FLOW-LQR: FLOW-LOCALIZED LINEAR QUADRATIC REGULATOR FOR MULTI-TURN IMAGE EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Instruction-based image editors enable iterative visual refinement, but successive edits can accumulate errors and degrade image quality. Our observations indicate that structured degradation arises even when an editor is asked to reconstruct an image without semantic changes, affecting both edited and unedited regions during subsequent editing. Preserving unedited pixels can limit this accumulation, but requires accurate localization, correction within edited regions, and coherent integration of generated and preserved content. We introduce **Flow-LQR**, a flow-localized linear quadratic regulator for stable multi-turn image editing. Complementary flow responses, calibrated by a reconstruction observation, identify the spatial region of the requested edit. We then reuse the flow predicted by the frozen diffusion transformer as an implicit representation of degradation. Latent residuals at the final denoising step provide disturbance observations in unchanged regions, and harmonic completion extends them into the edited region. A gain-preserving LQR coordinates generated and preserved content while compensating for this disturbance, followed by pixel restoration outside the localized region. We also introduce **MTE-Bench** for evaluating cumulative degradation over successive edits. Experiments demonstrate superior image preservation over open-source editors and the closed-source editor on both MTE-Bench and Banana100 benchmarks, with improved perceptual quality over the native backbone. Flow-LQR requires no backbone training or external segmentation models, and its disturbance observer reuses existing flow predictions without additional inference costs.

1 INTRODUCTION

Instruction-based image editors are evolving from isolated generation tools into visual interfaces that support extended interaction. A single natural-language request can now modify the appearance of an object, insert new content, or transform a scene (Brooks et al., 2022; Zhang et al., 2023; Black Forest Labs et al., 2025; Wu et al., 2025). Visual agents further extend this interaction by planning sequences of operations, invoking editing tools, and revising results through visual feedback (Ye et al., 2026a; Shen et al., 2026; Yao et al., 2026). In such workflows, the unit of computation extends beyond a single instruction-image pair to an iterative refinement process, in which successive requests can lead to multiple rounds of editing before the desired result is reached.

Although leading image editing models such as FLUX.2 and Qwen-Image-Edit are effective at individual edits (Black Forest Labs, 2025; Wu et al., 2025), their outputs can deteriorate under successive editing. Subjects can deform, textures become over-smoothed or over-sharpened, and colors drift (Tang et al., 2026; Liao et al., 2026). Appearance errors introduced during stochastic generation at each turn are carried into subsequent inputs and can accumulate into visible degradation, as illustrated in Figure 1(a). Despite its prevalence, this problem remains unresolved by existing approaches (Chen et al., 2025; Zhou et al., 2025). In addition, multi-turn evaluations have begun to emerge (Tang et al., 2026), but systematic assessment of cumulative degradation remains limited.

To investigate the source of this systematic degradation, we examine reconstruction without a semantic edit by asking the editor to reproduce the input image unchanged. Our observations indicate that even this operation introduces structured appearance degradation. These changes extend across the image, affecting both regions targeted by later edits and regions that should remain unchanged. As reconstruction or editing is repeated, visible degradation persists and accumulates instead of

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

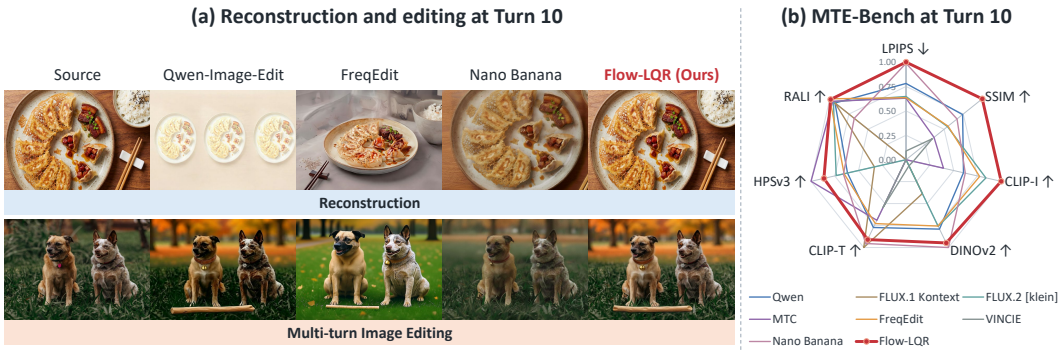

Figure 1: **Recursive reconstruction and multi-turn image editing.**

naturally recovering. This suggests that part of the degradation arises from the image generation process itself and can affect both edited and unedited content.

A natural remedy is to localize the requested edit and directly preserve pixels from the preceding image outside that region. Since edits are often spatially sparse, separating edited and unedited regions can substantially reduce the degradation carried forward. However, this simple strategy faces three challenges. First, instruction-based editors generally do not provide an explicit mask of the requested change, and pixel differences between the input and output mix semantic edits with global appearance drift, making accurate localization unreliable. Second, copying unchanged pixels leaves degradation within the edited region uncorrected, allowing it to accumulate over subsequent turns. Third, the newly generated region must be coherently integrated with the preserved surroundings. Direct compositing can introduce visible seams or inconsistencies in color, texture, and local structure across the boundary. Further generative refinement may reconcile these differences, but can also introduce additional appearance drift. Effective preservation therefore requires local coordination that integrates the requested change while controlling further degradation.

