PRELIMINARY PREPRINT VERSION – DO NOT CITE
This is not the version of record. The final version will appear in the Proceedings of IJCAI-ECAI 2026.

EKFEdit: Extended Kalman Filter for Training-Free Flow-Based Image Editing

Chengfu Zou[1], **Yaochen Li**[1*], **Yi Han**[1], **Wenlong Zhou**[1] and **Jia Shu**[1]
[1]School of Software Engineering, Xi'an Jiaotong University
{chengfuzou, winghan, 0long0, 2214322027}@stu.xjtu.edu.cn
yaochenli@mail.xjtu.edu.cn

Abstract

Despite recent advances in training-free text-guided image editing using pretrained rectified flow models, achieving a balance between text adherence and background preservation remains a challenge. Existing methods typically treat image editing as an open-loop trajectory generation task, adopting either an inversion-based or inversion-free paradigm. However, neither can perfectly achieve this goal. Specifically, inversion-based approaches accumulate inversion errors without feedback, whereas inversion-free methods exhibit unstable trajectories due to the lack of constraints. To address this, we propose EKFEdit, a novel framework that formulates image editing as a sequential state estimation problem in a nonlinear dynamic system, which models inversion-free editing trajectories as the system's prior dynamics and leverages an Extended Kalman Filter to adaptively fuse observations from external editing algorithms. EKFEdit enables adaptive trajectory correction, effectively combining the complementary strengths of the different paradigms. We instantiate this framework into two concrete approaches (EKFEdit-RF and EKFEdit-DNA) via different observation methods, respectively. Extensive experiments demonstrate that our approaches achieve new state-of-the-art performance compared to previous methods. Code is available at https://github.com/baiyeweiguang/EKFEdit.

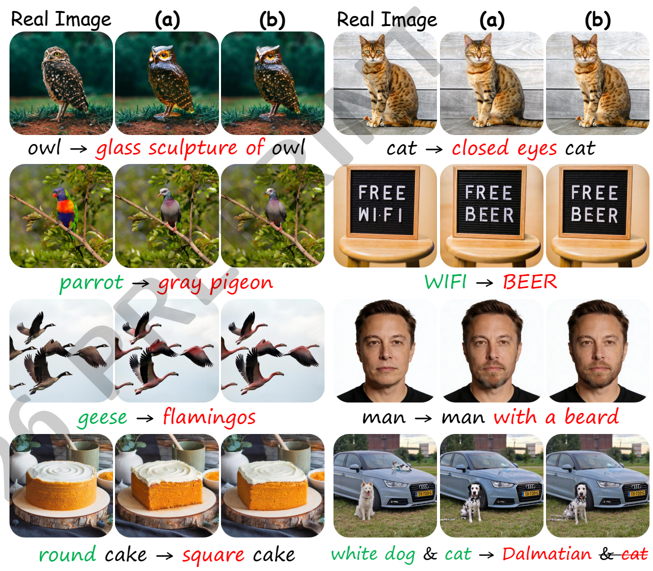

Figure 1: Representative results of EKFEdit-RF **(a)** and EKFEdit-DNA **(b)**.

1 Introduction

Rectified Flow (RF) models [Liu *et al.*, 2022; Lipman *et al.*, 2022; Albergo and Vanden-Eijnden, 2022] have recently emerged as a compelling alternative to Diffusion Models [Ho *et al.*, 2020; Rombach *et al.*, 2022] for text-to-image (T2I) generation. By constructing straight transport paths between noise and data distributions, RF models achieve superior generation quality with significantly fewer sampling steps. Building on pretrained large-scale RF models [Labs, 2024; Esser *et al.*, 2024; Stability AI, 2024], training-free image editing aims to modify visual content via text guidance while preserving the structural fidelity of the source image.

Previous editing approaches typically follow an inversion-based paradigm, which first inverts the source image into a noise representation and then reconstructs it under the guidance of a target prompt. This pipeline faces a fundamental limitation: the inversion process inevitably introduces discretization error. Since the exact velocity field at previous timestep is unknown during inversion, local approximations must be used. In the absence of an effective feedback mechanism, these errors are not rectified but instead accumulate along the trajectory. Consequently, this leads to background distortion and loss of details in the output.

To address reconstruction degradation caused by inversion, recent research has explored inversion-free approaches, such as FlowEdit [Kulikov *et al.*, 2025]. Instead of explicit inversion, these methods construct a direct transport path mapping the source distribution to the target distribution. By coupling velocity fields, they attempt to transport the image latent directly. Without accumulated inversion errors, inversion-free methods typically perform better at preserving unedited de-

*Corresponding author.

tails. However, they suffer from trajectory instability and dynamic oscillation. Lacking a structural anchor to guide the generation process, the open-loop editing path becomes susceptible to layout collapse or semantic drift, with results often tending to be either under-edited or over-edited.

Thus, a critical trade-off persists in existing work. Inversion-based methods struggle with the loss of background details due to cumulative inversion errors. On the other hand, inversion-free approaches excel at preserving local details, but often suffer from structural inconsistency and semantic misalignment. This raises a core question:

Can we leverage the detail preservation of inversion-free dynamics while stabilizing their trajectories to ensure robust structural and semantic alignment?

To tackle this challenge, we propose **E**xtended **K**alman **F**ilter **Edit**ing (**EKFEdit**), a novel framework that reformulates the image editing process as an optimal state estimation problem. We view the inversion-free trajectory as a noisy system dynamic and employ an Extended Kalman Filter (EKF) to stabilize it using outputs from auxiliary algorithms as observations. EKFEdit effectively corrects the deviations of the inversion-free path to realize its potential for high-fidelity editing while ensuring structural consistency. To ensure computational feasibility, we further implement a gradient-free Jacobian Diagonal Approximation, which enables efficient state estimation in high-dimensional latent space. Building on this approximation, we instantiate our framework with two variants, **EKFEdit-RF** and **EKFEdit-DNA** using RF-Solver [Wang *et al.*, 2024] and DNAEdit [Xie *et al.*, 2025] as external observers, respectively.

