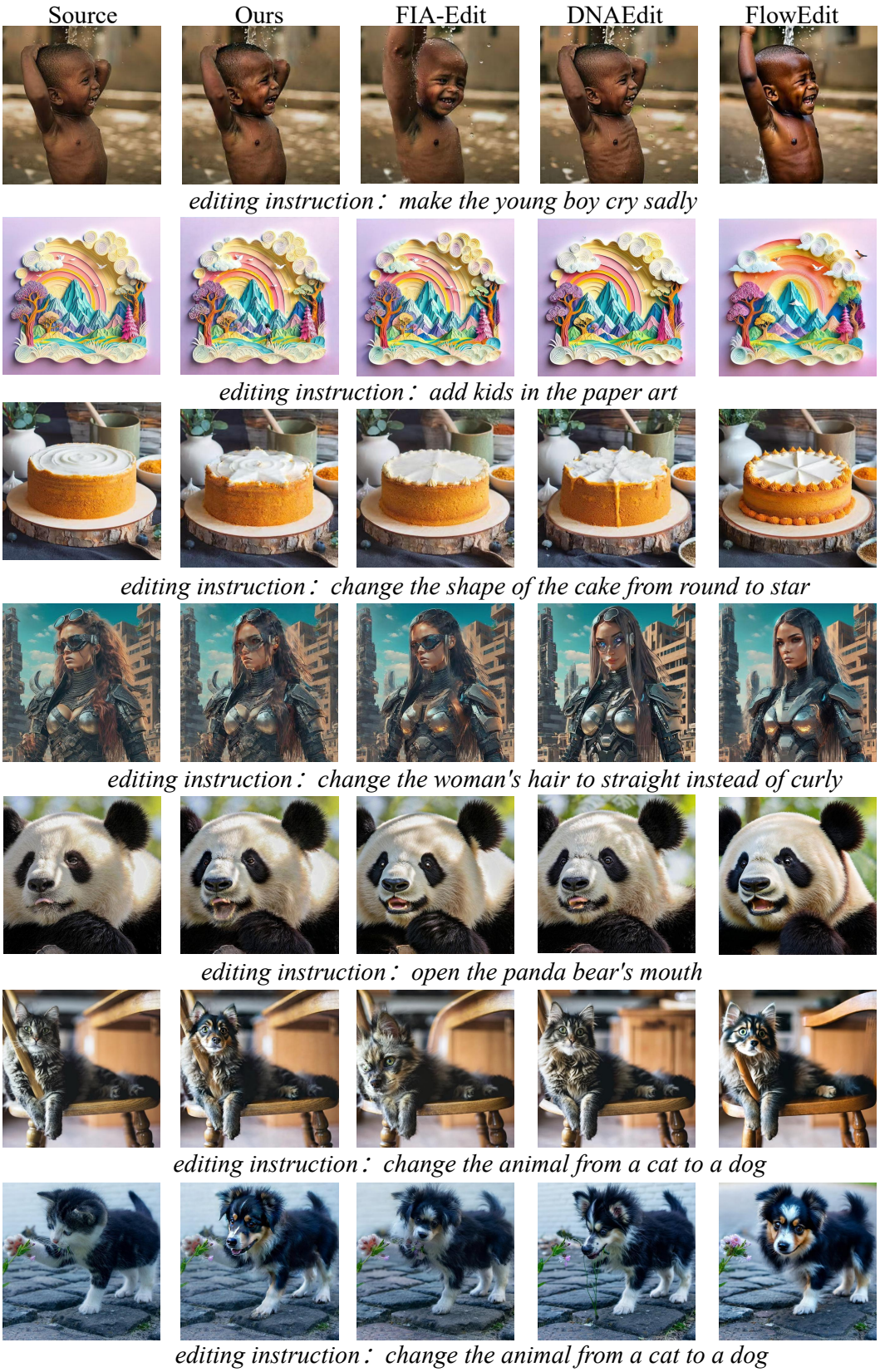


Figure 9: Additional qualitative comparison on DNA-Bench.

Table 11: DNA-Bench results across different target-prompt lengths.

Split	Method	Distance $\times 10^3 \downarrow$	PSNR ↑	LPIPS $\times 10^3 \downarrow$	MSE $\times 10^4 \downarrow$	SSIM $\times 10^2 \uparrow$	CLIP Whole ↑	CLIP Edited ↑
Short	FlowEdit	24.99	23.14	96.58	72.75	84.61	**27.97**	**21.95**
	DNAEdit	14.38	26.70	74.23	33.20	88.72	27.19	21.32
	FIA-Edit	11.54	27.30	58.04	31.07	88.71	27.16	21.05
	MatchEdit	**8.74**	**29.45**	**45.74**	**18.52**	**90.25**	27.27	20.57
Medium	FlowEdit	24.28	22.89	96.15	75.15	85.12	**29.25**	**24.31**
	DNAEdit	13.87	26.82	69.34	31.07	89.54	28.42	23.43
	FIA-Edit	10.57	27.12	55.45	29.61	89.50	28.68	23.12
	MatchEdit	**6.84**	**29.44**	**41.84**	**17.77**	**91.09**	28.66	23.01
Long	FlowEdit	26.08	21.23	114.44	98.69	81.87	**30.10**	**24.75**
	DNAEdit	14.38	25.33	82.30	41.29	87.32	29.15	23.93
	FIA-Edit	11.05	25.80	65.60	36.79	87.41	29.33	23.90
	MatchEdit	**6.28**	**28.41**	**47.45**	**19.74**	**89.77**	29.27	23.75

Table 12: DNA-Bench results across different levels of source-target prompt difference.

