WHERE TO LOOK, WHERE TO EDIT: ENHANCING EDITABILITY IN INVERSION-FREE EDITING

Anonymous authors
Paper under double-blind review

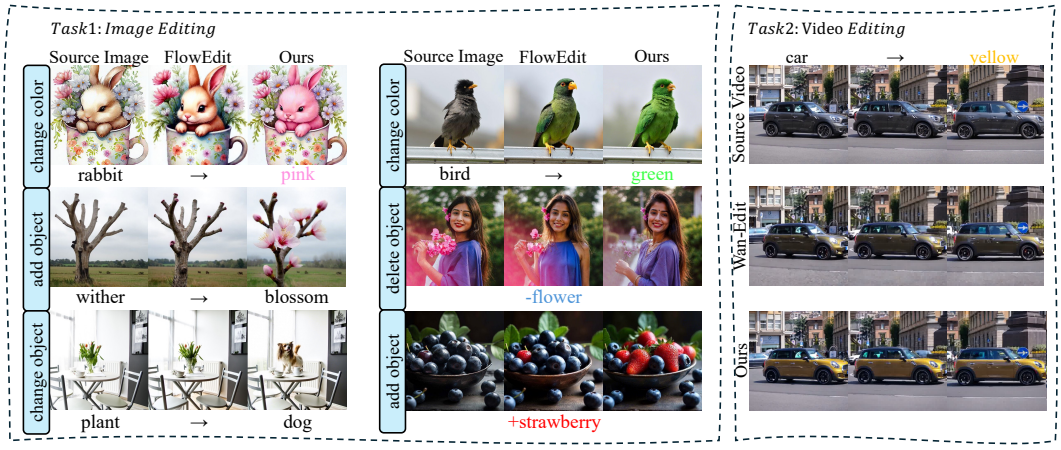

Figure 1: Representative editing results of SAATEdit. (Left) Comparison with FlowEdit on PIE-Bench across various editing types, *e.g.*, color change, object addition, deletion, and replacement. (Right) Comparison with WanEdit on FiVE-Bench for video editing. In both domains, our method significantly improves text alignment while preserving structural fidelity.

ABSTRACT

Inversion-free methods based on rectified-flow models enable efficient image and video editing by avoiding costly inversion. However, their limited editing strength often leads to poor alignment with target prompts. We find that this limitation comes from insufficient attention to target-prompt semantics within editing regions. Selectively increasing this attention can improve target-prompt guidance and enable stronger edits while maintaining structural fidelity. This motivates us to propose Spatially Adaptive Attention Transfer Edit (SAATEdit), a training-free and inversion-free framework for more effective image and video editing. SAATEdit extracts spatial priors from source-prompt-guided attention to localize editing regions and uses these priors to compute bounded spatial gains for targeted attention enhancement. Benefiting from spatial priors and targeted attention enhancement, SAATEdit achieves stronger edits and better alignment with target prompts while maintaining structural fidelity. Experiments on PIE-Bench and FiVE-Bench demonstrate that SAATEdit achieves superior image and video editing performance with minimal inference overhead.

1 INTRODUCTION

With the rapid progress in text-to-image (T2I) generation, the demand for text-driven image editing has grown significantly. The dominant training-free editing paradigm is built on inversion, where a real image is first inverted into its initial noise, and the editing condition is then injected during denoising (Hertz et al., 2023; Mokady et al., 2023; Ju et al., 2024; Wallace et al., 2023; Jiao et al., 2026; Xie et al., 2025). However, inversion is computationally expensive and inherently suffers from error accumulation. These approximation errors cause the reconstructed noise to deviate from

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.

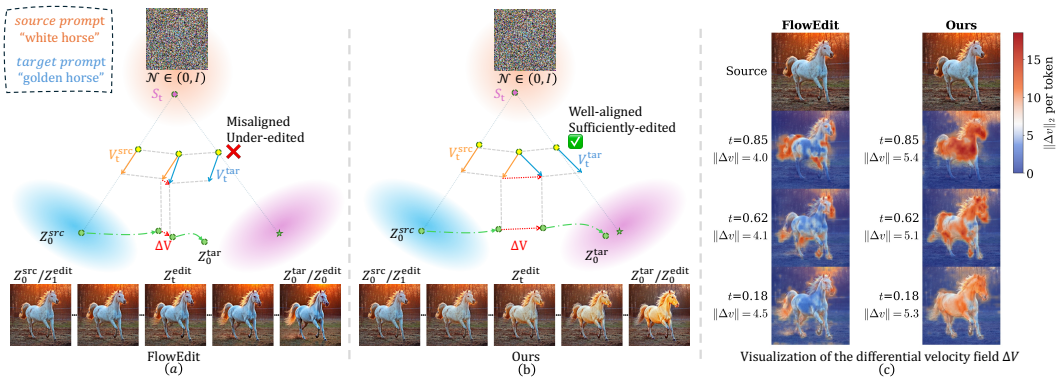

Figure 2: (a) In FlowEdit, the strong source image prior limits the model's attention to the target edit region. As the editing signal diffuses spatially, the target velocity v^t_{tar} closely aligns with the source velocity v^t_{src}, resulting in a weak differential velocity field ΔV and limited editing strength. (b) SAATEdit introduces targeted attention enhancement to adaptively amplify the signal within the edit region. This generates a sufficiently strong ΔV to drive semantic changes. (c) Magnitude of ΔV at different timesteps t. SAATEdit concentrates a strong differential signal precisely on the edit region, whereas the signal in FlowEdit remains weak and diffused.

the true distribution, which degrades the editing fidelity. To address this, the inversion-free method FlowEdit (Kulikov et al., 2025) removes this step entirely by guiding the editing via a differential velocity field defined between the source prompt and the target prompt, enabling low-cost and high-fidelity image editing.

Despite this progress, inversion-free editing remains limited in *editing strength*. As illustrated in Fig. 1, FlowEdit typically fails on editing tasks that require large semantic changes, such as color replacement. In these cases, the generated results are overly constrained by the source image. This limitation is also inherited by subsequent refinements such as FlowAlign(Kim et al., 2026), SNR-Edit (Jiang et al., 2026), and RIDGE (Gong et al., 2026). Their designs mainly aim to stabilize the editing trajectory to better preserve structural fidelity. However, this stabilization forces the intermediate latent states to anchor to the source image prior. Consequently, they fail to resolve the issue of insufficient editing strength and even further weaken the editing effect.

To investigate this limitation, we conduct a detailed analysis on the dynamics of inversion-free editing. As shown in Fig. 2, the editing process is driven by the differential velocity field ΔV. A major reason for the insufficient editing strength is the lack of attention to the target edit region. Specifically, the strong conditional prior from the source image forces the model to focus heavily on the original content. As a result, the model pays insufficient attention to the editing words in the target prompt, causing the editing signal to diffuse spatially rather than accurately localizing the intended region. This diffused signal directly weakens the target velocity field (v_{tar}), causing it to closely align with the source velocity field (v_{src}). Consequently, the resulting weak differential signal ΔV fails to drive the desired semantic changes, severely limiting the overall editing strength.

The above analysis motivates us to enhance the attention on target edit regions to improve the editing strength of inversion-free methods. Based on this motivation, we propose **SAATEdit** (Spatially Adaptive Attention Transfer Edit), a training-free and inversion-free framework for more effective image and video editing. To resolve the issue of inaccurate localization, SAATEdit incorporates an Attention Reference Transfer (ART) mechanism that extracts spatial priors from source-prompt-guided attention to precisely delineate the edit region. Building upon this localized reference, our method introduces Spatial Adaptive Enhancement (SAE) to compute bounded spatial gains for targeted attention enhancement. By adaptively amplifying the response within the edit region and suppressing the background, SAE ensures that the editing signal concentrates on the intended area rather than diffusing spatially. Working in synergy, these complementary mechanisms achieve accurate localization and strong editing capabilities without introducing any additional inference overhead.

