h-Flow: Flexible Flow-based Image Editing via Doob's *h*-Transform

Zehui Guo[1], Zhen Wang[2], Junwei Shu[1], Yang Li[1,*], Changbo Wang[1,*], and Long Chen[2]

[1] East China Normal University, Shanghai, China
51275901098@stu.ecnu.edu.cn, 51265901091@stu.ecnu.edu.cn,
yli@cs.ecnu.edu.cn, cbwang@cs.ecnu.edu.cn
[2] The Hong Kong University of Science and Technology, Hong Kong SAR, China
zhenwang@ust.hk, longchen@ust.hk

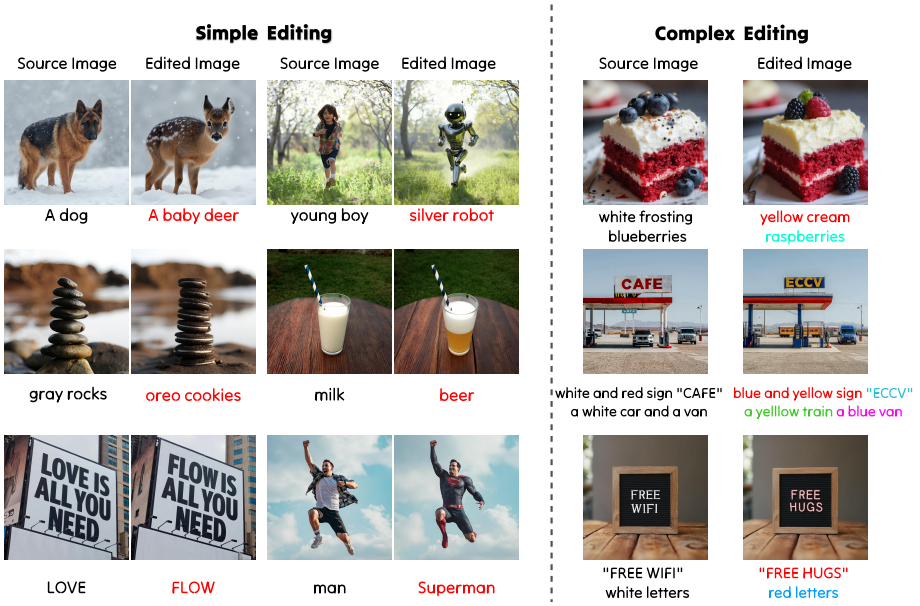


Fig. 1: *h*-Flow enables faithful text-based editing across diverse scenarios.

Abstract. Editing images with pre-trained text-to-image flow models typically requires carefully balancing target alignment with the desired prompt and source consistency with the original image. Existing approaches either rely on inversion-based pipelines or heuristic source-to-target trajectory constructions, which often depend on architecture-specific designs or are sensitive to hyperparameters. In this paper, we propose **h-flow**, a training-free and theoretically grounded flow-based editing framework. Inspired by Doob's *h*-Transform, we reformulate image

*Corresponding author.

editing as conditional generation under multiple terminal events corresponding to source consistency and target alignment. We first extend the classical h-Transform from SDE-based models to the deterministic RF framework by constructing an equivalent SDE with identical marginals. Within this formulation, we design dedicated h-functions for source consistency and target alignment, yielding closed-form reconstruction guidance and velocity-based semantic editing signals. We further introduce a velocity orthogonal decomposition to decouple reconstruction and editing directions, enabling a controllable trade-off between the two objectives. Extensive experiments demonstrate that h-flow achieves effective, robust, and flexible editing across diverse scenarios.

Keywords: Text-based Image Editing · Rectified Flow Model · Doob's h-Transform

1 Introduction

Rectified flow (RF) models [20,21] have emerged as a powerful generative paradigm for various visual generation tasks. Benefiting from large-scale pretraining, text-to-image RF models [8, 19] have been widely adopted for text-based image editing [18,28], which aims to modify the visual contents of a source image according to a target prompt. The central challenge of this task lies in balancing the two common goals: *target alignment* and *source consistency*. Specifically, the synthesized image should faithfully adhere to the target prompt while preserving the editing-irrelevant regions unchanged from the source image.

To achieve this trade-off, early flow-based editing approaches predominantly adopt a two-stage, inversion-based paradigm [17, 28, 32]. In the forward process, the source image is first inverted into a latent variable in the noise distribution, in the backward process, the edited image is generated from this latent under the guidance of the target prompt. Representative works mainly focus on improving inversion accuracy [17, 28] for better source reconstruction, and further inject source conditions (*e.g.*, attention maps [32]) during the forward process to enhance source consistency. However, these strategies are largely heuristic and often rely on architecture-specific analysis, limiting their scalability and general applicability. More recent studies have pioneered inversion-free paradigms [12, 16, 18], which directly construct transformation trajectories from the source image to the target image, enforcing strong structural consistency. Despite their empirical effectiveness, they remain sensitive to specific hyperparameter settings, such as random seeds and the number of flow steps, which constrain their robustness and generalization in practical scenarios.

Different from prior methods that are primarily motivated by empirical intuition without a unified theoretical grounding, we propose ***h*-flow**, a training-free and flexible flow-based editing approach built upon the intrinsic probabilistic structure of flow models. Specifically, inspired by Doob's h-Transform [6, 27, 29], we reformulate image editing as a conditional generation problem under multiple terminal events, corresponding to the two fundamental objectives of editing:

source consistency with the input image and *target alignment* with the target prompt. Since the classical h-Transform is defined for SDE-based stochastic processes, whereas RF models operate under a deterministic ODE formulation, we first construct an equivalent SDE that shares the same marginal distributions as the RF dynamics. This enables us to rigorously extend the h-Transform from diffusion/SDE-based models (*e.g.*, h-Edit [25]) to the ODE-based RF framework, providing a principled theoretical bridge. Within this unified formulation, we explicitly design dedicated h-functions to encode the two core objectives of image editing. Specifically, the h-function associated with source consistency imposes a terminal constraint on the source image, which naturally induces an exact, closed-form reconstruction guidance along the flow trajectory. In contrast, the h-function for target alignment models the desired semantic transition toward the target prompt. Under this design, the editing signal is characterized by the deviation of the editing velocity from the source velocity, thereby explicitly capturing the semantic shift between the source and target prompts in the velocity field.