In summary, our contributions are:

- We propose EKFEdit, a novel framework to interpret flow-based editing as an optimal state estimation problem. We instantiate it into two variants, EKFEdit-RF and EKFEdit-DNA, which achieve state-of-the-art performance by unifying the strengths of different editing methods.
- We develop an efficient, gradient-free EKF implementation using Jacobian Diagonal Approximation, minimizing the computational overhead for high-dimensional models.
- We further incorporate a Trajectory Alignment technique and dynamic noise scheduling into EKFEdit implementation, which effectively reduce ghosting artifacts and improve the editing performance.

2 Related Work

2.1 Inversion-Based Image Editing

To address reconstruction degradation, existing inversion-based methods focus on refining inversion accuracy. RF-Solver [Wang *et al.*, 2024] and FireFlow [Deng *et al.*, 2024] adopt high-order solvers for accurate velocity field estimation to alleviate inversion errors. FTEdit [Xu *et al.*, 2025] elevates the inversion precision via iterative average velocity. Other works [Zhu *et al.*, 2025b; Avrahami *et al.*, 2025; Xu *et al.*, 2024; Yan *et al.*, 2025] boost background consistency through complex model-specific attention mechanisms. By contrast, DNAEdit [Xie *et al.*, 2025] directly estimates initial noise in the noise space but still suffers from minor initial errors that amplify progressively during generation.

2.2 Inversion-Free Image Editing

To fully avoid inherent flaws of the inversion process, FlowEdit [Kulikov *et al.*, 2025] pioneered constructing an Ordinary Differential Equation (ODE) that directly maps source and target distributions, without requiring explicit estimation of the initial noise. However, for complex scenarios, FlowEdit experiences editing trajectory drift due to inconsistent velocity directions, often causing source structure damage (over-edited) or editing failure (under-edited). To mitigate this, TweezeEdit [Mao *et al.*, 2025] strengthens source preservation via regularization and SplitFlow [Yoon *et al.*, 2025] decomposes prompts and flow fields and aggregates outputs to alleviate the under-edited versus over-edited trade-off.

2.3 Control Theory in Generative Processes

Control theory has been effectively applied to model generative processes, with validated use in diffusion models [Berner *et al.*, 2022; Koo *et al.*, 2024; Rout *et al.*, 2024b; Zhu *et al.*, 2025a]. While previous flow-based control methods [Rout *et al.*, 2024a; Zhou *et al.*, 2025] focusing on optimizing the inversion process via linear quadratic regulators, FlowAlign [Kim *et al.*, 2025] extends optimal control to inversion-free editing, enforcing trajectory smoothness to enhance source consistency.

While Kalman-Edit [Chi *et al.*, 2025] introduces Kalman filtering to flow-based editing, it ignores time-varying covariance by employing a simple identity matrix as the state transition matrix and fundamentally acts as a regulator that tracks a static, pre-computed reference trajectory. This reliance on a decoupled reference fails to adapt to real-time deviations, often introducing artifacts when the editing direction conflicts with the reference.

In contrast, EKFEdit reformulates editing as an optimal state estimation problem within a nonlinear stochastic system. By leveraging the EKF to linearize system dynamics, our method enables real-time adaptive correction of trajectory drift, fundamentally diverging from the rigid trajectory tracking of the prior work.

3 Method

3.1 Preliminaries

Rectified Flow Models. Rectified Flow (RF) [Lipman *et al.*, 2022; Liu *et al.*, 2022] learns a transport map between distributions π_0 and π_1 via an ODE on $t \in [0, 1]$:

$$d\mathbf{z}_t = v_\theta(\mathbf{z}_t, t)dt, \tag{1}$$

where v_θ is a neural velocity field. To ensure stable training and efficient sampling, RF enforces straight-line trajectories via linear interpolation $\mathbf{z}_t = (1-t)\mathbf{z}_0 + t\mathbf{z}_1$.

Inversion-Based Image Editing. These methods seek to recover a latent noise $\mathbf{z}_T$ that reconstructs the source image.

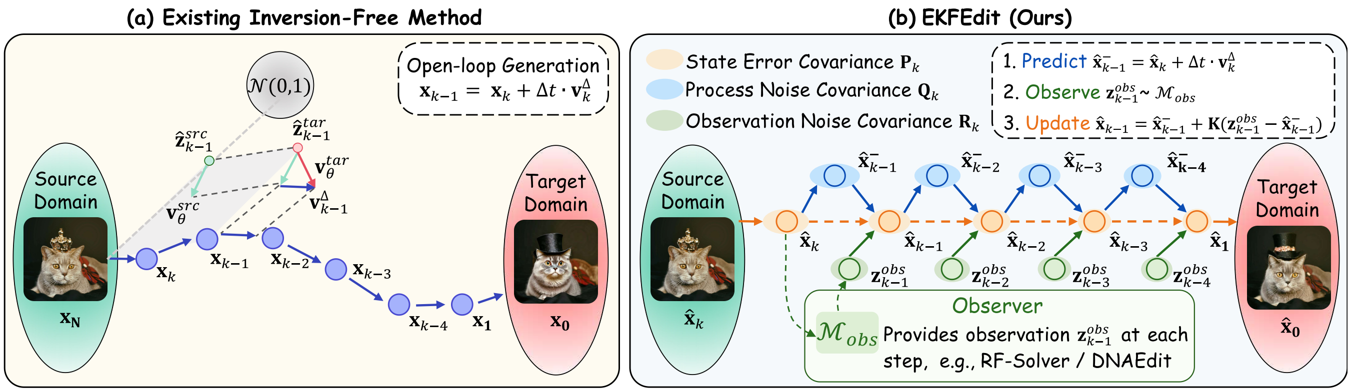

Figure 2: Overview of EKFEdit vs. existing inversion-free methods. EKFEdit fuses inversion-free dynamics with independent editing observations via EKF, yielding more stable trajectories and reduced semantic drift.