To address these challenges, we propose **Flow-LQR**, a flow-localized linear quadratic regulator for stable multi-turn image editing. First, we identify the requested edit from complementary early and late flow responses calibrated by a keep-image observation. Second, we reuse the same model responses to represent the degradation introduced during generation. The frozen diffusion transformer thereby serves as an implicit degradation encoder by representing changes from the current input through a latent residual at the final denoising step. We retain this discrepancy only in nominally unchanged regions, where it is less entangled with the requested semantic change, and use harmonic completion to infer a spatially coherent disturbance field inside the edit. This links localization and disturbance observation through a shared latent representation, without a separate learned encoder or additional model evaluations. Finally, a gain-preserving LQR coordinates the generated region with its surroundings while applying disturbance compensation and maintaining guidance toward the native candidate. We additionally introduce **MTE-Bench** to evaluate preservation, instruction alignment, and image quality throughout successive edits.

Our contributions are summarized below.

- We introduce **Flow-LQR**, a training-free, plug-and-play framework designed to improve multi-turn editing with existing flow-based image editors, requiring no backbone updates or external segmentation models.
- We unify edit localization and degradation modeling through cached DiT flow responses. Latent residuals and harmonic completion provide a disturbance estimate, which a gain-preserving LQR uses to coordinate the edit without weakening candidate guidance.
- We introduce **MTE-Bench**, a multi-turn image editing benchmark covering diverse natural scenes and sequences of interdependent editing instructions. It enables systematic evaluation of how image fidelity, perceptual quality, and instruction alignment evolve as edited outputs are recursively reused.
- Experiments on MTE-Bench demonstrate improved multi-turn image preservation over the evaluated open-source methods and the closed-source editor Nano Banana, with consistent gains across successive editing turns.

2 Related Work

Instruction-based Image Editing. Instruction-based image editing maps an input image and a natural-language request to a modified image while preserving content unrelated to the request. Building on latent diffusion representations (Rombach et al., 2022), early approaches control attention, infer edit masks, or guide latent features to maintain source correspondence, as in Prompt-to-Prompt, DiffEdit, ChatFace, and DragonDiffusion (Hertz et al., 2023; Couairon et al., 2022; Yue et al., 2023; Mou et al., 2024). InstructPix2Pix learns editing directly from paired images and instructions (Brooks et al., 2022), while MagicBrush provides manually annotated editing supervision (Zhang et al., 2023). Emu Edit and SmartEdit further improve instruction understanding through recognition tasks and multimodal reasoning (Sheynin et al., 2024; Huang et al., 2024). More recently, FLUX.1 Kontext, Qwen-Image, and FLUX.2 have expanded the scope of in-context generation and editing with flow-based models (Black Forest Labs et al., 2025; Wu et al., 2025; Black Forest Labs, 2025). Complementary inference methods control these models through attention injection, inversion-free transport, and local transport objectives, including Stable Flow, FlowEdit, and ChordEdit (Avrahami et al., 2025; Kulikov et al., 2025; Lu et al., 2026). These advances improve individual source-to-target transformations. Our focus is the cumulative appearance degradation that arises when their outputs are repeatedly reused as editing inputs.

Multi-turn Image Editing. Multi-turn editing requires an editor to execute the current request while retaining earlier changes and untouched content. MTC regulates trajectories and attention to improve sequential consistency (Zhou et al., 2025), and VINCIE learns in-context editing from temporally structured video examples (Qu et al., 2025). FreqEdit constructs an analytic reference velocity from the current input and uses its discrepancy from the editing velocity to adapt spatial frequency injection, with path compensation to preserve editability (Liao et al., 2026). Causal memory and multi-turn optimization offer further ways to retain editing history (Xu et al., 2026; Ye et al., 2026b), while Agent Banana, IMAGAgent, and PhotoAgent organize editing through planning and visual feedback (Ye et al., 2026a; Shen et al., 2026; Yao et al., 2026). Evaluation efforts such as EdiVal-Agent and Banana100 examine object-level behavior and degradation under repeated image transformations (Chen et al., 2025; Tang et al., 2026). Flow-LQR instead reuses the predicted flow from the editor for both localization and latent disturbance observation. Harmonic completion extends exterior observations into the edit, and a local regulator compensates for the inferred disturbance while preserving candidate guidance. We additionally introduce MTE-Bench to evaluate preservation, instruction alignment, and image quality throughout ten-turn editing sessions.

3 Preliminaries

Multi-turn Image Editing. At turn k, an instruction-based image editing model takes the preceding output I_{k-1} and a new request p_k to produce

$$I_k = \mathcal{E}_\theta(I_{k-1}, p_k, \epsilon_k), \tag{1}$$

where ϵ_k is the sampling noise. At each turn, we align the preceding output I_{k-1} to the working resolution and denote the resulting input by X, with latent representation $z_p = \mathrm{Enc}(X)$. Conditioned on this input and the current request, the frozen editor produces a native candidate z_c and its decoded image $Y = \mathrm{Dec}(z_c)$, which Flow-LQR refines before passing the result to the next turn.

