CSTAR-OPD: COUNTERFACTUAL SPATIO-TEMPORAL ALIGNMENT FOR REGION-AWARE ON-POLICY DISTILLATION IN LOCAL IMAGE EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Large instruction-based image editors can teach compact models, but local editing requires changing the target region while preserving its surroundings. Standard on-policy distillation (OPD) matches teacher and student at student-visited states, yet a global teacher target may propagate unintended changes outside the edit region. Uniform sampling across denoising states also ignores differences in their influence on the final edit. We introduce **CSTAR-OPD**, a **C**ounterfactual **S**patio-**T**emporal **A**lignment framework for **R**egion-aware OPD. At the same student-visited state, we evaluate a frozen teacher under both the edit instruction and an identity instruction. The student matches the edit-conditioned velocity inside the target mask and the identity-conditioned velocity outside it, using separate region-normalized losses so that the background does not dominate the objective solely by area. The teacher and mask are used only during training; inference requires neither. Analyses of semantic direction and 4B-to-9B model switching motivate sampling one student state per example from states 0–13 of a 50-step schedule. On GIE-Bench, CSTAR-OPD reduces outside-mask M-MSE by 17.7% relative to full-range global OPD while maintaining the same 88.98% functional correctness. These results support combining early-state supervision with region-specific teacher targets to balance editing and preservation.

1 INTRODUCTION

Instruction-based image editing aims to transform a source image according to a natural-language request while retaining other content not implicated by that request. This dual requirement is particularly strict for local editing, such as adding, removing, or replacing an object and modifying an object attribute. Recent editors have made substantial progress through large-scale paired training, diffusion-based attention control, and mask-guided generation (Brooks et al., 2023; Hertz et al., 2023; Couairon et al., 2023). However, successful image editing and source preservation remain distinct and often competing objectives. GIE-Bench evaluates these two objectives separately, highlighting the difficulty of simultaneously achieving accurate image editing and strong preservation of irrelevant regions (Qian et al., 2025).

For compact image editors, one promising way to improve editing capability is to transfer knowledge from a stronger teacher model. Recent on-policy distillation (OPD) methods provide an effective framework for such capability transfer (Li et al., 2026; Xu et al., 2026; Zhou et al., 2026). Instead of learning from trajectories generated offline by the teacher, OPD lets the student roll out its own denoising trajectory and queries the teacher at the latent states actually visited by the student. The student is then optimized to match the teacher's local transition or velocity field at these states, reducing the discrepancy between the states encountered during distillation and those encountered at inference time. This formulation has been shown to effectively transfer and consolidate specialized generative and editing capabilities into a unified student model.

However, applying OPD to image editing exposes both spatial and temporal misalignment as Fig. 1 shows. Spatially, existing formulations typically treat the teacher's edit-conditioned trajectory as a global supervision signal and align the student with the teacher over the entire latent space. For local editing, however, the target region should follow the teacher's editing behavior, whereas re-

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.

gions unrelated to the request should ideally preserve the source content. Consequently, unintended variations introduced by the teacher outside the target region may also be transferred to the student through global trajectory matching. A typical case is shown as Fig. 2.

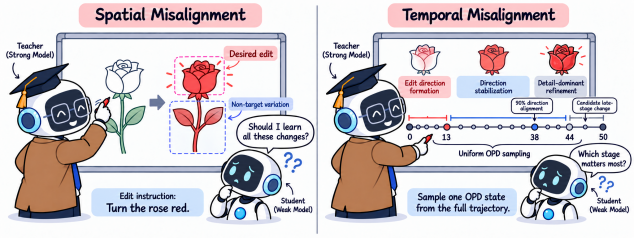


Figure 1: Illustration of misalignments in conventional teacher–student distillation

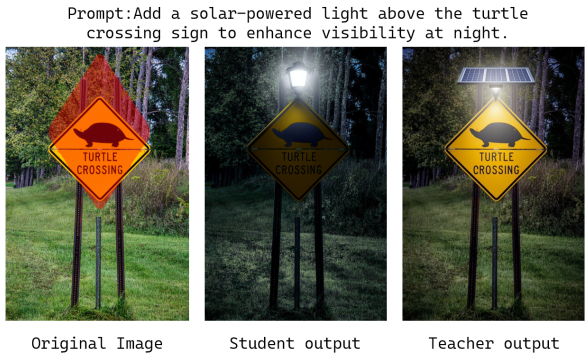


Figure 2: Both the student model and the teacher model made the same mistake during editing, changing the picture scene to an evening scene.

Temporally, the denoising trajectory is also stage-dependent. Different diffusion timesteps contribute different types of information. Earlier stages primarily establish coarse semantic structures and spatial layouts, whereas later stages progressively refine local attributes and visual details (Yi et al., 2024; Li et al., 2025). Accordingly, treating all timesteps equally during teacher–student optimization can mix supervision of different granularity. Image editing trajectories also exhibit substantial temporal redundancy, suggesting that not all stages contribute equally to the editing process (Yan et al., 2025). Nevertheless, existing OPD formulations typically sample teacher supervision across the entire trajectory, implicitly treating different editing stages as homogeneous. Since the editing dynamics evolve stage by stage, supervision from different timesteps may encode distinct and even competing behaviors. Forcing a shared student to jointly fit these heterogeneous signals can introduce cross-stage interference and lead to a compromised solution, degrading the final editing performance. Together with the spatial ambiguity described above, this motivates a spatio-temporally aligned distillation strategy that determines both where and when teacher supervision should be applied.

In this paper, we explore the edit phase of image edit model and propose **CSTAR-OPD**, a counterfactual spatio-temporal alignment framework for region-aware on-policy distillation. Spatially, CSTAR-OPD decomposes the supervision using a training-time mask. The edit region matches the transition predicted by the edit-conditioned teacher, whereas the other region matches a counterfactual preservation target. This successfully divides the learning objective of the model into editing and preservation, thereby avoiding learning the editing errors that may be leaked from the teacher model. The supervision is internalized by the student, so inference remains mask-free. Temporally, instead of conventional OPD training that applies supervision over the full denoising trajectory or samples timesteps across the entire trajectory, we perform OPD using only samples from the early-stage editing trajectory. This ensures that most of the training occurs at the positions where smaller models need to learn more and reduces the rollout transitions required to reach the sampled state.