To explicitly balance the two objectives, we further propose a velocity orthogonal decomposition strategy, which projects the editing velocity onto complementary subspaces. This design ensures that reconstruction and editing directions act independently, fully decoupling source preservation and target modification at the velocity level, and thus inherently achieving a controllable trade-off between source consistency and target alignment. Moreover, since h-flow modifies the backward process while remaining agnostic to the specific forward construction, our method can be seamlessly combined with different forward processes in a plug-and-play manner, further enhancing editing flexibility and performance. Extensive experiments demonstrate the effectiveness, robustness, and generalization capability of ***h*-flow** across diverse editing scenarios.

In summary, our contributions can be summarized as follows:

- We propose ***h*-flow**, a training-free and theoretically grounded flow-based image editing framework that reformulates editing via Doob's h-Transform.
- We design dedicated h-functions to explicitly encode source consistency and target alignment. Together with a velocity orthogonal decomposition strategy, our method inherently decouples reconstruction and editing to achieve a controllable trade-off, and can be flexibly combined with different forward processes in a plug-and-play manner.
- Extensive experiments and ablations demonstrate the effectiveness and robustness of ***h*-flow** under diverse editing scenarios.

2 Related work

Inversion-based Editing This paradigm first inverts the source image into a noise latent through a forward inversion process, and then re-generates the edited result under the target prompt in the backward process. To mitigate error accumulation during inversion and subsequent generation, a line of works [2,

24, 28, 32] focuses on improving inversion accuracy. For example, RF-Solver-Edit [32] designs higher-order ODE solvers to reduce discretization errors, while RF-Inversion [28] formulates inversion as a dynamic optimal control problem to refine the trajectory. However, even accurate inversion cannot fully guarantee source fidelity after editing. Therefore, another line of works [9, 17, 31, 32, 36] enhances source consistency by injecting additional source conditions during the backward generation process. Despite improved source preservation, these methods typically rely on heuristic designs and architecture-specific modifications, limiting their generality and flexibility.

Inversion-Free Editing. A growing body of work seeks to bypass explicit inversion. SDEdit [23] directly perturbs the source image with random noise to replace the inversion step. However, the noise strength introduces a hard trade-off between editability and source preservation. FlowEdit [18] pioneers a dual-trajectory paradigm that constructs a direct source-to-target ODE by coupling two noise-to-image flows, thereby avoiding inversion. Nevertheless, per-step random noise sampling can cause trajectory instability. FlowAlign [16] further improves stability within this framework by formulating source preservation as an optimal control problem and optimizing the trajectory accordingly. Despite these advances, such source-to-target constructions remain largely heuristic, lacking a unified theoretical foundation, and are sensitive to hyperparameter choices such as random seeds and flow steps.

Doob's h-Transform in Image Editing. Following [15, 22, 33, 38], we define bridges as stochastic processes conditioned on a predefined terminal event, derived from a base Markov process via Doob's h-transform [6, 27, 29]. Specifically, h-Edit [25] is the first to introduce the h-Transform into image editing. However, it is developed within the diffusion SDE framework and approximates the source constraint through a text prompt proxy, without enforcing an explicit hard constraint on the source image. In contrast, we extend the h-Transform to the deterministic flow matching framework, and derive an exact closed-form guidance for the source image constraint, resulting in a more principled, flexible, and controllable editing formulation.

3 Preliminaries

3.1 Rectified Flow

Rectified Flow (RF) [21] defines a forward process as a linear interpolation between a data sample $x_0 \sim p_0$ and Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$:

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

The conditional transition kernel is an isotropic Gaussian:

$$p(x_t \mid x_0) = \mathcal{N}\big(x_t;\, (1-t)x_0,\, t^2 I\big). \tag{2}$$

A velocity network $v_\theta(x_t, t)$ is trained via the flow matching objective [20] to approximate $\mathbb{E}[\epsilon - x_0 \mid x_t]$. Samples are generated by solving the ODE $dx_t/dt = v_\theta(x_t, t)$ backward from t=1 to t=0.

Applying the Tweedie formula to the RF kernel (2) gives the posterior mean $\mathbb{E}[x_0 \mid x_t]$, which we approximate using the learned velocity:

$$\hat{x}_0 \triangleq x_t - t\, v_\theta(x_t, t) \approx \mathbb{E}[x_0 \mid x_t]. \tag{3}$$

The approximation is exact when v_θ matches the true conditional expectation $\mathbb{E}[\epsilon - x_0 \mid x_t]$. From the Gaussian form of Eq. (2) and Eq. (3), one can derive the score–velocity identity:

$$\nabla_{x_t} \log p_t(x_t) = \frac{(1-t)\,\hat{x}_0 - x_t}{t^2}, \tag{4}$$

which will be used repeatedly in our derivations.

3.2 Inversion-based Image Editing

Given a source image I^{src}, a source prompt c^{src}, and a target prompt c^{edit}, text-based image editing aims to produce an output I^{edit} that (i) reflects the semantic content described by c^{edit} (*target alignment*), while (ii) preserving the structure and unedited regions of I^{src} (*source consistency*). Both images are encoded as $x_0^{src} = \mathcal{E}(I^{src})$ and $x_0^{edit} = \mathcal{E}(I^{edit})$ via a pretrained VAE encoder $\mathcal{E}$, and all operations are performed in this latent space.