We comprehensively evaluate our method on PIE-Bench (Ju et al., 2024) for image editing and FiVE-Bench (Li et al., 2025) for video editing. Compared to FlowEdit (Kulikov et al., 2025),

SAATEdit achieves better structural fidelity while significantly enhancing the overall editing strength, particularly in tasks requiring large semantic changes. This robust performance extends to the video domain, where SAATEdit substantially improves the editing capability over the Wan-Edit baseline. Furthermore, these improvements are achieved with negligible computational overhead, adding only 4% and 6.1% to the inference time for image and video editing, respectively.

2 Related Work

Inversion-Based Editing. Inversion-based methods first invert a real image to its initial noise and then re-generate it along the original generative trajectory under new text conditions, so the recovered noise must faithfully preserve the source structure while still permitting semantic modification. For diffusion models this is usually achieved through DDIM Inversion (Song et al., 2021), Null-Text Inversion (Mokady et al., 2023), or negative-prompt and dual-schedule inversion (Miyake et al., 2025; Huang et al., 2025), followed by injecting internal features (*e.g.*, cross-attention maps) during the editing forward pass (Hertz et al., 2023; Ju et al., 2024; Wallace et al., 2023). Rectified flow models formulate generation as a deterministic ODE and thus admit more exact inversion in principle, but truncation errors accumulated over discrete timesteps still shift the recovered noise away from the true distribution. Higher-order solvers (Wang et al., 2025b; Deng et al., 2025), redesigned inversion (Jiao et al., 2026), direct noise alignment (Xie et al., 2025), and restricting edits to a subset of layers (Avrahami et al., 2024; Rout et al., 2025) mitigate this error without eliminating it.

Inversion-Free Editing. Inversion-free methods avoid this overhead and error accumulation: SDEdit (Meng et al., 2021) perturbs the source image with random noise, while more recent works construct a differential velocity field between the source image and the target text (Kulikov et al., 2025; Yoon et al., 2025; Kim et al., 2026), at the price of discarding part of the source structural prior through FlowEdit's random noise initialization. Their structure-preserving designs—the source-similarity objective of FlowAlign (Kim et al., 2026), the flow decomposition of SplitFlow (Yoon et al., 2025), the structure-aware noise rectification of SNR-Edit (Jiang et al., 2026), and the internally guided re-noising of RIDGE (Gong et al., 2026)—anchor the intermediate latents to the source and thereby suppress the edit itself. Both lines are thus biased toward fidelity rather than editability, most evidently on tasks requiring large semantic changes.

Attention Manipulation for Editing. Manipulating attention maps is a common technique in text-driven image editing. Prompt-to-Prompt (Hertz et al., 2023) preserves image structure by injecting the cross-attention maps of the source prompt into those of the target prompt. Such methods act on the *separate* cross-attention module of a U-Net, where image queries attend only to text keys, so a single word's attention map can be replaced in isolation.

However, the above methods mainly focus on maintaining structural fidelity. In this work, we address the critical challenge of insufficient *editing strength* in inversion-free editing. Therefore, instead of restricting the generation to closely follow the source image, we utilize the attention mechanism to make the editing signal both stronger and more accurately localized.

3 Preliminaries

3.1 Rectified Flow Models

Rectified Flow (RF) (Liu et al., 2023; Lipman et al., 2023) models generation as the ODE $dZ_t = V(Z_t, t)dt$, whose velocity field transports a prior noise distribution π_1 to the data distribution π_0 along linear trajectories $Z_t = tS_1 + (1-t)X_0$, where $X_0 \sim \pi_0$ and $S_1 \sim \pi_1$. The ideal velocity field of this rectified path is $S_1 - X_0$, and a network $V_\theta(Z_t, t, c)$ conditioned on c (*e.g.*, a text prompt) is trained to approximate it. At inference time, data is generated by integrating the ODE backward from $t = 1$ to $t = 0$.

3.2 Inversion-Free Editing

To bypass the expensive inversion process, inversion-free methods like FlowEdit (Kulikov et al., 2025) utilize the differential velocity field ΔV between a source prompt c_{src} and a target prompt

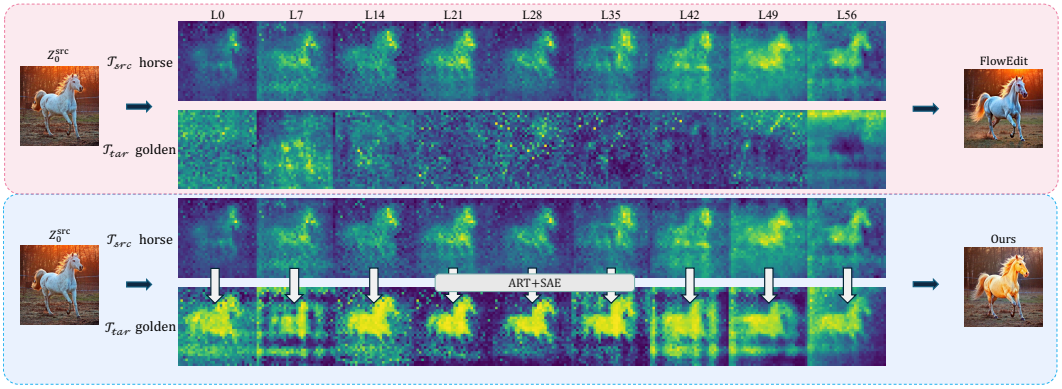

Figure 3: Attention score visualization of the source object token ($\mathcal{T}_{src}$, "horse") and the target editing token ($\mathcal{T}_{tar}$, "golden") across nine layers at the same timestep. Top: without ART the target editing token's attention is diffuse and carries no object shape. Bottom: ART transfers the source-branch attention scores into the target editing token, inheriting the same object-shaped spatial prior.

c_{tar} to guide the edit directly in the image space. Given a sequence of discrete timesteps $\{t_i\}$, we define $V_{t_i}^{src} := V_\theta(Z_{t_i}^{src}, t_i, c_{src})$ and $V_{t_i}^{tar} := V_\theta(Z_{t_i}^{tar}, t_i, c_{tar})$. To avoid inversion, the target latent at step i is approximated by:

$$Z_{t_i}^{tar} \approx Z_{t_i}^{edit} + Z_{t_i}^{src} - X_0, \tag{1}$$

where $Z_{t_i}^{src} = t_i S_{t_i} + (1-t_i)X_0$ is the linear interpolation between the source image and a randomly initialized noise, and the transition latent $Z_{t_i}^{edit}$, initialized as X_0, is updated iteratively by:

$$Z_{t_{i-1}}^{edit} = Z_{t_i}^{edit} + (t_{i-1} - t_i)\Delta V_{t_i}, \tag{2}$$

where $\Delta V_{t_i} = V_{t_i}^{tar} - V_{t_i}^{src}$, and $V_{t_i}^{tar} \approx V_\theta(Z_{t_i}^{edit} + Z_{t_i}^{src} - X_0, t_i, c_{tar})$ approximates the target velocity. More details are in (Kulikov et al., 2025).