Our contributions are threefold:

- We provide a systematic analysis of local-editing trajectories, showing that their semantic effect is phase-structured rather than uniform over denoising time, and identify an early stage sampling OPD. OPD supervision is concentrated on the empirically identified early editing stage, reducing the student rollout required to reach sampled supervision states.
- We introduce a counterfactual region-aware OPD objective that learns the teacher's editing transition inside the target mask and a preservation transition outside it, while requiring no mask at inference.
- We develop a single-query, early-window training strategy that is cheaper than dense or multi-state OPD and empirically yields a better edit–preservation trade-off than full-range single-state and uniform multi-state baselines across four local editing tasks.

2 Related Work

2.1 Instruction-Based Local Image Editing and Preservation

Text-guided image editing has been explored through attention control, mask-guided denoising, and instruction-based fine-tuning. Prompt-to-Prompt preserves spatial layouts through cross-attention control (Hertz et al., 2023), DiffEdit restricts denoising to automatically detected regions (Couairon et al., 2023), and InstructPix2Pix learns a general editor from synthetic image pairs (Brooks et al., 2023). Recent methods further introduce region- and timestep-aware conditioning for fine-grained control (Zhou et al., 2025). Meanwhile, GIE-Bench evaluates editing accuracy and non-target preservation separately (Qian et al., 2025). Unlike methods that use masks primarily for conditioning or inference control, CSTAR-OPD uses the edit region to determine the teacher supervision target: the student follows the teacher's editing behavior inside the mask and its counterfactual preservation behavior outside it.

2.2 Diffusion Distillation and On-Policy Supervision

Diffusion distillation has primarily focused on reducing sampling steps through progressive distillation, consistency learning, or distribution matching (Salimans & Ho, 2022; Song et al., 2023; Yin et al., 2024). Our objective instead is to transfer editing capability from a large teacher to a smaller student while retaining multi-step generation. Recent on-policy distillation methods supervise the student at states visited by its own rollout, using trajectory-level or transition-level matching (Fang et al., 2026; Li et al., 2026; Jiang et al., 2026; Liu et al., 2026). However, their supervision is generally spatially global. CSTAR-OPD preserves on-policy state coverage while constructing region-dependent targets from editing and preservation teacher trajectories.

2.3 Stage-Specific Dynamics Across Denoising Time

Denoising dynamics are temporally non-uniform: early steps primarily determine semantic structure and composition, whereas later steps increasingly refine appearance and texture (Tinaz et al., 2025; Li et al., 2025). Existing editing methods exploit this structure through timestep-dependent conditioning or inference-time intervention (Zhou et al., 2025; Troeshestova et al., 2026). In contrast, we analyze the editing trajectory to determine where teacher supervision is most informative during training. The resulting phase-aware sampling policy concentrates OPD queries in the early editing stage and is evaluated against full-range single-state and multi-state distillation baselines.

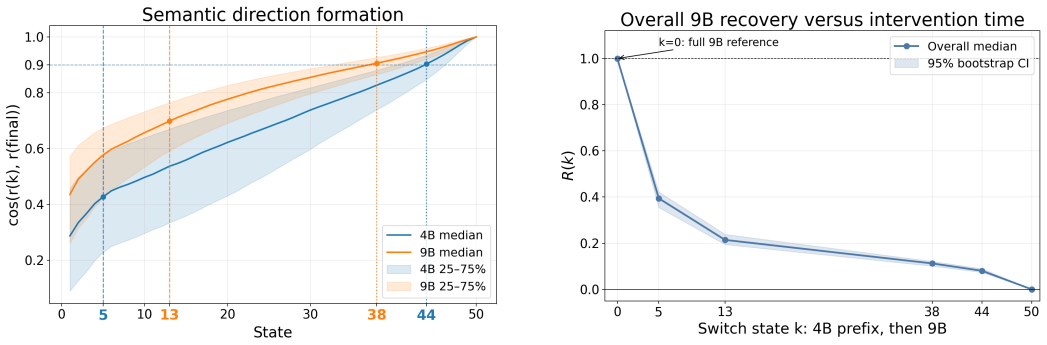
3 Do edit phases exist?

3.1 When Does the Semantic Editing Direction Emerge?

To investigate how does the editing behavior emerge in the diffusion trajectory, we define counterfactual edit residual $r_m(k)$.

$$r_m(k) = x_m^{\text{edit}}(k) - x_m^{\text{identity}}(k) \tag{1}$$

where m represents model, k represents timestamp. For the same model, the same sample, and the same initial noise, we run two trajectories each: one is the normal trajectory of the instruction



(a) Trajectory-stage analysis of 4B and 9B.

(b) Recoverability under 4B-to-9B switching.

Figure 3: **Editing trajectories exhibit non-uniform temporal structure.** Left: the teacher's edit-residual direction undergoes a statistically supported transition at state 13. Right: most recoverability is lost when the teacher is removed from the earliest denoising states. The solid line reports the median recovery over 1,000 samples, and the shaded region denotes the 95% bootstrap confidence interval of the median.

editing $x_m^{\text{edit}}(k)$, and the other is the trajectory under the condition of maintaining the original image $x_m^{\text{identity}}(k)$. And we define edit residual direction $C_m(k)$:

$$C_m(k) = cos(r_m(k), r_m(K)) \tag{2}$$

where K represents the last timestamp of diffusion trajectory. The analysis of 4,000 trajectories is shown by Fig. 3a. The 9B model maintains higher alignment with the final semantic direction throughout the early trajectory, indicating faster semantic direction formation than the 4B model. Its rapid formation regime also lasts longer, extending to state 13, whereas the corresponding regime of 4B ends near state 5. The detailed analysis can seen in Appendix.

3.2 When Can the 9B Model Redirect the 4B Editing Trajectory

Section 3.1 reveals a temporal gap between the editing dynamics of the 4B and 9B models: the 9B model establishes a stronger semantic direction early and reaches directional stability sooner. This gap is directly relevant to OPD. To examine when such teacher guidance is most valuable, we conduct a counterfactual 4B-to-9B switching experiment. We run the trajectory with the 4B model up to state k, switch to the 9B model for the remaining steps, and measure the recovery of the final 9B editing outcome using Eq. 3.

$$R(k) = \underset{i=1,\ldots,N}{\text{Median}} \left[1 - \frac{\left\| \mathbf{x}_i^{4\text{B}\to 9\text{B}}(k) - \mathbf{x}_i^{9\text{B}} \right\|_2}{\left\| \mathbf{x}_i^{4\text{B}} - \mathbf{x}_i^{9\text{B}} \right\|_2} \right] \tag{3}$$

where $R(k)$ denotes the overall recovery rate when the model is switched from 4B to 9B at state k, i indexes the evaluation samples and N is the total number of samples. The term $\mathbf{x}_i^{4\text{B}\to 9\text{B}}(k)$ denotes the final latent obtained by running the 4B model up to state k and the 9B model for the remaining denoising steps. The terms $\mathbf{x}_i^{4\text{B}}$ and $\mathbf{x}_i^{9\text{B}}$ denote the final latents produced by the complete 4B and 9B trajectories, respectively.