Inversion-based editing typically consists of two stages. The forward process maps the source image to an intermediate latent state x_{t_N} at some time $t_N \in (0, 1]$, from which the backward generation begins. This can be achieved either through iterative inversion, $x_{t_N} = x_0^{src} + \int_0^{x_{t_N}} v_\theta(x_t^{src}, t, c^{src})\, dt$ or via on-step noising $x_{t_N} = (1-t_N)\, x_0^{src} + t_N\, \epsilon$. The backward process then generates the edited latent from this intermediate state under the target prompt by integrating the conditional flow in reverse time:

$$x_0^{edit} = x_{t_N} - \int_0^{t_N} v_\theta(x_t, t, c^{edit})\, dt \tag{5}$$

3.3 Doob's h-Transform

Doob's h-Transform [6,27,29] is a classical tool for conditioning a Markov process on a future event. Consider an Itô SDE:

$$dx = f(x_t, t)\, dt + g(t)\, dw. \tag{6}$$

To condition this process on a terminal event A (e.g., $x_0 \in A$), the h-Transform introduces an additive drift correction proportional to the log-gradient of the harmonic function $h(x_t, t) = p_t(A \mid x_t)$:

$$dx = \left[f(x_t, t) + g^2(t)\, \nabla_{x_t} \log p_t(A \mid x_t)\right] dt + g(t)\, dw. \tag{7}$$

The corresponding probability-flow ODE (PF-ODE) [1], which shares the same marginal distributions, is:

$$\frac{dx}{dt} = \underbrace{f(x_t, t) - \frac{g^2(t)}{2}\, \nabla_{x_t} \log p_t(x_t)}_{\widetilde{f}\ \text{(unconditioned PF-ODE drift)}} + \frac{g^2(t)}{2}\, \nabla_{x_t} \log p_t(A \mid x_t). \tag{8}$$

The first part $\widetilde{f}$ is the standard (unconditioned) PF-ODE drift; the second part is the bridge guidance with a *positive* sign. When A corresponds to a class label, Eq. (8) recovers classifier guidance [4]; when combined with a null-condition baseline, it reduces to classifier-free guidance [10]. The h-transform thus provides a unified probabilistic foundation for guided generation.

The $g > 0$ requirement. The guidance terms in Eqs. (7)–(8) are scaled by $g^2(t)$ or $g^2(t)/2$. When $g(t) > 0$, the infinitesimal transition kernel has non-degenerate variance, and the h-transform reweights it smoothly. When $g \equiv 0$—as in Rectified Flow—the transition kernel collapses to a Dirac delta, and the h-transform is undefined. Resolving this incompatibility is the central technical challenge addressed in Sec. 4. The h-transform is presented above in forward time for clarity. When the process runs in reverse time ($t: 1 \to 0$), the terminal event $\{x_0 \in A\}$ lies at the process's endpoint; the h-transform applied to the reverse-time PF-ODE carries a negative sign, because the guidance must deflect the trajectory against its natural flow direction to reach the target at $t=0$.

4 Method

Problem formulation: editing as conditional generation. Given the source latent $x_0^{src} = \mathcal{E}(z^{src})$ and a target prompt c^{edit}, the goal is to generate an edited latent x_0^{edit} that satisfies both constraints simultaneously. We formulate this as sampling from:

$$x_0^{edit} \sim p(\cdot \mid x_0^{src}, c^{edit}), \tag{9}$$

where the conditional distribution encodes proximity to x_0^{src} in unedited regions and alignment with c^{edit} in edited regions. In the RF framework, this amounts to modifying the backward sampling ODE $dx_t/dt = v_\theta(x_t, t)$ so that the output remains close to x_0^{src} except where c^{edit} specifies changes. The two objectives are inherently in tension: the velocity direction needed to reconstruct x_0^{src} and the direction needed to align with c^{edit} are generally not parallel and may conflict. How to decouple and effectively balance them is the key of this problem.

4.1 Rectified Flow Bridge Construction

A rigorous mathematical tool for such conditioning is Doob's h-transform [6]. For a reverse-time generative SDE $dx_t = f\,dt + g\,d\bar{w}_t$ with marginals $\{p_t\}$, conditioning on a terminal event $\{x_0 \in A\}$ modifies the probability-flow ODE to:

$$\frac{dx_t}{dt} = \widetilde{f}(x_t, t) - \frac{g^2}{2}\,\nabla_{x_t} \log h(x_t, t), \tag{10}$$

where $\widetilde{f} = f - \frac{g^2}{2}\nabla \log p_t$ is the unconditioned PF-ODE drift, and $h(x_t, t)$ is a product of harmonic functions encoding the constraints [7] (detailed in the Supplementary Material). **The deterministic-ODE obstacle.** Rectified Flow (RF) generates samples via a purely deterministic ODE where the diffusion

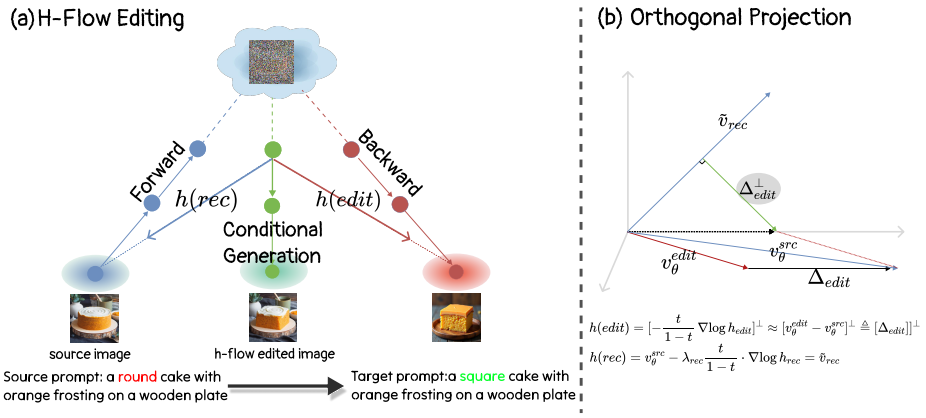


Fig. 2: Overview of h-Flow. (a) Dual-Guidance via h-Transforms. We formulate image editing as a conditional generation process governed by two terminal constraints. The generative trajectory is jointly guided by a reconstruction function h_{rec} (ensuring source structural fidelity) and an editing function h_{edit} (enforcing target semantic alignment). **(b) Orthogonal Velocity Projection.** To resolve the inherent tension between reconstruction and editing, we project the editing direction Δ_{edit} onto the orthogonal complement of the reconstruction velocity $\bar{v}_{rec}$. The isolated orthogonal component $\Delta_{edit}^{\perp}$ exclusively alters the semantic content without interfering with the source structure, enabling perfectly decoupled control.

coefficient $g(t) \equiv 0$. Consequently, the h-transform guidance term $\frac{g^2}{2}\nabla \log h$ vanishes identically. This structural incompatibility means the continuous-time h-transform cannot be directly applied to RF models.