Conditional Flow and Local Control. Rectified flow and flow matching parameterize generation with a conditional velocity field (Liu et al., 2022; Lipman et al., 2023). The training interpolation between a clean latent z_0 and noise ϵ is $\widetilde{z}_\sigma = (1-\sigma)z_0 + \sigma\epsilon$. At inference, the noisy latent follows

$$\frac{dz_\sigma}{d\sigma} = v_\theta(z_\sigma, \sigma, c), \qquad z_{i+1} = z_i + (\sigma_{i+1} - \sigma_i)v_i, \tag{2}$$

with decreasing noise levels $\sigma_{i+1} < \sigma_i$. Here c contains the image and instruction, and v_i includes classifier-free guidance from the editor (Ho & Salimans, 2022). At a fixed state, the predicted clean endpoint is $\widehat{z}_0 = z_\sigma - \sigma v$. An additive velocity correction u therefore gives

$$z_0(u) = \widehat{z}_0 - \sigma u. \tag{3}$$

This linear relation lets us balance endpoint targets and control effort with a quadratic objective. We derive the resulting one-interval LQR in the next section, holding the current predictions and targets fixed. We write $\|a\|_A^2 = a^\top A a$ for $A \succeq 0$.

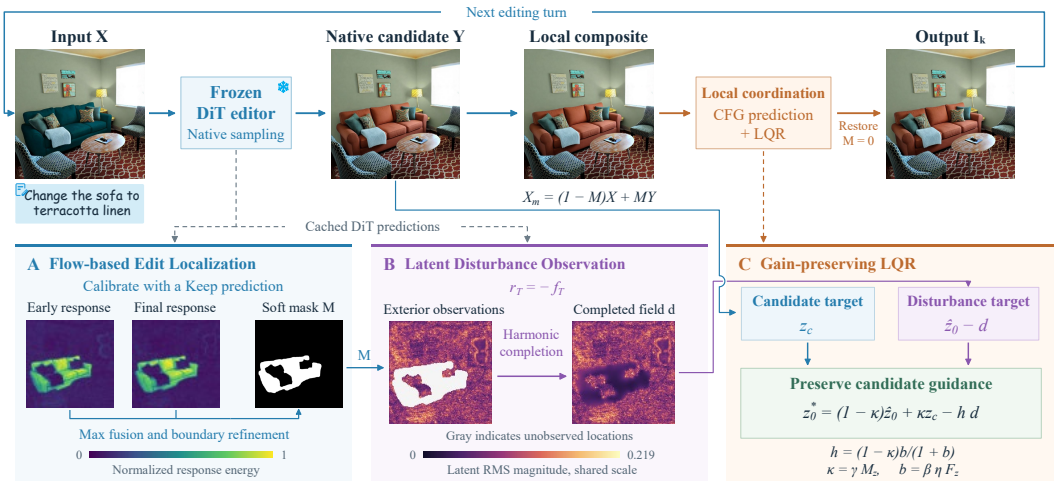

Figure 2: **Overview of Flow-LQR.** Cached DiT predictions support edit localization and latent disturbance estimation, followed by gain-preserving LQR coordination and exterior restoration.

4 FLOW-LQR

As shown in Figure 2, Flow-LQR consists of three components. First, calibrated early and late flow responses define the edit support. Second, the latent residual at the final denoising step provides disturbance observations outside that support, and harmonic completion extends them into the edited region. Both localization and disturbance estimation reuse the predictions from the frozen editor. Finally, a gain-preserving LQR coordinates the candidate with its preserved surroundings while compensating for the estimated disturbance.

4.1 FLOW-BASED EDIT LOCALIZATION

The localization module estimates where the current instruction changes the input image. We compare the predicted clean endpoint with the input latent, calibrate the discrepancy across channels, and combine observations from two sampling steps to construct the edit support.

Clean Endpoint Discrepancy. At sampling step i, the frozen editor predicts velocity v_i from the noisy latent z_i at noise level σ_i. Its estimated clean endpoint and the discrepancy from the current input latent z_p are

$$\widehat{z}_{0,i} = z_i - \sigma_i v_i, \qquad f_i = z_p - \widehat{z}_{0,i}. \tag{4}$$

The endpoint estimates the latent image produced at zero noise under the current prediction. Comparing it with z_p provides a spatial cue for the requested change. Both intended edits and prediction errors contribute to this discrepancy, so it does not directly provide a ground-truth segmentation. The same discrepancy can be expressed through an analytic reference velocity. A straight path from z_i to the current input latent z_p has average velocity

$$v_i^{\mathrm{ref}} = \frac{z_p - z_i}{0 - \sigma_i} = \frac{z_i - z_p}{\sigma_i}. \tag{5}$$

Substituting this reference into the velocity difference gives $f_i = \sigma_i(v_i - v_i^{\mathrm{ref}})$, which measures the predicted departure from the input in clean latent coordinates.