3.3 Joint Attention in Rectified-Flow Transformers

DiTs concatenate text and image tokens into a single sequence of length $L = L_{txt} + L_{img}$. Let H be the number of attention heads. For a single head, the attention layer first computes a joint **raw score matrix** $S \in \mathbb{R}^{L\times L}$ and subsequently a normalized **attention matrix** $A \in \mathbb{R}^{L\times L}$:

$$S = \frac{QK^\top}{\sqrt{d}}, \qquad A = \mathrm{softmax}(S), \tag{3}$$

where Q, K, V are the queries, keys, and values of the concatenated sequence, d is the head dimension. The joint raw score matrix S can be written as a block matrix whose blocks correspond to the four intra- and inter-modal interactions:

$$S = \begin{bmatrix} S^{txt-txt} & S^{txt-img} \\ S^{img-txt} & S^{img-img} \end{bmatrix}, \tag{4}$$

where the superscripts denote the query and key modalities. In this work, we focus on the image-to-text sub-block $S^{img-txt} \in \mathbb{R}^{L_{img}\times L_{txt}}$, which records the response of the image tokens to the textual information, and denote it by $M = S^{img-txt}$ for brevity. We write the entry of M that represents the raw score between the j-th image query and the i-th text key as $M_{j,i}$, where $j \in \{1, \ldots, L_{img}\}$ and $i \in \{1, \ldots, L_{txt}\}$. Thus, the i-th column of M represents the response of the entire image space to the i-th text token.

4 Method

4.1 Overview

Inversion-free editing is driven entirely by the differential velocity field $\Delta V_{t_i} = V_{t_i}^{tar} - V_{t_i}^{src}$ in Eq. (2). Due to the two coupled defects of inaccurate edit region identification and insufficient editing strength, this paradigm frequently fails on tasks requiring large semantic changes. To address

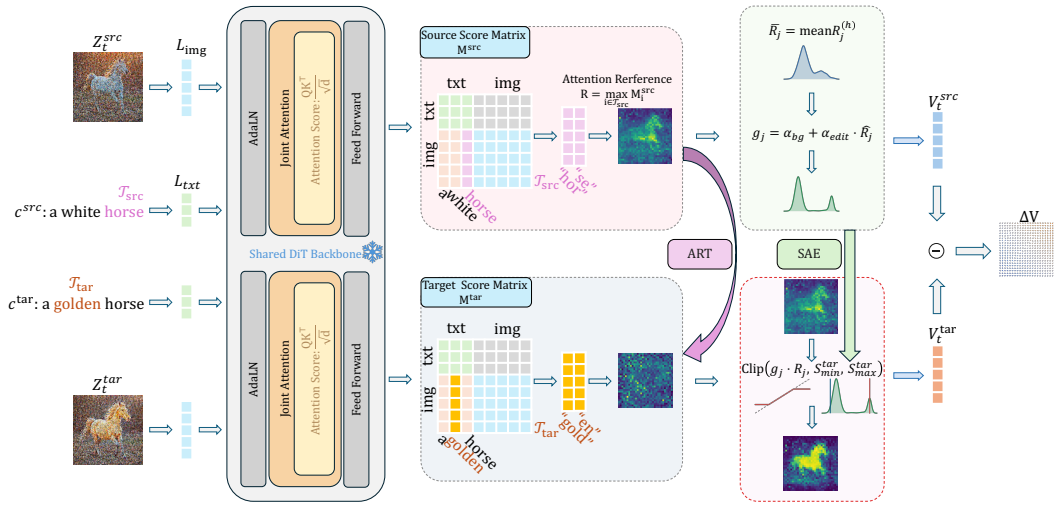

Figure 4: Overview of the proposed framework. The source and target branches share a frozen DiT backbone and each produces a velocity field through joint attention. Attention Reference Transfer (ART) transfers a reference attention pattern into the target editing tokens to specify the region to edit, and Spatial Adaptive Enhancement (SAE) derives a per-patch gain from that reference so that the edit is concentrated inside the region. The resulting edit direction is $\Delta V = v_{\text{tar}} - v_{\text{src}}$.

this issue, we propose SAATEdit, an inversion-free framework that proactively controls the edit region and editing strength by regulating the attention mechanisms. As illustrated in Fig. 4, SAATEdit relies on two synergistic mechanisms:

$$(R, \hat{M}^{tar}) = \text{ART}(M^{src}, \mathcal{T}_{src}, \mathcal{T}_{tar}), \quad \tilde{M}^{tar} = \text{SAE}(\hat{M}^{tar}, R), \tag{5}$$

where $R \in \mathbb{R}^{L_{img}}$ is the reference signal of the edit region extracted by Attention Reference Transfer (ART) and injected into the target editing tokens, yielding the injected sub-block $\hat{M}^{tar}$, $\tilde{M}^{tar}$ is the modulated version of $\hat{M}^{tar}$ produced by Spatial Adaptive Enhancement (SAE), and $\mathcal{T}$ represents the respective token index sets.

4.2 Attention Reference Transfer

Because the conditional prior of the source image is overwhelmingly strong and the target prompt conflicts with the source image content, the attention layers respond insufficiently to the target editing tokens, resulting in an ambiguously identified edit region. To mitigate this, we observe that the edit region is already explicitly and accurately identified in the source branch: the model can focus reliably in the attention maps on the source object it actually "looks at". As illustrated in Fig. 3, ART harnesses this by extracting the raw score columns of the source object tokens to construct a clean reference signal that conforms to the edit region.

Let M^{src} and M^{tar} denote the image-to-text sub-blocks of the source and target branches, respectively. Both sets are the subword tokens of the words naming the source object and the edit target, extracted from the prompt pair(Appendix C). Given the token set $\mathcal{T}_{src}$ of the source object, we aggregate its columns within M^{src} using a maximum operation over subwords. For each image token j, the reference signal is extracted as

$$R_j = \max_{i \in \mathcal{T}_{src}} M_{j,i}^{src}, \tag{6}$$

which forms a single consolidated reference vector $R \in \mathbb{R}^{L_{img}}$. This maximum aggregation implicitly differentiates the edit region and prevents dilution by the number of tokens. We then transfer this shared reference into the columns of M^{tar} that correspond to the target editing tokens $\mathcal{T}_{tar}$, yielding the ART-injected sub-block $\hat{M}^{tar}$:

$$\hat{M}_{j,i}^{tar} = R_j, \quad \forall i \in \mathcal{T}_{tar}. \tag{7}$$

All operations above are applied independently to every attention head and to every attention layer. The modified sub-block is substituted back into the joint raw score matrix of the target branch, giving the modified matrix $\hat{S}^{tar}$, which is normalized by the standard softmax, i.e., $\hat{A}^{tar} = \mathrm{softmax}(\hat{S}^{tar})$, before being multiplied by V.

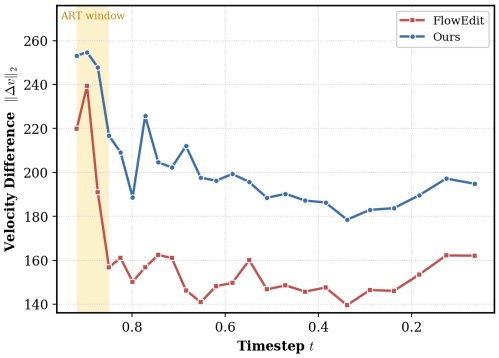

Figure 5: Velocity difference $\|\Delta V_{t_i}\|_2$ over timesteps t for FlowEdit and SAATEdit; the shaded region is the ART/SAE window, i.e., the early high-noise steps.

Window of ART. Applying ART over the entire generative trajectory would constrain the edit region too rigidly and push the attention scores away from the training distribution, thereby degrading both editability and generative fidelity. We therefore introduce a window parameter n_{art} and apply ART only during the early high-noise stage, marked by the shaded region in Fig. 5. Such early intervention suffices to accurately localize the edit region and induce sufficient divergence between V^{tar} and V^{src}, whereas subsequent steps are left unconstrained so that the model can refine the edit under its standard attention computation. We use $n_{art} = 4$ by default, and SAE is applied within the same window.