Equivalent SDE construction. To circumvent this, we exploit the property that infinitely many SDEs can share the exact same marginal distributions $\{p_t\}$ as a given ODE—they differ only in stochastic fluctuations that cancel out marginally. We construct an equivalent Itô SDE with diffusion coefficient $g_{eq}(t) = \sqrt{2t/(1-t)}$, chosen so that $g_{eq}(0)=0$ (no noise at the clean-data endpoint) and $g_{eq}(t)>0$ for $t>0$. We detail in Supplementary how this construction yields an equivalent process that perfectly matches the RF marginals. Substituting g_{eq} into the bridge PF-ODE (10) and noting that $g_{eq}^2/2 = t/(1-t)$ and $\widetilde{f}_{eq} = v_\theta$ (see the Supplementary Material), we obtain

$$\frac{dx_t}{dt} = v_\theta(x_t, t) - \frac{t}{1-t}\nabla_{x_t} \log h(x_t, t), \tag{11}$$

Recall that the harmonic function $h(x_t, t)$ encodes the probability of reaching a specified terminal event at $t=0$. In our image editing formulation, the desired terminal state is jointly governed by two simultaneous conditions: maintaining structural fidelity to the source image (src) and achieving semantic alignment with the target prompt ($edit$). By treating these two terminal constraints as conditionally independent events given the intermediate state x_t, the overall

harmonic function naturally factorizes into a product of two experts:

$$h = h_{rec} \cdot h_{edit} \quad \Longrightarrow \quad \nabla \log h = \nabla \log h_{rec} + \nabla \log h_{edit}. \tag{12}$$

In the log-gradient space, this factorization elegantly translates into an additive superposition of a reconstruction force and an editing force. Because our equivalent SDE strictly shares the RF marginals, this dual-guidance mechanism transfers exactly to the deterministic RF framework without altering the pre-trained model's unconditional prior. This formulation provides a rigorous mathematical bridge from abstract terminal goals to concrete ODE modifications, leading directly to the specific design of each guidance term.

4.2 *h*-Function Design

The RF bridge equation (11) requires specifying the harmonic function $h = h_{rec} \cdot h_{edit}$. We now derive closed-form expressions for each term, converting the abstract bridge equation into a computable editing ODE.

Source preservation (h_{rec}) The source constraint corresponds to the terminal event $\{x_0 = x_0^{src}\}$. Setting $h_{rec}(x_0, 0) = \delta(x_0 - x_0^{src})$ and applying the harmonic property [6] gives $h_{rec}(x_t, t) = p(x_0^{src} \mid x_t)$. Expanding via Bayes' rule and evaluating the gradients under RF's Gaussian kernel (see the Supplementary Material), we obtain:

$$\frac{t}{1-t} \cdot \nabla \log h_{rec} = \frac{x_0^{src} - \hat{x}_0}{t} = v_\theta^{src} - v_{rec}, \tag{13}$$

where $\hat{x}_0 = x_t - t\, v_\theta^{src} \approx \mathbb{E}[x_0 \mid x_t, c^{src}]$ is the Tweedie posterior mean and we define the **reconstruction velocity**:

$$v_{rec} \triangleq \frac{x_t - x_0^{src}}{t}. \tag{14}$$

This is the unique velocity that steers x_t straight ahead, arriving exactly at x_0^{src} at the end of the generative process (t=0). Substituting Eq. (13) into the RF bridge equation (11), the guidance term $-\frac{t}{1-t}\nabla \log h_{rec}$ becomes $v_{rec} - v_\theta^{src}$, and the conditioned velocity reduces to exactly v_{rec}. To allow tunable reconstruction strength, we relax the exact bridge and define:

$$\tilde{v}_{rec} \;=\; v_\theta^{src} \;+\; \lambda_{rec}\, v_{rec}, \tag{15}$$

where $\lambda_{rec} \geq 0$. We adopt this additive form rather than the residual alternative $v_\theta^{src} + \lambda(v_{rec} - v_\theta^{src})$, because the latter contaminates the error-free v_{rec} with model prediction error through the subtraction $v_{rec} - v_\theta^{src}$; the additive form preserves v_{rec} as a clean structural anchor (see Supplementary for further discussion on this parameterization).

Text-guided editing (h_{edit}) The editing constraint encodes the target prompt c^{edit}. We set $h_{edit}(x_0, 0) = p(c^{edit} \mid x_0)$, which propagates to $h_{edit}(x_t, t) = p(c^{edit} \mid x_t)$ by the harmonic property. Unlike h_{rec}, this score has no closed form. Expanding $\nabla \log h_{edit}$ via Bayes' rule yields a difference of conditional scores. To isolate the semantic change from c^{src} to c^{edit}, we replace the unconditional baseline $\nabla \log p_t(x_t)$ with the source-conditioned score $\nabla \log p(x_t \mid c^{src})$, effectively computing $\nabla \log[p(c^{edit} \mid x_t)/p(c^{src} \mid x_t)]$. Approximating both conditional scores via their Tweedie estimates and multiplying by the bridge coefficient $t/(1-t)$, all time-dependent factors cancel exactly (As shown in the Supplementary Material), yielding:

$$-\frac{t}{1-t}\nabla \log h_{edit} \;\approx\; v_\theta^{edit} - v_\theta^{src} \;\triangleq\; \Delta_{edit}. \tag{16}$$

We call Δ_{edit} the **editing direction**. Compared to the standard CFG direction $v_\theta^{edit} - v_\theta^{\varnothing}$, our formulation discards the shared content between c^{edit} and c^{src} and retains only the novel semantic change.

4.3 Orthogonal Decomposition

The product-of-experts framework combines the two guidance terms additively. However, in high-dimensional velocity space, $\Delta_{edit} = v_\theta^{edit} - v_\theta^{src}$ may retain a component along the reconstruction-guided direction $\tilde{v}_{rec}$ that interferes with source preservation. We project Δ_{edit} onto the orthogonal complement of $\tilde{v}_{rec}$:

$$\Delta_{edit}^{\perp} = \Delta_{edit} - \frac{\langle \Delta_{edit},\, \tilde{v}_{rec} \rangle}{\|\tilde{v}_{rec}\|^2}\, \tilde{v}_{rec}. \tag{17}$$

By construction, $\langle \Delta_{edit}^{\perp},\, \tilde{v}_{rec} \rangle = 0$: the editing signal now operates exclusively in directions orthogonal to the reconstruction flow. This is analogous to projected gradient descent in constrained optimization, where updates are projected onto the feasible subspace.