Channel Calibration. Latent channels differ in their sensitivity to incidental changes, so a large endpoint discrepancy does not always indicate a requested edit. We estimate this sensitivity with one additional keep-image prediction at the first state of the native trajectory,

$$v_1^{\mathrm{keep}} = v_\theta(z_1, \sigma_1, X, p_{\mathrm{keep}}), \qquad r_1 = \sigma_1(v_1^{\mathrm{keep}} - v_1^{\mathrm{ref}}). \tag{6}$$

For channel c, let $m_c = \frac{1}{N}\sum_j r_1(j,c)^2$ denote its mean squared reconstruction discrepancy over N spatial tokens, and let $\bar{m}$ be the mean across channels. We use these statistics to normalize the edit

response,

$$a_i(j) = \sqrt{\frac{1}{C}\sum_{c=1}^{C} \frac{f_i(j,c)^2}{(1-\lambda)m_c + \lambda\bar{m} + \varepsilon_{\text{cal}}}}, \tag{7}$$

where $\lambda \in [0,1]$ regularizes the channel scales toward their shared mean, while $\varepsilon_{\text{cal}} > 0$ stabilizes the normalization. These scales downweight channels that fluctuate strongly under the keep-image instruction, so the resulting score measures the edit response relative to the reconstruction baseline.

Complementary Flow Aggregation. Flow responses evolve throughout denoising, and different stages can reveal different parts of the requested edit. At higher noise levels, early predictions can indicate the emerging spatial extent of the change. As denoising progresses, later predictions can resolve object boundaries and appearance details that were less evident earlier. We therefore combine an early prediction at i_e with the final prediction at T. After normalizing each response map, we take their pointwise maximum,

$$q_i(j) = \frac{a_i(j)}{\max(\max_\ell a_i(\ell), \varepsilon_{\text{norm}})}, \qquad s(j) = \max\{q_{i_e}(j), q_T(j)\}. \tag{8}$$

This aggregation retains regions with a strong response at either stage, allowing the two observations to provide complementary coverage. Both predictions are cached from the same native editing trajectory and add no model evaluations.

We upsample the aggregated score to the image grid, identify a high-response core through adaptive thresholding, and extend it into connected regions with weaker responses. Boundary refinement and feathering then produce the soft edit support $M \in [0,1]^{H\times W}$ used for both compositing and disturbance observation. Sampling choices and mask-processing parameters are specified in Appendix A.

4.2 DiT-based Latent Disturbance Observation

Localizing the requested edit allows us to preserve the preceding image outside the edit support, preventing incidental changes in those regions from propagating to subsequent turns. The generated content inside the support, however, can still suffer from degradation. We therefore use discrepancies in nominally unchanged regions to characterize the disturbance introduced during generation and infer its spatial pattern within the edit. Reusing latent residuals from the frozen DiT provides this representation without an external encoder or additional model evaluations, forming the basis for subsequent LQR compensation.

Exterior Disturbance Observation. We construct the latent residual from the cached DiT prediction at the final denoising step,

$$r_T = \sigma_T\big(v_T^{\text{ref}} - v_T\big) = \widehat{z}_{0,T} - z_p = -f_T. \tag{9}$$

Within the edit support, this residual contains the requested content change as well as incidental degradation. We therefore retain observations only in the unedited region, where the content should remain unchanged. Let Ω_z denote the corresponding latent locations outside the support and a surrounding safety margin. The observed disturbance is

$$d_j^{\text{obs}} = r_{T,j}, \qquad j \in \Omega_z. \tag{10}$$

These exterior observations provide evidence of generation-induced drift while limiting interference from the intended edit.

Latent Harmonic Completion. To infer the disturbance inside the edit, we model its spatially coherent component with a smooth latent field constrained by the exterior observations. For the spatial neighborhood graph $\mathcal{G} = (V, \mathcal{E})$, we solve

$$\widehat{d} = \arg\min_D \frac{1}{2} \sum_{(j,\ell)\in\mathcal{E}} \|D_j - D_\ell\|_2^2 \quad \text{s.t.} \quad D_j = r_{T,j}, \quad j \in \Omega_z. \tag{11}$$

The constraints preserve the observed exterior, while the smoothness objective extends the disturbance across the unobserved region with minimal local variation. The completed field is mapped to the latent layout of the editor as d, which supplies the disturbance target for the LQR. This transfer relies on spatial continuity of the disturbance. Solver details and observation-availability checks are provided in Appendix D.

4.3 GAIN-PRESERVING LINEAR QUADRATIC REGULATOR

Given the edit support and completed disturbance, we next regulate a local refinement of the composite to integrate the generated content with its preserved surroundings.

