BEYOND TEMPORAL SMOOTHING: SPATIAL ENERGY BUDGETS STABILIZE ONE-STEP DIFFUSION EDITING

Anonymous authors
Paper under double-blind review



Figure 1: **One-step editing by BudEdit.** It converts background residual energy into a spatial budget and allocates the constructed injection only to edit-relevant regions for stronger, more coherent edits.

ABSTRACT

One-step text-guided diffusion editing is efficient but prone to spatially misallocated updates that distort the edited object and alter the background. Existing methods often improve stability by averaging the editing field across timesteps. We instead identify spatial energy misallocation as a distinct and measurable failure mode: across two independent noise draws, the residual field is essentially unrepeatable, making the background field unreliable for direct transport, while its total energy still sets a usable magnitude for the draw at hand. BudEdit turns that magnitude into an explicit budget and reallocates it to edit-relevant regions selected jointly by residual energy and cross-attention, controlling where editing energy is spent rather than averaging over timesteps. The resulting training-free, inversion-free editor spends the budget on transport and reuses it to scale a correction in a lower-noise gated refinement. The budgeted injection field matches its prescribed budget exactly and vanishes on the identified background support, by construction. On PIE-Bench with SD-Turbo, BudEdit outperforms ChordEdit under each method's reported default settings on all 11 evaluated metrics, including a 2.1 dB gain in background PSNR, 31% lower DINO, and 36% lower LPIPS, while improving all five editing-quality metrics and reporting the lowest runtime in the comparison.

1 INTRODUCTION

Distilled one-step generators such as SD-Turbo (Sauer et al., 2024), InstaFlow (Liu et al., 2024), and SwiftBrush (Nguyen & Tran, 2024) have made real-time synthesis practical, but extending that speed to text-guided editing has proven harder: the editing signal is the difference between two independently conditioned model outputs (source and target prompts), and integrating this field in a single large step can produce object distortion and background degradation. Existing training-free editors (Kulikov et al., 2025; Xu et al., 2024) stay stable by distributing the update over many steps, while trained one-step editors (Nguyen et al., 2025) require task-specific training and therefore target a different setting from training-free approaches.

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.

Recent one-step editors have explored different approaches to stabilizing this regime. ChordEdit (Lu et al., 2026) attributes the instability to the volatility of the single-timestep field and proposes temporal smoothing as a remedy. Other recent methods infer an edit mask and amplify conditional transport, as in WhereEdit (Hu et al., 2026), or refine the transport trajectory with the Riemannian residual line search of RRLS (Yi et al., 2026), neither of which controls where the single-draw field deposits its energy. This motivates a complementary question: can the spatial distribution of the editing field be controlled explicitly rather than averaged uniformly across timesteps?

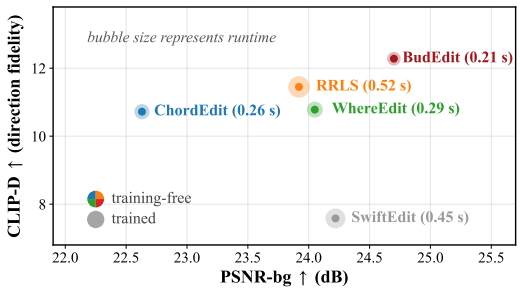

Figure 2: **One-step editors on PIE-Bench.** PSNR-bg vs. CLIP-D, bubble size = runtime; BudEdit leads both axes and reports the lowest runtime in this comparison.

We identify spatial energy misallocation as a measurable failure mode of one-step editing. Across two independent noise draws the residual field is essentially unrepeatable, with region-level cosine similarity near 0.1 in both the injection support and its complement, and its amplitude map is no more stable. The source-anchored displacement is substantially more reproducible and is therefore used as the injection direction. Although the residual remains draw-dependent, BudEdit uses its realized energy as a budget scale for the current draw and its high-energy support as a set of candidate edit locations for that draw. Accordingly, the raw residual is not transported globally; it is retained only as an attenuated, support-restricted base term. The appendix reports these two-draw measurements.

We therefore introduce BudEdit, a training-free and inversion-free editor that converts unreliable background residual energy into an image-dependent budget and reallocates it to edit-relevant regions. A residual-energy support and a cross-attention localization signal, both read off the transport forward pass at no additional UNet evaluation, determine where the budget is allocated, while a constrained allocation problem determines how it is distributed. The resulting closed-form construction provides exact budget matching on nonempty injection support and hard-support control for the injected field. A second low-noise pass then cleans up the transported image: within an attention-derived gate, GatedRefine scales its correction by the same budget and restores source detail.

Our contributions are summarized as follows:

- We identify spatial energy misallocation as a measurable failure mode in one-step editing, complementing existing views centered on temporal instability.
- We formulate spatial energy budgeting as a constrained allocation problem and derive a closed-form solution with exact construction-level budget matching on nonempty injection support and hard-support control.
- We propose BudEdit, which sets its budget from background residual energy and injects it where the energy support meets a prompt-difference attention gate, outperforming the temporal-smoothing baseline on all 11 PIE-Bench metrics.

2 Related Work

Multi-step editing established the prevailing recipe for stability: shape how the edit signal enters the denoiser and pay for the control with steps, typically 20–50 evaluations per edit. Attention manipulation and spatial feature injection guide the edit from inside the model (Hertz et al., 2023; Tumanyan et al., 2023), while inversion-based pipelines reconstruct the source along a guided trajectory (Song et al., 2021a; Mokady et al., 2023). Distilled generators (Salimans & Ho, 2022; Song et al., 2023; Sauer et al., 2024; Liu et al., 2024; Nguyen & Tran, 2024) offer real-time synthesis, but their editing signal, a difference of two independently conditioned predictions, must be integrated in a single large step; existing inversion-free editors (Kulikov et al., 2025; Xu et al., 2024) stay above interactive rates, and FlowDC (Jiang et al., 2026) likewise operates over multiple steps while decaying the velocity components orthogonal to the edit direction for multi-target complex editing on its own Complex-PIE-Bench; trained editors (Brooks et al., 2023; Nguyen et al., 2025) require task-specific training rather than the training-free operation studied here.

Among training-free one-step editors, ChordEdit (Lu et al., 2026) sets the reference: it frames the task as dynamic optimal transport, attributes instability to the volatility of the single-timestep field, and stabilizes integration by averaging observable residuals at two timesteps. WhereEdit (Hu et al., 2026) performs mask-aware local latent editing and RRLS (Yi et al., 2026) refines the transport with a Riemannian residual line search. These methods leave open the question of whether measured residual energy can be controlled explicitly in space.

Across these approaches, stabilization remains implicit. To our knowledge, no training-free one-step editor formulates the spatial allocation of measured residual energy as an image-dependent budget with construction-level support constraints.

Our work makes this energy explicit. In the Benamou–Brenier formulation (Benamou & Brenier, 2000), kinetic energy is the fundamental transport cost, and low-energy paths underlie optimal-transport flow matching (Lipman et al., 2023); editing, however, has not treated measured residual energy as a controllable quantity. We do: the single-draw editing field is read as a noisy measurement, its draw-sensitive background energy is converted into an image-dependent budget, and the budget is re-injected where the model's own attention indicates edit relevance. This reframes stabilization from uniform temporal smoothing to explicit spatial allocation of editing energy.

3 METHOD

3.1 PRELIMINARIES

Following Lu et al. (2026), we use the probability-flow ODE framework (Song et al., 2021b). A text-to-image model conditioned on c induces a drift $v(x_t, t, c)$; editing transports the source latent x_{src} from the source-conditioned data distribution $p(x \mid c_{\mathrm{src}})$ toward the target-conditioned one $p(x \mid c_{\mathrm{tar}})$. At current latent x_τ, which is x_{src} in the single-step setting, and time t, the model is queried at $z \sim K_t(\cdot \mid x_\tau)$. Its output $Q(z, t, c)$ (noise prediction for SD-Turbo, velocity for flow models) is mapped by the time-only linear map $\mathcal{T}_t$ into a common comparison field, and the source/target predictions share the same z.

At fixed (x_τ, t), write $Q_z^c \equiv Q(z, t, c)$. The algorithm computes a single-draw residual, while its population counterpart averages over the noise distribution:

$$\widehat{R}_z \equiv \widehat{R}(x_\tau, t; z) = \mathcal{T}_t(Q_z^{c_{\mathrm{tar}}} - Q_z^{c_{\mathrm{src}}}), \qquad \overline{R} \equiv \overline{R}(x_\tau, t) = \mathbb{E}_{z \sim K_t(\cdot \mid x_\tau)}\left[\widehat{R}_z\right]. \tag{1}$$

We define the population residual field and its draw-dependent deviation as

$$u_t(x_\tau) := \overline{R}(x_\tau, t), \qquad \epsilon_t(z) := \widehat{R}_z - u_t(x_\tau). \tag{2}$$

By construction, $\mathbb{E}_{z \sim K_t(\cdot \mid x_\tau)}[\epsilon_t(z)] = 0$. Thus, $\widehat{R}_z = u_t(x_\tau) + \epsilon_t(z)$ decomposes the single-draw residual into its population mean and draw-dependent deviation. When the noise draw is fixed, we suppress z and write $\widehat{R}(x_\tau, t)$.

Throughout, $\|\cdot\|_C$ denotes the per-pixel channel-wise norm of a latent field, and $\|h\|^2 \equiv \sum_{(i,j)} \|h(i,j)\|_C^2$ denotes its squared L_2 norm over the domain Ω, corresponding to the entire image. We refer to this quantity as its total energy.

3.2 AN ENERGY DIAGNOSIS OF ONE-STEP INSTABILITY

For the time-dependent ODE $\dot{x} = v(x, t) \equiv v_t(x)$, assume that v is C^2 with bounded derivatives along the exact trajectory and that the Euler and exact solutions start from the same state. The local truncation error of one explicit Euler step of size Δt is bounded by the total derivative of the vector field:

$$\|x_{\mathrm{true}} - x_{\mathrm{euler}}\| \le \frac{(\Delta t)^2}{2} \sup_{s \in [t, t+\Delta t]} \|\partial_s v(x(s), s) + D_x v(x(s), s) v(x(s), s)\|. \tag{3}$$

A useful upper bound replaces the total derivative norm by $\sup_s \|\partial_s v(x(s), s)\| + \sup_s \|D_x v(x(s), s)\| \sup_s \|v(x(s), s)\|$. Thus, field magnitude is one component of the one-step error. On the background support this argues for suppression, since a zero field removes its magnitude-dependent contribution. Inside the injection support the construction raises field magnitude instead, trading a larger local discretization error for a semantic transition completed in a single step.

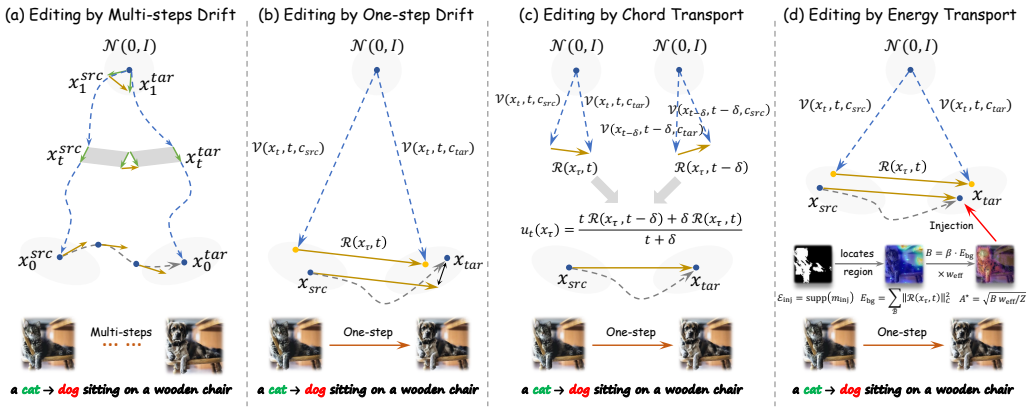

Figure 3: **Comparison of editing paradigms.** (a) multi-step editing buys stability by spreading the update over many small steps; (b) one-step editing integrates the measured field in a single large step; (c) chord transport forms the one-step update from a time-weighted average of residual fields at two timesteps; (d) energy transport converts background residual energy into a budget and spends it inside the effective injection support.

For the conceptual diagnosis, let $\mathcal{E}^\star$ denote the semantic edit region and let $\mathcal{B}^\star = \Omega \setminus \mathcal{E}^\star$ denote its background. We write the pointwise decomposition $\widehat{R}(x_\tau, t; z) = u_t(x_\tau) + \epsilon_t(z)$. On $\mathcal{B}^\star$, the desired semantic transport component underlying u_t is expected to be small, but the full population residual need not vanish: it may contain systematic prompt-pair response as well as draw-sensitive variation. We treat noise dominance as an operational property, verified empirically rather than derived from the model: raw residual amplitudes are draw-sensitive, and the background residual field is unreliable for direct transport. The corresponding two-draw measurement is reported in the Appendix. For $0 < \delta < t_s$, ChordEdit's chord control field averages residuals across the two timesteps t_s and $t_s - \delta$:

$$\widehat{u}_{t_s}(x_\tau) = \frac{t_s\, \widehat{R}(x_\tau, t_s - \delta) + \delta\, \widehat{R}(x_\tau, t_s)}{t_s + \delta}. \tag{4}$$

The average is applied uniformly across edited and background regions, so it reduces the variance of the field without deciding where its energy lands.