The projection decomposes Δ_{edit} into two interpretable components:

$$\Delta_{edit} = \underbrace{\frac{\langle \Delta_{edit},\, \tilde{v}_{rec} \rangle}{\|\tilde{v}_{rec}\|^2}\, \tilde{v}_{rec}}_{\text{parallel to } \tilde{v}_{rec} \text{ (discarded)}} + \underbrace{\Delta_{edit}^{\perp}}_{\text{orthogonal to } \tilde{v}_{rec} \text{ (kept)}}. \tag{18}$$

The parallel component overlaps with the reconstruction direction and would either redundantly reinforce or destructively counteract it; discarding it eliminates the fidelity–editability entanglement. The orthogonal component carries only the *novel semantic content* of c^{edit} that is absent from the source-preserving trajectory.

Varying λ_{edit} now modifies the edit without degrading fidelity, and varying λ_{rec} adjusts fidelity without suppressing the edit direction. The two controls are fully decoupled. The computational cost is one inner product and one vector subtraction per step.

Algorithm 1 h-Flow: RF Bridge Editing

Require: Source z^{src}, prompts c^{src} and c^{edit}, v_θ, VAE $\mathcal{E}/\mathcal{D}$, λ_{rec}, λ_{edit}, schedule $\{t_N > \cdots > t_0{=}0\}$

$x_0^{src} \leftarrow \mathcal{E}(z^{src})$
$x_{t_N} \leftarrow \textsc{Invert}(x_0^{src}, v_\theta, c^{src})$
for $n = N, \ldots, 1$ **do**
  $t \leftarrow t_n; \quad \delta t \leftarrow t_n - t_{n-1}$
  $v^{src} \leftarrow v_\theta(x_t, t, c^{src})$
  $v_{rec} \leftarrow (x_t - x_0^{src}) \,/\, t$
  $\tilde{v}_{rec} \leftarrow v^{src} + \lambda_{rec} \cdot v_{rec}$
  $v^{edit} \leftarrow v_\theta(x_t, t, c^{edit})$
  $\Delta_{edit} \leftarrow v^{edit} - v^{src}$
  $\Delta_{edit}^{\perp} \leftarrow \Delta_{edit} - \dfrac{\langle \Delta_{edit},\, \tilde{v}_{rec} \rangle}{\|\tilde{v}_{rec}\|^2}\, \tilde{v}_{rec}$
  $v \leftarrow \tilde{v}_{rec} + \lambda_{edit} \cdot \Delta_{edit}^{\perp}$
  $x_{t_{n-1}} \leftarrow x_t - v \cdot \delta t$
end for
return $\mathcal{D}(x_{t_0})$

4.4 The Complete *h*-Flow Framework

Starting from the RF bridge equation (11) and incorporating the reconstruction guidance (15), editing direction (16), and orthogonal projection (17), we arrive at the editing ODE:

$$\frac{dx_t}{dt} = \underbrace{v_\theta^{src} + \lambda_{rec}\, v_{rec}}_{\tilde{v}_{rec}} + \lambda_{edit}\, \Delta_{edit}^{\perp}. \tag{19}$$

By construction, $\tilde{v}_{rec}$ and $\Delta_{edit}^{\perp}$ are orthogonal: the two controls λ_{rec} and λ_{edit} act on independent subspaces without mutual interference.

The complete procedure is given in Algorithm 1. The method is training-free and gradient-free; its only interface to the pre-trained model is two forward passes of v_θ per step. The backward editing mechanism is agnostic to the forward initialization—any inversion, noise injection, or hybrid strategy is compatible, and improvements in the forward process translate directly to better results.

5 Experiments

5.1 Experimental Setup

Dataset. We evaluate on the PIE-Bench [14] and PIE-Bench++ [11] datasets, which contain 700 diverse images covering humans, animals, and objects in varied environments. Each sample includes source and target text descriptions and an annotated mask indicating the editing region. PIE-Bench focuses on single-aspect edits (e.g., object addition/removal, replacement, color, material, style,

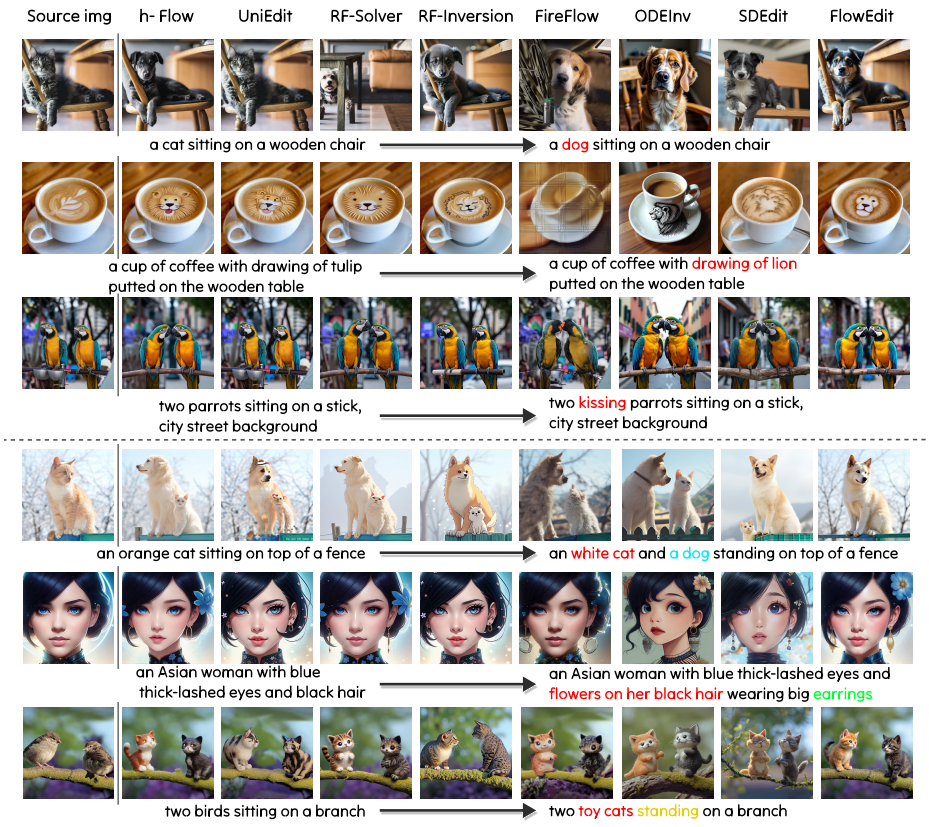


Fig. 3: Qualitative comparisons. The source prompt and the target prompt is shown on the bottom for each example. The first column displays the real input image. The subsequent columns present the editing results of our proposed h-Flow, followed by various state-of-the-art baselines.

background, action, texture, and position), while PIE-Bench++ extends the same images with multi-aspect prompts requiring simultaneous modifications to multiple objects or attributes, enabling evaluation under more complex editing scenarios.