However, the discrete inversion usually approximates the velocity at the unknown state $\mathbf{z}_k$ using the known state $\mathbf{z}_{k+1}$:

$$v_\theta(\mathbf{z}_k) \approx v_\theta(\mathbf{z}_{k+1}, t_{k+1}). \tag{2}$$

This approximation introduces local truncation errors that accumulate along the trajectory. Consequently, the inverted noise diverges from the standard Gaussian ($\mathbf{z}_T \notin \mathcal{N}(0, I)$), leading to background distortion and detail loss in the final output.

Inversion-Free Editing. Leveraging the linear property of Rectified Flow, inversion-free methods, such as FlowEdit [Kulikov *et al.*, 2025], bypass latent inversion by constructing a transition path directly in the image space. The latent in the direct path, Z_t^{FE}, is approximated via parallelogram law:

$$Z_t^{\mathrm{FE}} \approx X^{\mathrm{src}} + \hat{Z}_t^{\mathrm{tar}} - \hat{Z}_t^{\mathrm{src}}. \tag{3}$$

This trajectory evolves under an ODE driven by the relative velocity difference:

$$dZ_t^{\mathrm{FE}} = \left[v_\theta(\hat{Z}_t^{\mathrm{tar}}, t, c^{\mathrm{tar}}) - v_\theta(\hat{Z}_t^{\mathrm{src}}, t, c^{\mathrm{src}}) \right] dt. \tag{4}$$

Here, the pseudo-source $\hat{Z}_t^{\mathrm{src}}$ is linearly interpolated from the known source X^{src}, while the unknown pseudo-target is approximated using the current state: $\hat{Z}_t^{\mathrm{tar}} \approx Z_t^{\mathrm{FE}} + \hat{Z}_t^{\mathrm{src}} - X^{\mathrm{src}}$. Iteratively updating Z_t^{FE} using this velocity difference gradually maps the source image toward the target. This process is shown in Figure 2 (a).

Extended Kalman Filter. The Extended Kalman Filter (EKF) is a recursive estimator for nonlinear dynamical systems, which propagates uncertainty by locally linearizing the transition and observation functions. Consider a discrete-time nonlinear system:

$$\begin{aligned} \mathbf{x}_k &= f(\mathbf{x}_{k-1}) + \mathbf{w}_k, \qquad && \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k), \\ \mathbf{z}_k &= h(\mathbf{x}_k) + \boldsymbol{\nu}_k, && \boldsymbol{\nu}_k \sim \mathcal{N}(0, \mathbf{R}_k), \end{aligned} \tag{5}$$

where $\mathbf{x}_k$ denotes the system state, $\mathbf{z}_k$ is the observation, and $f(\cdot), h(\cdot)$ are the nonlinear transition and observation function, respectively.

EKF linearizes these functions via first-order Taylor expansion, computing Jacobian matrices $\mathbf{F}_k = \frac{\partial f}{\partial \mathbf{x}}|_{\hat{\mathbf{x}}_{k-1}}$ and $\mathbf{H}_k = \frac{\partial h}{\partial \mathbf{x}}|_{\hat{\mathbf{x}}_k^-}$. Here, $\mathbf{P}_k$ denotes the posterior state estimation error covariance, $\mathbf{P}_k^-$ denotes its predicted counterpart, while $\mathbf{Q}_k$ and $\mathbf{R}_k$ denote the process and observation noise covariances, respectively. State estimation proceeds recursively:

Prediction:

$$\begin{aligned} \hat{\mathbf{x}}_k^- &= f(\hat{\mathbf{x}}_{k-1}), \\ \mathbf{P}_k^- &= \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_{k-1}. \end{aligned} \tag{6}$$

Update:

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_k^- \mathbf{H}_k^T \left(\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k \right)^{-1}, \\ \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k \left(\mathbf{z}_k - h(\hat{\mathbf{x}}_k^-) \right). \end{aligned} \tag{7}$$

3.2 Overall Framework of EKFEdit

Existing flow-based image editing methods operate fundamentally in an open-loop manner. Without feedback to actively correct deviations, inversion-based approaches accumulate errors over the trajectory. Inversion-free methods assume that the coupled velocity fields would produce the expected output, but do not account for the discrepancy between actual and expected outputs. In practice, the lack of explicit constraints often leads to trajectory drift and makes it hard to reach the optimal balance between text adherence and structural preservation.

To address this limitation, we argue that image editing should not be viewed as a rigid trajectory generation task, but rather as an optimal state estimation problem within a stochastic dynamic system. Under this perspective, the inversion-free trajectory can serve as a noisy prior of the system's state, while outputs from auxiliary editing algorithms provide external observations.

Based on this insight, we propose **EKFEdit**, a unified framework that employs the Extended Kalman Filter to dynamically fuse these two information sources. As illustrated in Figure 2 (b), EKFEdit adaptively weights the predicted state with external observations via Kalman gain. This method transforms the editing process into an estimation system that dynamically corrects trajectory drift and oscillations of the inversion-free prior. Moreover, it avoids reliance on background information embedded with accumulated errors unlike inversion-based methods.

Since generative flow models denoise from $t = 1$ to $t = 0$, we apply EKF on the reverse timeline, where each iteration estimates the transition from t_k to t_{k-1} through two steps:

- **Prediction Step:** This step predicts a prior state transition using the inversion-free trajectory as the system dynamics.
- **Update Step:** This rectifies the predicted state by fusing it with external observations via the Kalman gain, dynamically injecting semantic guidance to correct trajectory drifts.

3.3 State Space Modeling on Latent Space

We define the high-dimensional latent representation of the image as system state $\mathbf{x}_k$, which is sequentially estimated by fusing predictions with observations.