4.3 Spatial Adaptive Enhancement

Although ART localizes the edit region accurately, editing tasks with a large semantic change often require a higher editing strength. Due to the strong image prior, the model's response within the identified region still remains insufficiently strong. Uniformly amplifying the entire score matrix is, however, not viable, as it would simultaneously amplify the response in the background regions. To overcome this, SAE adaptively modulates a gain coefficient from spatial information. Writing $R^{(h)}$ for the reference of Eq. (6) obtained from the raw score sub-block of the h-th head, we first average it over all heads and normalize the result to the unit interval:

$$\bar{R}_j = \frac{1}{H}\sum_{h=1}^{H} R_j^{(h)} \in \mathbb{R}^{L_{img}}, \qquad \hat{R}_j = \frac{\bar{R}_j - \min(\bar{R})}{\max(\bar{R}) - \min(\bar{R})}. \tag{8}$$

This normalized map is subsequently mapped to a bounded spatial gain through an affine transformation:

$$g_j = \alpha_{bg} + \alpha_{edit} \cdot \hat{R}_j, \quad g_j \in [\alpha_{bg}, \alpha_{bg} + \alpha_{edit}], \tag{9}$$

where α_{bg} and α_{edit} set the floor of the gain and the in-region amplification, respectively. Derived from the spatial information implicit in the reference signal, this gain replaces the uniform injection in Eq. (7) with a spatially adaptive scheme. Modulating the raw scores without a bound may, however, push the injected values outside the numerical range seen during training and thus induce out-of-distribution (OOD) artifacts. We therefore constrain the modulated values to the range of the original target score sub-block:

$$\tilde{M}^{tar}_{j,i} = \mathrm{Clip}\Big(g_j \cdot \hat{M}^{tar}_{j,i},\, S^{tar}_{\min},\, S^{tar}_{\max}\Big), \quad \forall i \in \mathcal{T}_{tar}, \tag{10}$$

where $\mathrm{Clip}(x, a, b) = \min(\max(x, a), b)$, and $S^{tar}_{\max}$ and $S^{tar}_{\min}$ are the maximum and minimum of the target raw score sub-block before any modification; both the bounds and the injected values are those of the respective head. Finally, the modulated sub-block $\tilde{M}^{tar}$ replaces $\hat{M}^{tar}$ in the joint matrix of the target branch, giving $\tilde{S}^{tar}$, which is normalized by the standard softmax. By scaling and bounding the raw scores, SAE reallocates the attention mass across the image tokens, strengthening the edit while preserving the background in practice (Tab. 2). Both ART and SAE reuse the forward passes that the base inversion-free editor already performs and add no additional ones.

Table 1: Comparison on FiVE-Bench (Li et al., 2025) in video editing. Red, blue, and yellow denote the top 3 performers. Full results, including motion and perceptual quality, are provided in Appendix D.

Method	Structure Distance ↓	Background PSNR ↑	Background LPIPS ↓	Background SSIM ↑	CLIP Whole ↑	Overall FiVE-Acc ↑
TokenFlow (Geyer et al., 2024)	35.62	19.06	263.61	72.51	26.46	27.43
DMT (Yatim et al., 2023)	85.95	14.71	404.60	51.64	26.66	48.42
VidToMe (Li et al., 2023)	22.37	21.15	263.91	70.69	26.84	26.77
AnyV2V (Ku et al., 2024)	71.36	15.90	348.59	50.77	24.89	38.02
VideoGrain (Yang et al., 2025)	12.40	27.05	185.21	79.13	25.69	37.23
Pyramid-Edit (Li et al., 2025)	28.65	20.84	276.59	71.72	26.82	43.84
Wan-Edit (Li et al., 2025)	11.92	25.16	91.65	82.97	27.19	49.52
Wan-Edit + Ours	12.06	25.38	88.62	82.94	27.41	59.04

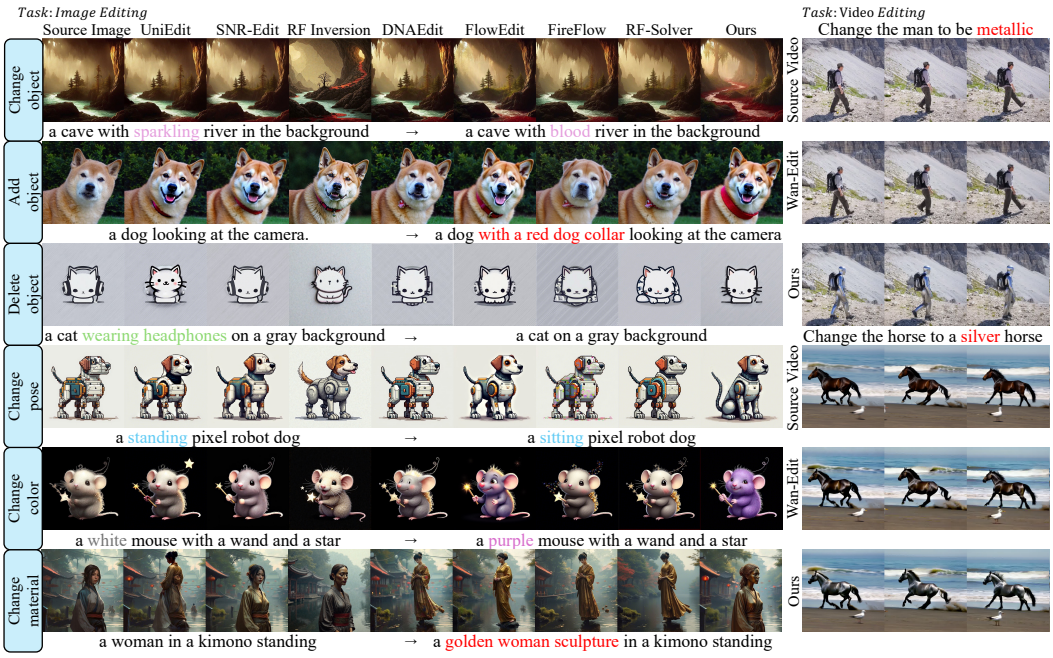

Figure 6: Qualitative comparison. Left: image editing across six task types (object change, addition, deletion, pose, color, and material), comparing our method with inversion-based editors (UniEdit, SNR-Edit, RF Inversion, DNAEdit, FireFlow, RF-Solver) and the inversion-free FlowEdit. Right: video editing on FiVE-Bench, comparing our method with Wan-Edit.

5 Experiments

5.1 Experimental Setup

Datasets. For image editing we use PIE-Bench (Ju et al., 2024), 700 images over ten edit types whose object masks separate the edited region from the background, so that editability and fidelity can be measured independently. For video editing we use FiVE-Bench (Li et al., 2025), 100 videos with 420 editing prompt pairs over six edit types; its SAM2-generated masks (Ravi et al., 2025) support temporal-consistency and background evaluation, and it reports FiVE-Acc, a VLM-judged success rate that measures whether the requested edit was actually carried out.

Baselines. For image editing we compare against training-free editors under the same FLUX.1-dev backbone: RF Inversion (Rout et al., 2025), DNAEdit (Xie et al., 2025), and UniEdit (Jiao et al.,

Table 2: Comparison on PIE-Bench (Ju et al., 2024) in image editing. Methods are ordered by editability, from weakest to strongest. Red, blue, yellow represent top 3 performers.