As shown in Fig. 3b, the recovery score decreases sharply from switch point 0 to 5 and 13. Thus, after only the first 13 states are generated by the 4B model, a full 9B continuation can recover only 21.4% of the endpoint gap. This result indicates that the editing dynamics of the two models already diverge substantially within states 0–13, and that the deviation introduced by the early 4B trajectory is difficult to redirect through later 9B denoising. Combined with the semantic-direction analysis, where the 9B model exhibits stronger early alignment and remains in its rapid direction-formation regime until state 13, we identify states 0–13 as a high-value OPD window. Teacher velocity supervision in this interval helps the student form the correct semantic editing trajectory early, before errors accumulate and lead to larger deviations from the teacher.

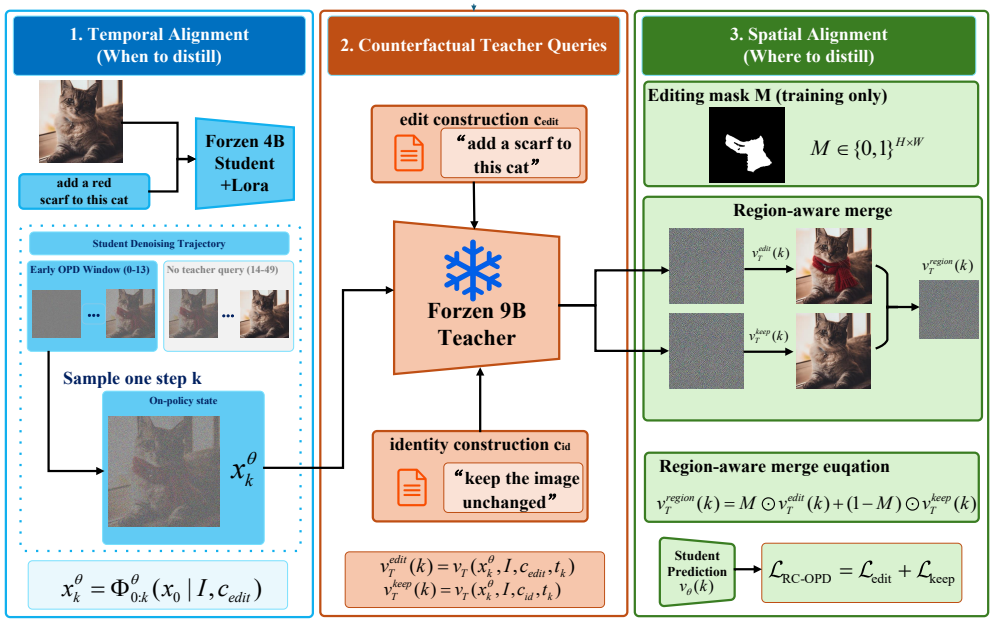


Figure 4: **Overview of CSTAR-OPD.** *Left:* The 4B student, conditioned on the source image and edit instruction, is rolled out to a randomly selected early state $\mathbf{x}_k^\theta$ ($0 \le k \le 13$). *Middle:* At that same state and timestep, the frozen 9B teacher predicts an edit field and an identity-conditioned preservation field. *Right:* A source-image mask selects the edit field inside the target region and the preservation field elsewhere; the student is trained with separately normalized regional losses. The teacher, identity instruction, and mask are used only during training.

4 Region-Aware Counterfactual On-Policy Distillation

The trajectory analysis in Sec. 3 shows that the effect of teacher supervision depends on the denoising stage. Local image editing also poses a spatial challenge: matching an edit-conditioned teacher field everywhere may transfer unintended changes to regions that should be preserved. Restricting supervision to the edit mask alone leaves the complementary region without a preservation target. We address both issues with **CSTAR-OPD** (**C**ounterfactual **S**patio-**T**emporal **A**lignment for **R**egion-aware **O**n-**P**olicy **D**istillation). As illustrated in Fig. 4, the method selects an early state visited by the current student, evaluates two teacher fields at that state, and matches each field in its corresponding spatial region. The mask changes the training objective, whereas the student receives no mask at inference.

4.1 Early-Stage On-Policy State Selection

Each training example consists of a source image I, an edit instruction c_{edit}, and a pixel-space edit mask M_{pix}. We use the base FLUX.2 Klein 9B model as the frozen teacher and the base 4B model (Black Forest Labs, 2026) with a trainable transformer LoRA (Hu et al., 2022) as the student. The two models have compatible packed latent fields. We encode I with the student's frozen VAE, normalizing and packing its latent representation into source-image tokens $\mathbf{z}_I$. At each denoising step, the current noisy tokens are concatenated with these fixed source tokens as image inputs. The predicted velocity is taken only from the noisy-token positions. Teacher and student receive the same $\mathbf{z}_I$ and noisy state, while their respective text encoders encode the instruction.

We sample packed Gaussian noise $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, with the shape of the source latent, and construct a $K = 50$ transition schedule. Let t_j denote its j-th model timestep and $\mathcal{S}_j$ represent the corresponding scheduler update. Given a sampled index k, the current student generates the state

$$\begin{aligned} \mathbf{x}_{j+1}^\theta &= \mathcal{S}_j\left(\mathbf{x}_j^\theta, \mathbf{v}_\theta^{\text{edit}}(\mathbf{x}_j^\theta, \mathbf{z}_I, t_j)\right), \quad j = 0, \dots, k-1, \\ \mathbf{x}_k^\theta &= \Phi_\theta^{0:k}(\mathbf{x}_0 \mid I, c_{\text{edit}}) \end{aligned} \tag{4}$$

where $\mathbf{v}_\theta^{\text{edit}}$ is the student's classifier-free-guided field under c_{edit}. This rollout uses the same source-image conditioning and scheduler as student inference. It is performed without gradient tracking, and $\mathbf{x}_k^\theta$ is detached before field matching. Thusthe student is supervised at a state visited by its *current* policy, while gradients update the student's prediction at that state rather than propagating through its preceding rollout.