Prediction with Inversion-Free Dynamics. Following the approach in FlowEdit [Kulikov *et al.*, 2025], we treat the coupling velocity field associated with the direct path from source image distribution to target image distribution as the system's state transition function. The prediction step aims to estimate the prior state $\hat{\mathbf{x}}_{k-1}^-$ at timestep t_{k-1} based on the state at t_k, which is defined as:

$$\hat{\mathbf{x}}_{k-1}^- = f(\hat{\mathbf{x}}_k) = \hat{\mathbf{x}}_k + \Delta t \cdot v^\Delta(\hat{\mathbf{x}}_k, t_k), \tag{8}$$

where $\Delta t = t_{k-1} - t_k$, and v^Δ represents the relative velocity difference:

$$v^\Delta = v_\theta(\hat{\mathbf{z}}_k^{\text{tar}}, c^{\text{tar}}; t_k) - v_\theta(\hat{\mathbf{z}}_k^{\text{src}}, c^{\text{src}}; t_k), \tag{9}$$

where $\hat{\mathbf{z}}_k^{\text{tar}}$ denotes the intermediate state of the noise-to-target trajectory estimated via the parallelogram law: $\hat{\mathbf{z}}_k^{\text{tar}} = \hat{\mathbf{x}}_k + \hat{\mathbf{z}}_k^{\text{src}} - \mathbf{x}_N$, and $\hat{\mathbf{z}}_k^{\text{src}}$ is obtained by forward process as $\hat{\mathbf{z}}_k^{\text{src}} = (1-t)\mathbf{x}_N + t\mathbf{N}_t$, where $\mathbf{N}_t$ is a random Gaussian noise.

The predicted error covariance $\mathbf{P}_{k-1}^-$ follows Eq. (6), with the transition Jacobian $\mathbf{F}_k = \frac{\partial f}{\partial \mathbf{x}}|_{\hat{\mathbf{x}}_k} = \mathbf{I} + \Delta t \frac{\partial \mathbf{v}^\Delta}{\partial \mathbf{x}}|_{\hat{\mathbf{x}}_k}$.

Update using Independent Observation Fusion. To correct drift in predicted trajectories, we introduce an independent observer module $\mathcal{M}_{obs}$, which produces a target-aware latent observation at the same timestep as the state to be estimated. Given its internal state $\mathbf{x}_k^{obs}$, the current EKF state $\hat{\mathbf{x}}_k$, and the source-target prompts, the observer outputs

$$\mathbf{z}_{k-1}^{obs} = \mathcal{M}_{obs}(\mathbf{x}_k^{obs}, \hat{\mathbf{x}}_k, c^{src}, c^{tar}, t_k). \tag{10}$$

The observation lies in the same latent space as $\mathbf{x}_{k-1}$ and serves as an alternative estimate carrying the target semantics.

We define the observation function as direct observation, given by $\mathbf{z}_{k-1} = \mathbf{I}\mathbf{x}_{k-1} + \boldsymbol{\nu}_{k-1}$. State update is completed through Kalman gain $\mathbf{K}_{k-1}$:

$$\hat{\mathbf{x}}_{k-1} = \hat{\mathbf{x}}_{k-1}^- + \mathbf{K}_{k-1}(\mathbf{z}_{k-1}^{\text{obs}} - \hat{\mathbf{x}}_{k-1}^-). \tag{11}$$

It indicates that the posterior state is a weighted sum of prediction and observation, with weights dynamically determined by $\mathbf{K}$. The final edited state is equivalent to the last posterior state $\hat{\mathbf{x}}_0$.

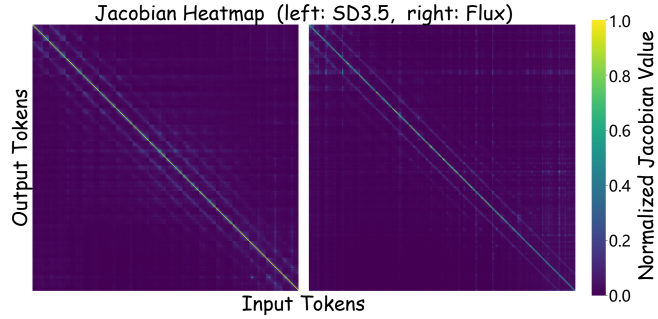

Figure 3: Jacobian energy visualization for SD3.5 and Flux on the same 16×16 latent-token grid, i.e., 256 tokens per model. Each entry denotes the aggregated Jacobian-block norm between an input and output token.

3.4 Jacobian Diagonal Approximation

Diagonal Assumption via Locality. The state dimension in diffusion models is typically very large ($d \sim 10^4 - 10^5$), making the construction and propagation of a full-rank Jacobian computationally prohibitive. Fortunately, recent studies [Kamb and Ganguli, 2025; Lukoianov *et al.*, 2025] show that diffusion denoisers exhibit strong spatial locality: each output element is primarily sensitive to a limited neighborhood of the input. This property suggests that the dominant local dependency can be captured by diagonal or near-diagonal Jacobian components. We empirically verify this observation in Figure 3 on two representative T2I backbones, SD3.5 [Stability AI, 2024] and Flux [Labs, 2024]. The token-level Jacobian energy maps show that gradient energy is concentrated around aligned input-output tokens in both models. Motivated by this locality, we assume approximate independence between state dimensions within minute timesteps and estimate only the diagonal entries of $\mathbf{F}_k$.

Gradient-Free Approximation. We estimate the diagonal of the transition Jacobian with a finite-difference approximation, avoiding memory-intensive backpropagation:

$$\begin{aligned}\operatorname{diag}(\mathbf{F}_k) \approx \mathbf{1} + \frac{\Delta t}{\epsilon}\Big(\big(&v_\theta(\hat{\mathbf{z}}_k^{\text{tar}} + \epsilon\mathbf{u}, c^{\text{tar}}; \mathbf{t}_k) \\ &- v_\theta(\hat{\mathbf{z}}_k^{\text{tar}}, c^{\text{tar}}; \mathbf{t}_k)\big) \odot \mathbf{u}^{-1}\Big).\end{aligned} \tag{12}$$

In practice, we set $\mathbf{u} = v_\theta(\hat{\mathbf{z}}_k^{\text{tar}}, c^{\text{tar}}; t_k)$, use a small ϵ, and clip the estimated values for stability. This requires only one additional inference step.

Tensorized Implementation. Consistent with the diagonal assumption, we model $\mathbf{P}, \mathbf{Q}, \mathbf{R}$ as diagonal tensors matching the shape of $\mathbf{x}$. This reformulation converts matrix multiplications into element-wise operations, significantly boosting inference efficiency by reducing space complexity from $O(d^2)$ to $O(d)$.