Accordingly, BudEdit separates how much editing energy is administered from where and how it is spent: the spatially aggregated background residual energy from the current draw sets the per-image budget scale, while the effective injection support and allocation profile determine its spatial deployment. The next section formalizes this allocation as a constrained optimization with a closed-form solution. Figure 3 contrasts this spatial budgeting view with temporal smoothing.

3.3 Spatial Energy Budgeting

The analysis above suggests a two-stage construction: the aggregate residual energy measured on the operational background support scales an image-dependent budget, which is then allocated within an effective edit support according to an explicit spatial profile.

Harvesting the budget. The algorithm estimates a data-dependent energy support $\widehat{\mathcal{E}}$ from the residual field: it thresholds the per-pixel residual energy and then applies a narrow boundary extension, represented by a soft mask $m \in [0,1]^{H\times W}$ whose nonzero support is $\mathrm{supp}(m) = \widehat{\mathcal{E}}$. Let $\widehat{\mathcal{B}} = \Omega \setminus \widehat{\mathcal{E}}$ denote the corresponding operational background support.

Let $\Phi \in [0,1]^{H\times W}$ denote the peak-normalized cross-attention localization signal and $\widehat{x}_{\mathrm{tar}}$ the target-conditioned latent prediction, both read off the same UNet forward pass; the construction of Φ is detailed in Section 3.4. The residual-energy support $\widehat{\mathcal{E}}$ provides candidate edit locations, while Φ gates and weights them. With $\mathbf{1}[\cdot]$ denoting an indicator, we define the mask pointwise for $(i,j) \in \Omega$ and set its active support as follows:

$$m_{\mathrm{inj}}(i,j) = m(i,j)\,\mathbf{1}[\Phi(i,j) > 0.5]\,\mathbf{1}[\|\widehat{x}_{\mathrm{tar}}(i,j) - x_{\mathrm{src}}(i,j)\|_C > 0]\,, \quad \mathcal{E}_{\mathrm{inj}} = \mathrm{supp}(m_{\mathrm{inj}}). \tag{5}$$

The smoothed attention readout $w = \mathrm{blur}(\Phi)$ specifies the within-support allocation profile. Together with the soft injection mask, it gives

$$w_{\mathrm{eff}}(i,j) = m_{\mathrm{inj}}(i,j)^2\, w(i,j). \tag{6}$$

Because m_{inj} scales field amplitude, its square enters the energy profile. The nonnegative smoother satisfies $w(i,j) > 0$ wherever $\Phi(i,j) > 0.5$, so Eq. 5 gives $w_{\mathrm{eff}} > 0$ throughout $\mathcal{E}_{\mathrm{inj}}$.

Energy is harvested from the operational background support, without applying the attention gate:

$$E_{\mathrm{bg}} = \sum_{(i,j)\in\widehat{\mathcal{B}}} \|\widehat{R}(i,j)\|_C^2, \qquad B = \beta E_{\mathrm{bg}}. \tag{7}$$

For $\beta \geq 0$, $B \geq 0$. The quantity E_{bg} sets an image-dependent scale rather than a conserved quantity: $\beta = 1$ equates the budget with the measured background energy, while $\beta > 1$ favors stronger editing. We use $\beta = 4.0$ and report the sweep in the Appendix. On nonempty support, Eq. 7 makes the mean injected energy density scale with $|\widehat{\mathcal{B}}|/|\mathcal{E}_{\mathrm{inj}}|$, giving a small target on a large canvas a denser correction.

Allocation as information projection. Assume that $\mathcal{E}_{\mathrm{inj}}$ is nonempty and $B > 0$. Eq. 5 makes the displacement nonzero on this support, so

$$d(i,j) = \frac{\widehat{x}_{\mathrm{tar}}(i,j) - x_{\mathrm{src}}(i,j)}{\|\widehat{x}_{\mathrm{tar}}(i,j) - x_{\mathrm{src}}(i,j)\|_C}, \quad \|d(i,j)\|_C = 1, \quad (i,j) \in \mathcal{E}_{\mathrm{inj}}. \tag{8}$$

The injection field takes the form $f = A \odot d$, where $A \geq 0$ is the unknown scalar amplitude to be solved for. We further define, for $f \not\equiv 0$ on $\mathcal{E}_{\mathrm{inj}}$,

$$\mu_{w_{\mathrm{eff}}}(i,j) = \frac{w_{\mathrm{eff}}(i,j)}{\sum_{(k,l)\in\mathcal{E}_{\mathrm{inj}}} w_{\mathrm{eff}}(k,l)}, \qquad p_f(i,j) = \frac{\|f(i,j)\|_C^2}{\sum_{(k,l)\in\mathcal{E}_{\mathrm{inj}}} \|f(k,l)\|_C^2}. \tag{9}$$

For $B > 0$, the budgeted solution is

$$(f^\star, A^\star) \;\in\; \arg\min_{f,A}\; D_{\mathrm{KL}}\big(p_f \,\big\|\, \mu_{w_{\mathrm{eff}}}\big) \text{ s.t. } \begin{array}{ll} \mathrm{supp}(f) \subseteq \mathcal{E}_{\mathrm{inj}}, & f|_{\mathcal{E}_{\mathrm{inj}}} = A \odot d, \\ A(i,j) \geq 0, & \sum_{(i,j)\in\mathcal{E}_{\mathrm{inj}}} A(i,j)^2 \;=\; B, \end{array} \tag{10}$$

The distributions induced by feasible solutions of Eq. 10 include $\mu_{w_{\mathrm{eff}}}$, so the minimum is attained exactly at $p_{f^\star} = \mu_{w_{\mathrm{eff}}}$. With this and the fixed-budget constraint, the nonnegativity condition yields the unique amplitude map on $\mathcal{E}_{\mathrm{inj}}$:

$$A^\star(i,j) = \sqrt{\frac{B\, w_{\mathrm{eff}}(i,j)}{\sum_{(k,l)\in\mathcal{E}_{\mathrm{inj}}} w_{\mathrm{eff}}(k,l)}}, \qquad f^\star = A^\star \odot d. \tag{11}$$

For $B = 0$ or $\mathcal{E}_{\mathrm{inj}} = \varnothing$, we set $f^\star = 0$; the implementation uses the same fallback if the normalizing sum vanishes, a defensive case excluded by the positivity above. The nominal transport field is

$$F = \alpha\, m_{\mathrm{inj}} \odot \widehat{R} + f^\star, \qquad x_{\mathrm{edit}} = x_{\mathrm{src}} + F, \tag{12}$$

where $\alpha = 0.7$. All image-dependent quantities in Eqs. 5–12 are computed from the current image and prompts; α, β, and other implementation constants are fixed globally, while the Appendix summarizes the remaining construction rules.

Exact injection-budget matching and background support. Under the assumptions above, the unique solution $f^\star$ of Eq. 10 satisfies

$$\sum_{(i,j)\in\mathcal{E}_{\mathrm{inj}}} \|f^\star(i,j)\|_C^2 = B, \qquad f^\star|_{\widehat{\mathcal{B}}} = 0. \tag{13}$$

For $B > 0$, Eq. 13 holds exactly; for $B = 0$, both sides are zero by convention. These guarantees apply to the administered injection field $f^\star$. The full field F also contains the attenuated residual term, while refinement and fusion determine the final image-space transition.

A transport-field bound on the injection support. By the triangle inequality and $0 \le m_{\text{inj}} \le 1$, the composed field F of Eq. 12 satisfies:

$$\sum_{(i,j)\in\mathcal{E}_{\text{inj}}} \|F(i,j)\|_C^2 \;\le\; \left(\alpha\sqrt{\sum_{(i,j)\in\mathcal{E}_{\text{inj}}} \|\widehat{R}(i,j)\|_C^2} + \sqrt{B}\right)^2. \tag{14}$$

The bound controls the nominal composed field; the ODE discretization term is governed by the local-error analysis above, while fixed calibration, refinement, and output blending determine the final image-space transition.

3.4 Localization from Prompt-Difference Attention

Cross-attention maps bind spatial positions to text tokens (Hertz et al., 2023; Chefer et al., 2023; Hu et al., 2026). A capturing attention processor, swapped into the forward pass already required for transport, yields an attention signal $\Phi \in [0,1]^{H\times W}$ without an additional UNet evaluation: maps of the differing tokens are averaged over attention heads, upsampled to latent resolution, averaged over layers, peak-normalized, and finally max-combined per pixel across the two prompt sides. The blurred readout $w = \text{blur}(\Phi)$ supplies the unweighted allocation signal. Together with the soft injection mask, it forms the effective allocation weight w_{eff} of Eq. 6; the same measurement supplies the pointwise hard attention gate of Eq. 5.

Algorithm 1: BudEdit: energy transport.

Input: $x_{\text{src}}, c_{\text{src}}, c_{\text{tar}}, t_s, t_e, \alpha, \beta$
$z \leftarrow \text{noisify}(x_{\text{src}}, t_s)$
$(Q_z^{c_{\text{src}}}, Q_z^{c_{\text{tar}}}, \widehat{x}_{\text{tar}}, \Phi) \leftarrow \text{UNet}(z, t_s; \cdot)$
$\widehat{R} \leftarrow \mathcal{T}_{t_s}(Q_z^{c_{\text{tar}}} - Q_z^{c_{\text{src}}})$
$\widehat{\mathcal{E}} \leftarrow \text{Extend}(\text{EnergySupport}(\widehat{R}))$; $\widehat{\mathcal{B}} \leftarrow \Omega \setminus \widehat{\mathcal{E}}$
$m \leftarrow \text{Soften}(\widehat{\mathcal{E}})$; $w \leftarrow \text{blur}(\Phi)$
$(m_{\text{inj}}, \mathcal{E}_{\text{inj}}) \leftarrow$ Eq. 5; $w_{\text{eff}} \leftarrow m_{\text{inj}}^2 \odot w$
$Z \leftarrow \sum_{(i,j)\in\mathcal{E}_{\text{inj}}} w_{\text{eff}}(i,j)$
$E_{\text{bg}} \leftarrow \sum_{(i,j)\in\widehat{\mathcal{B}}} \|\widehat{R}(i,j)\|_C^2$; $B \leftarrow \beta E_{\text{bg}}$
if $\mathcal{E}_{\text{inj}}=\varnothing$ **or** $B=0$ **or** $Z=0$: $f^\star \leftarrow 0$
else: $d \leftarrow \text{unit}(\widehat{x}_{\text{tar}} - x_{\text{src}})$ on $\mathcal{E}_{\text{inj}}$; $f^\star \leftarrow A^\star \odot d$ by Eq. 11
$x_{\text{edit}} \leftarrow x_{\text{src}} + \alpha m_{\text{inj}} \odot \widehat{R} + f^\star$
$z_e \leftarrow \text{noisify}(x_{\text{edit}}, t_e)$
$(Q_{z_e}^{c_{\text{src}}}, Q_{z_e}^{c_{\text{tar}}}, \widetilde{x}_{\text{tar},e}, \Phi_{\text{pin}}) \leftarrow \text{UNet}(z_e, t_e; \cdot)$
$\widetilde{R}_e \leftarrow \mathcal{T}_{t_e}(Q_{z_e}^{c_{\text{tar}}} - Q_{z_e}^{c_{\text{src}}})$
$(x_{\text{refined}}, \Psi) \leftarrow \text{GatedRefine}(\cdot)$ by Eq. 16
$x_{\text{out}} \leftarrow \Psi \odot x_{\text{refined}} + (1-\Psi) \odot x_{\text{src}}$; **return** x_{out}

Attention plays three roles: defining the hard attention gate, shaping the effective profile w_{eff} within the region, and gating the second-pass refinement of Section 3.5. Energy support supplies candidate edit locations; cross-attention gates and weights the injected field.

3.5 Budget-Aware Gated Refinement

Second-pass cleanup. Given the transported prediction x_{edit}, we introduce a second-pass cleanup, implemented as a single forward pass at the lower noise level t_e that reuses the noise realization of the transport pass:

$$z_e = \text{noisify}(x_{\text{edit}}, t_e), \qquad \widetilde{R}_e = \mathcal{T}_{t_e}\big(Q_{z_e}^{c_{\text{tar}}} - Q_{z_e}^{c_{\text{src}}}\big). \tag{15}$$

The same query also returns a target-conditioned prediction $\widetilde{x}_{\text{tar},e}$ and a fresh attention readout Φ_{pin}. Thresholding Φ_{pin}, dilating the result, and intersecting it with the energy support of $\widetilde{R}_e$ yields the cleanup gate. Starting from $\widetilde{x}_{\text{tar},e}$, within this gate we reinject a fraction of B as a residual correction, continuing the budgeted edit, and apply a gated local extrapolation away from x_{edit} to recover reliable but under-edited content.

Gated refinement and fusion. The cleanup candidate is further refined with a low-amplitude high-frequency source anchor and an attention-modulated micro-swap, which restore local texture and stabilize uncertain local structure. We write the refinement and its output gate as

$$(x_{\text{refined}}, \Psi) = \text{GatedRefine}\big(x_{\text{edit}}, x_{\text{src}}, m, \widetilde{x}_{\text{tar},e}, \widetilde{R}_e, \Phi_{\text{pin}};\, B\big). \tag{16}$$

Here $\Psi \in [0,1]^{H\times W}$ retains the first-pass edit support where second-pass attention admits it, and adds extensions jointly supported by the second-pass residual and attention; soft boundaries and a graded confidence ramp govern final fusion:

$$x_{\text{out}} = \Psi \odot x_{\text{refined}} + (1-\Psi) \odot x_{\text{src}}. \tag{17}$$

Table 1: **Quantitative comparison on PIE-Bench.** T-free: Training-free. I-free: Inversion-free. Best and second-best among one-step methods are highlighted in yellow and blue, ties included.