Method	Type	Structure Distance ↓	Background PSNR ↑	Background MSE ↓	CLIP Whole ↑	CLIP Edit ↑
UniEdit (Jiao et al., 2026)	Inversion	16.24	24.79	5.99	25.39	21.97
SNR-Edit (Jiang et al., 2026)	Inv.-Free	17.42	24.53	5.51	25.34	21.99
RF Inv. (Rout et al., 2025)	Inversion	42.29	20.20	13.92	24.57	22.20
FireFlow (Deng et al., 2025)	Inversion	27.21	23.08	7.07	26.28	22.88
DNAEdit (Xie et al., 2025)	Inversion	26.00	22.71	8.06	26.27	23.02
RF-Solver (Wang et al., 2025b)	Inversion	34.68	22.30	8.80	26.32	23.12
FlowEdit (Kulikov et al., 2025)	Inv.-Free	27.66	21.94	9.18	26.09	22.78
FlowEdit + Ours	Inv.-Free	27.57	22.83	8.27	26.51	23.14

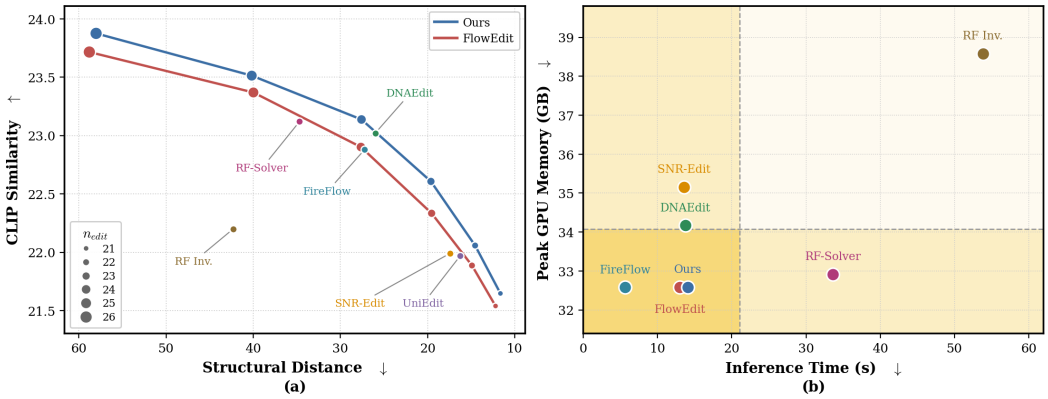

Figure 7: (a) Pareto frontier on PIE-Bench. Marker sizes indicate the number of editing steps n_{edit}. SAATEdit consistently outperforms FlowEdit under all settings. (b) Efficiency on PIE-Bench, peak GPU memory against inference time. SAATEdit improves the editing result while adding only 4% inference time over FlowEdit, with peak GPU memory unchanged.

2026); the higher-order solvers RF-Solver (Wang et al., 2025b) and FireFlow (Deng et al., 2025); and the inversion-free methods FlowEdit (Kulikov et al., 2025) and SNR-Edit (Jiang et al., 2026). FlowEdit is our direct baseline and shares the same backbone, prompts, and sampling schedule, so it isolates the effect of our two mechanisms. For video editing we compare against Wan-Edit and Pyramid-Edit (Li et al., 2025), the attention-injection methods TokenFlow (Geyer et al., 2024), VidToMe (Li et al., 2023), and VideoGrain (Yang et al., 2025), and the inversion-based frameworks AnyV2V (Ku et al., 2024) and DMT (Yatim et al., 2023).

Implementation Details. We implement SAATEdit on FLUX.1-dev (Labs, 2024) for image editing and on Wan2.1 (Wang et al., 2025a) for video editing, without any training or fine-tuning. Both mechanisms act on all attention layers over the first $n_{art} = 4$ editing steps, with the SAE gains fixed to $\alpha_{bg} = 0.5$ and $\alpha_{edit} = 1.5$. Other settings are same to FlowEdit (Kulikov et al., 2025) for image and Wan-Edit (Li et al., 2025) for video, and the complete parameter set is listed in Appendix B.

5.2 Image Editing

Quantitative Comparison. Tab. 2 reports the PIE-Bench results. Relative to its own baseline FlowEdit, SAATEdit improves both structure and semantic alignment: CLIP-Edit rises from 22.78 to 23.14 and CLIP-Whole from 26.09 to 26.51, while Structure Distance, PSNR and MSE also improve (27.66→27.57, 21.94→22.83, 9.18→8.27). Editing capability and structural fidelity therefore improve together rather than trading against each other, and SAATEdit attains the highest CLIP-Edit and CLIP-Whole in the table. Fig. 7(a) shows the same behaviour over the whole edit-strength

Table 3: Ablation Study on PIE-Bench (Ju et al., 2024). Best in each column is in bold.

Method	**Structure** Distance ↓	**Background** PSNR ↑	MSE ↓	**CLIP** Whole ↑	Edit ↑
FlowEdit (Kulikov et al., 2025)	27.66	21.94	9.18	26.09	22.78
+ ART	30.33	21.95	9.64	26.47	23.08
+ ART + SAE **(ours)**	**27.57**	**22.83**	**8.27**	**26.51**	**23.14**

sweep: with n_{edit} varying from 21 to 26, SAATEdit stays above FlowEdit at every setting, and Fig. 7(b) shows that this gain comes at negligible cost: 4% more inference time and unchanged peak memory over FlowEdit.

Qualitative Evaluation. Fig. 6 (left) compares the edits. Fidelity-oriented baselines often return the source image nearly intact when the edit requires a large semantic change, while methods with stronger edits tend to alter regions that were not meant to change. SAATEdit changes the target object clearly and leaves the rest of the image consistent with the source.

5.3 Video Editing

Quantitative and Qualitative Comparison. Tab. 1 reports the FiVE-Bench results, where the dominant gain is in editing success: FiVE-Acc rises from 49.52 to 59.04, a relative improvement of 19%. Because FiVE-Acc is judged by a VLM on each sample independently rather than by a metric we selected, it is direct evidence that the requested edits are carried out more often. The gain is not bought with structure: PSNR and LPIPS improve (25.16→25.38, 91.65→88.62), Structure Distance and SSIM are essentially unchanged, and the only added cost is a 6.1% increase in inference time. Qualitatively, Fig. 6 (right) shows that Wan-Edit often applies the requested attribute only partially, whereas SAATEdit realizes the target concept while keeping the background temporally consistent.

5.4 Ablation Study

Table 3 evaluates the individual contributions of the two proposed mechanisms. Applying ART alone improves the CLIP-Edit score from 22.78 to 23.08, indicating that the transferred reference effectively guides the edit. However, this improvement comes at a cost to structural fidelity. Without adaptive modulation, the editing signal inadvertently affects the background, worsening the Structure Distance from 27.66 to 30.33. By introducing SAE to adaptively modulate the spatial weights, the Structure Distance is restored to 27.57, while the CLIP-Edit score further increases to 23.14. These results demonstrate that the proposed two mechanisms are highly complementary: ART provides an accurate spatial reference, while SAE concentrates the editing strength on the target region to prevent background degradation.

6 Limitations

Despite its effectiveness, SAATEdit has two limitations. First, extracting raw attention score matrices requires architecture-specific adaptation to the backbone's attention layers. Second, when the target image requires large-scale structural changes, the attention reference extracted from the source image can no longer provide an effective spatial prior.