3.5 Instantiation of EKFEdit

A simplified overview of the EKFEdit procedure is provided in Algorithm 1. Following previous work [Kulikov *et al.*, 2025; Xie *et al.*, 2025], we keep the optional start step T_s to control when to perform editing.

Algorithm 1 EKFEdit Framework

Input: Source image I_{src}, Source/Target text $c_{\text{src}}, c_{\text{tar}}$, Number of sampling steps N, Start Step T_s, Process/Observation Noise $\mathbf{Q}$, $\mathbf{R}$, $\mathbf{P}_{\text{init}}$
Output: I_{tar}

$\mathbf{z}_{\text{src}} \leftarrow \text{VAE}(I_{\text{src}})$;
$\hat{\mathbf{x}}_N \leftarrow \mathbf{z}_{\text{src}}$; $\quad z_{N+1}^{\text{obs}} \leftarrow \mathbf{z}_{\text{src}}$; $\quad \mathbf{P}_N \leftarrow \mathbf{P}_{\text{init}}$;
Init $\mathbf{x}_N^{obs}$ according to specific Observer.
for $k = N - T_s$ **down to** 1 **do**
  // Prediction Step
  $\hat{\mathbf{x}}_{k-1}^{-} \leftarrow f_{\text{flow}}(\hat{\mathbf{x}}_k, \mathbf{z}_{\text{src}}, \Delta t)$ Eq. (8)
  $\mathbf{F}_k \leftarrow \text{DiagJac}(\hat{\mathbf{x}}_k, \epsilon)$ Eq. (12)
  $\mathbf{P}_{k-1}^{-} \leftarrow \mathbf{F}_k \mathbf{P}_k \mathbf{F}_k^T + \mathbf{Q}_k$
  // Update Step
  $\mathbf{z}_{k-1}^{\text{obs}} \leftarrow \mathcal{M}_{\text{obs}}(\mathbf{x}_k^{\text{obs}}, \hat{\mathbf{x}}_k, c^{\text{src}}, c^{\text{tar}}, t_k)$ // varies with the observer implementation
  $\mathbf{K}_{k-1} \leftarrow \mathbf{P}_{k-1}^{-}(\mathbf{P}_{k-1}^{-} + \mathbf{R}_{k-1})^{-1}$
  $\hat{\mathbf{x}}_{k-1} \leftarrow \hat{\mathbf{x}}_{k-1}^{-} + \mathbf{K}_{k-1}(\mathbf{z}_{k-1}^{\text{obs}} - \hat{\mathbf{x}}_{k-1}^{-})$
  $\mathbf{P}_{k-1} \leftarrow (\mathbf{I} - \mathbf{K}_{k-1})\mathbf{P}_{k-1}^{-}$
  $\mathbf{x}_{k-1}^{\text{obs}} \leftarrow \text{Align}(\mathbf{x}_k^{\text{obs}}, \hat{\mathbf{x}}_{k-1})$ Eq. (14)
end for
return $\text{VAE}_{\text{dec}}(\hat{\mathbf{x}}_0)$

Dynamic Noise Schedule. Following prior observations that velocity estimation is less reliable at high noise levels [Kulikov *et al.*, 2025; Xie *et al.*, 2025], we use time-varying diagonal covariances:

$$\sigma_Q(t_k) = \lambda_1 t_k, \qquad \sigma_R(t_k) = \lambda_2(1 - t_k). \tag{13}$$

This schedule assigns larger process uncertainty in early steps and gradually reduces the reliance on external observations in later steps, mitigating accumulated observation errors.

Trajectory Alignment. Direct fusion of the predicted state $\hat{\mathbf{x}}_k$ and the external observation can lead to ghosting artifacts when their underlying velocity fields conflict. Inspired by DNAEdit, we align the internal state of observer $\mathbf{x}_t^{obs}$ towards the posterior state $\hat{\mathbf{x}}_t$ to ensure geometric consistency before each prediction loop:

$$\mathbf{x}_{t-1}^{\text{obs}} = \mathbf{x}_t^{\text{obs}} + \Delta t \cdot (\eta \cdot \mathbf{v}_t^{\text{tar}} + (1 - \eta) \cdot \frac{\mathbf{x}_t^{\text{obs}} - \hat{\mathbf{x}}_t}{t_k}), \tag{14}$$

where $\mathbf{v}_t^{tar}$ denotes the target velocity estimated by the external observer at time t, and $\eta \in [0, 1]$ is a balancing factor. This constraint forces the observation path $\mathbf{x}^{obs}$ to geometrically converge towards the predicted trajectory $\hat{\mathbf{x}}$, mitigating directional conflicts.

EKFEdit-RF. We first implement EKFEdit with the RF-Solver [Wang *et al.*, 2024] as the observer. During the RF-Solver inversion stage, we cache the source velocity $\mathbf{v}_t^{\text{src}}$ along the inversion trajectory, and then compute the target velocity $\mathbf{v}_t^{\text{tar}}$ from the aligned observer state during the EKFEdit loop. All velocity fields are estimated by the Second-order Solver within RF-Solver, and the attention injection mechanism is retained in the update of $\mathbf{z}^{edit}$ to enhance structure preservation. The observation is constructed as:

$$\mathbf{z}_{k-1}^{obs} = \hat{\mathbf{x}}_k + \Delta t \cdot (\mathbf{v}_t^{\text{tar}} - \mathbf{v}_t^{\text{src}}). \tag{15}$$

EKFEdit-DNA. We further adapt the framework to DNAEdit [Xie *et al.*, 2025]. In this variant, the observer first performs Direct Noise Alignment to obtain a noise-aligned trajectory and cache the DNA correction terms. During the EKFEdit loop, the target velocity $\mathbf{v}_t^{\text{tar}}$ is computed on the DNA-specific residual-corrected latent, while the cached DNA correction serves as the source-side term $\mathbf{v}_t^{\text{src}}$. EKFEdit-DNA therefore inherits the same observation construction Eq. (15) and alignment Eq. (14), but obtains its observation from DNAEdit-specific noise alignment rather than RF-Solver inversion. Please refer to the project page for details.