Type	Methods	Background						Edited					Properties		Efficiency			
		PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑	T-free	I-free	Step	NFE	Runtime(s)↓	VRAM(MB)↓
Multi-step (>10 steps)	SpotEdit	27.21	13.92	69.39	0.88	41.55	96.64	24.64	22.51	17.63	67.01	60.04	✓	✓	50	50	11.37	26384
	FlowEdit	22.12	8.91	104.71	0.84	26.88	95.12	26.01	22.78	10.90	69.39	63.72	✓	✓	33	33	9.39	20180
	KV-Edit	34.91	0.52	11.26	0.95	16.57	97.06	24.56	20.83	7.95	71.11	62.10	✓	✗	24	48	9.42	31618
	PnPInversion	22.32	8.29	112.77	0.79	28.11	95.03	25.42	22.53	10.43	68.82	61.17	✓	✗	50	100	15.03	9290
Few-step (2–10 steps)	TurboEdit	22.55	9.25	101.93	0.80	29.87	95.24	26.54	23.04	14.62	71.11	63.83	✓	✓	4	4	0.91	19574
	InstantEdit	28.61	2.79	45.21	0.87	12.72	97.41	24.89	21.93	13.06	67.38	60.41	✓	✗	4	8	0.84	7812
	InfEdit	28.11	6.98	56.09	0.85	16.90	97.32	24.87	22.07	9.65	67.54	60.83	✓	✓	4	4	1.29	6794
One-step (1 step)	SwiftEdit	24.22	5.41	76.39	0.82	15.14	96.34	24.48	21.54	7.56	66.52	60.05	✗	✗	1	2	0.45	19500
	ChordEdit	22.63	7.98	122.66	0.77	28.81	95.04	24.83	22.15	10.72	67.21	60.19	✓	✓	1	2	0.26	7333
	WhereEdit	24.05	7.97	108.22	0.80	31.03	94.93	25.18	22.66	10.78	64.28	56.39	✓	✓	1	3	0.29	7339
	RRLS	23.92	6.27	111.40	0.81	25.07	95.21	25.38	22.09	11.47	65.90	57.79	✓	✓	1	4	0.52	7333
	BudEdit	**24.71**	**5.54**	**78.24**	**0.82**	**20.01**	**96.17**	**25.25**	**22.52**	**12.28**	**67.77**	**60.31**	✓	✓	1	2	**0.21**	**7319**

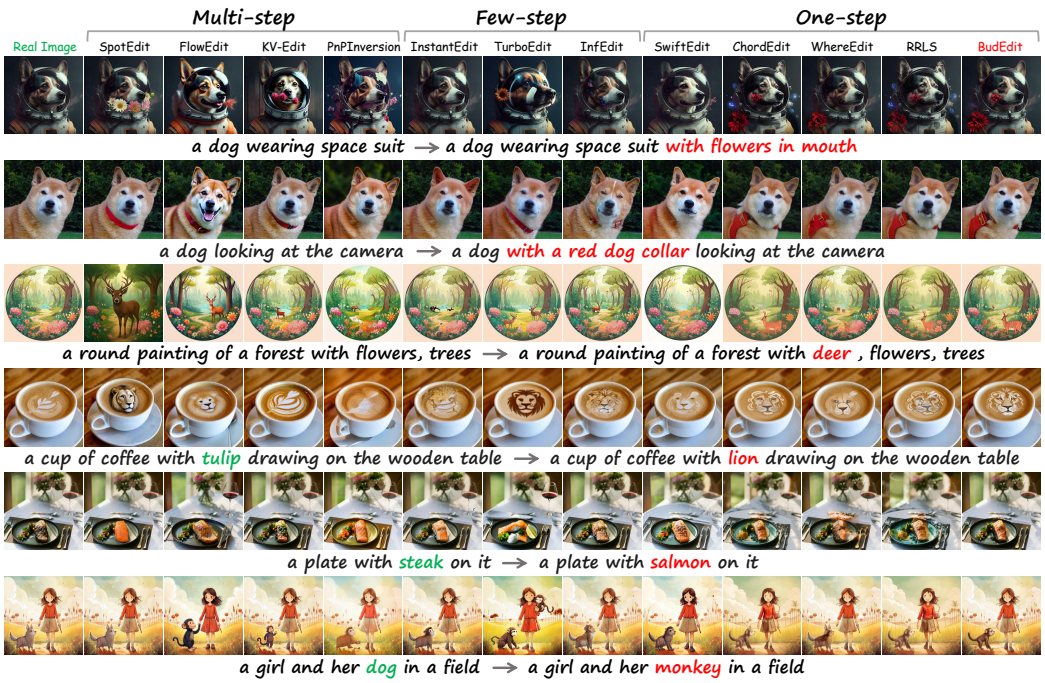

Figure 4: **Qualitative comparison with the methods of Table 1**. Several baselines exhibit background color shifts, ghosting, or structural distortion, while BudEdit better preserves the source surroundings and produces detailed edits with clean boundaries, while preserving the intended edit semantics throughout.

4 Experiments

4.1 Setup

Dataset and baselines. PIE-Bench (Ju et al., 2024): 700 images, 10 editing categories, all 512×512, seed fixed at 42 for all methods. Baselines: multi-step (SpotEdit (Qin et al., 2026), FlowEdit (Kulikov et al., 2025), KV-Edit (Zhu et al., 2025), PnPInversion (Ju et al., 2024)), few-step (TurboEdit (Deutch et al., 2024), InstantEdit (Gong et al., 2025), InfEdit (Xu et al., 2024)), and one-step (trained SwiftEdit (Nguyen et al., 2025); training-free ChordEdit (Lu et al., 2026), WhereEdit (Hu et al., 2026), RRLS (Yi et al., 2026)). All methods are evaluated using their respective reported default configurations.

Metrics. Eleven metrics, one per column of Table 1: background-fidelity (PSNR, MSE, LPIPS (Zhang et al., 2018), SSIM, DINO (Caron et al., 2021), CLIP (Radford et al., 2021)) and editing-quality (CLIP-W, CLIP-E, CLIP-D (Gal et al., 2022), MUSIQ, MUSIQ-E (Ke et al., 2021)).

Implementation. In Table 1, Step counts sampling updates, while the number of function evaluations (NFE) counts UNet forward evaluations, with the source and target conditions batched into

one evaluation. Like ChordEdit, BudEdit uses one sampling step and two UNet evaluations, with $t_s = 0.78$, $t_e = 0.38$, $\alpha = 0.7$, $\beta = 4.0$; runtimes are measured on an RTX 4090.

4.2 Main Results

Quantitative Analysis. Table 1 compares BudEdit with existing editors on PIE-Bench. Among one-step methods, BudEdit achieves the highest PSNR, CLIP-D, MUSIQ, and MUSIQ-E, while using two UNet evaluations and reporting the lowest runtime of 0.21 s per image. Among training-free one-step editors, BudEdit leads all six background-fidelity metrics of Table 1 and exceeds ChordEdit on all five editing-quality metrics; SwiftEdit reaches its background numbers with task-specific training. The few-step and multi-step methods provide a broader quality–compute reference rather than a same-budget ablation: they use 4–100 evaluations and report 0.84–15.03 s per image, compared with BudEdit's two evaluations and 0.21 s.

Qualitative Analysis. Figure 4 shows the corresponding qualitative comparison. In the lion latte-art example, BudEdit produces a clearly delineated mane while preserving the cup and table appearance of the source image; in the steak-to-salmon example it preserves the dark plate and side dishes. These examples complement Table 1: BudEdit combines strong background fidelity with the highest CLIP-D among one-step methods while retaining fine target texture and clean edit boundaries.

Comparison with Chord transport. In Table 2, we examine the role of temporal smoothing in ChordEdit. Disabling its chord window (δ=0) raises CLIP-E from 22.15 to 22.54, but lowers MUSIQ-E from 60.19 to 55.97 and background PSNR from 22.63 to 20.13 dB. Figure 5 further shows visible background and structural distortions in the window-off outputs. CLIP-E alone therefore does not fully reflect edit quality, and we include MUSIQ-E as a complementary measure of edited-region perceptual quality. Without relying on temporal smoothing, BudEdit improves all five metrics in Table 2 over ChordEdit's smoothed default. Compared with the window-off row, BudEdit gives up only 0.02 CLIP-E while improving background PSNR by 4.58 dB and MUSIQ-E by 4.34 points.

Table 2: **ChordEdit chord window ablation.** δ=0.15 is default; δ=0 disables the window.

Config	PSNR↑	MSE↓	CLIP-W↑	CLIP-E↑	MUSIQ-E↑
δ=0	20.13	13.34	25.11	**22.54**	55.97
δ=0.15	22.63	7.98	24.83	22.15	60.19
BudEdit	**24.71**	**5.54**	**25.25**	22.52	**60.31**



Figure 5: **ChordEdit window ablation.** ChordEdit with δ=0 distorts the scene; BudEdit preserves scene.

Matched effective timestep. ChordEdit's default chord window assigns six sevenths of its weight to $t_s - \delta$. With t_s=0.9 and δ=0.15, its weighted-average time is $t_s - 6\delta/7 = 0.771$, close to BudEdit's direct timestep t_s=0.78. This proximity is consistent with temporal smoothing shifting ChordEdit toward a lower effective timestep, while the mechanisms remain different: ChordEdit averages fields globally over time, whereas BudEdit evaluates directly at t_s=0.78 and controls the trade-off through spatial energy budgeting. This reading is corroborated by an independent reproduction of ChordEdit (Li et al., 2026).

4.3 Verification of the Theory's Quantities

Localization derives from the model itself. Figure 6 illustrates this model-derived localization on two examples. The GT Mask is included for visual comparison only; editing uses no external or human-annotated mask, and the Derived Mask is produced by the model itself. In the Overlap panel, green marks the common support of the Derived Mask and the GT Mask, while red and blue mark regions selected by only one of them. The appendix reports the overlap between the derived support and the annotated region over all 700 pairs.

4.4 Component Ablations

Table 3 isolates budgeted energy reallocation from second-pass cleanup. Setting β=0 drops the

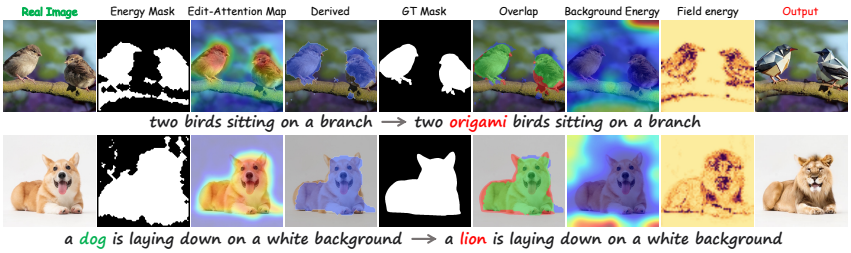

Figure 6: **Annotation-free localization.** Two edits, from birds to origami birds and from a dog to a lion. The columns show the Real Image, Energy Mask, Edit-Attention Map, Derived Mask, GT Mask, Overlap, background energy, field energy of the composed transport field, and output.

budgeted injection field $f^\star$ from Eq. 12, leaving only the attenuated residual term; disabling cleanup drops the second UNet evaluation, halving NFE. Each factor helps alone. Energy reallocation alone raises CLIP-E/CLIP-D by 0.31/1.57 without cleanup; cleanup gives the larger editing-quality gain, +1.30/+5.87 and +1.08 CLIP-W. They also interact: with cleanup on, the reallocation gains grow to 0.71/2.93, and the full configuration attains the best CLIP-W, CLIP-E and CLIP-D. Transporting less energy changes the background less, so the no-energy, no-cleanup setting holds the highest PSNR and SSIM. Cleanup recovers under-editing with an extra forward pass, whereas energy reallocation goes further within a single forward pass: it moves the measured background energy into the injection support, tightening editing strength without an additional evaluation, unlike temporal smoothing.

Table 3: **Factorial ablation of energy reallocation and cleanup.**Bold: column best.

Energy	Cleanup	PSNR↑	SSIM↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	NFE↓
✗	✗	**26.14**	**0.85**	23.51	20.51	3.48	**1**
✓	✗	25.73	0.84	23.73	20.82	5.05	**1**
✗	✓	25.92	0.83	24.59	21.81	9.35	2
✓	✓	24.71	0.82	**25.25**	**22.52**	**12.28**	2

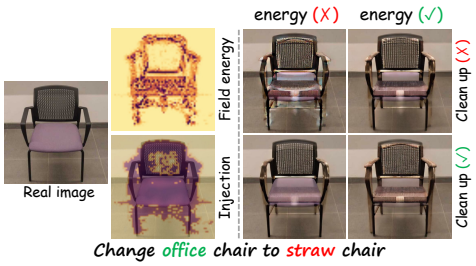

Figure 7: **Qualitative and energy analysis.** Injection: energy of $f^\star$; field energy: energy of the transport field.