Based on Sec. 3, we select one query index independently for each training example from

$$k \sim \mathcal{U}\big(\{0, 1, \ldots, 13\}\big) \tag{5}$$

Here k counts *completed* student transitions: $k = 0$ queries the initial noise at t_0 without taking a rollout step, and $k = 13$ queries the state after 13 transitions at t_{13}. Both indices therefore contribute a teacher supervision target. Only one state and one pair of teacher fields are evaluated per example and optimization update. Sampling the early window concentrates the query budget on the stage identified in our analysis; we compare this choice with full-range and multi-state sampling in Sec. 5.2.

4.2 Counterfactual Field Decomposition

At the selected sample step $(\mathbf{x}_k^\theta, \mathbf{z}_I, t_k)$, we query the frozen teacher with the requested edit c_{edit} and a fixed identity instruction c_{id} that asks it to reproduce the source image without changes. Both predictions therefore share the noisy state, source image, timestep, and teacher parameters; only the text instruction differs. In our implementation, fields used by the scheduler and loss include classifier-free guidance (CFG) (Ho & Salimans, 2022). For $a \in \{T, \theta\}$ and an instruction c, define

$$\mathbf{v}_a^c(\mathbf{x}, \mathbf{z}_I, t) = \mathbf{f}_a^\emptyset(\mathbf{x}, \mathbf{z}_I, t) + s\big[\mathbf{f}_a^c(\mathbf{x}, \mathbf{z}_I, t) - \mathbf{f}_a^\emptyset(\mathbf{x}, \mathbf{z}_I, t)\big], \tag{6}$$

where $\mathbf{f}_a^c$ and $\mathbf{f}_a^\emptyset$ are the model's raw conditional and empty-prompt predictions, respectively, and $s = 4$ is the guidance scale. Each model uses its own text encoder and empty-prompt prediction. The teacher's edit, identity, and empty-prompt branches are evaluated at the shared state; the student is evaluated with edit and empty-prompt branches. We abbreviate the two resulting guided teacher fields as

$$\begin{aligned} \mathbf{v}_T^{\text{edit}}(k) &= \mathbf{v}_T^{c_{\text{edit}}}(\mathbf{x}_k^\theta, \mathbf{z}_I, t_k), \\ \mathbf{v}_T^{\text{keep}}(k) &= \mathbf{v}_T^{c_{\text{id}}}(\mathbf{x}_k^\theta, \mathbf{z}_I, t_k). \end{aligned} \tag{7}$$

We define their difference as $\Delta\mathbf{v}_T(k)$:

$$\Delta\mathbf{v}_T(k) = \mathbf{v}_T^{\text{edit}}(k) - \mathbf{v}_T^{\text{keep}}(k). \tag{8}$$

It shows how the teacher's guided field changes when the instruction changes from preserving the source image to performing the requested edit, while the student state, source image, and timestep remain fixed. Both teacher predictions use the same empty prompt and guidance scale. Thus, the shared empty prompt contribution cancels in the subtraction, while the guidance scale remains. The identity instruction encourages the teacher to preserve the source content, but does not guarantee an exact copy. Both fields are evaluated directly at the same student state, without generating a separate teacher trajectory or using an edited target image.

4.3 Region-Aligned Field Matching

We convert M_{pix} into weights aligned with the packed latent-field elements. Specifically, we resize the mask to the pre-packing latent grid with area interpolation, preserve the 2×2 patch offsets when patchifying, and repeat these offsets across the latent channels before packing. Let $M_q \in [0, 1]$ be the resulting edit weight for scalar field element q; $1 - M_q$ is its preservation weight. This construction retains sub-token spatial detail instead of assigning a single mask value to an entire packed token.

The two teacher fields define a conceptual region-dependent target:

$$\begin{aligned} \mathbf{v}_T^{\text{region}}(k) &= M \odot \mathbf{v}_T^{\text{edit}}(k) + (1 - M) \odot \mathbf{v}_T^{\text{keep}}(k) \\ &= \mathbf{v}_T^{\text{keep}}(k) + M \odot \Delta\mathbf{v}_T(k). \end{aligned} \tag{9}$$

The student predicts a guided field at the *same* detached state using only the edit instruction:

$$\mathbf{v}_\theta(k) = \mathbf{v}_\theta^{c_{\text{edit}}}(\mathbf{x}_k^\theta, \mathbf{z}_I, t_k). \tag{10}$$

Equation 9 describes the spatial assignment and is used as a diagnostic target; we do not minimize a global MSE against it. Instead, the code applies the mask as *linear loss weights* and normalizes the edit and preservation regions separately:

$$\mathcal{L}_{\text{edit}} = \frac{\sum_q M_q \big(v_{\theta,q}(k) - v^{\text{edit}}_{T,q}(k)\big)^2}{\max\{1, \sum_q M_q\}},$$
$$\mathcal{L}_{\text{keep}} = \frac{\sum_q (1 - M_q) \big(v_{\theta,q}(k) - v^{\text{keep}}_{T,q}(k)\big)^2}{\max\{1, \sum_q (1 - M_q)\}}. \tag{11}$$

The index q covers packed tokens and channels. The denominator matches the implementation's clamp for a potentially empty region. In the main configuration, both loss weights are one, giving

$$\mathcal{L}_{\text{CSTAR-OPD}} = \mathcal{L}_{\text{edit}} + \mathcal{L}_{\text{keep}}. \tag{12}$$

Separate normalization keeps the relative contribution of a small edit region from vanishing as its area decreases. For fractional mask values, these weighted losses are generally *not equivalent* to MSE against the interpolated field in Eq. 9; the two supervision errors are weighted independently.

The teacher, both text encoders, the VAE, and the student's base transformer weights are frozen; only the student's transformer LoRA is updated. At inference, the adapted 4B student runs the full denoising schedule with I and c_{edit}. The teacher, identity instruction, and training mask are absent, and the inference interface matches that of the original student.

5 Experiments

We perform OPD training on FLUX.2-klein-base-4B using FLUX.2-klein-base-9B as the teacher across four editing tasks from the CoCoEdit dataset: object addition, object removal, object replacement, and attribute modification. The LoRA is applied only to the 4B student Transformer, while the 9B teacher, VAE, and text encoder remain frozen. The default LoRA configuration uses rank $r = 64$, alpha $= 64$, and dropout $= 0$, giving a scaling factor of $\alpha/r = 1$. We then evaluate the trained model on GIE-Bench.

Training dataset We employ **CoCoEdit-40K** (Wu et al., 2026) as the training dataset for OPD. The dataset contains 39,901 samples spanning seven editing task categories. For training, we select four local editing tasks: object addition, object removal, object replacement, and attribute editing. Among them, there are 6000 pieces of training data for the item replacement task, and 5000 pieces of data for the other tasks. In total, there are 21,000 pieces of training data.