Local Coordination. We first combine the preceding image X and native candidate Y using

$$X_m = (1 - M) \odot X + M \odot Y. \tag{12}$$

Let F be a spatial dilation of $M > 0$ that provides room to coordinate the compositing boundary. Mapping M and F to the latent grid gives M_z and F_z. Using the original round noise ϵ, we initialize the composite at the final positive noise level σ,

$$z_m = F_z \odot \mathrm{Enc}(X_m) + (1 - F_z) \odot z_p, \qquad z_\sigma = (1 - \sigma) z_m + \sigma\epsilon. \tag{13}$$

A single CFG prediction gives $\widehat{z}_0 = z_\sigma - \sigma v_{\mathrm{CFG}}$, the uncontrolled clean endpoint of this local coordination step.

Edit and Disturbance Targets. We use the native candidate z_c as the target for preserving the requested edit and $\widehat{z}_0 - d$ as the target for disturbance removal. Under the local dynamics in Eq. (3), adding control u gives the endpoint $\widehat{z}_0 - \sigma u$, whose deviations from these targets are $e - \sigma u$ and $d - \sigma u$, with $e = \widehat{z}_0 - z_c$. Balancing the two deviations against the magnitude of the endpoint correction gives the quadratic objective

$$J_b(u) = \underbrace{\tfrac{1}{2} \left\| e - \sigma u \right\|_{A_b}^2}_{\text{candidate target}} + \underbrace{\tfrac{1}{2} \left\| d - \sigma u \right\|_{B}^2}_{\text{disturbance target}} + \underbrace{\tfrac{1}{2} \sigma^2 \left\| u \right\|_{2}^2}_{\text{control effort}}. \tag{14}$$

The diagonal weights A_b and $B = \mathrm{diag}(b)$ determine the relative contributions of candidate retention and disturbance removal, with $b = \beta\eta F_z$ restricting compensation to the coordination region. Here β sets the compensation strength, while η enables it only when exterior observations are available.

Gain-preserving Feedback. With a fixed candidate penalty, increasing the disturbance weight reduces the relative influence of the candidate target. We therefore specify the desired candidate coefficient as $\kappa = \gamma M_z$, with $0 \le \gamma < 1$, and adjust the penalty according to

$$A_b = \mathrm{diag}\left(\frac{\kappa(1 + b)}{1 - \kappa} \right). \tag{15}$$

Proposition 1 (Gain-preserving terminal feedback)**.** *For fixed predictions, masks, and feedback strength, with $\sigma > 0$, $0 \le \kappa < 1$, and $b \ge 0$, Eq. (14) has the unique minimizer*

$$\begin{aligned} u^* &= (I + A_b + B)^{-1} (A_b e + Bd)/\sigma \\ &= \frac{\kappa \odot e}{\sigma} + \frac{(1 - \kappa) \odot b}{1 + b} \odot \frac{d}{\sigma}. \end{aligned} \tag{16}$$

Its unprojected endpoint is

$$z_0^* = (1 - \kappa) \odot \widehat{z}_0 + \kappa \odot z_c - \frac{(1 - \kappa) \odot b}{1 + b} \odot d. \tag{17}$$

The coefficient of the candidate is exactly κ, independent of b.

The parameter β therefore adjusts disturbance compensation without weakening candidate guidance, with smaller values reducing both the correction strength and local sensitivity to errors in d (Appendix C.4).

Output Restoration. We apply the optimal control u^* to the predicted velocity and execute the final Euler interval. After clamping the latent exterior of F_z to the preceding-image reference and decoding the result as Y_r, we form the output

$$I_k = (1 - M) \odot X + M \odot Y_r. \tag{18}$$

The output combines the regulated content within the edit support with the unchanged input pixels where $M = 0$, then becomes the input for the next editing turn.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Evaluation Benchmarks. We construct MTE-Bench using 100 natural photographs collected from Pexels, covering people, animals, indoor scenes, and everyday objects. For each image, we construct ten progressive editing instructions using scene-specific templates grounded in the visible content. The sequences cover object addition and removal, attribute modification, material changes, and background replacement, with later requests building on earlier edits to assess cumulative degradation and retention of editing history. We additionally report the multi-turn editing comparison on Banana100 to evaluate cumulative degradation (Tang et al., 2026). Dataset construction and evaluation protocols are detailed in Appendix.

Evaluation Metrics. We evaluate multi-turn image editing across four complementary dimensions of image fidelity, semantic consistency, text alignment, and perceptual quality. LPIPS and SSIM assess perceptual and structural fidelity, respectively (Zhang et al., 2018; Wang et al., 2004), while CLIP-I and DINOv2 measure semantic similarity between edited and source images (Radford et al., 2021; Oquab et al., 2023). We further use CLIP-T to measure image-text alignment and HPSv3 and RALI to assess overall perceptual quality (Ma et al., 2025; Zhao et al., 2026). Metric computation details are provided in Appendix A.

Baselines. We compare Flow-LQR with the open-source image editors Qwen-Image-Edit (Wu et al., 2025), FLUX.1 Kontext (Black Forest Labs et al., 2025), and FLUX.2 [klein] (Black Forest Labs, 2025). The comparison also includes the multi-turn editing methods Multi-Turn Consistent Image Editing (MTC) (Zhou et al., 2025), FreqEdit (Liao et al., 2026), and VINCIE (Qu et al., 2025), together with the closed-source editor Nano Banana. We use the respective official implementations and documented inference settings as the basis for baseline evaluation, with model-specific adaptations detailed in Appendix B.