Figure 7 visualizes this interaction: the injection energy map shows where $f^\star$ spends the budget, and the field energy map shows the resulting field of Eq. 12. For the *office chair* → *straw chair* edit, budgeted energy reallocation alone overlays straw texture on the cushion, cleanup alone yields only a slight change to the armrests, and only the full configuration converts the chair coherently.

4.5 Cross-Dataset and Cross-Backbone Analysis

Table 4 crosses two datasets, PIE-Bench (Ju et al., 2024) and the multi-aspect PIE-Bench++ of ParallelEdits (Huang et al., 2024), with two one-step backbones, SD-Turbo (Sauer et al., 2024) and SwiftBrushV2 (Dao et al., 2024), using the same hyperparameters throughout. BudEdit achieves the best background preservation and editing quality across all four settings, improving background PSNR by over 2 dB. The appendix reports all eleven metrics.

Table 4: **Quantitative comparison on different datasets and backbones.** Bold: column best.

Dataset	Backbone	PSNR↑		CLIP-E↑	
		ChordEdit	**BudEdit**	ChordEdit	**BudEdit**
PIE-Bench	SD-Turbo	22.63	**24.71**	22.15	**22.52**
	SwiftBrushV2	22.93	**25.49**	21.18	**21.26**
PIE-Bench++	SD-Turbo	22.87	**25.06**	24.12	**24.27**
	SwiftBrushV2	23.16	**25.83**	22.46	**22.70**

5 Conclusion

BudEdit addresses spatially misallocated updates in one-step diffusion editing by converting single-draw background residual energy into an image-dependent budget and re-injecting it into the regions selected jointly by residual energy and cross-attention, with exact injection-budget matching and zero injection on the operational background support. On PIE-Bench, it outperforms the temporal-smoothing approach of ChordEdit on all 11 reported metrics, including a 2.1 dB PSNR gain, at the lowest runtime in the comparison. Together, these results establish explicit spatial energy budgeting as an alternative to temporal smoothing for stable, efficient, controllable one-step editing.

Ethics Statement

This work studies text-guided image editing using two public benchmarks, PIE-Bench (Ju et al., 2024) and PIE-Bench++ (Huang et al., 2024), and publicly released pre-trained backbones; no new data are collected. The human preference study in the appendix was a voluntary, anonymous comparison in which participants judged only editing outcomes, with no personal information recorded. As with any image-editing method, BudEdit could be misused to alter photographs misleadingly; we regard provenance and watermarking tooling as the appropriate mitigation.

Reproducibility Statement

BudEdit requires no training or per-image optimization; reproducing our results requires only the released backbones and the procedure described here. Section 3.3 and Section 3.5 define the method, Algorithm 1 summarizes the inference flow, and Section 4.1 lists the global hyperparameters and evaluation hardware. The appendix provides additional construction details, exact metric definitions, and sensitivity analyses for the global hyperparameters.

AI Use Statement

We used a general-purpose language-model assistant for surface-level language editing: checking grammar, spelling, and punctuation of author-written text and flagging unclear phrasing for the authors to rewrite. This falls under the recommended-disclosure category of editing a paper to improve readability. We did not use generative AI tools for any required-disclosure task: the problem formulation, theoretical claims and proofs, implementation, experimental design, and interpretation of results are entirely the authors' own. No synthetic data were generated and no qualitative data analysis was performed by the language-model assistant. Every suggested language edit was reviewed by an author, and we take full responsibility for the final content of this paper.

References

Jean-David Benamou and Yann Brenier. A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393, 2000. doi: 10.1007/s002110050002.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. InstructPix2Pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18392–18402, 2023.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9650–9660, 2021.

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-Excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics*, 42(4), 2023. doi: 10.1145/3592116.

Trung Dao, Thuan Hoang Nguyen, Thanh Le, Duc Vu, Khoi Nguyen, Cuong Pham, and Anh Tran. SwiftBrush v2: Make your one-step diffusion model better than its teacher. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 176–192, 2024.

Gilad Deutch, Rinon Gal, Daniel Garibi, Or Patashnik, and Daniel Cohen-Or. TurboEdit: Text-based image editing using few-step diffusion models. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–12. ACM, 2024. doi: 10.1145/3680528.3687612.

Rinon Gal, Or Patashnik, Haggai Maron, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. StyleGAN-NADA: CLIP-guided domain adaptation of image generators. *ACM Transactions on Graphics*, 41(4):1–13, jul 2022. doi: 10.1145/3528223.3530164.

Yiming Gong, Zhen Zhu, and Minjia Zhang. InstantEdit: Text-guided few-step image editing with piecewise rectified flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16808–16817, 2025.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross-attention control. In *International Conference on Learning Representations (ICLR)*, 2023.

Ming Hu, Mingyu Dou, Jianfu Yin, Miaomiao Zhang, Cong Hu, Yao Wang, Bingliang Hu, and Quan Wang. WhereEdit: Mask-aware local latent editing for one-step image editing. *arXiv preprint arXiv:2607.20883*, 2026.

Mingzhen Huang, Jialing Cai, Shan Jia, Vishnu Suresh Lokhande, and Siwei Lyu. ParallelEdits: Efficient multi-aspect text-driven image editing with attention grouping. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 22569–22595, 2024. doi: 10.52202/079017-0710.

Yilei Jiang, Zhen Wang, Yanghao Wang, Jun Yu, Yueting Zhuang, Jun Xiao, and Long Chen. FlowDC: Flow-based decoupling-decay for complex image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25757–25766, 2026.

Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. PnP inversion: Boosting diffusion-based editing with 3 lines of code. In *International Conference on Learning Representations (ICLR)*, 2024.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 5148–5157, 2021.

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.

Minghan Li, Jeremy Moebel, and Mengyu Wang. Rethinking one-step image editing through chordedit: Reproduction, simplification, and new insights. *arXiv preprint arXiv:2606.14042*, 2026.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. InstaFlow: One step is enough for high-quality diffusion-based text-to-image generation. In *International Conference on Learning Representations (ICLR)*, 2024.

Liangsi Lu, Xuhang Chen, Minzhe Guo, Shichu Li, Jingchao Wang, and Yang Shi. ChordEdit: One-step low-energy transport for image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14398–14407, 2026.

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.

Thuan Hoang Nguyen and Anh Tran. SwiftBrush: One-step text-to-image diffusion model with variational score distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7807–7816, 2024.

Trong-Tung Nguyen, Quang Nguyen, Khoi Nguyen, Anh Tran, and Cuong Pham. SwiftEdit: Lightning fast text-guided image editing via one-step diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21492–21501, 2025.

Zhibin Qin, Zhenxiong Tan, Zeqing Wang, Songhua Liu, and Xinchao Wang. SpotEdit: Selective region editing in diffusion transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18683–18692, 2026.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 87–103, 2024.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021a.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021b.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32211–32252. PMLR, 2023.

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

Sihan Xu, Yidong Huang, Jiayi Pan, Ziqiao Ma, and Joyce Chai. Inversion-free image editing with language-guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9452–9461, 2024.

Hongzhu Yi, Zhongtian Luo, Tong Li, Yiyan Fan, and Jungang Xu. Bridging the manifold gap: Riemannian residual line search for one-step image editing. *arXiv preprint arXiv:2606.24844*, 2026.

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 (CVPR)*, pp. 586–595, 2018.

Tianrui Zhu, Shiyi Zhang, Jiawei Shao, and Yansong Tang. KV-Edit: Training-free image editing for precise background preservation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16607–16617, 2025.

SUPPLEMENTARY MATERIAL

A IMPLEMENTATION DETAILS

This section summarizes the construction rules that Section 3 names but does not expand on: the energy support and first-pass masks, the second-pass gated update, the structure-restoration operations, and the pin mask Ψ. It also specifies the measurement protocol used throughout the experiments. The four global hyperparameters t_s, t_e, α, and β are given in Section 4.1 and swept in Appendices F–I, so they are not repeated below.

Every coefficient in this section is fixed globally. Beyond the swept hyperparameters t_s, t_e, α, and β, these constants were fixed once on a development subset of PIE-Bench and then frozen: the same values are used for PIE-Bench++, for SwiftBrushV2, and for every ablation and sensitivity run reported here, with no per-setting retuning.

Energy support and first-pass masks. The residual-energy support $\widehat{\mathcal{E}}$ collects the pixels whose channel-wise residual energy exceeds an energy threshold relative to the per-image maximum of $\|\widehat{R}(i,j)\|_C^2$, followed by one dilation with an all-ones 3×3 filter. The thresholding rule is fixed across experiments and applied per image to the first-pass residual. The soft mask m equals 1 on the thresholded support and 0.5 on the added one-pixel ring, and the smoothed attention readout $w = \mathrm{blur}(\Phi)$ uses a Gaussian of scale 1.5 with a 7×7 kernel.

Differing tokens. The two prompts are tokenized to the same length and compared as token-id sequences; the differing set is the union of the tokens inserted or replaced on the target side and those deleted or replaced on the source side, with padding and boundary tokens excluded. Each side yields one map, normalized to its own peak before the pixelwise maximum, so Φ records how strongly a pixel is attended by any differing token rather than the difference of two attention magnitudes. When one side is empty, as in a pure insertion, the maximum reduces to the single remaining map; when no differing token survives, all valid target tokens are used.

Target-conditioned prediction. $\widehat{x}_{\mathrm{tar}}$ and $\widetilde{x}_{\mathrm{tar},e}$ are the target-conditioned latent estimates obtained from the corresponding model outputs using the backbone's standard prediction-to-latent conversion at the queried level; no additional network evaluation is involved. Only the displacement $\widehat{x}_{\mathrm{tar}} - x_{\mathrm{src}}$ enters the first-pass construction, through the indicator of Eq. 5 and the normalization of Eq. 8, while $\widetilde{x}_{\mathrm{tar},e}$ serves as the starting point of the second-pass refinement.

Second-pass gated update (Section 3.5). The second query returns a target-conditioned prediction $\widetilde{x}_{\mathrm{tar},e}$, a residual $\widetilde{R}_e$, and an attention readout Φ_{pin}. The cleanup gate intersects an energy support derived from $\widetilde{R}_e$, constructed analogously to $\widehat{\mathcal{E}}$, with the second-pass attention support. Starting from $\widetilde{x}_{\mathrm{tar},e}$, a residual correction along the normalized second-pass residual is scaled by a fixed fraction of the budget within this gate, followed by gated local extrapolation away from x_{edit} to strengthen under-edited updates. The cleanup gate applies only to these two updates; later structure restoration is written back through Ψ in Eq. 17.

Structure restoration (Section 3.5). The cleanup candidate is supplemented with a low-amplitude high-frequency source anchor that restores texture attenuated by transport. A Φ_{pin}-modulated micro-swap further restores uncertain local detail. These operations form the refined candidate x_{refined} before final source fusion.

Pin mask Ψ (Section 3.5). $\Psi \in [0,1]^{H\times W}$. Its support retains the pixels with $m(i,j) > \frac{1}{2}$ that the dilated second-pass attention support admits, takes the maximum with the extension admitted jointly by $\widetilde{R}_e$ and that attention support, and is then given a one-latent-pixel soft boundary and weighted by a coverage-adaptive confidence ramp. Unsupported locations therefore remain blended with the source image.

Measurement protocol. All timings and metrics are computed at 512×512 with centre cropping, in `float32`, one image per forward pass with the source and target conditions batched into that

single evaluation as defined in Section 4.1, on a single GPU; the noise seed is fixed at 42 for every method, as stated in Section 4.1. We selected this common seed in a preliminary comparison because it gave consistently strong aggregate performance across the evaluated one-step editors. To make the remaining seed variation explicit, Appendix B reports the mean and standard deviation over seeds 41, 42, and 43 for the one-step comparison. The reported per-image runtime is wall-clock time for the complete edit, including both UNet evaluations, mask construction, and VAE decoding.

B Multi-seed Evaluation and Seed Selection

Table A1 evaluates all one-step methods under the same three noise seeds, 41, 42, and 43. The results show that BudEdit retains the strongest multi-seed mean in PSNR, CLIP-D, MUSIQ, and MUSIQ-E, while the other editors retain their expected strengths on individual preservation or semantic metrics. The main comparison fixes seed 42 for every method, rather than choosing a separate favorable seed per method, and the multi-seed summary makes the resulting variation visible.

Table A1: **Multi-seed one-step comparison on PIE-Bench.** Values are mean±standard deviation over noise seeds 41, 42, and 43, evaluated on the full 700-image set; as in Table 1, the five mask-restricted background metrics use the 556-image subset defined in Appendix D. The main Table 1 reports the common seed 42 for every method; no method receives a separately selected seed. Metric scales and decimal precision follow Table 1, including background CLIP on the $\times 100$ scale. Best and second-best displayed means are highlighted in yellow and blue, ties included.