Evaluation benchmark dataset We evaluate our method on **GIE-Bench** Qian et al. (2025), a benchmark for text-guided image editing. The benchmark contains 1,080 image-instruction pairs from 856 unique images, spanning 20 visual content categories and nine editing tasks, with 120 examples per task. It evaluates both functional correctness and content preservation, measuring whether the requested edit is completed and whether unrelated content remains unchanged. Our study focuses on unintended changes outside the target region in local image editing. We therefore evaluate five relevant tasks: object addition, object removal, object replacement, attribute editing, and color change. These tasks contain 600 benchmark examples; after excluding one example with an invalid image link, our evaluation comprises 599 examples.

5.1 Main Results

The masked metrics compare the edited image with the source image outside the target region, while the edit success rate measures whether the requested change was completed. Table 1 compares four models: the original FLUX.2-Klein-base-4B model, Base OPD, and our method. Base OPD uniformly samples one state from steps 0–49 and matches the edit conditioned teacher field over the entire latent field. Our method uniformly samples one state from the early window of steps 0–13 and applies region aligned supervision: the student matches the teacher's edit field inside the target mask and its identity conditioned field outside the mask.

Table 1: Results on GIE-Bench for the 4B and 9B base model, Base OPD, and our method. M- denotes evaluation outside the target edit mask. Arrows indicate the preferred direction.

Method	M-CLIP↑	M-SSIM↑	M-MSE↓	M-PSNR↑	Functional Correctness↑
4B	0.9777	0.9018	820.01	22.89	87.98%
9B	0.9791	0.9096	**605.15**	**24.04**	**90.98%**
Base OPD	0.9777	0.9199	791.31	23.27	88.98%
Ours	**0.9798**	**0.9259**	651.40	23.63	88.98%

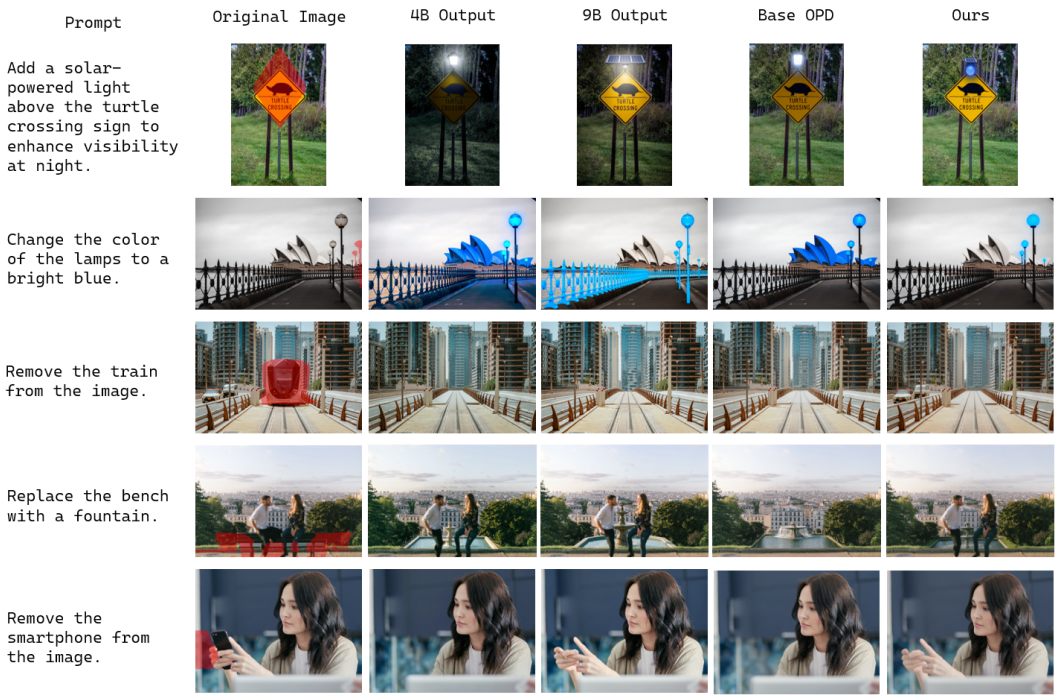


Figure 5: Qualitative comparison of 4B, 9B, Base OPD, and our method across diverse local image editing tasks. Our method accurately performs the requested edits while better preserving non-target regions, leading to reduced editing leakage and improved visual consistency.

Compared with the original 4B model, our method improves M-SSIM from 0.9018 to 0.9259 and M-CLIP from 0.9777 to 0.9798, while reducing M-MSE by 20.56% and increasing M-PSNR by 0.74 dB. The edit success rate also improves from 87.98% to 88.98%. Compared with Base OPD, our method further improves M-SSIM and M-CLIP, reduces M-MSE by 17.68%, and increases M-PSNR by 0.36 dB, with the same edit success rate of 88.98%. These results indicate that early-stage sampling and region-aligned supervision improve content preservation without sacrificing editing success.

Qualitative results in Fig. 5 show that our OPD method not only ensures the success rate of editing but also significantly reduces the problem of editing diffusion. Moreover, in many editing cases, it exceeds the editing retention capability of the 9B model, achieving more precise editing of the target. At the same time, no mask is required during reasoning.

5.2 Ablation Study

Experimental setup We vary three factors: the sampling range, the number of sampled student states per example, and the use of region-aware supervision. The full range includes states 0–49, while the early range includes states 0–13. All variants use the same 9B teacher and 4B student architecture. Table 2 reports content preservation and functional correctness on GIE-Bench.

Table 2: Ablation on GIE-Bench. N is the number of sampled states per training example; Region indicates region-aware supervision. M- denotes evaluation outside the edit mask.

Sampling Range	N	Region	M-CLIP↑	M-SSIM↑	M-MSE↓	M-PSNR↑	Functional Correctness↑
0–49	1	No	0.9777	0.9199	791.31	23.27	**88.98%**
0–13	1	No	0.9779	0.9200	801.40	23.15	88.65%
0–49	1	Yes	0.9799	0.9267	622.32	23.92	**88.98%**
0–13	1	Yes	0.9798	0.9259	651.40	23.63	**88.98%**
0–13	3	Yes	0.9824	0.9323	550.59	23.93	88.48%
0–49	3	Yes	**0.9827**	**0.9353**	**549.33**	**24.82**	87.15%

Effect of temporal and spatial alignment With global supervision and one sampled state, restricting the sampling range from 0–49 to 0–13 yields similar results: M-SSIM changes from 0.9199 to 0.9200, M-MSE from 791.31 to 801.40, and functional correctness from 88.98% to 88.65%. Under uniform sampling, the expected number of student rollout transitions per queried state falls from 24.5 to 6.5, a reduction of 73.5%. This measures rollout length rather than total training time.