7 Conclusion

In this paper, we identify that the primary failure of inversion-free editing stems from inadequate attention to the target edit region, which severely limits the overall editing strength. To address this, we propose SAATEdit, a training-free framework that explicitly modulates attention to enhance both localization and intensity. Specifically, the ART module transfers spatial priors to accurately delineate the edit region, while the SAE module adaptively amplifies the editing signal within this targeted area. Extensive evaluations on both image (PIE-Bench) and video (FiVE-Bench) benchmarks demonstrate that SAATEdit significantly boosts text alignment and overall editing capability compared to current image and video editing methods.

AI USE STATEMENT

In this work, we used generative AI tools for the translation and polishing of the manuscript, and for assisting with the implementation of the experimental codebase. We have not used generative AI tools to formulate the research idea, to design the proposed method, or to produce the experimental results, and no other task requiring disclosure is applicable to this work. All AI-assisted code was reviewed and tested for correctness by the authors against the officially released baseline implementations, and every AI-polished passage was checked against the authors' original intent. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our research advances inversion-free text-driven image editing by improving editability without sacrificing structural fidelity, offering more controllable creative tools for artists and designers. However, like any advanced generative technology, it carries potential risks of misuse, such as the creation of deceptive content, deepfakes, or misinformation. Because SAATEdit operates entirely on frozen, publicly released backbones without any training or additional parameters, it neither lowers the cost of producing manipulated media nor introduces failure modes beyond those of the base generative models. To mitigate these risks, we strongly advocate for the integration of our framework with robust digital watermarking techniques and provenance tracking systems. Furthermore, by open-sourcing our code, we aim to facilitate the broader research community in developing more advanced detection algorithms for AI-manipulated media.

REPRODUCIBILITY STATEMENT

We provide the complete inference-time procedure of SAATEdit in Algorithm 1 (Appendix A), and a full listing of the hyperparameters used for SAATEdit and for every compared method in Appendix B, including the number of editing steps, the sampling steps, the source and target guidance scales, the ART window, and the SAE gains. The datasets, evaluation protocols, and complete per-category results are described in Appendix D and Appendix E, and further qualitative evidence is given in Appendix F. SAATEdit is training-free and model-agnostic: both mechanisms operate inside the attention computation of a frozen backbone and reuse forward passes that the base editor already performs, so every result can be reproduced with the publicly released FLUX.1-dev and Wan2.1 checkpoints together with the PIE-Bench and FiVE-Bench benchmarks, without any fine-tuning or additional parameters.

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. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7877–7888, 2024. URL https://api.semanticscholar.org/CorpusID:274165561.

Yingying Deng, Xiangyu He, Changwang Mei, Peisong Wang, and Fan Tang. Fireflow: Fast inversion of rectified flow for image semantic editing. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=JFafMSAjUm.

Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. TokenFlow: Consistent diffusion features for consistent video editing. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

Ruiliang Gong, Zhen Wang, Yanghao Wang, and Long Chen. Ridge: Re-noising with internal dynamic guidance for image editing. 2026. URL https://api.semanticscholar.org/CorpusID:290833197.

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 (ICLR)*, 2023.

Jiancheng Huang, Yi Huang, Jianzhuang Liu, Donghao Zhou, Yifan Liu, and Shifeng Chen. Dual-schedule inversion: Training-and tuning-free inversion for real image editing. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 660–669. IEEE, 2025.

Lifan Jiang, Boxi Wu, Yuhang Pei, Tianrun Wu, Yongyuan Chen, Yan Zhao, Shiyu Yu, and Deng Cai. SNR-Edit: Structure-aware noise rectification for inversion-free flow-based editing. *arXiv preprint arXiv:2601.19180*, 2026.

Guanlong Jiao, Biqing Huang, Kuan-Chieh Wang, and Renjie Liao. UniEdit-Flow: Unleashing inversion and editing in the era of flow models. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026.

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 (ICLR)*, 2024.

Jeongsol Kim, Yeobin Hong, Jonghyun Park, and Jong Chul Ye. FlowAlign: Trajectory-regularized, inversion-free flow-based image editing. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026.

Max W.F. Ku, Cong Wei, Weiming Ren, Huan Yang, and Wenhu Chen. Anyv2v: A tuning-free framework for any video-to-video editing tasks. *Trans. Mach. Learn. Res.*, 2024, 2024. URL https://api.semanticscholar.org/CorpusID:268554083.

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

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

Minghan Li, Chenxi Xie, Yichen Wu, Lei Zhang, and Mengyu Wang. Five-bench: A fine-grained video editing benchmark for evaluating emerging diffusion and rectified flow models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16672–16681, 2025.

Xirui Li, Chao Ma, Xiaokang Yang, and Ming-Hsuan Yang. Vidtome: Video token merging for zero-shot video editing. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7486–7495, 2023. URL https://api.semanticscholar.org/CorpusID:266348383.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://api.semanticscholar.org/CorpusID:245704504.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6038–6047, 2023.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollar, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Ha6RTeWMd0.

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.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

Bram Wallace, Akash Gokul, and Nikhil Naik. EDICT: Exact diffusion inversion via coupled transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22532–22541, 2023.

Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Xiaofeng Meng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *CoRR*, abs/2503.20314, March 2025a. URL https://doi.org/10.48550/arXiv.2503.20314.

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 *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=uDreZphNky.

Chenxi Xie, Minghan Li, Shuai Li, Yuhui Wu, Qiaosi Yi, and Lei Zhang. DNAEdit: Direct noise alignment for text-guided rectified flow editing. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*, 2025.

Xiangpeng Yang, Linchao Zhu, Hehe Fan, and Yi Yang. Videograin: Modulating space-time attention for multi-grained video editing. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=SSslAtcPB6.

Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. Space-time diffusion features for zero-shot text-driven motion transfer. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8466–8476, 2023. URL https://api.semanticscholar.org/CorpusID:265466643.

Sung-Hoon Yoon, Minghan Li, Gaspard Beaudouin, Congcong Wen, Muhammad Rafay Azhar, and Mengyu Wang. Splitflow: Flow decomposition for inversion-free text-to-image editing. *arXiv preprint arXiv:2510.25970*, 2025.

APPENDIX

In the supplemental document, we provide:

- §A Full Algorithm.
- §B Experiment Details.
- §C Token-set Construction.
- §D Full Evaluation Results On FiVE-Bench.
- §E Per-Category Results on PIE-Bench.
- §F More Qualitative Results.
- §G Additional Ablation Study.

A FULL ALGORITHM

In this section, we provide the formal pseudocode for our proposed SAATEdit framework. Algorithm 1 outlines the complete inference-time editing pipeline, which regulates the attention computation through two synergistic mechanisms: Attention Reference Transfer (ART) and Spatial Adaptive Enhancement (SAE). ART is employed to transfer the reference attention pattern of the source object tokens into the columns of the target editing tokens, ensuring that the edit region is explicitly identified at the attention level. Simultaneously, SAE derives a bounded, spatially adaptive gain from the same reference, concentrating the editing strength on the target object while suppressing the background. By regulating these two complementary aspects — where to edit and how strongly — in a unified manner, SAATEdit alleviates the insufficient editing strength and achieves effective, inversion-free editing for both images and videos.