5.2 COMPARISON WITH PRIOR METHODS

Qualitative Results. Figure 3 compares successive edits of a portrait and a vase arrangement. In the portrait sequence, Flow-LQR retains facial identity, pose, and framing while changing clothing, accessories, and background. The FLUX outputs show pronounced changes in facial texture and contrast, FreqEdit alters the pose and appearance, and MTC replaces the original portrait with a different composition. VINCIE also retains eyeglasses after their removal was requested. In the vase sequence, Flow-LQR preserves the ring-shaped silhouette and surrounding objects while updating material, flowers, and background. Several competing outputs deform the vase, place the flower inside the central opening rather than the vase mouth, or omit the requested flower. These differences highlight the value of restricting updates to the intended region while maintaining earlier edits and scene structure. Flow-LQR couples this localized preservation with coordination anchored to the native candidate, allowing the requested change to remain part of the regulated output.

Quantitative Results. Tables 1 and 2 organize the comparison around source fidelity, text alignment, and perceptual quality across successive editing turns. On MTE-Bench, Flow-LQR leads the full-benchmark methods on the reference-based fidelity measures at both checkpoints. Its advantage over the native editor persists as edited outputs become the inputs to later requests, while its perceptual-quality scores remain competitive with the strongest baselines. On Banana100, Flow-LQR also leads the reference-based fidelity measures while improving text alignment and perceptual quality over the native editor. Its stronger structural and semantic preservation persists across the reported checkpoints, indicating that the benefit extends beyond the scenes and instruction sequences in MTE-Bench. Relative to Nano Banana, Flow-LQR better preserves source content and achieves stronger perceptual-quality scores, while Nano Banana retains an advantage in text alignment. MTC leads the reference-free quality measures but exhibits weaker fidelity. This contrast favors evaluating preservation, alignment, and quality jointly, and is consistent with the aim of compensating appearance drift without weakening guidance toward the native edit.



Figure 3: **Multi-turn qualitative comparison on MTE-Bench.** Successive rows show editing checkpoints with abbreviated instructions.

Table 1: **Multi-turn editing on MTE-Bench.** Best and second-best results are bold and underlined.

	Method	Turn 5							Turn 10						
		LPIPS↓	SSIM↑	CLIP-I↑	DINOv2↑	CLIP-T↑	HPSv3↑	RALI↑	LPIPS↓	SSIM↑	CLIP-I↑	DINOv2↑	CLIP-T↑	HPSv3↑	RALI↑
Multi-Turn Editing	Qwen-Image-Edit	0.234	0.752	0.882	0.856	0.215	7.471	4.082	0.516	0.520	0.828	0.762	0.251	7.371	4.117
	FLUX.1 Kontext	0.472	0.493	0.816	0.783	0.225	7.125	4.085	0.709	0.330	0.764	0.623	**0.256**	6.929	4.115
	FLUX.2 [klein]	0.276	0.716	0.896	0.862	0.204	7.632	4.097	0.549	0.469	0.853	0.751	0.234	7.557	4.139
	MTC	0.519	0.453	0.821	0.519	0.224	7.600	4.034	0.553	0.424	0.806	0.490	0.249	**7.971**	4.147
	FreqEdit	0.330	0.638	0.894	0.855	0.217	7.569	**4.107**	0.552	0.471	0.846	0.750	0.250	7.427	**4.189**
	VINCIE	0.499	0.544	0.817	0.619	**0.228**	6.663	3.390	0.687	0.418	0.772	0.514	0.246	6.408	3.125
	Nano Banana	0.292	0.592	0.885	0.872	0.227	7.729	3.930	0.465	0.501	0.830	0.834	0.255	7.438	3.849
	Flow-LQR (Ours)	**0.172**	**0.828**	**0.912**	**0.891**	0.227	**7.699**	4.098	**0.462**	**0.586**	**0.870**	**0.816**	0.254	7.759	4.179

Human Preference Evaluation. We conducted a human evaluation with 20 participants, each rating 50 groups of outputs from all eight methods on a five-point scale for fidelity, text alignment, and aesthetics. Flow-LQR achieves the highest mean fidelity and aesthetic ratings, with text alignment close to the highest-rated method. These results support improved visual preservation and quality while maintaining instruction alignment. The protocol and mean ratings are presented in Appendix E.

5.3 Ablation Study

Table 3 compares the predefined configurations on MTE-Bench. All share the frozen backbone, sampling settings, and prescribed seeds, while each configuration follows its own recursive trajectory. We next analyze the effectiveness of flow-based localization, latent disturbance modeling, and gain-preserving regulation in turn.

Effectiveness of Flow-based Localization. Adding flow-based localization to the native editor improves structural and semantic fidelity at both checkpoints, together with higher HPSv3 scores. This stage uses direct local compositing to preserve content outside the estimated edit region. The

Table 2: **Recursive evaluation on Banana100.** Best and second-best results are bold and underlined. Evaluation protocols are detailed in Appendix A.