4 Experiments

4.1 Implementation and Compared Methods

Implementation Details. We employ the pretrained SD3.5-medium [Stability AI, 2024] as the base T2I model. For our methods, we set the sampling steps to $N = 50$ and start step $T_s = 17$ with hyperparameters $\eta = 0.8$, $\lambda_1 = 0.1$, and $\lambda_2 = 1.0$. For all compared methods, we adopt the default settings in their official implementations. Image resolution is fixed at 512×512. Classifier-Free Guidance (CFG) [Ho and Salimans, 2022] is important for balancing editing effects, where higher values enhance image-text alignment but often compromise background preservation. To ensure a fair comparison, we group experiments by different CFG settings to fairly evaluate background preservation under similar text adherence levels.

Compared Methods. We benchmark against current advanced training-free flow-based methods, covering inversion-based RF-Inversion [Rout *et al.*, 2024a], RF-Solver [Wang *et al.*, 2024] and DNAEdit [Xie *et al.*, 2025], as well as inversion-free FlowEdit [Kulikov *et al.*, 2025] and FlowAlign [Kim *et al.*, 2025].

4.2 Qualitative Comparison

Figure 4 shows the visual comparison between our method and competitive methods. In general, our methods achieve better structural and background preservation while accurately executing the target edits. In contrast, FlowEdit shows clear structural deviations, such as altering the object pose in the *bear* case. RF-Solver fails to effectively preserve background details, while DNAEdit exhibits editing failures, resulting in either failure to generate the expected content (*white dog, cat* $\rightarrow$ *husky, lion*) or severe grid-like artifacts (*cat, crown* $\rightarrow$ *rabbit, hat*). The failure case of DNAEdit is partially inherited by EKFEdit-DNA, reflecting that EKFEdit cannot fully reject arbitrary observation errors. Nevertheless, EKFEdit is not a pure observer-following scheme: the Kalman gain downweights observations in stable regions with low predicted uncertainty, but may still preserve observer influence when errors overlap with regions requiring substantial semantic changes. Similarly, in the *photo* $\rightarrow$ *watercolor* case, the orientation of the woman's face remains consistent with the corresponding observer methods, empirically validating that observations provide significant semantic guidance within the EKFEdit framework.

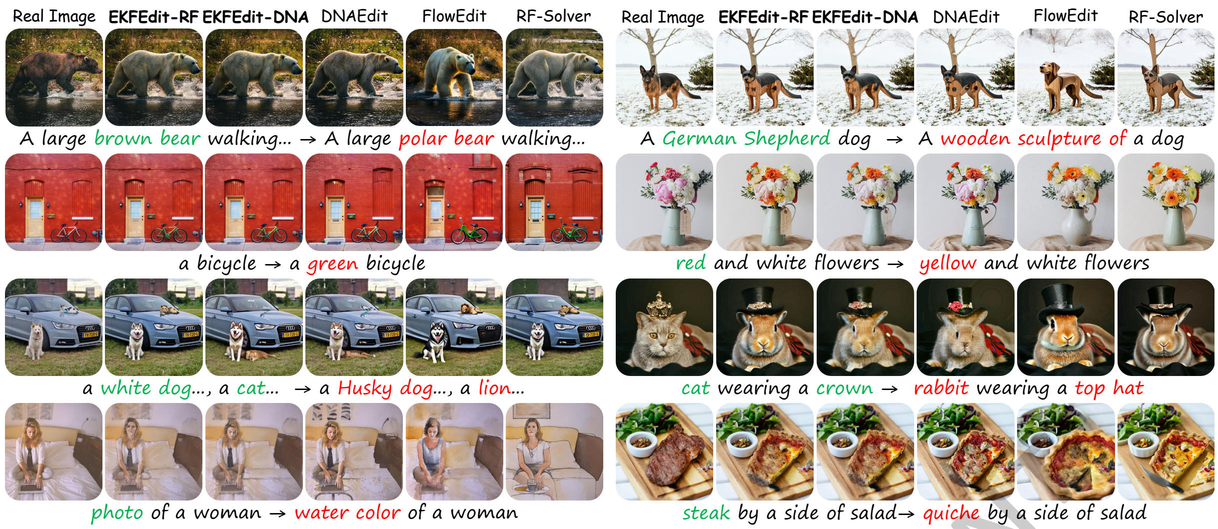

Figure 4: Qualitative comparison of competitive training-free flow based image editing methods.

Method	**CFG**	**Structure**	**Background Preservation**				**Semantic**	**Rank**
		Distance ↓	PSNR ↑	LPIPS ↓	MSE ↓	SSIM ↑	CLIP ↑	Avg. ↓
RF-Inversion	3.5	36.775	18.094	0.291	0.0172	0.601	29.888	6.667
RF-Solver	5.5	34.646	18.957	0.232	0.0137	0.718	30.222	4.000
FlowEdit	13.5	32.275	18.037	0.214	0.0170	0.718	30.422	4.333
DNAEdit	7.5	31.928	18.878	0.248	0.0139	0.719	30.177	4.333
FlowAlign	13.5	34.702	18.780	0.233	0.0154	0.736	28.961	5.167
EKFEdit-RF	12.5	16.828	19.770	0.183	0.0115	0.750	30.321	1.167
EKFEdit-DNA	10.5	19.734	19.058	0.193	0.0134	0.740	30.238	2.167
RF-Inversion	2.5	35.501	18.494	0.287	0.0158	0.607	29.359	6.667
RF-Solver	3.5	17.070	20.929	0.196	0.00874	0.754	29.636	4.667
FlowEdit	7.5	11.853	21.162	0.151	0.00864	0.775	29.951	3.167
DNAEdit	3.5	15.502	21.704	0.196	0.00735	0.777	29.229	3.833
FlowAlign	10.5	26.471	20.078	0.201	0.0114	0.744	29.207	6.167
EKFEdit-RF	7.5	9.985	21.894	0.145	0.00718	0.785	29.812	2.000
EKFEdit-DNA	5.5	9.370	22.142	0.142	0.00682	0.786	29.774	1.333

Table 1: Quantitative comparison on FlowEdit dataset. Red ( ), blue ( ), yellow ( ) represent the top 3 performers.