Method	Background						Edited				
	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
SwiftEdit	24.22±0.00	5.41±0.00	76.38±0.02	0.82±0.00	15.13±0.01	96.34±0.01	24.48±0.00	21.55±0.01	7.58±0.01	66.52±0.00	60.05±0.01
ChordEdit	22.40±0.21	8.39±0.36	125.94±3.20	0.77±0.00	29.73±0.98	94.94±0.09	24.81±0.02	22.18±0.04	10.86±0.18	67.52±0.33	60.17±0.28
WhereEdit	23.90±0.17	8.06±0.10	110.10±2.03	0.80±0.00	31.24±0.20	94.89±0.06	25.17±0.03	22.65±0.09	10.81±0.07	64.00±0.26	56.18±0.18
RRLS	23.61±0.27	6.59±0.38	115.22±3.57	0.81±0.00	25.83±0.84	95.09±0.11	25.39±0.02	22.20±0.11	11.53±0.12	65.52±0.41	57.47±0.35
BudEdit	**24.46±0.25**	**5.78±0.29**	**81.08±3.12**	**0.82±0.00**	**20.64±0.64**	**96.11±0.06**	**25.21±0.07**	**22.51±0.06**	**12.34±0.22**	**67.64±0.15**	**60.32±0.14**

C Datasets

PIE-Bench. PIE-Bench (Ju et al., 2024) contains 700 images covering ten editing categories: random (140 images), change object (80), add object (80), delete object (80), change content (40), change pose (40), change color (40), change material (40), change background (80), and change style (80). Within the nine non-random categories the images are split evenly between natural photographs and artificial scenes, the latter being human-made artwork rather than generated imagery, and within each of those groups evenly across animal, human, indoor, and outdoor content. Every image carries five annotations: a source prompt, a target prompt, an editing instruction, the edited word, and a hand-annotated mask of the region the edit is expected to occupy. All mask-restricted background metrics in this paper are computed against that hand-annotated mask, so the preservation numbers are measured on a region the benchmark defines rather than one the method chooses.

PIE-Bench++. PIE-Bench++ (Huang et al., 2024) rebuilds the same 700 images around *multi-aspect* editing, where one instruction changes several attributes or objects at once. It keeps the same ten categories as PIE-Bench – random, plus change object, content, pose, color, material, background, and style, together with add object and delete object – and re-annotates the prompts so that 24% of the targets change a single aspect, 57% change two, and 19% change more than two. Each item carries a source prompt, a target prompt, an edit action recording the position in the source prompt together with the edit type and the operation (swap, add, or delete), and a mapping from each edited object to the attributes changed on it.

The second dataset is the harder of the two for the mechanism studied here: a multi-aspect instruction spreads the edit over several disjoint regions, so the support that the energy measurement has to recover is fragmented rather than compact, and a single mask covers proportionally more of the frame. The cross-dataset comparison in Section 4.5 therefore tests whether spatial energy budgeting

survives a change in how the editing energy is distributed over the image, not only a change in image content.

D Evaluation Protocol and Metric Definitions

All eleven metrics are computed by a single evaluation script, so every method in this paper – baselines included – is measured under one implementation rather than by copying numbers across papers. Every metric below operates in image space: wherever x_{src} or x_{out} appears in this section it denotes the decoded image, not the latent of the same name in Section 3. Let $M_{\mathrm{gt}} \in \{0,1\}^{512\times512}$ be the hand-annotated PIE-Bench editing mask, x_{src} the source image, and x_{out} the edited output. Unless specified otherwise in Table A2, a region-restricted full-reference metric multiplies the source and output by the region indicator and evaluates them without cropping. Background CLIP instead pools patch tokens outside the edit mask. Five metrics restrict to the background, $1 - M_{\mathrm{gt}}$ (PSNR, MSE, LPIPS, SSIM, and CLIP), and two restrict to the edited region, M_{gt} (CLIP-E and MUSIQ-E); the remaining four (DINO, CLIP-W, CLIP-D, and MUSIQ) are evaluated on the full image without region masking. The Background/Edited column groups of Table 1 in the main paper reflect what each metric is meant to measure, not which pixels it reads. This mask is used for evaluation only; the editing method never reads it. Table A2 lists the region, computation, and reported scale for each metric.

Table A2: **Metric definitions.** Region, computation, and reported scale for each of the eleven metrics of Table 1. Arrows give the direction of improvement.

Metric	Region	Computation	Scale
PSNR↑	$1 - M_{\mathrm{gt}}$	Peak signal-to-noise ratio, data range 1	—
MSE↓	$1 - M_{\mathrm{gt}}$	Mean squared error on $[0, 1]$ intensities	$\times10^3$
LPIPS↓	$1 - M_{\mathrm{gt}}$	Learned perceptual patch similarity, SqueezeNet backbone	$\times10^3$
SSIM↑	$1 - M_{\mathrm{gt}}$	Structural similarity, data range 1	—
DINO↓	whole	Mean squared error between the patch-token cosine self-similarity matrices of the last DINO ViT-B/8 block, computed on both images resized to 224 without cropping	$\times10^3$
CLIP↑	$1 - M_{\mathrm{gt}}$	CLIP ViT-L/14 patch tokens pooled outside the mask; cosine between the pooled source and output features	$\times10^2$
CLIP-W↑	whole	Cosine between the CLIP image embedding and the target-prompt text embedding	$\times10^2$
CLIP-E↑	M_{gt}	Same as CLIP-W, with the background zeroed on the image side	$\times10^2$
CLIP-D↑	whole	Cosine between the normalized image-side and text-side difference vectors, Eq. 18	$\times10^2$
MUSIQ↑	whole	No-reference image quality assessment	—
MUSIQ-E↑	M_{gt}	MUSIQ applied to the edited-region bounding-box crop	—

CLIP-D is the bi-directional CLIP score: it asks whether the change the method made points the way the instruction asked, rather than whether the output merely resembles the target prompt. Writing g_{img} and g_{txt} for the CLIP image and text encoders and c_{src}, c_{tar} for the two prompts,

$$\text{CLIP-D} = \cos\left(\frac{g_{\mathrm{img}}(x_{\mathrm{out}}) - g_{\mathrm{img}}(x_{\mathrm{src}})}{\|g_{\mathrm{img}}(x_{\mathrm{out}}) - g_{\mathrm{img}}(x_{\mathrm{src}})\|}, \; \frac{g_{\mathrm{txt}}(c_{\mathrm{tar}}) - g_{\mathrm{txt}}(c_{\mathrm{src}})}{\|g_{\mathrm{txt}}(c_{\mathrm{tar}}) - g_{\mathrm{txt}}(c_{\mathrm{src}})\|} \right). \tag{18}$$

If either difference vector has zero norm, we define CLIP-D as zero; otherwise Eq. 18 applies. This is why CLIP-D separates methods that CLIP-E cannot: an output that leaves the source almost unchanged can still score well on CLIP-E if the source already partly matches the target prompt, but it has no displacement to align, so its CLIP-D collapses.

Sample counts. The five background metrics require a non-empty background region. Exclusion is decided per image, by whether the annotated mask covers the entire frame; this is most common in the style and background categories, where no background is left to preserve, but it is not confined to them. On PIE-Bench this excludes 144 of the 700 images, leaving 556 for PSNR, MSE, LPIPS, SSIM, and CLIP: 78 of 80 change-style, 27 of 80 add-object, 19 of 140 random, 10 of 80 change-background, 5 of 40 change-pose, 4 of 40 change-content, and 1 of 40 change-color images, with no exclusions among change-object, delete-object, or change-material. On PIE-Bench++, which carries the same ten categories, the same rule excludes 529 images and leaves 171; the largest contributors are change-style (80 of 80), random (127 of 140), change-object and add-object (78 of 80 each), and change-color (38 of 40), with the remaining 128 spread over delete-object, material, pose, content, and change-background. The six remaining metrics use all 700 images on both benchmarks. This split is reported by the evaluation script and is identical for every method.

Scaling conventions. PSNR, SSIM, MUSIQ, and MUSIQ-E are reported unscaled; MSE, LPIPS, and DINO are multiplied by 10^3; the four CLIP-based scores by 10^2. SSIM is on the $[0, 1]$ scale. These conventions differ from the ones used in the original PIE-Bench tables, which report MSE at $\times 10^4$ and SSIM at $\times 10^2$, so values here should be compared within this paper rather than against numbers transcribed from elsewhere.

Qualitative background consistency. Figure A1 provides a visual counterpart to the mask-restricted background metrics. In the highlighted non-edited regions, BudEdit preserves the source appearance with limited color drift and texture disruption while completing the requested edit. Notably, despite using a one-step transport, BudEdit maintains visually coherent backgrounds alongside the multi-step DiT-based editors SpotEdit and KV-Edit in these examples.

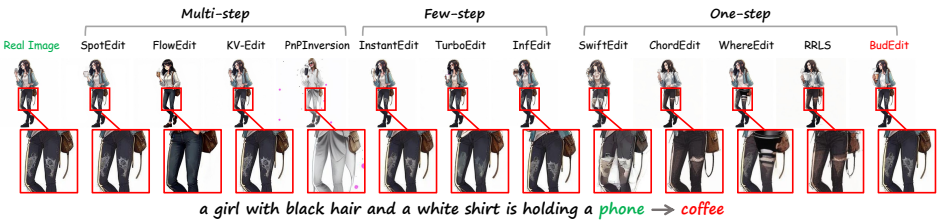

Figure A1: **Qualitative background-preservation comparison.** The first column is the Real Image. Red boxes and enlarged crops highlight a non-edited region while the instruction changes "phone" to "coffee." BudEdit (Ours) preserves the source appearance in the highlighted region while completing the edit.

E Human Preference Study

Automatic metrics quantify complementary aspects of one-step editing, including background similarity, semantic alignment, and perceptual quality, but they do not fully capture which results human observers prefer. As shown in Figure A2, we compared the four training-free one-step editors of Table 1 of the main paper (BudEdit, ChordEdit, WhereEdit, and RRLS) in a blind preference study with 20 participants on 20 randomly sampled PIE-Bench cases. Each participant evaluated the same 20 cases. For each case, participants saw the source image, target instruction, and four anonymized outputs; method identities were hidden and output positions were randomized.

For each case, participants selected one output under three criteria: *instruction faithfulness*, which measures whether the target edit is correctly realized; *background preservation*, which measures the fidelity of the non-edited regions relative to the source image; and *editing quality*, which measures the visual coherence and naturalness of the edited region. Each criterion therefore contains $20 \times 20 = 400$ valid choices.

BudEdit receives the most votes across all three criteria: $140/400$ votes for instruction faithfulness, $193/400$ votes for background preservation, and $157/400$ votes for editing quality. The largest margin is in background preservation, where BudEdit receives 48.25% of the votes compared with 31.00% for WhereEdit. BudEdit also leads ChordEdit by 9.75 percentage points in instruction faithfulness and 12.75 percentage points in editing quality. This vote ranking is consistent with the

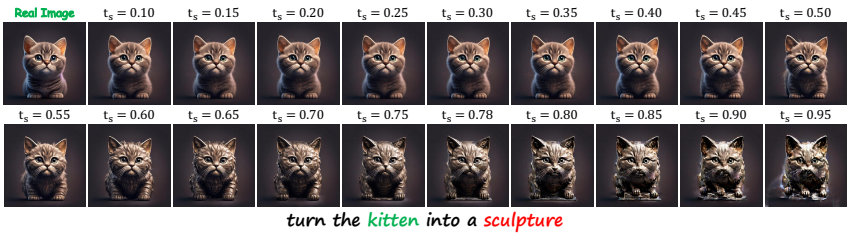

Figure A3: **Qualitative sensitivity to the transport timestep t_s.** Outputs across the same t_s sweep as Table A3, with all other hyperparameters fixed.

background-preservation and editing metrics reported in the main paper, and is visualized in Figure A2.

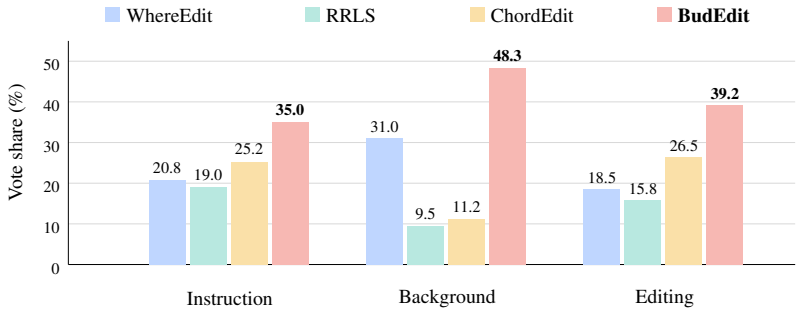

Figure A2: **Human preference study.** Twenty participants evaluate 20 randomly sampled PIE-Bench cases in a four-way blind comparison. Bars show vote share; each criterion aggregates 400 choices.

F Sensitivity to the Transport Timestep t_s

Table A3 varies the transport timestep t_s from 0.10 to 0.95 in increments of 0.05 and includes the default t_s=0.78 as an additional setting, while fixing $(t_e, \alpha, \beta) = (0.38, 0.7, 4.0)$. The table shows a preservation–editing trade-off. Edit-direction similarity, measured by CLIP-D, rises overall from 2.95 and peaks at 13.24, while DINO, a background-fidelity metric with a lower-is-better direction, reaches 28.47 at the largest timestep. In contrast, background fidelity is strongest at an intermediate timestep: PSNR peaks at 27.90 dB for t_s=0.20 and then decreases to 23.13 dB at t_s=0.95, while LPIPS increases from its minimum of 50.70 at t_s=0.25 to 98.07. The edit gain is already close to saturation at large timesteps: CLIP-D increases by 1.39 points from 0.70 to 0.78, but changes by only 0.18 points from 0.85 to 0.95, whereas PSNR drops by 0.85 dB over the same interval. The default t_s=0.78 provides a balanced operating point: larger timesteps offer only limited additional directional gain while incurring a continuing cost in background fidelity. Figure A3 shows the corresponding qualitative outputs.