	Method	Turn 5							Turn 10						
		LPIPS↓	SSIM↑	CLIP-I↑	DINOv2↑	CLIP-T↑	HPSv3↑	RALI↑	LPIPS↓	SSIM↑	CLIP-I↑	DINOv2↑	CLIP-T↑	HPSv3↑	RALI↑
Multi-Turn Editing	Qwen-Image-Edit	0.429	0.452	0.881	0.745	0.243	5.584	3.812	0.442	0.411	0.882	0.700	0.240	6.080	3.999
	FLUX.1 Kontext	0.548	0.238	0.837	0.811	0.230	5.297	2.895	0.638	0.197	0.786	0.673	0.217	4.629	2.428
	FLUX.2 [klein]	0.386	0.424	0.879	0.794	0.245	5.732	3.789	0.495	0.347	0.851	0.712	0.232	5.804	3.856
	MTC	0.629	0.344	0.703	0.257	0.152	**7.142**	**4.179**	0.651	0.329	0.683	0.226	0.147	**7.271**	**4.237**
	FreqEdit (Qwen)	0.384	0.621	0.855	0.743	0.237	5.710	3.877	0.428	0.535	0.832	0.670	0.238	5.665	4.051
	VINCIE	0.688	0.268	0.801	0.478	0.213	4.080	2.560	0.814	0.224	0.749	0.248	0.179	3.332	2.199
	Nano Banana	0.316	0.500	0.894	0.816	**0.260**	6.341	3.709	0.340	0.429	0.904	0.856	**0.267**	6.920	3.685
	Flow-LQR (Ours)	**0.283**	**0.625**	**0.916**	**0.820**	0.249	6.374	3.910	**0.304**	**0.584**	**0.908**	**0.863**	0.243	7.015	4.062

Table 3: **Cumulative component ablations on MTE-Bench.** Components are added in order, best and second-best results are bold and underlined.

Method	Turn 5							Turn 10						
	LPIPS↓	SSIM↑	CLIP-I↑	DINOv2↑	CLIP-T↑	HPSv3↑	RALI↑	LPIPS↓	SSIM↑	CLIP-I↑	DINOv2↑	CLIP-T↑	HPSv3↑	RALI↑
Baseline	0.234	0.752	0.882	0.856	0.215	7.471	4.082	0.516	0.520	0.828	0.762	0.251	7.371	4.117
+ Flow-based Localization	0.181	0.824	0.907	0.884	0.206	7.500	4.089	0.463	0.584	0.860	0.803	0.240	7.548	4.157
+ Latent Disturbance Modeling	0.174	0.820	0.911	0.887	0.211	7.646	4.091	**0.462**	0.579	0.866	0.813	0.253	7.392	4.112
+ Gain-preserving Regulation	**0.172**	**0.828**	**0.912**	**0.891**	**0.227**	**7.699**	**4.098**	**0.462**	**0.586**	**0.870**	**0.816**	**0.254**	**7.759**	**4.179**

improvement therefore characterizes the contribution of localized preservation before disturbance modeling or regulation is introduced. Text alignment and quality measures remain important for assessing whether stronger source preservation also retains the requested changes.

Effectiveness of Latent Disturbance Modeling. Latent disturbance modeling characterizes the appearance errors that accumulate during multi-turn editing, but it does not directly modify the edited image. To evaluate this component, we combine it with CFG-style guidance that applies the estimated disturbance as a correction. This intermediate configuration excludes LQR guidance, allowing us to distinguish disturbance modeling with a simple guidance rule from the gain-preserving regulation used by the complete method.

This variant reduces perceptual drift and improves semantic fidelity and text alignment at both checkpoints. Structural similarity and later perceptual-quality scores nevertheless show trade-offs relative to localization alone, indicating that applying a disturbance estimate without coordinated regulation does not uniformly improve the edited output. This motivates balancing disturbance correction with guidance toward the native candidate.

Effectiveness of Gain-preserving Regulation. The final stage is the complete Flow-LQR method, which coordinates the native candidate and the disturbance target while preserving candidate guidance. Table 3 shows that the proposed regulator improves structural and semantic fidelity, instruction alignment, and perceptual quality over CFG-style disturbance guidance at both checkpoints. It also recovers the structural consistency and later visual quality lost under the simpler correction. These gains support the benefit of jointly balancing the native-candidate and disturbance targets, allowing degradation compensation to maintain guidance toward the requested edit. Efficiency and limitations are discussed in Appendix F.

6 Conclusion

We presented Flow-LQR, a training-free framework that uses a shared flow representation to localize edits and model cumulative degradation. Cached DiT predictions supply both the edit response and exterior latent residuals, which harmonic completion extends into the edited region to guide a gain-preserving LQR. On MTE-Bench, this combination improves preservation across ten-turn sessions relative to the evaluated open-source editors and Nano Banana, with higher perceptual quality than the native backbone. The complete pipeline requires three additional transformer evaluations per turn for calibration and coordination, while disturbance observation reuses existing predictions without further model evaluations. Future work will address non-smooth degradation and uncertain support, together with more direct evaluation of semantic fidelity over long interactions.