4.3 Quantitative Comparison

Dataset. We conduct quantitative evaluations on the FlowEdit dataset [Kulikov *et al.*, 2025], which consists of 76 diverse real images. Each image is paired with a source prompt and several handcrafted target prompts, forming 280 text-image pairs in total. Additionally, we utilize PIE-Bench [Ju *et al.*, 2023], which contains 700 synthetic and natural images with short paired prompts.

Metrics. To quantitatively evaluate editing quality, we adopt metrics consistent with existing benchmarks: MSE, PSNR [Gonzalez, 2009], SSIM [Wang *et al.*, 2004] and LPIPS [Zhang *et al.*, 2018] for background preservation; Structure Distance [Ju *et al.*, 2023] for structural integrity; and CLIP Similarity Score (CLIP) [Wu *et al.*, 2021] for semantic adherence between edited images and target text.

Results. Table 1 reports the quantitative performance of our method and baselines on the FlowEdit dataset, as well as the average rank of each method across all metrics. All methods within each CFG group yield comparable CLIP similarity scores[1] for fair comparison. Despite achieving the highest CLIP scores, FlowEdit consistently obtains the lower PSNR, which stems from the absence of trajectory constraints and consequent semantic drift in non-edited regions. Other methods fail to achieve top-2 performance across all metrics. In contrast, our method achieves top-3 CLIP scores while outperforming all baselines on all four background preservation metrics and Structure Distance, fully demonstrating the robust performance of our method. Quantitative performance on PIE-Bench is presented in Table 2. In this table, CLIP_W refers to the CLIP similarity between the whole image and the target text, and CLIP_E corresponds to the CLIP similarity calculated on the masked edited area.

[1] FlowAlign failed to achieve a higher CLIP score even when tuning the CFG scale across a wide range of values.

Method	CFG	Structure	Background Preservation				Semantic		Rank
		Distance ↓	PSNR ↑	LPIPS ↓	MSE ↓	SSIM ↑	CLIP_W ↑	CLIP_E ↑	Avg. ↓
RF-Inversion	3.5	38.973	22.318	0.169	0.00844	0.751	26.692	23.455	6.857
RF-Solver	5.5	25.093	23.510	0.107	0.00633	0.842	27.350	24.188	4.429
FlowEdit	13.5	21.985	23.470	0.0868	0.00668	0.856	27.672	24.212	3.571
DNAEdit	7.5	24.724	23.676	0.106	0.00628	0.850	27.142	23.906	4.571
FlowAlign	13.5	35.352	24.176	0.0614	0.00531	0.869	26.698	23.153	4.413
EKFEdit-RF	12.5	13.802	26.164	0.0614	0.00406	0.885	27.273	23.909	1.857
EKFEdit-DNA	10.5	17.159	24.585	0.0707	0.00513	0.875	27.294	24.039	2.423
RF-Inversion	1.5	52.318	22.318	0.197	0.00844	0.697	25.883	22.641	7.000
RF-Solver	3.5	17.058	25.821	0.0786	0.00388	0.874	26.655	23.469	4.000
FlowEdit	7.5	8.538	28.169	0.0376	0.00244	0.913	26.423	23.068	2.714
DNAEdit	3.5	13.098	27.185	0.0654	0.00295	0.900	26.010	22.863	4.429
FlowAlign	10.5	30.681	25.423	0.0449	0.00412	0.885	26.28	22.726	5.143
EKFEdit-RF	7.5	7.501	28.541	0.0358	0.00217	0.916	26.361	23.059	2.286
EKFEdit-DNA	5.5	6.790	28.896	0.0321	0.00201	0.920	25.953	22.650	2.429

Table 2: Quantitative comparison on PIE-Bench. Red ( ), blue ( ), yellow ( ) represent the top 3 performers.

Fusion Strategy	CLIP ↑	PSNR ↑	LPIPS ↓	Struct. Dist. ↓
Prediction Only	**29.95**	<u>21.16</u>	<u>0.15</u>	<u>11.85</u>
Observation Only	29.64	20.93	0.20	17.07
Static Fusion	29.71	21.03	0.19	14.35
EKFEdit (Ours)	<u>29.81</u>	**21.89**	**0.14**	**9.98**

Table 3: Quantitative comparison for ablation over different fusion approaches.

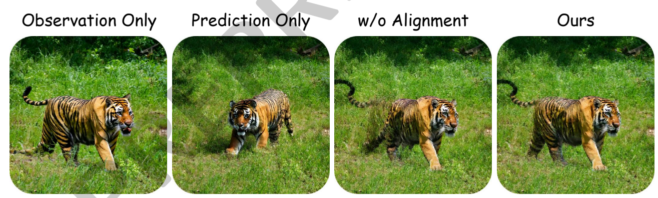

Figure 5: Visual comparison on the impact of trajectory alignment. Without alignment, conflicting guidance causes severe artifacts.

Component Settings			Metrics			
Jacobian Approx.	Noise Sched.	Trajectory Alignment	CLIP ↑	PSNR ↑	LPIPS ↓	Struct. Dist. ↓
✓	×	×	**30.09**	20.39	0.19	15.98
✓	×	✓	29.85	21.51	0.15	11.04
✓	✓	×	29.75	<u>21.72</u>	<u>0.15</u>	<u>10.57</u>
Id	✓	✓	29.87	21.39	0.15	11.39
✓	Rev	✓	<u>29.88</u>	21.50	0.17	11.51
✓	✓	✓	29.81	**21.89**	**0.14**	**9.98**

Table 4: Ablation of key components. We evaluate Jacobian approximation (vs. Identity Matrix [Id]), noise scheduling (vs. Reversed scheduling [Rev]), and alignment. × indicates component removal.

4.4 Ablation Study

Effectiveness of Kalman Fusion. Table 3 validates the superiority of EKF fusion. Unlike Static Fusion ($\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + 0.5(\mathbf{z}_k^{obs} - \hat{\mathbf{x}}_k^-)$) which yields suboptimal results using fixed weights, our method adaptively fuses prediction and observation with Kalman gain, achieving superior performance in semantic alignment and structural preservation.