G Sensitivity to the Budget Multiplier β

Table A4 sweeps the budget multiplier $\beta \in \{0, 1, \ldots, 8\}$ with $(t_s, t_e, \alpha) = (0.78, 0.38, 0.7)$ fixed. Increasing β moves the operating point from preservation toward editing: PSNR decreases from 25.92 to 24.18 dB and LPIPS increases from 65.62 to 84.04, while CLIP-D increases from 9.35 to 12.75 and CLIP-E from 21.81 to 22.56. In contrast, the edited-region perceptual score MUSIQ-E decreases from 60.67 to 59.10, reflecting a cost in local perceptual quality as more budget is allocated to editing. The directional editing gain is strongly front-loaded: the first increment adds 1.80 CLIP-D points, whereas the final increment adds only 0.09. Beyond β=4, CLIP-D increases by just 0.47 points in total, but PSNR decreases by another 0.53 dB and LPIPS increases by 5.80. At the default β=4, the table already captures most of the measured CLIP-D gain before higher budgets incur a larger cost in background fidelity. Figure A4 shows the corresponding qualitative outputs, and Figure A5 plots the marginal trade-off across the sweep. The latter makes the post-β=4 knee

Table A3: **Sensitivity to the transport timestep t_s.** All other hyperparameters fixed at $(t_e, \alpha, \beta) = (0.38, 0.7, 4.0)$. The default used throughout the main paper, t_s=0.78, is shaded.

t_s	Background						Edited				
	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
0.10	27.64	2.99	53.49	0.85	9.05	98.12	23.14	20.50	2.95	69.74	62.57
0.15	27.83	2.87	51.85	0.85	8.73	98.10	23.19	20.55	3.21	69.69	62.44
0.20	27.90	2.84	51.08	0.85	8.49	98.17	23.17	20.55	3.48	69.63	62.38
0.25	27.87	2.83	50.70	0.85	8.42	98.11	23.26	20.58	4.06	69.65	62.45
0.30	27.84	2.83	50.75	0.86	8.78	98.10	23.37	20.68	4.72	69.66	62.39
0.35	27.71	2.92	51.16	0.85	8.76	97.98	23.40	20.80	5.18	69.60	62.32
0.40	27.47	3.04	52.71	0.85	9.37	97.84	23.62	20.96	6.04	69.56	62.19
0.45	27.26	3.21	54.57	0.85	10.07	97.61	23.79	21.11	6.64	69.46	62.12
0.50	26.95	3.42	56.87	0.85	10.98	97.50	23.92	21.19	7.41	69.37	61.93
0.55	26.68	3.65	59.53	0.84	11.88	97.21	24.13	21.52	8.44	69.22	61.71
0.60	26.32	3.93	62.74	0.84	13.25	97.02	24.38	21.69	9.22	69.03	61.33
0.65	25.97	4.28	66.23	0.84	14.78	96.70	24.62	21.95	10.29	68.82	61.15
0.70	25.39	4.74	71.03	0.83	16.92	96.41	24.83	22.08	10.89	68.45	60.60
0.75	24.97	5.24	75.54	0.82	18.85	96.26	25.09	22.36	11.76	68.03	60.42
0.78	24.71	5.54	78.24	0.82	20.01	96.17	25.25	22.52	12.28	67.77	60.31
0.80	24.50	5.81	80.66	0.82	21.04	96.01	25.26	22.53	12.46	67.50	59.86
0.85	23.98	6.49	86.71	0.82	23.62	95.60	25.30	22.55	12.92	66.84	58.74
0.90	23.37	7.33	94.65	0.81	27.10	95.30	25.37	22.57	13.24	65.94	57.78
0.95	23.13	7.60	98.07	0.81	28.47	95.22	25.33	22.50	13.10	65.52	57.24

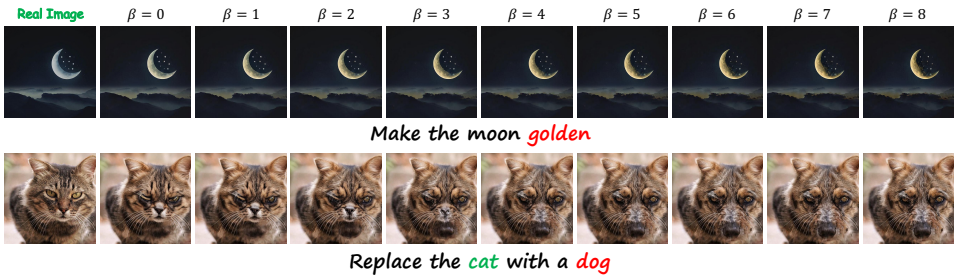

Figure A4: **Qualitative sensitivity to the budget multiplier β.** Outputs across the same β sweep as Table A4, with the instruction and other hyperparameters fixed.

explicit: CLIP-E rises by only 0.04 points from β=4 to 8, whereas PSNR falls by 0.53 dB. Together with the 0.47-point CLIP-D gain in Table A4, this shows that larger budgets yield diminishing editing returns while background preservation continues to deteriorate.

Table A4: **Sensitivity to the budget multiplier β.** All other hyperparameters fixed at $(t_s, t_e, \alpha) = (0.78, 0.38, 0.7)$. The default β=4 used throughout the main paper is shaded.

β	Background						Edited				
	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
0	25.92	4.28	65.62	0.83	14.81	97.04	24.59	21.81	9.35	68.47	60.67
1	25.34	4.83	71.62	0.83	17.20	96.50	24.94	22.22	11.15	68.16	60.61
2	25.05	5.10	74.41	0.83	18.34	96.29	25.04	22.29	11.70	68.02	60.51
3	24.86	5.32	76.49	0.83	19.24	96.15	25.17	22.41	11.98	67.86	60.43
4	24.71	5.54	78.24	0.82	20.01	96.17	25.25	22.52	12.28	67.77	60.31
5	24.56	5.68	79.96	0.82	20.71	95.91	25.27	22.54	12.39	67.60	59.79
6	24.41	5.85	81.48	0.82	21.35	95.83	25.28	22.55	12.55	67.52	59.34
7	24.29	6.00	82.83	0.82	21.92	95.76	25.31	22.57	12.66	67.42	59.20
8	24.18	6.15	84.04	0.82	22.44	95.69	25.30	22.56	12.75	67.31	59.10

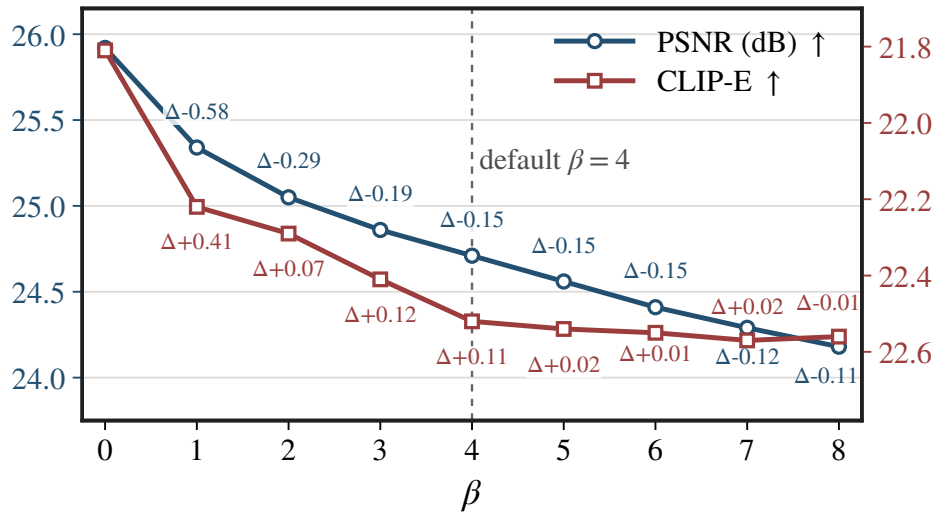

Figure A5: **Marginal budget sweep.** Edit gain saturates by β=4, while background cost persists. Same sweep as Table A4.

H Sensitivity to the Residual Attenuation α

Table A5 sweeps the residual attenuation α from 0.0 to 1.2 in increments of 0.1, while fixing the remaining hyperparameters at $(t_s, t_e, \beta) = (0.78, 0.38, 4.0)$. The table shows a preservation–editing trade-off. As α increases from 0.0 to 1.2, PSNR decreases from 26.30 to 23.44 dB, LPIPS increases from 63.46 to 91.13, and DINO increases from 12.12 to 28.74, indicating progressively larger changes in the background. At the same time, the editing-direction score CLIP-D rises from 6.44 to 12.98, while CLIP-E increases from 21.27 to 22.63. These gains saturate after the default α=0.7: increasing α from 0.7 to 1.2 adds only 0.70 CLIP-D points but costs 1.27 dB in PSNR and increases LPIPS by 12.89. MUSIQ-E decreases smoothly with α, indicating an increasing local perceptual-quality cost. At the default α=0.7, the table captures most of the directional editing gain without the larger preservation cost observed at higher attenuation values. Figure A6 shows the corresponding qualitative outputs, and Table A5 gives the same sweep numerically: the background measures fall throughout, while CLIP-D flattens just past the default, which is the 0.70-point gain over the last five settings quantified above.

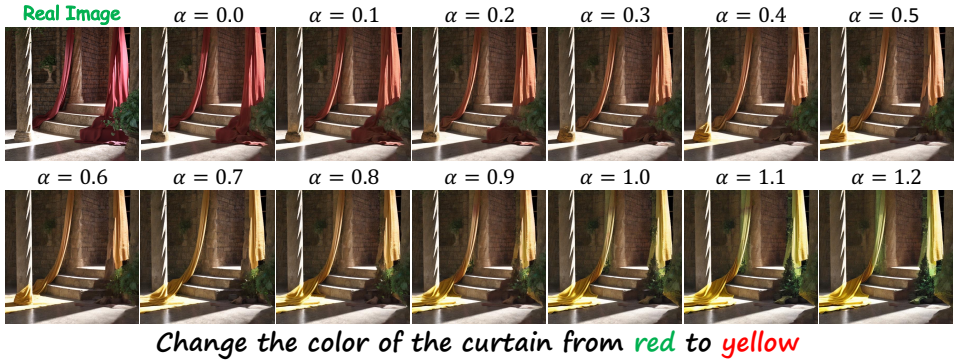

Figure A6: **Qualitative sensitivity to the residual attenuation α.** Outputs across the same α sweep as Table A5, with all other hyperparameters fixed.

Table A5: **Sensitivity to the residual attenuation α.** All other hyperparameters fixed at $(t_s, t_e, \beta) = (0.78, 0.38, 4.0)$. The default used throughout the main paper, α=0.7, is shaded.

α	Background						Edited				
	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
0.0	26.30	3.91	63.46	0.84	12.12	97.43	23.86	21.27	6.44	69.32	61.76
0.1	26.22	3.97	64.30	0.83	12.70	97.30	24.05	21.44	7.36	69.19	61.59
0.2	26.02	4.13	65.95	0.83	13.39	97.10	24.33	21.67	8.56	69.07	61.44
0.3	25.80	4.34	68.04	0.83	14.40	96.92	24.56	21.86	9.56	68.83	61.24
0.4	25.55	4.57	70.27	0.83	15.60	96.69	24.78	22.06	10.55	68.61	60.92
0.5	25.24	4.86	73.20	0.83	17.04	96.43	24.99	22.21	11.22	68.36	60.55
0.6	24.96	5.17	75.90	0.83	18.70	96.22	25.14	22.34	11.83	68.09	60.43
0.7	24.71	5.54	78.24	0.82	20.01	96.17	25.25	22.52	12.28	67.77	60.31
0.8	24.44	5.86	80.81	0.82	21.54	95.88	25.27	22.54	12.48	67.42	59.84
0.9	24.17	6.23	83.29	0.82	23.33	95.71	25.30	22.57	12.67	67.12	59.37
1.0	23.91	6.65	86.16	0.82	25.01	95.58	25.33	22.59	12.85	66.77	58.91
1.1	23.69	7.04	88.52	0.81	26.69	95.45	25.36	22.62	12.93	66.37	58.44
1.2	23.44	7.44	91.13	0.81	28.74	95.34	25.41	22.63	12.98	65.81	57.97

I Sensitivity to the Refinement Timestep t_e

Table A6 varies the refinement timestep t_e from 0.05 to 0.60, while fixing $(t_s, \alpha, \beta) = (0.78, 0.7, 4.0)$. The sweep is most favorable for background fidelity around t_e=0.10: PSNR peaks at 25.93 dB and LPIPS reaches 61.52, before both metrics progressively deteriorate toward 22.52 dB and 106.02 at t_e=0.60. DINO follows the same pattern, increasing from its minimum of 14.29 to 31.15. In parallel, CLIP-E increases from 21.25 to 22.95 and CLIP-D from 8.60 to 12.85. The default t_e=0.38 provides a balanced operating point between background preservation and edited-region alignment; larger values yield only limited additional directional gains at a substantially higher preservation cost. Figure A7 shows the corresponding qualitative outputs. Table A6 is where the two branches of the trade-off separate cleanly: every background measure deteriorates monotonically once t_e passes its 0.10 optimum, while CLIP-W and CLIP-E keep rising past the default and CLIP-D levels off at 12.87 near t_e=0.55, so the chosen point is selected for the balance rather than for either branch on its own.