Metrics. Following [14], we report two families of metrics. *Source Consistency* is measured on non-edited regions (defined by the provided masks) via PSNR, LPIPS [37], SSIM [34], and MSE, and on the full image via structure distance [30]. *Target Alignment* is measured by CLIP similarity [26] between the edited image and the target prompt, both on the entire image (CLIP-Entire) and on the edited region (CLIP-Edited). Besides above standard metrics, we further evaluated *Image Quality* with HPSv2 [35] and Aesthetic Score (AS) [5], which quantify perceptual fidelity and visual quality.

Baselines. We compare h-Flow against seven baselines across two categories. *Inversion-based*: RF-Inversion [28], RF-Solver [32], UniEdit-Flow [13], FireFlow [3],

Table 1: Quantitative comparison on PIE-Bench and PIE-Bench++. Best in **bold**, second best underlined, third best wavy. "Rank" indicates average rank across metrics.

Method	Rank↓	*Source Consistency*					*Target Alignment*		*Image Quality*	
		PSNR↑	LPIPS↓	SSIM↑	MSE↓	Distance↓	CLIP-Entire↑	CLIP-Edited↑	HPS↑	AS↑
PIE-Bench										
Inversion-Free										
SDEdit [23]	6.06	18.01	0.240	0.650	0.021	0.063	24.67	22.54	27.08	5.05
FlowEdit [18]	4.39	21.92	0.111	0.833	0.009	0.028	25.19	22.54	27.70	4.87
Inversion-Based										
OdeInv	5.44	13.44	0.349	0.620	0.071	0.130	**26.56**	**23.83**	**28.73**	4.92
FireFlow [3]	5.89	18.15	0.207	0.738	0.026	0.057	25.69	22.56	26.45	4.63
Rf-Inversion [28]	5.00	20.59	0.186	0.706	0.013	0.042	25.26	22.34	28.21	4.98
UniEdit-Flow [13]	3.00	**29.45**	**0.060**	**0.900**	**0.002**	**0.011**	25.80	22.34	26.77	4.96
RF-Solver [32]	3.22	22.90	0.140	0.819	0.007	0.031	26.09	22.68	27.24	**5.48**
h-Flow (ours)	**2.33**	23.88	0.110	0.840	0.006	0.026	25.80	22.80	27.79	4.98
PIE-Bench++										
Inversion-Free										
SDEdit [23]	7.11	19.92	0.2300	0.680	0.014	0.090	22.50	21.80	27.08	4.00
FlowEdit [18]	4.11	22.27	0.1087	0.841	0.009	0.031	24.94	24.24	27.70	4.86
Inversion-Based										
OdeInv	5.44	17.21	0.2400	0.720	0.028	0.080	25.50	**24.92**	**28.20**	4.73
FireFlow [3]	5.67	18.95	0.1877	0.760	0.021	0.060	25.71	24.77	26.45	4.56
Rf-Inversion [28]	4.44	21.13	0.1800	0.730	0.011	0.040	25.23	24.49	27.93	5.18
UniEdit-Flow [13]	3.56	**29.46**	**0.0600**	**0.900**	0.018	**0.018**	25.37	24.39	26.96	4.89
RF-Solver [32]	2.89	23.69	0.1285	0.832	0.006	0.033	**25.79**	24.89	27.24	**5.43**
h-Flow (ours)	**2.78**	24.30	0.1090	0.850	**0.005**	0.029	25.48	24.54	27.67	4.92

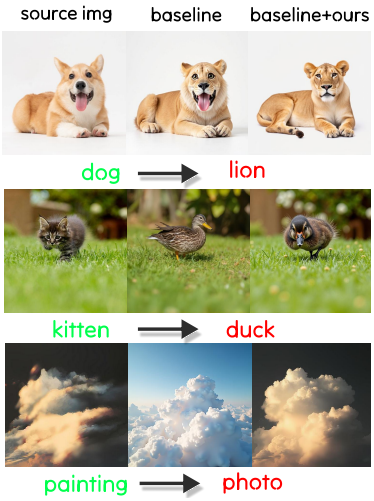
and standard ODE inversion (OdeInv). *Inversion-free*: FlowEdit [18]. SDEdit [23]. All methods use officially released code with recommended hyperparameters and are evaluated on the same FLUX.1-dev backbone.

Implementation Details

We evaluate on FLUX.1-dev. For the backward editing process, we use N=28 Euler steps with a schedule on $[0, 1]$. Unless otherwise specified, the forward process defaults to RF-Inversion [28] with n=24 steps. The hyperparameters λ_{rec} and λ_{edit} are set to `1.2` and `1.5`, respectively.

5.2 Experimental Results

Qualitative evaluation. As shown in Figure 3, we can observe 1) *Superior semantic alignment.* h-Flow accurately executes complex prompts, including subject translation (*e.g.*, cat to dog in row 1, col 2) and localized element additions (*e.g.*, adding flowers and earrings in row 5, col 2), avoiding the loss of subject identity commonly seen in ODEInv (col 7) and SDEdit (col 8). 2) *Strict structural preservation.* Compared to FireFlow (col 6), our method perfectly preserves the background and unedited regions. 3) *Optimal editing-fidelity balance.* Both visually and quantitatively, h-Flow demonstrates outstanding performance in balancing editability and fidelity. For instance, UniEdit (row 3, col 3) fails to achieve the editing action of "kissing" for the parrots, whereas our method (row



Method	*Source Consistency*		*Target Alignment*	
	SSIM↑	Distance↓	CLIP-Entire↑	CLIP-Edited↑
RF-Inversion [28]	0.70	**0.070**	25.26	22.34
+ h-Flow (ours)	**0.73**	0.082	**26.10**	**23.60**
SDEdit [23]	0.65	0.063	24.67	22.54
+ h-Flow (ours)	**0.65**	**0.063**	**25.28**	**23.10**
ODE-Inv	0.62	0.130	**26.56**	**23.83**
+ h-Flow (ours)	**0.70**	**0.077**	26.44	23.65

Table 2: ↑ Quantitative comparison of the compatibility of h-Flow with diverse forward processes on PIE-Bench.