Effectiveness of Key Components. Table 4 validates the efficacy of key components in EKFEdit: Jacobian Diagonal Approximation, Dynamic Noise Schedule and Trajectory Alignment. First, regarding linearization, replacing our Jacobian Diagonal Approximation with an Identity Matrix leads to performance degradation, confirming the necessity of capturing local non-linearities. Second, improvements across metrics confirm the effectiveness of our Dynamic Noise Schedule. Notably, the Reversed Schedule, which increases the process noise and reduces observation noise sequentially, results in structural degradation, indicating that the estimator should prioritize prior dynamics over observations in later stages. Finally, Trajectory Alignment effectively mitigates ghosting artifacts caused by conflicting velocity fields between the prior and observer, as visualized in Figure 5.

5 Conclusion

In this paper, we introduce EKFEdit, a unified framework that reformulates training-free flow-based image editing as a stochastic state estimation problem. By dynamically fusing inversion-free priors with external observations via the Extended Kalman Filter, our approach effectively balances the trade-off between text adherence and structural preservation. We further enable efficient high-dimensional inference through a gradient-free Jacobian diagonal approximation. To demonstrate the framework's versatility, we present two concrete instantiations, EKFEdit-RF and EKFEdit-DNA. Extensive experiments on standard benchmarks confirm that both variants consistently improve the balance between semantic alignment and structural preservation.

A current limitation of EKFEdit is its additional inference overhead from joint prediction-observation execution, which we aim to reduce via parallelization in future work.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62495084 and 62473307).

References

[Albergo and Vanden-Eijnden, 2022] Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

[Avrahami *et al.*, 2025] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7877–7888, 2025.

[Berner *et al.*, 2022] Julius Berner, Lorenz Richter, and Karen Ullrich. An optimal control perspective on diffusion-based generative modeling. *arXiv preprint arXiv:2211.01364*, 2022.

[Chi *et al.*, 2025] Haozhe Chi, Zhicheng Sun, Yang Jin, Yi Ma, Jing Wang, and Yadong Mu. Enhancing consistency of flow-based image editing through kalman control. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[Deng *et al.*, 2024] 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.

[Esser *et al.*, 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.

[Gonzalez, 2009] Rafael C Gonzalez. *Digital image processing*. Pearson education india, 2009.

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

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

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

[Kamb and Ganguli, 2025] Mason Kamb and Surya Ganguli. An analytic theory of creativity in convolutional diffusion models. In *International Conference on Machine Learning*, pages 28795–28831. PMLR, 2025.

[Kim *et al.*, 2025] Jeongsol Kim, Yeobin Hong, Jonghyun Park, and Jong Chul Ye. Flowalign: Trajectory-regularized, inversion-free flow-based image editing. *arXiv preprint arXiv:2505.23145*, 2025.

[Koo *et al.*, 2024] Juil Koo, Chanho Park, and Minhyuk Sung. Posterior distillation sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13352–13361, 2024.

[Kulikov *et al.*, 2025] 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*, pages 19721–19730, 2025.

[Labs, 2024] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. Accessed: 2025-12-22.

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

[Liu *et al.*, 2022] 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.

[Lukoianov *et al.*, 2025] Artem Lukoianov, Chenyang Yuan, Justin Solomon, and Vincent Sitzmann. Locality in image diffusion models emerges from data statistics. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[Mao *et al.*, 2025] Jianda Mao, Kaibo Wang, Yang Xiang, and Kani Chen. Tweezeedit: Consistent and efficient image editing with path regularization. *arXiv preprint arXiv:2508.10498*, 2025.

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

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

[Rout *et al.*, 2024b] Litu Rout, Yujia Chen, Nataniel Ruiz, Abhishek Kumar, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Rb-modulation: Training-free personalization of diffusion models using stochastic optimal control. *arXiv preprint arXiv:2405.17401*, 2024.

[Stability AI, 2024] Stability AI. Stable diffusion 3.5 medium. https://huggingface.co/stabilityai/stable-diffusion-3.5-medium, 2024. Accessed: 2025-12-22.

[Wang *et al.*, 2004] 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.

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

[Wu *et al.*, 2021] Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. Godiva: Generating open-domain videos from natural descriptions. *arXiv preprint arXiv:2104.14806*, 2021.

[Xie *et al.*, 2025] Chenxi Xie, Minghan Li, Shuai Li, Yuhui Wu, Qiaosi Yi, and Lei Zhang. Dnaedit: Direct noise alignment for text-guided rectified flow editing. *arXiv preprint arXiv:2506.01430*, 2025.

[Xu *et al.*, 2024] Yu Xu, Fan Tang, Juan Cao, Yuxin Zhang, Xiaoyu Kong, Jintao Li, Oliver Deussen, and Tong-Yee Lee. Headrouter: A training-free image editing framework for mm-dits by adaptively routing attention heads. *arXiv preprint arXiv:2411.15034*, 2024.

[Xu *et al.*, 2025] 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28479–28489, 2025.

[Yan *et al.*, 2025] Zexuan Yan, Yue Ma, Chang Zou, Wenteng Chen, Qifeng Chen, and Linfeng Zhang. Eedit: Rethinking the spatial and temporal redundancy for efficient image editing. *arXiv preprint arXiv:2503.10270*, 2025.

[Yoon *et al.*, 2025] 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.

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

[Zhou *et al.*, 2025] Zijun Zhou, Yingying Deng, Xiangyu He, Weiming Dong, and Fan Tang. Multi-turn consistent image editing. *arXiv preprint arXiv:2505.04320*, 2025.

[Zhu *et al.*, 2025a] Kaizhen Zhu, Mokai Pan, Yuexin Ma, Yanwei Fu, Jingyi Yu, Jingya Wang, and Ye Shi. Unidb: A unified diffusion bridge framework via stochastic optimal control. *arXiv preprint arXiv:2502.05749*, 2025.

[Zhu *et al.*, 2025b] Tianrui Zhu, Shiyi Zhang, Jiawei Shao, and Yansong Tang. Kv-edit: Training-free image editing for precise background preservation. *arXiv preprint arXiv:2502.17363*, 2025.