AI USE STATEMENT

In preparing the final manuscript, we used generative AI tools only for language polishing and improving readability. For the final version, all research tasks beyond language polishing were independently completed by the authors without generative AI assistance. We have reviewed all AI-assisted language revisions to ensure that they preserve the intended meaning and technical accuracy. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Omri Avrahami, Or Patashnik, Ohad Fried, Egor Nemchinov, Kfir Aberman, Dani Lischinski, and Daniel Cohen-Or. Stable flow: Vital layers for training-free image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

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.

Black Forest Labs. FLUX.2: Frontier visual intelligence. Official model and inference repository, 2025. URL `https://github.com/black-forest-labs/flux2`.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2022.

Tianyu Chen, Yasi Zhang, Zhi Zhang, Peiyu Yu, Shu Wang, Zhendong Wang, Kevin Lin, Xiaofei Wang, Zhengyuan Yang, Linjie Li, et al. Edival-agent: An object-centric framework for automated, fine-grained evaluation of multi-turn editing, 2025.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance, 2022.

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, 2022.

Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, and Ying Shan. Smartedit: Exploring complex instruction-based image editing with multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

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*, 2025.

Yucheng Liao, Jiajun Liang, Kaiqian Cui, Baoquan Zhao, Haoran Xie, Wei Liu, Qing Li, and Xudong Mao. Freqedit: Preserving high-frequency features for robust multi-turn image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 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*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022.

Liangsi Lu, Xuhang Chen, Minzhe Guo, Shichu Li, Jingchao Wang, and Yang Shi. Chordedit: One-step low-energy transport for image editing, 2026.

Yuhang Ma, Yunhao Shui, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. HPSv3: Towards wide-spectrum human preference score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://arxiv.org/abs/2508.03789.

Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. Dragondiffusion: Enabling drag-style manipulation on diffusion models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2307.02421.

Maxime Oquab, Timothee Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision, 2023.

Leigang Qu, Feng Cheng, Ziyan Yang, Qi Zhao, Shanchuan Lin, Yichun Shi, Yicong Li, Wenjie Wang, Tat-Seng Chua, and Lu Jiang. Vincie: Unlocking in-context image editing from video, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Fei Shen, Chengyu Xie, Lihong Wang, Zhanyi Zhang, Xin Jiang, Xiaoyu Du, and Jinhui Tang. Imagagent: Orchestrating multi-turn image editing via constraint-aware planning and reflection, 2026.

Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu edit: Precise image editing via recognition and generation tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Kenan Tang, Praveen Arunshankar, Andong Hua, Anthony Yang, and Yao Qin. Banana100: Breaking nr-iqa metrics by 100 iterative image replications with nano banana pro, 2026.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Shengming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report, 2025.

Hang Xu, Xiaoxiao Ma, Guohui Zhang, Yu Hu, Siming Fu, Jie Huang, Lin Song, Haoyang Huang, Nan Duan, and Feng Zhao. Anchoredit: Maintaining temporal consistency in multi-turn image editing via causal memory. *arXiv preprint arXiv:2606.11751*, 2026. URL https://arxiv.org/abs/2606.11751.

Mingde Yao, Zhiyuan You, Tam-King Man, Menglu Wang, and Tianfan Xue. Photoagent: Agentic photo editing with exploratory visual aesthetic planning, 2026. URL https://arxiv.org/abs/2602.22809.

Ruijie Ye, Jiayi Zhang, Zhuoxin Liu, Zihao Zhu, Siyuan Yang, Li Li, Tianfu Fu, Franck Dernoncourt, Yue Zhao, Jiacheng Zhu, Ryan Rossi, Wenhao Chai, and Zhengzhong Tu. Agent banana: High-fidelity image editing with agentic thinking and tooling, 2026a.

Yuxiao Ye, Haoran He, Fangyuan Kong, Xintao Wang, Pengfei Wan, Kun Gai, and Ling Pan. Edit-r2: Context-aware reinforcement learning for multi-turn image editing. arXiv:2606.05950, 2026b. URL https://arxiv.org/abs/2606.05950.

Dongxu Yue, Qin Guo, Munan Ning, Jiaxi Cui, Yuesheng Zhu, and Li Yuan. Chatface: Chat-guided real face editing via diffusion latent space manipulation, 2023. URL https://arxiv.org/abs/2305.14742.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *Advances in Neural Information Processing Systems*, 2023.

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*, pp. 586–595, 2018.

Shijie Zhao, Xuanyu Zhang, Weiqi Li, Junlin Li, Li Zhang, Tianfan Xue, and Jian Zhang. Reasoning as representation: Rethinking visual reinforcement learning in image quality assessment. *arXiv preprint arXiv:2510.11369*, 2026. URL https://arxiv.org/abs/2510.11369.

Zijun Zhou, Yingying Deng, Xiangyu He, Weiming Dong, and Fan Tang. Multi-turn consistent image editing, 2025.