Fig. 4: ← Qualitative comparison of h-Flow integrated with different forward processes (baselines from top to bottom: RF-Inversion, SDEdit, and ODE-Inv).

3, col 2) successfully implements this action while maintaining high structural consistency, achieving top-tier aesthetic quality and text alignment.

Quantitative evaluation From Table 1, we have the following observations 1) For both benchmarks, UniEdit achieves the highest source consistency (*e.g.*, highest PSNR and lowest MSE) but yields suboptimal target alignment and lower human preference scores (HPS). 2) Conversely, ODEInv achieves high target alignment (*e.g.*, best CLIP-Edited scores) but severely degrades image quality and structural preservation, as evidenced by its lowest PSNR and highest Distance metrics. 3) Our proposed h-Flow achieves a significantly better trade-off. It ranks first in average across all 9 metrics on both PIE-Bench and PIE-Bench++, maintaining highly competitive source consistency while delivering top-tier target alignment and aesthetic quality (AS).

5.3 Ablation Study

Compatibility with Forward Processes. Since our backward editing ODE is agnostic to the forward process, h-Flow can be integrated with diverse forward initialization strategies in a plug-and-play manner. To validate this, we combine h-Flow with three distinct forward processes: RF-Inversion, SDEdit (noise injection), and ODE-Inv. As shown in Table 2, applying h-Flow consistently optimizes the trade-off between editability and fidelity across all settings. For RF-Inversion and SDEdit, h-Flow substantially improves target text alignment (CLIP-Edited gains of +1.26 and +0.56, respectively) while strictly preserving structural metrics. Notably, for ODE-Inv, the base forward process achieves high editability but suffers from severe structural degradation (SSIM drops to 0.62, Distance spikes to 0.130). In this case, integrating h-Flow acts as a powerful structural stabilizer, massively recovering source fidelity (SSIM surges to 0.70, Distance drops to 0.077) with only a negligible fluctuation in text alignment.

Table 3: Effect of reconstruction strength λ_{rec} (fixed λ_{edit}=1.5).

λ_{rec}	SSIM↑	Distance↓	CLIP-Entire↑	CLIP-Edited↑
0.0	0.830	0.026	**25.97**	**22.92**
1.2	0.840	0.026	25.80	22.80
1.5	**0.856**	**0.025**	25.65	22.73

Table 4: Effect of editing strength λ_{edit} (fixed λ_{rec}=1.2).

λ_{edit}	SSIM↑	Distance↓	CLIP-Entire↑	CLIP-Edited↑
0.0	**0.846**	**0.020**	22.84	20.49
1.5	0.840	0.026	25.80	22.80
3.0	0.804	0.034	**26.26**	**23.49**

As shown in Figure 4, h-Flow consistently delivers high-quality edits, robustly compensating for the inherent weaknesses of each base inversion method.

Hyperparameters. 1) Effect of λ_{rec}. Table 3 sweeps λ_{rec} with λ_{edit}=1.5 fixed. As λ_{rec} increases from 0 to 1.5, confirming that the reconstruction guidance progressively anchors the trajectory to the source. We set 1.2 for trade-off. 2) Effect of λ_{edit}. Table 4 sweeps λ_{edit} with λ_{rec}=1.2 fixed. Increasing λ_{edit} from 0 to 1.5 significantly boosts editability (CLIP-Edited 20.49 → 22.80) with only a modest fidelity drop. Further increasing to 3.0 yields diminishing returns in editing but exacerbates fidelity degradation, thus we set λ_{edit}=1.5 as a suitable trade-off. Crucially, comparing Tables 3 and 4 demonstrates successful objective decoupling via our decomposition: sweeping λ_{rec} barely affects editability (a mere 0.19-point drop in CLIP-Edited), whereas sweeping λ_{edit} drives a massive 3.0-point swing, where each parameter predominantly controls its intended axis.

Effect of orthogonal projection. Table 5 ablates projecting Δ_{edit} onto the orthogonal complement of $\tilde{v}_{rec}$. Under fixed hyperparameters (λ_{rec}=1.2, λ_{edit}=1.5), the projection significantly improves CLIP-Entire (+0.43) while keeping structural source fidelity strictly preserved (Distance remains completely unchanged at 0.026, and SSIM slightly improves to 0.840). Mechanistically, the raw Δ_{edit} contains components along $\tilde{v}_{rec}$ that inadvertently counteract reconstruction guidance. Eliminating this interference focuses the editing strictly on orthogonal semantic shifts, achieving enhanced text alignment at zero structural cost.

Table 5: Effect of orthogonal projection. The projection improves editing effectiveness without sacrificing source consistency.

Variant	SSIM↑	Distance↓	CLIP-Entire↑	CLIP-Edited↑
w/o projection	0.834	**0.026**	25.37	22.79
w/ projection	**0.840**	**0.026**	**25.80**	**22.80**

6 Conclusion

We presented **h-flow**, a principled flow-based image editing method that reformulates editing through Doob's h-Transform and extends it to the deterministic rectified flow framework. By explicitly modeling source consistency and target alignment with dedicated h-functions and decoupling their effects via velocity orthogonal decomposition, our approach achieves a controllable trade-off between reconstruction and semantic editing while remaining training-free and plug-and-play. Future work will explore extending this framework to other modalities such as video or 3D generation and incorporating richer combinations of control conditions for more general multimodal editing.