Split	Method	Distance $\times 10^3 \downarrow$	PSNR ↑	LPIPS $\times 10^3 \downarrow$	MSE $\times 10^4 \downarrow$	SSIM $\times 10^2 \uparrow$	CLIP Whole ↑	CLIP Edited ↑
Low	FlowEdit	25.78	22.83	100.07	77.22	84.47	**29.08**	**23.27**
	DNAEdit	14.27	26.70	74.05	32.89	88.86	28.26	22.68
	FIA-Edit	11.40	27.36	57.34	29.87	89.02	28.32	22.40
	MatchEdit	**8.18**	**29.64**	**44.35**	**17.61**	**90.65**	28.35	22.27
High	FlowEdit	24.45	22.14	102.99	85.33	83.45	**29.13**	**24.08**
	DNAEdit	14.16	25.92	75.67	37.07	88.30	28.25	23.11
	FIA-Edit	10.71	26.10	61.73	35.17	88.10	28.46	22.99
	MatchEdit	**6.39**	**28.52**	**45.53**	**19.91**	**90.09**	28.45	22.62

A.7 Additional Generalization Results

A.7.1 DNA-Bench Results (Long-Prompt Editing)

Fig. 9 presents additional qualitative comparisons on DNA-Bench. The examples cover facial-expression editing, object insertion, shape modification, hairstyle editing, localized editing, and category-level transformation. Across these cases, MatchEdit generally preserves source pose, framing, and surrounding content while producing varying degrees of the requested semantic change.

Tabs. 11 and 12 break down the DNA-Bench performance across prompt lengths and source-target textual discrepancies. MatchEdit secures the top preservation metrics across all length tertiles (Short, Medium, Long) and discrepancy levels (Low, High). For instance, in the Long prompt split, compared to FIA-Edit, MatchEdit reduces MSE by 17.05 and improves SSIM by 2.36 percentage points, showing stronger source preservation under long-prompt editing.

A.7.2 MSE-Bench Results (Multi-Turn Editing)

Sequential Editing Analysis. Tab. 13 traces the cumulative trajectory of multi-turn edits. MatchEdit sustains the highest cumulative CLIP across all five turns. Notably, at Turn 5, MatchEdit outpaces FlowEdit by 1.66 in cumulative CLIP and 9.44 percentage points in SSIM Source, indicating reduced accumulation of structural degradation across sequential editing turns.

Direct Turn-5 Analysis. Tab. 14 evaluates the direct Turn-5 anchor. Even without iterative stepping, MatchEdit provides superior cumulative target alignment (27.89) alongside the highest SSIM

Table 13: Per-turn results on MSE-Bench sequential editing.

Method	Turn	Cum. CLIP ↑	Current CLIP ↑	Prior Retention ↑	CLIP-I Source ↑	CLIP-I Previous ↑	SSIM Source ↑	MSE Source ↓
FIA-Edit	1	29.90	24.31	–	91.39	91.39	83.32	0.0056
	2	29.98	22.74	24.49	87.37	95.27	74.98	0.0112
	3	29.88	21.17	23.74	85.21	96.85	67.74	0.0177
	4	29.03	20.20	23.09	83.43	97.15	61.64	0.0249
	5	28.27	20.75	22.55	81.57	97.53	56.10	0.0327
FlowEdit	1	30.08	24.59	–	87.82	87.82	75.13	0.0145
	2	30.15	22.77	24.65	84.59	95.09	65.10	0.0281
	3	29.67	21.18	23.78	82.65	96.87	57.91	0.0409
	4	28.76	20.22	22.93	80.84	97.56	52.47	0.0528
	5	28.15	21.17	22.42	79.50	97.68	48.15	0.0634
MatchEdit	1	30.49	24.91	–	92.27	92.27	86.98	0.0029
	2	31.32	23.40	25.25	86.99	94.78	78.59	0.0068
	3	31.39	21.90	24.44	83.61	96.74	70.93	0.0113
	4	30.80	20.84	23.72	81.59	97.45	64.06	0.0163
	5	29.81	21.69	22.96	80.25	98.15	57.59	0.0215

Source (85.56) and lowest MSE (0.0038). These results suggest that the advantages of MatchEdit are not solely due to sequential staging and are also observed under direct editing.

Table 14: Direct Turn-5 editing from the original image on MSE-Bench.

Method	Cum. CLIP ↑	Current CLIP ↑	Prior Retention ↑	CLIP-I Source ↑	SSIM Source ↑	MSE Source ↓
FIA-Edit	26.91	20.81	22.21	**89.02**	<u>82.71</u>	<u>0.0062</u>
FlowEdit	<u>27.52</u>	**21.36**	<u>22.46</u>	85.55	74.78	0.0151
MatchEdit	**27.89**	<u>21.19</u>	**22.47**	<u>87.75</u>	**85.56**	**0.0038**