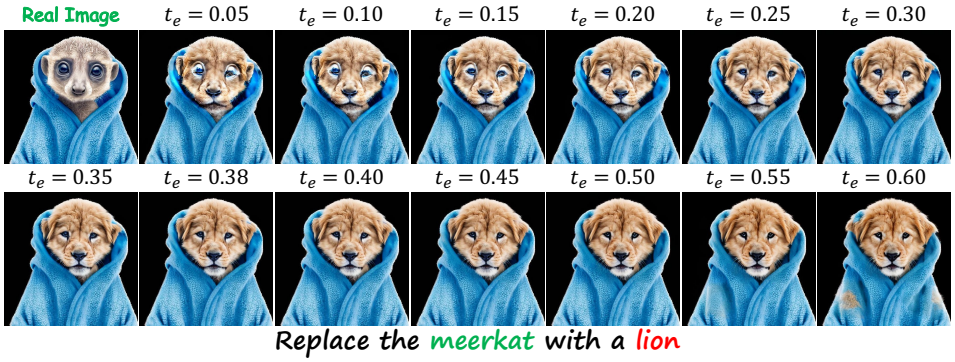

Figure A7: **Qualitative sensitivity to the refinement timestep t_e.** Outputs across the same t_e sweep as Table A6, with all other hyperparameters fixed.

Table A6: **Sensitivity to the refinement timestep t_e.** All other hyperparameters fixed at $(t_s, \alpha, \beta) = (0.78, 0.7, 4.0)$. The default t_e=0.38 is shaded.

t_e	Background						Edited				
	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
0.05	25.84	4.31	62.34	0.84	14.77	97.15	24.26	21.25	8.60	65.22	56.78
0.10	25.93	4.24	61.52	0.84	14.29	97.19	24.34	21.40	9.19	66.42	58.01
0.15	25.91	4.26	62.43	0.84	14.39	97.11	24.43	21.53	9.85	67.14	58.84
0.20	25.80	4.38	64.62	0.84	15.02	97.09	24.63	21.84	10.38	67.57	60.23
0.25	25.56	4.59	67.63	0.83	16.03	96.86	24.83	22.00	10.93	67.76	60.09
0.30	25.27	4.93	71.44	0.83	17.32	96.62	25.01	22.16	11.58	67.83	59.92
0.35	24.95	5.26	75.36	0.83	18.86	96.39	25.16	22.37	12.03	67.81	60.14
0.38	24.71	5.54	78.24	0.82	20.01	96.17	25.25	22.52	12.28	67.77	60.31
0.40	24.54	5.75	80.28	0.82	20.97	96.03	25.29	22.54	12.39	67.72	59.98
0.45	24.05	6.37	86.28	0.81	23.16	95.76	25.36	22.66	12.75	67.65	59.84
0.50	23.55	7.12	92.72	0.81	25.65	95.45	25.50	22.78	12.78	67.55	59.71
0.55	23.01	7.99	99.36	0.80	28.14	95.17	25.54	22.92	12.87	67.44	59.58
0.60	22.52	8.96	106.02	0.80	31.15	94.91	25.55	22.95	12.85	67.28	59.49

J Cross-dataset and Cross-backbone Generalization

Table A7 reports the cross-dataset and cross-backbone comparison over the same four settings as Table 4 of the main paper – each of the two datasets evaluated with both SD-Turbo and SwiftBrushV2 – run for both ChordEdit and BudEdit, giving eight rows in total. Each method keeps its own default hyperparameters fixed across the four dataset–backbone settings, under the same evaluation protocol. Bold denotes the better method within each dataset–backbone setting.

Table A7: **Cross-dataset and cross-backbone comparison.** ChordEdit and BudEdit on two datasets and two backbones, with 11 evaluation metrics. Each setting is run on the full 700-image set; the five mask-restricted background metrics use the non-empty-background subset defined in Appendix D (556 of 700 on PIE-Bench, 171 of 700 on PIE-Bench++).

Dataset	Backbone	Method	Background						Edited				
			PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
PIE-Bench	SD-Turbo	ChordEdit	22.63	7.98	122.66	0.77	28.81	95.04	24.83	22.15	10.72	67.21	60.19
		BudEdit	**24.71**	**5.54**	**78.24**	**0.82**	**20.01**	**96.17**	**25.25**	**22.52**	**12.28**	**67.77**	**60.31**
	SwiftBrushV2	ChordEdit	22.93	7.34	120.51	0.77	27.34	95.49	23.74	21.18	6.83	66.62	59.30
		BudEdit	**25.49**	**4.74**	**73.92**	**0.83**	**15.91**	**96.64**	**24.01**	**21.26**	**8.37**	**67.39**	**59.43**
PIE-Bench++	SD-Turbo	ChordEdit	22.87	7.67	113.97	0.78	33.39	93.79	24.77	24.12	13.60	66.18	64.43
		BudEdit	**25.06**	**5.43**	**74.61**	**0.83**	**23.29**	**95.03**	**25.01**	**24.27**	**14.65**	**66.91**	**64.94**
	SwiftBrushV2	ChordEdit	23.16	6.93	110.94	0.79	30.20	94.31	23.15	22.46	9.35	65.80	64.09
		BudEdit	**25.83**	**4.48**	**66.84**	**0.84**	**17.52**	**95.59**	**23.41**	**22.70**	**10.55**	**66.59**	**64.79**

K Additional Diagnostics and Component Ablation

This section presents three additional diagnostics. The two-draw diagnostic measures the reproducibility of the single-draw residual field, the localization diagnostic compares the derived support against the annotated edit region, and the component ablation varies, one at a time, the three places where a set enters the construction of Section 3.3.

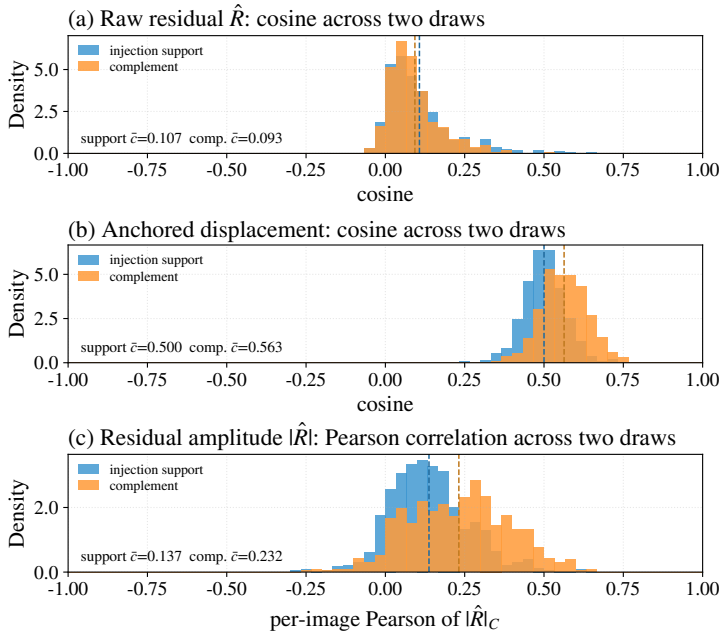

Figure A8: **Two-draw transport.** (a) Raw residual $\widehat{R}$; (b) anchored displacement $\widehat{x}_{\mathrm{tar}} - x_{\mathrm{src}}$; (c) per-pixel residual amplitude $\|\widehat{R}\|_C$. Blue/orange: the first draw's effective injection support and its complement.

Signal–noise decomposition. For each source–target pair, Figure A8 compares quantities obtained from two independent noise draws at t_s, evaluated on the effective injection support from the first draw and its complement, which contains the operational background support $\widehat{\mathcal{B}}$. Panel (a) measures the cross-draw cosine of the raw residual $\widehat{R}$. Its near-zero means, 0.107 on the injection support and 0.093 on its complement, show that the exact residual is strongly draw-sensitive. The residual is therefore not used as the sole full-image transport field: BudEdit suppresses it outside the operational support and retains only an attenuated, support-restricted base term inside the injection region.

Panel (b) measures the anchored displacement $\widehat{x}_{\mathrm{tar}} - x_{\mathrm{src}}$, whose pointwise unit normalization supplies the injection direction d. Its cross-draw cosines, 0.500 and 0.563, are substantially higher than those of the raw residual. Thus, while the measured displacement field remains draw-dependent, it provides a more consistent direction for the budgeted injection than the raw residual alone. Panel (c) measures the per-image Pearson correlation of the residual amplitude map $\|\widehat{R}\|_C$ across the two draws. The low correlations, 0.137 and 0.232, show that the spatial magnitude of the raw residual is also draw-sensitive. Together, these diagnostics motivate the asymmetric construction of BudEdit: the residual supplies the support, an attenuated local base term, and the background-derived budget magnitude, whereas the budgeted injection follows the anchored direction inside the effective support. The scalar the construction consumes is itself draw-dependent: over the 700 pairs, two independent draws give E_{bg} with a Pearson correlation of 0.320 and a median relative deviation of 58.6%. The budget is therefore image-dependent in the sense of Section 3.3, and within an image additionally draw-dependent rather than a draw-invariant constant. Conditioned on fixed support and allocation weights, budget variation rescales the injection amplitude without changing the spatial profile.

Derived support versus annotation. The injection support $\mathcal{E}_{\mathrm{inj}}$ of Eq. 5 is compared against the PIE-Bench annotation M_{gt}, which the method never reads. The derived support is taken at the latent resolution and upsampled to 512×512 by nearest neighbour; precision, recall and the intersection over union are pooled over all 700 pairs. The derived support attains precision 0.611 at recall 0.713, for an IoU of 0.490: it recovers most of the annotated region while extending about 17% beyond it in area. The outward extension is expected rather than incidental. The energy support is dilated once and carries a one-pixel soft ring (Appendix A), each latent cell covers an 8×8 block of image pixels, and a real edit needs a boundary margin for shadow and contact that a tightly drawn annotation does

not include. What the construction requires is that the budget be spent where the edit is, not that the support coincide with a human outline, and the output blend of Eq. 17 further restricts which of those pixels reach the final image.

Table A8: **Component ablations on PIE-Bench, one row per design choice.** The construction of Section 3.3 uses a set in three distinct places: the *spending domain* $\mathcal{E}_{\mathrm{inj}}$ that supports the injected field $f^\star$, the *collection domain* $\widehat{\mathcal{B}}$ from which the budget B is measured, and the *normalizing denominator* $\sum w_{\mathrm{eff}}$ that sets the scale of the amplitude $A^\star$. Each row varies exactly one of the three and leaves the other two as in BudEdit. AO: spending domain $\{\Phi > 0.5\}$ alone. EO: spending domain $\widehat{\mathcal{E}}$ alone. IU: denominator taken over $\widehat{\mathcal{E}}$ with the ungated weight. IS: collection domain drawn from inside $\mathcal{E}_{\mathrm{inj}}$. Bold: column best, ties included; shading: BudEdit.

	Background						Edited				
Configuration	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
AO	24.78	**5.28**	76.86	**0.82**	**19.28**	96.10	25.13	22.38	11.87	67.02	59.78
EO	24.35	5.82	82.31	**0.82**	20.62	95.83	25.21	22.48	12.16	67.51	59.46
IU	**24.87**	5.31	**76.69**	**0.82**	19.43	96.13	25.14	22.39	12.00	67.64	59.88
IS	21.98	11.27	118.58	0.79	37.38	94.30	25.13	22.38	11.43	62.05	54.13
BudEdit	24.71	5.54	78.24	**0.82**	20.01	**96.17**	**25.25**	**22.52**	**12.28**	**67.77**	**60.31**

Spending domain. The two single-signal variants trade against each other: AO preserves the background better (PSNR 24.78) but edits less (CLIP-D 11.87), while EO edits harder (CLIP-D 12.16) at a clear background cost (PSNR 24.35, LPIPS 82.31). Intersecting the two signals keeps the editing gain of one and the background of the other: BudEdit is the best row of the table on all five edited-region metrics and on background CLIP, while its PSNR of 24.71 is within 0.07 dB of AO.

Normalizing denominator. Changing only the denominator rescales the injected field without altering its spatial profile, so IU is a smaller realized budget rather than a different allocation. Its eleven values lie within 1.3% of the β=3 row of Table A4, which prescribes three quarters of the default budget, indicating behavior similar to a milder injection. The per-metric signs agree: PSNR, MSE, LPIPS and DINO all improve, SSIM is unchanged, only background CLIP edges down by 0.04, while all five edited-region metrics weaken. The gap between IU and BudEdit is consistent with the budget-matching property of Section 3.3 doing work.

Collection domain. Drawing E_{bg} from inside the spending domain is not a larger budget but a mis-scaled one. IS falls outside the β sweep's range: its LPIPS is 34.54 higher than the β=8 row, while its MUSIQ and MUSIQ-E lie below, and its DINO above, the corresponding sweep extrema. The energy inside the support is dominated by the displacement the edit is asking for, so measuring the budget there leaves it uninformative and the injection damages structure. What the construction requires is therefore that the collection domain lie *outside* the spending domain, not merely that it be broader.

Gated refinement. Table A9 isolates the effect of GatedRefine while retaining the same budgeted transport, low-noise second pass, noise realization, and two UNet evaluations. The variant without GatedRefine directly uses the second-pass prediction.

Table A9: **Ablation of** GatedRefine **on PIE-Bench.** Bold: column best; shading: full BudEdit.