7 Acknowledgement

This work was supported by the National Natural Science Foundation of China (62572194, 62472178, 62376244) and the Shanghai Frontiers Science Center of Molecule Intelligent Syntheses. This article was also supported by the Key Technology Research and Development Program Project of the Shanghai Science and Technology Commission under Grants (25511107200, 25511102400). Long Chen was supported by Hong Kong SAR Research Grants Council (RGC) General Research Fund (16219025), National Natural Science Foundation of China (NSFC) Young Scientists Fund Category B (62522216), Hong Kong SAR Research Grants Council (RGC) Early Career Scheme (26208924), and National Natural Science Foundation of China (NSFC) Young Scientists Fund Category C (62402408).

References

1. Chen, S., Chewi, S., Lee, H., Li, Y., Lu, J., Salim, A.: The probability flow ode is provably fast. NeurIPS **36**, 68552–68575 (2023)
2. Dao, T., Wang, Z., Pham, K.T., Chen, L.: Steerflow: Steering rectified flows for faithful inversion-based image editing. arXiv preprint arXiv:2604.01715 (2026)
3. Deng, Y., He, X., Mei, C., Wang, P., Tang, F.: Fireflow: Fast inversion of rectified flow for image semantic editing. In: ICML (2025)
4. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. NeurIPS **34**, 8780–8794 (2021)
5. discus0434: Aesthetic predictor v2.5: A siglip-based aesthetic score predictor. `https://github.com/discus0434/aesthetic-predictor-v2-5` (2024)
6. Doob, J.L., et al.: Classical potential theory and its probabilistic counterpart, vol. 19. Springer (1984)
7. Du, Y., Durkan, C., Strudel, R., Tenenbaum, J.B., Dieleman, S., Fergus, R., Sohl-Dickstein, J., Doucet, A., Grathwohl, W.S.: Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In: ICML. pp. 8489–8510. PMLR (2023)
8. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: ICML (2024)
9. Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross-attention control. In: ICLR (2023)
10. Ho, J., Salimans, T.: Classifier-free diffusion guidance. In: NeurIPS Workshop on Deep Generative Models and Downstream Applications (2021)
11. Huang, M., Cai, J., Jia, S., Lokhande, V.S., Lyu, S.: Paralleledits: Efficient multi-aspect text-driven image editing with attention grouping. NeurIPS **37**, 22569–22595 (2024)
12. Jiang, Y., Wang, Z., Wang, Y., Yu, J., Zhuang, Y., Xiao, J., Chen, L.: Flowdc: Flow-based decoupling-decay for complex image editing. In: CVPR. pp. 25757–25766 (2026)
13. Jiao, G., Huang, B., Wang, K.C., Liao, R.: Uniedit-flow: Unleashing inversion and editing in the era of flow models (2025), `https://arxiv.org/abs/2504.13109`
14. Ju, X., Zeng, A., Bian, Y., Liu, S., Xu, Q.: Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In: ICLR (2024)

15. Kieu, D., Do, K., Nguyen, T., Nguyen, D., Nguyen, T.: Bidirectional diffusion bridge models. In: KDD. pp. 1139–1148 (2025)
16. Kim, J., Hong, Y., Park, J., Ye, J.C.: Flowalign: Trajectory-regularized, inversion-free flow-based image editing. arXiv preprint arXiv:2505.23145 (2025)
17. Kim, J., Park, J., Song, Y., Kwak, N., Rhee, W.: Reflex: Text-guided editing of real images in rectified flow via mid-step feature extraction and attention adaptation. In: ICCV. pp. 15939–15948 (2025)
18. Kulikov, V., Kleiner, M., Huberman-Spiegelglas, I., Michaeli, T.: Flowedit: Inversion-free text-based editing using pre-trained flow models. In: ICCV. pp. 19721–19730 (2025)
19. Labs, B.F.: Flux. `https://github.com/black-forest-labs/flux` (2024)
20. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. arXiv preprint arXiv:2210.02747 (2022)
21. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: ICLR (2023)
22. Liu, X., Wu, L., Ye, M., Liu, Q.: Learning diffusion bridges on constrained domains. In: ICLR (2023)
23. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.Y., Ermon, S.: Sdedit: Guided image synthesis and editing with stochastic differential equations. In: ICLR (2022)
24. Mokady, R., Hertz, A., Aberman, K., Pritch, Y., Cohen-Or, D.: Null-text inversion for editing real images using guided diffusion models. In: CVPR. pp. 6038–6047 (2023)
25. Nguyen, T., Do, K., Kieu, D., Nguyen, T.: h-edit: Effective and flexible diffusion-based editing via doob's h-transform. In: CVPR. pp. 28490–28501 (2025)
26. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763. PMLR (2021)
27. Rogers, L.C.G., Williams, D.: Diffusions, Markov processes, and martingales, vol. 2. Cambridge university press (2000)
28. Rout, L., Chen, Y., Ruiz, N., Caramanis, C., Shakkottai, S., Chu, W.S.: Semantic image inversion and editing using rectified stochastic differential equations. In: ICLR (2025)
29. Särkkä, S., Solin, A.: Applied stochastic differential equations, vol. 10. Cambridge University Press (2019)
30. Tumanyan, N., Bar-Tal, O., Bagon, S., Dekel, T.: Splicing vit features for semantic appearance transfer. In: CVPR. pp. 10748–10757 (2022)
31. Tumanyan, N., Geyer, M., Bagon, S., Dekel, T.: Plug-and-play diffusion features for text-driven image-to-image translation. In: CVPR. pp. 1921–1930 (2023)
32. Wang, J., Pu, J., Qi, Z., Guo, J., Ma, Y., Huang, N., Chen, Y., Li, X., Shan, Y.: Taming rectified flow for inversion and editing. In: ICML. pp. 64044–64058. PMLR (2025)
33. Wang, Y., Jiang, Z., Wang, Z., Chen, L.: Coarse-guided visual generation via weighted h-transform sampling. arXiv preprint arXiv:2603.12057 (2026)
34. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
35. Wu, X., Hao, Y., Sun, K., Chen, Y., Zhu, F., Zhao, R., Li, H.: Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. arXiv preprint arXiv:2306.09341 (2023)

36. Xu, Y., Wang, Z., Li, K., Xiao, J., Chen, L.: Freetuner: Any subject in any style with training-free diffusion. ACM Transactions on Multimedia Computing, Communications and Applications (2024)
37. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR. pp. 586–595 (2018)
38. Zhou, L., Lou, A., Khanna, S., Ermon, S.: Denoising diffusion bridge models. In: ICLR (2024)