Region-aware supervision improves preservation at both sampling ranges. With one state sampled from 0–49, it reduces M-MSE from 791.31 to 622.32 (21.36%), increases M-SSIM from 0.9199 to 0.9267, and raises M-PSNR from 23.27 to 23.92 dB, while functional correctness remains at 88.98%. Within the early range, it reduces M-MSE from 801.40 to 651.40 (18.72%) and increases functional correctness from 88.65% to 88.98%. These comparisons hold the sampling range and number of sampled states fixed, supporting the benefit of distinct teacher targets for the edit and preservation regions.

Comparing the two region-aware settings with one sampled state, the full range achieves slightly stronger preservation: M-MSE is 622.32 versus 651.40, and M-PSNR is 23.92 versus 23.63 dB. Both achieve 88.98% functional correctness. Thus, restricting sampling to early states reduces the expected student rollout length while giving up some preservation performance.

Effect of sampling more states With region-aware supervision over 0–13, increasing the number of sampled states from one to three improves M-SSIM from 0.9259 to 0.9323 and reduces M-MSE from 651.40 to 550.59, while functional correctness decreases from 88.98% to 88.48%. At three sampled states, extending the range to 0–49 yields the strongest preservation results in the table, including an M-MSE of 549.33 and an M-PSNR of 24.82 dB, but functional correctness falls to 87.15%.

We select one state from 0–13 for the final model. This setting ties for the highest functional correctness, improves every preservation metric over global OPD with the same early range, and uses fewer expected student rollout transitions than full-range sampling. The ablation also shows that full-range region-aware supervision can achieve stronger preservation when additional rollout computation is acceptable.

6 Conclusion

We presented CSTAR-OPD, a region-aware on-policy distillation method for local image editing. Semantic-direction and model-switching analyses motivate supervising the 4B student at an early state, while counterfactual teacher guidance separates editing inside the target region from preservation outside it. With one sampled state per example, CSTAR-OPD reduces outside-mask M-MSE by 17.7% relative to global OPD while maintaining the same 88.98% functional correctness. The method requires no mask at inference, offering a practical balance between edit fidelity and preservation.

AI use statement

Generative AI tools were used solely to improve the grammar, clarity, and readability of author-written text. The authors reviewed all suggested edits to ensure that they preserved the intended technical meaning. The authors take full responsibility for the final content of this paper.

Ethics statement

Instruction-guided image editing can be misused to create misleading or non-consensual alterations. Our work evaluates edit fidelity and preservation on existing benchmarks; these metrics do not establish safety for deployment or address privacy and bias in the underlying models and data. Practical use should therefore include appropriate consent, provenance, and safeguards against harmful edits.

Reproducibility statement

The Method section defines the CSTAR-OPD objective and temporal sampling procedure. The Experimental Setup section describes the teacher and student models, datasets, training configuration, evaluation metrics, and comparison methods. The trajectory analyses and ablation studies document the procedures underlying our design choices. These descriptions provide a basis for reproducing the reported comparisons.

References

Black Forest Labs. FLUX.2 [klein]: Towards interactive visual intelligence, 2026. URL https://bfl.ai/blog/flux2-klein-towards-interactive-visual-intelligence.

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*, 2023.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. In *International Conference on Learning Representations*, 2023.

Zhen Fang, Wenxuan Huang, Yu Zeng, Yiming Zhao, Shuang Chen, Kaituo Feng, Yunlong Lin, Lin Chen, Zehui Chen, Shaosheng Cao, and Feng Zhao. Flow-OPD: On-policy distillation for flow matching models. *arXiv preprint arXiv:2605.08063*, 2026.

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*, 2023.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. URL https://arxiv.org/abs/2207.12598.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Dengyang Jiang, Xin Jin, Dongyang Liu, Zanyi Wang, Mingzhe Zheng, Ruoyi Du, Xiangpeng Yang, Qilong Wu, Zhen Li, Peng Gao, Harry Yang, and Steven C. H. Hoi. D-OPSD: On-policy self-distillation for continuously tuning step-distilled diffusion models. *arXiv preprint arXiv:2605.05204*, 2026.

Ming Li, Xin Gu, Fan Chen, Xiaoying Xing, Longyin Wen, Chen Chen, and Sijie Zhu. Superedit: Rectifying and facilitating supervision for instruction-based image editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19206–19215, 2025.

Quanhao Li, Junqiu Yu, Kaixun Jiang, Yujie Wei, Zhen Xing, Pandeng Li, Ruihang Chu, Shiwei Zhang, Yu Liu, and Zuxuan Wu. DiffusionOPD: A unified perspective of on-policy distillation in diffusion models. *arXiv preprint arXiv:2605.15055*, 2026.

Yangshuai Liu, Zheming Li, Jiaao Li, Kang He, Ziliang Lai, Zhitai Liu, and Chengru Song. Anchoring instruction outside mask: Exact reference caching for efficient in-context diffusion transformers. *arXiv preprint arXiv:2608.21229*, 2026.

Yusu Qian, Jiasen Lu, Tsu-Jui Fu, Xinze Wang, Chen Chen, Yinfei Yang, Wenze Hu, and Zhe Gan. Gie-bench: Towards grounded evaluation for text-guided image editing, 2025.

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

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, 2023.

Berk Tinaz, Zalan Fabian, and Mahdi Soltanolkotabi. Emergence and evolution of interpretable concepts in diffusion models. In *Advances in Neural Information Processing Systems*, 2025.

Lidia Troeshestova, Alexander Ustyuzhanin, and Sergey Kastryulin. DuET: Dual expert trajectories for diffusion image editing. *arXiv preprint arXiv:2606.13303*, 2026.

Yuhui Wu, Chenxi Xie, Ruibin Li, Liyi Chen, Qiaosi Yi, and Lei Zhang. CoCoEdit: Content-consistent image editing via region regularized reinforcement learning. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.14068.

Yixian Xu, Kaiyuan Gao, Yuxiang Chen, Yilei Chen, Zecheng Tang, Zihao Liu, Zikai Zhou, Deqing Li, Hao Meng, Kuan Cao, Jiahao Li, Jie Zhang, Liang Peng, Lihan Jiang, Ningyuan Tang, Shengming Yin, Tianhe Wu, Xiaoyue Chen, Yan Shu, Yanran Zhang, Yi Wang, Yu Wu, Yujia Wu, Zekai Zhang, Zhendong Wang, Xiao Xu, Kun Yan, and Chenfei Wu. Qwen-image-2.0-rl technical report. *arXiv preprint arXiv:2606.27608*, 2026.