Configuration	Background						Edited				
	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
w/o GatedRefine	23.94	6.27	90.14	0.81	25.88	95.34	**25.36**	**22.66**	11.74	**68.62**	**61.29**
BudEdit (Full)	**24.71**	**5.54**	**78.24**	**0.82**	**20.01**	**96.17**	25.25	22.52	**12.28**	67.77	60.31

Compared with the variant without GatedRefine, the full model improves all six background-fidelity metrics, increasing PSNR from 23.94 to 24.71 dB and reducing MSE, LPIPS, and DINO by 11.6%, 13.2%, and 22.7%, respectively. It also raises CLIP-D from 11.74 to 12.28, demonstrating better preservation of source structure and directional edit alignment. Conversely, removing GatedRefine yields slightly higher CLIP-W, CLIP-E, MUSIQ, and MUSIQ-E, revealing an

application-dependent fidelity–editability trade-off. As an inference-time module, GatedRefine can be enabled when source-structure preservation is prioritized and omitted when higher target-prompt similarity and perceptual quality are preferred; we retain it by default for balanced performance.

L CHORDEDIT'S CHORD WINDOW AT TWO TIMESTEPS

ChordEdit (Lu et al., 2026) stabilizes one-step editing with a chord window of width δ, which averages the control field over a short interval of timesteps. This is a temporal mechanism: it does not decide *where* the editing energy goes, only how smoothly it is accumulated in time. Table 1 reports ChordEdit at its default δ=0.15, while Table 2 additionally includes δ=0 at t_s=0.9; here we evaluate both settings under two timesteps, reported in Table A10: its own default (t_s=0.9, t_e=0.3) and BudEdit's timestep (t_s=0.78, t_e=0.38), to isolate the chord window's own contribution at each operating point. BudEdit, run at its own timestep as in Table 1, is placed alongside for reference. The t_s=0.9, δ=0.15 row is exactly the ChordEdit entry of Table 1, and the t_s=0.9, δ=0 row is the window-off configuration of Table 2, so the two tables read as one continuous experiment.

Table A10: **ChordEdit with and without its chord window, at its own default timestep and at BudEdit's timestep, alongside BudEdit.** Bold: column best; shading: BudEdit.

Method	Background						Edited				
	PSNR↑	MSE↓	LPIPS↓	SSIM↑	DINO↓	CLIP↑	CLIP-W↑	CLIP-E↑	CLIP-D↑	MUSIQ↑	MUSIQ-E↑
ChordEdit, t_s=0.9, δ=0	20.13	13.34	159.82	0.73	45.55	93.41	25.11	**22.54**	11.86	64.13	55.97
ChordEdit, t_s=0.9, δ=0.15 (default)	22.63	7.98	122.66	0.77	28.81	95.04	24.83	22.15	10.72	67.21	60.19
ChordEdit, t_s=0.78, δ=0	21.27	10.74	148.10	0.74	38.31	93.98	25.17	22.45	12.01	65.52	59.89
ChordEdit, t_s=0.78, δ=0.15	22.86	7.60	126.81	0.76	28.08	95.05	24.78	22.07	9.99	66.97	60.05
BudEdit	**24.71**	**5.54**	**78.24**	**0.82**	**20.01**	**96.17**	**25.25**	22.52	**12.28**	**67.77**	**60.31**

Within ChordEdit's own default timestep, enabling the chord window (δ=0 → 0.15) improves every background metric, matching the δ-ablation in Table 2: PSNR rises from 20.13 to 22.63 dB, MSE from 13.34 to 7.98, LPIPS from 159.82 to 122.66, SSIM from 0.73 to 0.77, DINO from 45.55 to 28.81, and background CLIP from 93.41 to 95.04. The same pattern holds when ChordEdit is instead run at BudEdit's timestep: enabling the chord window again improves background preservation. Across both timesteps, BudEdit wins every metric against three of the four ChordEdit settings, and ten of the eleven against the fourth; the sole exception anywhere is CLIP-E at t_s=0.9 with the window off, where ChordEdit reaches 22.54 against BudEdit's 22.52, the same 0.02 gap the main paper reports in Section 4.2. That section's "Matched effective timestep" paragraph relates the two methods' timesteps rather than equalizing them.

Temporal smoothing moves ChordEdit along a trade-off rather than off it. At its own default timestep, turning the chord window on raises background PSNR by 2.50 dB, lowers DINO from 45.55 to 28.81, and adds 3.08 MUSIQ, but costs 0.28 CLIP-W, 0.39 CLIP-E and 1.14 CLIP-D: the smoothed field is gentler everywhere, including inside the region the edit was supposed to change. Neither of the two settings dominates the other.

Spatial energy budgeting is not on that curve. Against ChordEdit's smoothed default, BudEdit preserves the background better, by 2.08 dB PSNR and 44.42 LPIPS, and it also edits more strongly, by 0.42 CLIP-W, 0.37 CLIP-E and 1.56 CLIP-D; against the *unsmoothed* default it keeps CLIP-D ahead by 0.42 while giving up the same 0.02 CLIP-E. At BudEdit's own timestep both editing margins over the window-off setting turn positive, +0.07 CLIP-E and +0.27 CLIP-D. BudEdit also scores highest of all five rows on both perceptual measures, MUSIQ and MUSIQ-E. This is what the two mechanisms turn on. Attenuating the field uniformly in time makes preservation and editing strength move in opposite directions, because the same attenuation applies inside the edited region as outside it. Using background residual energy to scale an injection inside the attention-gated support separates the two, so both can improve at once.

M NOTATION

Table A11 collects the symbols used in the main paper and in this supplement; the named operators of Algorithm 1 (`noisify`, `UNet`, EnergySupport, Extend, Soften, blur, unit, and GatedRefine)

are defined in the main text and in Appendix A rather than here. A few typographic conventions are used consistently and are worth stating once, because several symbols differ only by their decoration. Calligraphic capitals denote *regions*, that is, subsets of the latent pixel domain; the one exception is $\mathcal{T}_t$, which is an operator. An upright subscript names a semantic role and carries no claim about tensor order, so E_{bg} is a scalar while m_{inj} and w_{eff} are fields. A superscript $\star$ marks either an idealized quantity the algorithm never observes, as in $\mathcal{E}^\star$ and $\mathcal{B}^\star$, or the solution of the allocation program, as in $A^\star$ and $f^\star$, which the algorithm does compute. A hat marks a single-draw or algorithm-estimated quantity. Thus $\mathcal{B}^\star$, the semantic background, $\widehat{\mathcal{B}}$, the operational background support the algorithm actually measures on, and B, the administered budget scalar, are three distinct objects.

Table A11: **Notation.** Symbols used in the main paper and this supplement, grouped by role. Section numbers refer to the main paper.

Symbol	Meaning
Regions (Sections 3.2–3.3)	
Ω	Full latent pixel domain, of size $H \times W$
$\lvert\mathcal{S}\rvert$	Cardinality of a region $\mathcal{S}$, i.e. its number of latent pixels
$\mathcal{E}^\star$	Semantic edit region; idealized and unobservable, used only in the conceptual diagnosis
$\mathcal{B}^\star$	Its background, $\Omega \setminus \mathcal{E}^\star$; also idealized
$\widehat{\mathcal{E}}$	Residual-energy support: thresholded per-pixel residual energy with a narrow boundary extension
$\widehat{\mathcal{B}}$	Operational background support, $\Omega \setminus \widehat{\mathcal{E}}$; the domain on which the budget is collected
$\mathcal{E}_{\mathrm{inj}}$	Effective injection support, $\mathrm{supp}(m_{\mathrm{inj}})$; the domain on which the budget is spent
Fields and residuals (Sections 3.1–3.3, 3.5)	
$\widehat{R}_z \equiv \widehat{R}(x_\tau, t; z)$	Single-draw residual field, the editing signal at one noise draw
$\overline{R}$	Its population mean over the noise distribution
$u_t(x_\tau)$	Population residual field, $u_t := \overline{R}(x_\tau, t)$
$\epsilon_t(z)$	Draw-dependent deviation, $\widehat{R}_z - u_t(x_\tau)$, with zero mean by construction
$\widehat{u}_{t_s}(x_\tau)$	ChordEdit's chord control field at t_s; the time-weighted average of the two single-draw residuals at t_s and $t_s - \delta$, used only for comparison
$\widetilde{R}_e$	Second-pass residual, measured at the lower noise level t_e
F	Nominal transport field, $\alpha\, m_{\mathrm{inj}} \odot \widehat{R} + f^\star$
f	Candidate injection field; the optimization variable of the allocation program
$f^\star$	Budgeted injection field, $A^\star \odot d$; the unique solution of that program
A, $A^\star$	Nonnegative amplitude map: the optimization variable, and its optimum
d	Pointwise unit direction towards the target-conditioned prediction, defined on $\mathcal{E}_{\mathrm{inj}}$ with $\lVert d(i,j)\rVert_C = 1$
$v(x_t, t, c)$	Model drift induced by the conditioning c
$p(x \mid c)$	Conditional data distribution; editing runs between the source- and target-conditioned ones
(i,j), (k,l)	Latent pixel indices; (k,l) is used for the summation index under a normalization
Energy and budget (Sections 3.1, 3.3)	
$\lVert\cdot\rVert_C$	Per-pixel channel-wise norm in the common comparison space
$\lVert h\rVert^2$	Total energy of a field h, $\sum_{(i,j)} \lVert h(i,j)\rVert_C^2$
$\lVert\cdot\rVert$	Euclidean norm of a vector; the induced operator norm when applied to a Jacobian
E_{bg}	Background residual energy, measured on $\widehat{\mathcal{B}}$ without the attention gate
B	Administered budget, $B = \beta E_{\mathrm{bg}}$; a nonnegative scalar in energy units, matched exactly by $f^\star$ on nonempty $\mathcal{E}_{\mathrm{inj}}$

Symbol	Meaning
Z	Allocation normalizer, $\sum_{(i,j)\in\mathcal{E}_{\text{inj}}} w_{\text{eff}}(i,j)$
Masks and allocation weights (Sections 3.3–3.5)	
m	Soft mask in $[0,1]^{H\times W}$ with $\text{supp}(m) = \widehat{\mathcal{E}}$
$\mathbf{1}[\cdot]$	Indicator function; equals one when its condition holds and zero otherwise
$m_{\text{inj}}(i,j)$	Soft injection mask, $m(i,j)\mathbf{1}[\Phi(i,j) > 0.5]\mathbf{1}[\|\widehat{x}_{\text{tar}}(i,j) - x_{\text{src}}(i,j)\|_C > 0]$; acts on field amplitude
w	Smoothed attention readout, $\text{blur}(\Phi)$; the unweighted allocation signal
w_{eff}	Effective allocation weight, $m_{\text{inj}}^2 \odot w$; the square converts amplitude into energy
$\mu_{w_{\text{eff}}}$	Target distribution, w_{eff} normalized over $\mathcal{E}_{\text{inj}}$
p_f	Normalized energy distribution of the field f
Ψ	Output gate (pin mask), $\Psi \in [0,1]^{H\times W}$; retains the admitted first-pass edit support, adds extensions jointly supported by the second-pass residual and attention, and controls the final source fusion
M_{gt}	Evaluation editing mask used only to define the five background and two edited-region metrics of Appendix D, and never read by the method
Attention (Sections 3.4–3.5)	
Φ	Localization signal in $[0,1]^{H\times W}$, aggregated from the cross-attention maps of the tokens that differ between the two prompts
Φ_{pin}	Second-pass attention readout used to construct the cleanup gate, modulate the micro-swap, and construct the pin mask Ψ
Latents, predictions and conditions (Sections 3.1, 3.3, 3.5)	
x_t	General state on the ODE trajectory, indexed by time
x_τ	Current latent queried by the method; equals x_{src} in the single-step setting
x_{src}	Source latent, or source image where an image-space metric is meant
$\widehat{x}_{\text{tar}}$	Target-conditioned prediction from the transport pass
$\widetilde{x}_{\text{tar},e}$	Target-conditioned prediction from the second pass; the starting point of GatedRefine
x_{edit}	Transported latent, $x_{\text{src}} + F$
x_{refined}	Output of GatedRefine
x_{out}	Final output after the pin-mask blend
$z \sim K_t(\cdot \mid x_\tau)$	Noise draw and its noising kernel at time t
z_e	Re-noised latent for the second pass at t_e
$Q(z,t,c)$, Q_z^c	Model output: noise prediction for SD-Turbo, velocity for flow models
$\mathcal{T}_t$	Time-only linear map into the common comparison space
c_{src}, c_{tar}	Source and target conditioning prompts
Constants and operators	
t_s	Timestep at which the field is evaluated; 0.78 for BudEdit (Section 4.1). The chord field of Section 3.2 and the matched-timestep paragraph of Section 4.2 reuse this symbol for ChordEdit's own timestep, whose default is 0.9; Appendix L labels those runs explicitly
t_e	Refinement timestep, i.e. the second-pass noise level; 0.38
α	Attenuation applied to the measured residual; 0.7
β	Budget multiplier; 4.0, dimensionless
δ	ChordEdit's chord-window width; a baseline hyperparameter
Δt	Explicit Euler step size in the local-error bound
D_{KL}	Kullback–Leibler divergence
$\odot$	Hadamard, i.e. elementwise, product; a 2-D mask acts on a latent field by broadcasting over channels