Algorithm 1 Full algorithm for SAATEdit

Input: $X_0, \{t_i\}_{i=0}^{n_{edit}}$, prompts c_{src}, c_{tar}, token sets $\mathcal{T}_{src}, \mathcal{T}_{tar}$, ART steps n_{art}, gains $\alpha_{bg}, \alpha_{edit}$
Output: Edited image Z_0^{edit}
Init: $Z_{t_{n_{edit}}}^{edit} \leftarrow X_0$
for $i = n_{edit}$ **to** 1 **do**
  $S_{t_i} \sim \mathcal{N}(0, I)$
  $Z_{t_i}^{src} \leftarrow t_i S_{t_i} + (1 - t_i) X_0$
  $Z_{t_i}^{tar} \leftarrow Z_{t_i}^{edit} + Z_{t_i}^{src} - X_0$
  $V_{t_i}^{src} \leftarrow V_\theta(Z_{t_i}^{src}, t_i, c_{src})$
  if $i > n_{edit} - n_{art}$ **then** ▷ shared ART and SAE window
    $(R, \hat{M}^{tar}) \leftarrow \mathrm{ART}(M^{src}, \mathcal{T}_{src}, \mathcal{T}_{tar})$
    $g \leftarrow \mathrm{Gain}(R)$ ▷ spatial gain, Eqs. 8 and 9
    $\tilde{M}^{tar} \leftarrow \mathrm{Clip}(g \odot \hat{M}^{tar},\ S^{tar}_{\min},\ S^{tar}_{\max})$
    $V_{t_i}^{tar} \leftarrow V_\theta(Z_{t_i}^{tar}, t_i, c_{tar};\ \tilde{M}^{tar} \to S^{tar}[:, \mathcal{T}_{tar}])$
  else
    $V_{t_i}^{tar} \leftarrow V_\theta(Z_{t_i}^{tar}, t_i, c_{tar})$
  end if
  $\Delta V_{t_i} \leftarrow V_{t_i}^{tar} - V_{t_i}^{src}$
  $Z_{t_{i-1}}^{edit} \leftarrow Z_{t_i}^{edit} + (t_{i-1} - t_i)\Delta V_{t_i}$
end for
Return: Z_0^{edit}

B EXPERIMENT DETAILS

All experiments in this paper are run on an NVIDIA L40 GPU with a fixed seed of 42. Similar to existing methods, SAATEdit has a hyperparameter n_{edit} controlling the number of editing steps, together with the total number of sampling steps and the source/target guidance scales. In Tab. 2 (main paper), we compare SAATEdit with several baseline methods, including FlowEdit, DNAEdit, UniEdit, SNR-Edit, and RF Inv. Below, we provide detailed descriptions of the hyperparameters used for the FLUX backbone. All results presented (see the qualitative comparison figure, the Δv visualization and Pareto figures in the main paper, and the additional results in this appendix) are obtained using the specified hyperparameter settings. For each method presented in this paper, we use a fixed set of hyperparameters throughout all experiments.

Table 4: Hyperparameters on the FLUX backbone for the main PIE-Bench comparison. All methods use 28 total sampling steps and seed 42.

Method	n_{edit}	src@guidance	tar@guidance	Additional
FlowEdit	24	1.5	5.5	—
DNAEdit	26	1.0	2.5	MVG $= 0.8$
UniEdit	15	1.0	1.0	$\alpha = 0.6$, $\omega = 5.0$, inject $= 0$, layers 0–0
SNR-Edit	24	1.5	5.5	SPE: noise 0.9, feature 0.1
SAATEdit (ours)	24	1.5	5.5	Layers: all; first $n_{art} = 4$; $\alpha_{\text{bg}}, \alpha_{\text{edit}} = 0.5, 1.5$

For all methods, we adopt the officially recommended hyperparameters whenever available. In cases where multiple configurations are suggested, we select intermediate values within the provided ranges to ensure a fair and balanced comparison. For DNAEdit, we additionally set the MVG parameter to 0.8, and for UniEdit we use the default configuration of the official repository. For SNR-Edit, the structured prior is enabled with a noise scale of 0.9 and a feature scale of 0.1.

Table 5: Hyperparameters for the additional PIE-Bench baselines. Both use seed 42.

Method	Steps	guidance	Additional
RF Inv.	28	—	$\gamma = 0.5$, $\tau = 7$, $\eta = 0.9$, SDE enabled
RF-Solver	30	2.0	inject $= 4.5$
FireFlow	8	2.0	inject $= 1$, layers 0–37

Similarly, in Tab. 2 (main paper), we additionally compare SAATEdit with RF-Solver and FireFlow on PIE-Bench. Additionally, the Layer Range in FireFlow is set to 0–37.

C TOKEN-SET CONSTRUCTION

ART and SAE act on two token sets: the subword tokens of the word naming the edited entity in the source prompt ($\mathcal{T}_{src}$) and of the word naming the new content in the target prompt ($\mathcal{T}_{tar}$). Both are read directly off the prompt pair by an off-the-shelf LLM: given any source prompt and target prompt, the same extraction applies, with no edit-type label, no annotation from the benchmark, and no parser output.

The two returned words are then mapped to token indices deterministically: we re-tokenize the prompt with the backbone's text tokenizer, strip the subword boundary markers, concatenate the tokens, and locate the anchor word as a case-insensitive contiguous span, returning the corresponding index set. Because the span is matched explicitly, text-padding tokens can never be included. For FLUX.1-dev the indices refer to the T5-XXL sequence that feeds the joint attention, and for Wan2.1 to the UMT5 sequence.

The extraction prompt is given verbatim below.

```
You are extracting two attention anchor words from a source/target
prompt pair.

Given a SOURCE prompt describing the original content and a TARGET
prompt describing the intended edit, identify:

- SRC_WORD: the word in the SOURCE prompt naming what changes.
- TAR_WORD: the word in the TARGET prompt naming the new content that
  should appear or be emphasized.

Rules, in priority order:
1. LITERAL: each returned word must occur verbatim in its own prompt,
   matched case-insensitively as a contiguous span of subword tokens.
   Never paraphrase and never inflect. If no such word exists, return
   null for that side.
2. SPECIFICITY: anchor the entity that is edited, not its owner,
   container, or the surrounding scene. If the edit concerns an object
   that a person wears, holds, or rides, choose the object, not the
   person.
   "Change the skier's outfit to purple" -> SRC_WORD = "outfit".
   "Change the white horse to black" -> SRC_WORD = "horse".
   "Change the rollerblades to metallic" -> SRC_WORD = "rollerblades".
3. SOURCE ANCHOR: SRC_WORD names that entity in the SOURCE prompt.
   "a city bus" -> "a purple city bus": SRC_WORD = "bus".
4. TARGET ANCHOR: TAR_WORD names the new content in the TARGET prompt,
   i.e. a new object or the new value of the changed attribute.
   "... -> a purple city bus": TAR_WORD = "purple".
   "a city bus" -> "a jeep": TAR_WORD = "jeep".
5. ONE-SIDED EDITS: if the edited entity disappears from the TARGET
   prompt without being replaced, TAR_WORD is that same word when it
   still occurs there (typically negated, e.g. "without a backpack"),
   otherwise null. If instead a new thing appears that was absent from
   SOURCE prompt, SRC_WORD is null and TAR_WORD is that content.
6. WORD FORM: a multi-word anchor is allowed only when the phrase is a
   single lexical unit ("carbon fiber"); otherwise return the
   head word.
   Prefer a noun. Never return more than two tokens.

Return JSON only:
{"src_word": <string|null>, "tar_word": <string|null>,
 "src_reason": <one short clause>, "tar_reason": <one short clause>}
```

D FULL EVALUATION RESULTS ON FIVE-BENCH

In this section, we present the comprehensive evaluation results on FiVE-Bench. Tab. 6 and 7 show the quantitative performance of SAATEdit compared to existing methods across various video editing metrics, demonstrating the robust spatio-temporal consistency and precise semantic alignment of our method in video generation scenario; Tab. 8 further reports the per-category results of our method.

Table 6: Full comparison on FiVE-Bench (Li et al., 2025) for structure, background preservation, and CLIP similarity. Red, blue, and yellow represent the top 3 performers.