Zexuan Yan, Yue Ma, Chang Zou, Wenteng Chen, Qifeng Chen, and Linfeng Zhang. Eedit: Rethinking the spatial and temporal redundancy for efficient image editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17474–17484, 2025.

Xuanyu Yi, Zike Wu, Qingshan Xu, Pan Zhou, Joo-Hwee Lim, and Hanwang Zhang. Diffusion time-step curriculum for one image to 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9948–9958, 2024.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6613–6623, June 2024.

Jun Zhou, Jiahao Li, Zunnan Xu, Hanhui Li, Yiji Cheng, Fa-Ting Hong, Qin Lin, Qinglin Lu, and Xiaodan Liang. Fireedit: Fine-grained instruction-based image editing via region-aware vision language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13093–13103, June 2025.

Wei Zhou, Xiongwei Zhu, Zelin Xu, Bo Dong, Lixue Gong, Yongyuan Liang, Meng Chu, Leigang Qu, Lingdong Kong, Wei Liu, and Tat-Seng Chua. Danceopd: On-policy generative field distillation. *arXiv preprint arXiv:2606.27377*, 2026.

A Additional Details on Editing-Direction Formation and Recovery

A.1 Semantic Editing-Direction Analysis

Paired trajectories and state indexing. We analyze 4,000 matched examples, comprising 1,000 examples each of object addition, removal, replacement, and attribute editing. For each example and each model $m \in \{4\mathrm{B}, 9\mathrm{B}\}$, we compare an edit-conditioned trajectory with an identity-conditioned trajectory generated from the same initial noise and source image. The analysis verifies matching sample identifiers, latent shapes, initial-noise hashes, and noise schedules; all 4,000 matched examples pass validation. Each trajectory has $K = 50$ denoising transitions and $K + 1$ latent states: state 0 precedes the first update, and transition k maps state k to state $k + 1$. Thus, the final edit residual is $r_{m,i}(50)$, whereas the final model update is transition $49 \to 50$.

Per-example alignment and aggregation. Expanding Eqs. (1)–(2) to make the example index explicit, we compute

$$C_{m,i}(k) = \frac{\langle r_{m,i}(k), r_{m,i}(50)\rangle}{\|r_{m,i}(k)\|_2 \, \|r_{m,i}(50)\|_2}, \qquad \widetilde{C}_m(k) = \operatorname*{median}_i C_{m,i}(k). \tag{13}$$

Here $r_{m,i}(k)$ is the difference between the edit- and identity-conditioned packed latents for example i. The plotted solid curves are $\widetilde{C}_m(k)$; their shaded bands are the 25th–75th percentiles of the *example-level* $C_{m,i}(k)$ values, rather than confidence intervals for the median. Because the two trajectories share their initial state, $r_{m,i}(0) = 0$ and $C_{m,i}(0)$ is undefined. Consequently, the direction curves and breakpoint fits begin at state 1. By definition, $C_{m,i}(50) = 1$ up to floating-point error. Alignment with the final latent residual is an operational proxy for the editing direction; it does not by itself measure edit success or the magnitude of the residual.

Breakpoint and stability criteria. For each model, we fit the median curve at states $1, \ldots, 50$ with a continuous one-breakpoint regression,

$$\widetilde{C}_m(k) = a_m + b_m k + c_m (k - \tau_m)_+ + \epsilon_k, \qquad (u)_+ = \max(u, 0). \tag{14}$$

We scan integer candidates τ_m, requiring at least five observed states on either side, and select the candidate with the smallest Bayesian information criterion (BIC). The linear fit has two parameters; the hinge fit is charged four parameters, counting its estimated breakpoint. We report $\Delta\mathrm{BIC} = \mathrm{BIC}_{\mathrm{linear}} - \mathrm{BIC}_{\mathrm{hinge}}$ and treat $\Delta\mathrm{BIC} \geq 6$ as support for the fitted change in slope. The pre- and post-breakpoint slopes are b_m and $b_m + c_m$, respectively. Separately, the operational stability state is

$$k_m^{0.9} = \min \left\{ k \geq 1 : \widetilde{C}_m(k + j) \geq 0.9 \text{ for all } j = 0, \ldots, 4 \right\}. \tag{15}$$

The value 0.9 and the five-state window are analysis thresholds, not learned model parameters. We resample examples with replacement 300 times, recompute the median curve and both states, and report the 2.5th–97.5th percentiles of their bootstrap distributions. The BIC calculation uses the 50 statewise median observations; the bootstrap resamples examples, not individual timesteps.

Model	τ_m (95% CI)	Pre-slope	Post-slope	ΔBIC	$k_m^{0.9}$ (95% CI)
4B	5 [5, 5]	0.0356	0.0121	90.15	44 [44, 45]
9B	13 [13, 14]	0.0204	0.0075	105.25	38 [37, 38]

Table 3: Overall semantic-direction results on 4,000 paired examples. τ_m is the fitted change in slope, whereas $k_m^{0.9}$ is the first state of five consecutive median-alignment values of at least 0.9.

At state 1, the median alignment is 0.436 for 9B and 0.288 for 4B; at state 13, it is 0.698 and 0.537, respectively. Thus, 9B has greater alignment with its own final editing residual at these early states, and it reaches the operational stability threshold six states before 4B. The fitted pre-breakpoint slope is numerically larger for 4B than for 9B, so "faster formation" here refers to stronger alignment earlier in the trajectory and earlier stabilization, not to a larger instantaneous slope of the 9B curve. The 9B median curve changes slope at state 13; the 4B curve changes slope at state 5. These change points indicate a slower subsequent increase, not completion of the edit.

Task	n	$\tau_{4\mathrm{B}}$	$k_{4\mathrm{B}}^{0.9}$	$\tau_{9\mathrm{B}}$	$k_{9\mathrm{B}}^{0.9}$
Addition	1,000	9	43	14	37
Removal	1,000	5	46	7	36
Replacement	1,000	6	42	16	36
Attribute	1,000	39	46	14	42

Table 4: Task-stratified change points and operational stability states, computed independently from each task's median alignment curve.

Table 4 shows that the 9B model reaches the alignment threshold earlier than 4B in each task. The fitted change point is more task dependent. In particular, the 4B attribute curve has a fitted breakpoint at state 39 and increases in slope after that point. Therefore, the overall state-5 change point should not be interpreted as a universal boundary for every editing instruction.

Direction versus accumulated edit magnitude. The analysis also measures residual magnitude separately from directional alignment. Let $A^{\text{in}}_{m,i}(k)$ be the root-mean-square magnitude of $r_{m,i}(k)$ over spatial tokens inside the edit mask, using the resized mask as a soft token weight, and define within-example progress $P^{\text{in}}_{m,i}(k) = A^{\text{in}}_{m,i}(k)/(A^{\text{in}}_{m,i}(50)+10^{-8})$. At state 13, the median $P^{\text{in}}_{m,i}(13)$ is only 0.054 for 4B and 0.059 for 9B, while their median direction alignments are already 0.537 and 0.698. Thus, substantial alignment with the final residual appears before most of the final mask-inside residual magnitude has accumulated. This comparison does not imply that velocity magnitude is irrelevant to OPD: the training objective matches teacher velocity vectors, including both direction and magnitude.

A.2 Counterfactual 4B-to-9B Switching

Switch protocol and metric. We evaluate a stratified set of 1,000 examples, with 250 examples per task. For a switch at state k, the hybrid trajectory starts from the cached 4B edit-conditioned state $x_i^{4\text{B}}(k)$ and runs the 9B edit model through scheduler transitions $k, \dots, 49$. The 4B and 9B reference trajectories use matched prompts, source images, initial noise, timesteps, and noise schedules. The implementation checks compatibility of their cached states and scheduler before continuation. We compare final packed latents, without decoding intermediate noisy states. All 1,000 examples have results for the five recorded switch states $5, 13, 38, 44, 50$.

For an individual example, the implementation computes a relative ℓ_2 distance to the 9B endpoint,

$$d_{\text{rel}}(x, y) = \frac{\|x - y\|_2}{\max(\|y\|_2, 10^{-12})}, \qquad R_i(k) = 1 - \frac{d_{\text{rel}}(x_i^{4\text{B}\to 9\text{B}}(k), x_i^{9\text{B}})}{\max\big(d_{\text{rel}}(x_i^{4\text{B}}, x_i^{9\text{B}}), 10^{-12}\big)}. \tag{16}$$

The plotted $R(k)$ is the median of the $R_i(k)$ values across examples, as in Eq. 3. The common normalization by $\|x_i^{9\text{B}}\|_2$ cancels when the denominator is nonzero, yielding the simpler expression in the main text. Values are not clipped to $[0, 1]$; a negative value would indicate a hybrid endpoint farther from 9B than the complete 4B endpoint. This statistic measures final-latent proximity to the 9B reference, not image-editing correctness directly.

Defined endpoints and uncertainty. The recorded recovery CSV contains switch states $5, 13, 38, 44, 50$. The plotted state 0 is the defined full-9B reference endpoint, with $R_i(0) = 1$; it is not an additional recorded hybrid-continuation row in that CSV. State 50 is the complete 4B trajectory, giving $R_i(50) = 0$ by construction. At each recorded state, the analysis draws 1,000 examples with replacement 300 times, computes the median recovery in each resample, and uses the 2.5th and 97.5th percentiles as a 95% bootstrap confidence interval for the median. This interval describes uncertainty in the estimated median across examples; it is distinct from the interquartile bands in the semantic-direction plot. Lines and shaded regions between recorded switch states are visual interpolation and do not represent additional interventions.

Switch state k	Median recovery $R(k)$	95% bootstrap CI
0	1.000	Defined reference
5	0.394	$[0.355, 0.422]$
13	0.214	$[0.194, 0.238]$
38	0.112	$[0.101, 0.121]$
44	0.080	$[0.074, 0.089]$
50	0.000	Defined baseline

Table 5: Overall recovery on 1,000 examples. State 0 is a defined full-9B reference, while states 5–50 are recorded switch states.

As shown in Table 5, the median recovery falls to 0.394 after five 4B transitions and to 0.214 after 13. Thus, delaying 9B intervention to state 13 leaves a substantially smaller improvement toward the 9B endpoint than starting from the shared initial state. The remaining 0.214 recovery also shows that a 9B suffix can still alter the endpoint after state 13.

Paired interval losses. To summarize the effect of delaying intervention between tested states $a < b$, we compute the loss separately for each matched example before aggregating:

$$L_i(a, b) = R_i(a) - R_i(b), \qquad \widetilde{L}(a, b) = \operatorname*{median}_i L_i(a, b), \qquad D(a, b) = \frac{\widetilde{L}(a, b)}{b - a}. \tag{17}$$

Task	n	$R(5)$	$R(13)$	$R(38)$	$R(44)$
Addition	250	0.315	0.151	0.078	0.057
Replacement	250	0.322	0.187	0.100	0.071
Removal	250	0.403	0.200	0.104	0.092
Attribute	250	0.647	0.413	0.176	0.118

Table 6: Task-stratified median recovery. Common anchors $R(0) = 1$ and $R(50) = 0$ are omitted. Attribute editing retains more recovery at states 5 and 13 than the other three tasks.

For the unrecorded state-0 hybrid anchor, $L_i(0, 5) = 1 - R_i(5)$. Table 7 reports the resulting overall statistics. In particular, the 0–5 median loss is 0.606, while the measured 44–50 interval still has a median loss of 0.080. The intervals have unequal lengths, so their total and per-transition losses should be interpreted separately. Moreover, $\mathrm{median}_i[R_i(a) - R_i(b)]$ generally differs from $R(a) - R(b)$; median interval losses need not add to the endpoint loss.

Excluded 9B interval	Transitions	Median paired loss	Loss/transition
0–5	5	0.6059	0.1212
5–13	8	0.0788	0.0098
13–38	25	0.0695	0.0028
38–44	6	0.0192	0.0032
44–50	6	0.0804	0.0134

Table 7: Overall paired recovery losses for adjacent tested states. The 0–5 row uses the defined full-9B reference; all other rows compare two recorded switch states on the same 1,000 examples.

Implications and scope. The direction analysis identifies an early change in the 9B median alignment curve, and the switch experiment shows that a 4B prefix can substantially reduce subsequent recovery toward the 9B endpoint. Together, these observations motivate allocating OPD teacher queries to the early states, where the student's semantic editing trajectory is still being established. The switch intervention replaces the entire remaining student trajectory with 9B updates, whereas OPD matches teacher and student velocity fields locally at selected student-visited states. Consequently, the switch results locate a sensitive interval but do not establish that early-only OPD maximizes editing success. A matched-budget comparison of early, late, and full-range OPD sampling is required for that training claim. The nonzero 44–50 loss and task variation further show that later states remain relevant to the final 9B endpoint.