Method	**Structure**	**Background Preservation**				**CLIP Similarity**	
	Distance ↓	PSNR ↑	LPIPS ↓	MSE ↓	SSIM ↑	Whole ↑	Edit ↑
TokenFlow (Geyer et al., 2024)	35.62	19.06	263.61	138.65	72.51	26.46	21.15
DMT (Yatim et al., 2023)	85.95	14.71	404.60	372.78	51.64	26.66	21.44
VidToMe (Li et al., 2023)	22.37	21.15	263.91	88.75	70.69	26.84	21.05
AnyV2V (Ku et al., 2024)	71.36	15.90	348.59	342.97	50.77	24.89	19.72
VideoGrain (Yang et al., 2025)	12.40	27.05	185.21	25.10	79.13	25.69	20.31
Pyramid-Edit (Li et al., 2025)	28.65	20.84	276.59	95.63	71.72	26.82	20.20
Wan-Edit (Li et al., 2025)	11.92	25.16	91.65	41.58	82.97	27.19	21.45
Wan-Edit + Ours	12.06	25.38	88.62	35.11	82.94	27.41	21.52

Table 7: Full comparison on FiVE-Bench (Li et al., 2025) for perceptual quality, motion fidelity, and FiVE-Acc. Red, blue, and yellow represent the top 3 performers.

Method	**Quality & Motion**			**FiVE-Acc**				
	NIQE ↓	MFS ↑	Time ↓	YN ↑	MC ↑	FiVE-∪ ↑	FiVE-∩ ↑	FiVE-Acc ↑
TokenFlow (Geyer et al., 2024)	4.01	89.00	8.04	19.36	35.51	36.68	18.18	27.43
DMT (Yatim et al., 2023)	5.24	82.30	25.98	34.78	62.06	62.98	33.86	48.42
VidToMe (Li et al., 2023)	4.68	90.06	3.25	20.03	33.50	36.20	17.34	26.77
AnyV2V (Ku et al., 2024)	5.04	60.36	6.11	30.62	45.42	48.96	27.09	38.02
VideoGrain (Yang et al., 2025)	4.08	88.57	27.12	30.50	43.97	44.30	30.17	37.23
Pyramid-Edit (Li et al., 2025)	5.48	80.59	1.44	33.67	54.01	56.36	31.31	43.84
Wan-Edit (Li et al., 2025)	5.66	88.55	4.43	40.33	58.71	60.38	38.66	49.52
Wan-Edit + Ours	5.71	88.24	4.70	49.64	68.74	70.41	47.61	59.04

Table 8: Per-category results of our method on FiVE-Bench.

Category	n	Struct ↓	PSNR ↑	LPIPS ↓	SSIM ↑	CLIP$_{wh}$ ↑	FiVE-Acc ↑
1 Rigid replacement	100	13.13	25.09	89.40	82.58	27.19	54.60
2 Non-rigid replacement	100	14.34	24.60	94.50	81.99	26.92	55.60
3 Color	100	10.92	25.90	85.50	83.64	28.27	81.30
4 Material	100	10.50	25.76	86.00	83.46	27.79	48.00
5 Add object	10	9.33	25.11	93.40	80.26	25.12	83.30
6 Delete object	10	8.05	27.08	75.00	86.35	24.15	5.00
All	420	12.06	25.38	88.62	82.94	27.41	59.04

E PER-CATEGORY RESULTS ON PIE-BENCH

Table 9: Per-category results of our method on PIE-Bench (Ju et al., 2024), with all categories evaluated under the same setting as Tab. 2. The background-preservation metrics are computed over the images whose unedited region is non-empty, which excludes most of the object-addition, object-deletion, and style samples.

Category		n	**Structure**	**Background Preservation**			**CLIP Similarity**	
			Distance ↓	PSNR ↑	LPIPS ↓	SSIM ↑	Whole ↑	Edit ↑
0	Random	140	25.56	24.44	9.42	0.861	25.56	22.67
1	Change object	80	21.81	23.37	9.84	0.848	25.20	20.17
2	Add object	80	28.26	22.29	10.99	0.847	27.96	25.75
3	Delete object	80	30.39	19.36	16.23	0.764	25.69	19.11
4	Change attribute (content)	40	22.47	23.44	10.14	0.851	26.09	22.18
5	Change attribute (pose)	40	23.82	23.81	9.49	0.848	26.64	22.63
6	Change attribute (color)	40	28.63	22.81	11.10	0.842	27.48	22.35
7	Change attribute (material)	40	18.76	25.55	9.28	0.883	26.13	23.58
8	Change background	80	31.69	21.52	9.64	0.851	26.84	24.10
9	Change style	80	37.56	20.45	18.44	0.831	28.41	28.28
All		700	27.57	22.83	10.83	0.841	26.51	23.14

F More Qualitative Results

To further demonstrate the versatility and visual quality of our proposed method, we provide additional qualitative results in this section. Fig. 8 shows extended video editing comparisons, highlighting the temporal stability of SAATEdit. Fig. 9 presents a broader range of image editing scenarios, where SAATEdit consistently executes complex text instructions while rigorously protecting unedited background structures.



Figure 8: Additional qualitative results on video editing. Each block shows five consecutive frames of a source video together with the results of Wan-Edit and our method.



Figure 9: Additional qualitative results on image editing. Rows cover object addition, object deletion, pose change, style transfer, and background replacement; columns compare the source image with UniEdit, SNR-Edit, RF Inversion, DNAEdit, FlowEdit, FireFlow, RF-Solver, and our method.

G Additional Ablation Study

Tab. 10 isolates the two design choices behind our two mechanisms: the window over which they are applied, and whether the gain that modulates the transferred reference is spatially adaptive. For comparability with the main text, we report the same five metrics as in Tabs. 2 and 3.

Effect of the ART window. Applying ART and SAE only during the first $n_{art} = 4$ editing steps is clearly preferable to applying them over the whole editing trajectory. Extending the window consistently degrades background preservation (Structure Distance $27.57 \rightarrow 29.94$, PSNR $22.83 \rightarrow 22.74$, MSE $8.27 \rightarrow 8.58$) without buying any additional editability (CLIP-Edit $23.14 \rightarrow 23.06$). Once the edit region has been localized, continuing to inject the transferred reference constrains the spatial extent of the edit too rigidly and pushes the attention scores away from the training distribution, while the remaining unconstrained steps are what allow the model to refine the edit. Our default setting also improves on the FlowEdit baseline on all five metrics.

Table 10: Effect of the ART/SAE window and of the spatial gain on PIE-Bench (Ju et al., 2024). All rows use $T = 28$ sampling steps, $n_{edit} = 24$, and seed 42; the FlowEdit row quotes the implementation used in the main comparison.

Variant	Struct ↓	PSNR ↑	MSE ↓	CLIP-W ↑	CLIP-E ↑
FlowEdit (no ART, no SAE)	27.66	21.94	9.18	26.09	22.78
ART+SAE, first 4 steps **(ours)**	**27.57**	**22.83**	**8.27**	**26.51**	**23.14**
ART+SAE, all editing steps	29.94	22.74	8.58	26.54	23.06
ART+SAE, constant gain 2.0	29.27	22.05	8.96	26.15	22.85

Effect of the spatial gain. We also compare our spatially adaptive gain g_j with a constant gain of 2.0, i.e., the largest value g_j can take. A gain that does not depend on the transferred reference amplifies the response in the background regions together with the target object, so the edit is no longer confined to the edit region; consistently, the constant gain is worse on all five metrics, degrading Structure Distance from 27.57 to 29.27 and CLIP-Edit from 23.14 to 22.85. Spatial adaptivity of the gain is therefore what makes the stronger edit of ART compatible with background preservation.