CATD-MoE: Conflict-Aware Timestep-Dynamic K/V Mixture-of-Experts for Instruction-guided Image Editing

Anonymous authors
Paper under double-blind review

Abstract

Instruction-guided image editing models based on diffusion Transformers generally face a trade-off between editing flexibility and source image fidelity. Existing approaches typically rely on fixed K/V injection or manually defined task labels, making them less capable of adapting to dynamic editing requirements across different denoising stages and network depths. To address this issue, we propose **CATD-MoE**, a **Conflict-Aware Timestep-Dynamic K/V Mixture-of-Experts** framework that requires no retraining of the diffusion backbone. Specifically, we model the source image K/V and target denoising-state K/V as two memory experts, while their continuous interpolation forms a controlled bridging region. A lightweight Router dynamically determines their relative contributions based on denoising progress, mask properties, K/V conflict, and text representations. All diffusion backbone parameters remain frozen, with only the routing network being trained. During training, counterfactual candidates are constructed offline using target images, and Pareto-optimal supervision is derived according to editing quality, source image fidelity, and boundary quality, enabling the model to learn dynamic K/V mixing ratios. Experimental results on the independent PIE-Bench benchmark show that CATD-MoE effectively improves non-edited region fidelity and reduces boundary errors while maintaining high semantic consistency with the editing instructions. Further fine-grained evaluations and ablation studies demonstrate that joint spatiotemporal adaptation and K/V conflict awareness are key contributors to the observed performance gains.

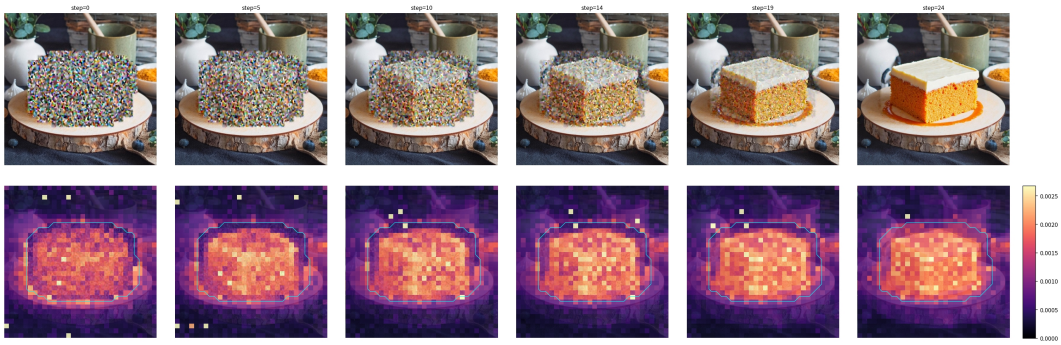

Figure 1: **Attention evolution across denoising timesteps.** Attention responses remain active around the edited region and in non-edited background areas at six representative denoising stages, revealing the concurrent demands of semantic modification and source content preservation. This observation motivates adaptive K/V memory allocation across the denoising process.

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.

1 INTRODUCTION

Natural image editing aims to modify specified content according to user instructions while preserving the structure, texture, and semantics of non-edited regions. Existing instruction-guided methods mainly include conditional diffusion-based editing, attention/inversion-based editing, and source-preserving approaches based on feature or K/V reuse (Meng et al., 2022; Rombach et al., 2022; Hertz et al., 2023; Tumanyan et al., 2023; Fu et al., 2024; Huang et al., 2024; Geng et al., 2024; Zhao et al., 2024; Zhu et al., 2025; Kim et al., 2025; Kulikov et al., 2025). Representative methods such as InstructPix2Pix (Brooks et al., 2023), Prompt-to-Prompt (Hertz et al., 2023), and Null-text Inversion (Mokady et al., 2023) improve instruction following, attention control, and inversion fidelity, while recent K/V-based methods further enhance background consistency by reusing source image representations (Zhu et al., 2025; Kim et al., 2025).

However, local editing still involves a fundamental trade-off between semantic editability and source image preservation. Strong source constraints may suppress the desired modification, whereas weak constraints can introduce unintended background drift (Cao et al., 2023; Wallace et al., 2023; Dong et al., 2023; Ju et al., 2024; Mou et al., 2024b; Huberman-Spiegelglas et al., 2024; Rout et al., 2025; Zhu et al., 2025; Kulikov et al., 2025). As shown in Figure 1, attention responses at six denoising timesteps remain active both around the edited region and in non-edited background areas, suggesting that semantic modification and source preservation coexist throughout denoising. Therefore, a globally fixed K/V injection strength is unlikely to be optimal across different timesteps and network depths.

Existing routing methods such as AnyEdit (Yu et al., 2025) and CARE-Edit (Wang et al., 2026) mainly select experts according to predefined tasks or multimodal conditions (Riquelme et al., 2021; Fedus et al., 2022; Zhou et al., 2022; Wu et al., 2024; Zhang et al., 2023b; Huang et al., 2023; Mou et al., 2024c). Such signals describe what edit is required, but do not directly reflect the evolving discrepancy between source memory and the current target representation. We therefore ask whether source image K/V can be adaptively controlled according to the real-time source–target state while keeping the diffusion backbone frozen.

Based on this motivation, we propose **CATD-MoE**, a **C**onflict-**A**ware **T**imestep-**D**ynamic K/V **M**ixture-**o**f-**E**xperts framework. CATD-MoE uses the discrepancy between source and target K/V as a routing signal and dynamically adjusts their contributions across denoising timesteps and Transformer block groups through a lightweight router. Our main contributions are:

- **Conflict-aware memory routing:** We use online source target K/V discrepancy to guide adaptive memory allocation without manually defined editing categories.
- **Fine-grained dynamic routing:** We extend fixed K/V mixing to timestep $\times$ block-group routing, enabling adaptive control across both denoising stages and network depth.
- **Frozen-backbone lightweight adaptation:** The diffusion backbone remains frozen, while only a lightweight router is trained using timestep, mask, text, conflict, and preservation information.
- **Counterfactual Pareto supervision:** Candidate routing strategies are evaluated during training according to editing, preservation, and boundary quality to derive supervision, while no target image is required at inference.

2 RELATED WORK

2.1 INSTRUCTION-GUIDED IMAGE EDITING

Instruction-guided image editing aims to modify specified regions according to natural language instructions while preserving the content and structure of non-edited areas. A representative approach, InstructPix2Pix (Brooks et al., 2023), trains a conditional diffusion model on large-scale synthetic data to directly learn the mapping from images and textual instructions to edited results. Subsequent work improves editing controllability, instruction following, and supervision quality through attention control, inversion, large-scale editing data, multimodal reasoning, and refined training objectives (Mokady et al., 2023; Kawar et al., 2023; Zhang et al., 2023c; Fu et al., 2024; Zhang et al.,

2023a; Huang et al., 2024; Sheynin et al., 2024; Geng et al., 2024; Zhang et al., 2024; Zhao et al., 2024; Xu et al., 2025; Zhou et al., 2025; Chen et al., 2025; Li et al., 2025; Simsar et al., 2025).

To mitigate background drift, feature and memory reuse methods reuse source representations during target denoising (Tumanyan et al., 2023; Cao et al., 2023; Wallace et al., 2023; Dong et al., 2023; Ju et al., 2024; Mou et al., 2024b; Xu et al., 2024; Brack et al., 2024; Mou et al., 2024a; Shi et al., 2024; Huberman-Spiegelglas et al., 2024; Rout et al., 2025; Kim et al., 2025; Kulikov et al., 2025; Gong et al., 2025). However, these approaches typically employ fixed or predefined reuse strategies, making them difficult to adapt to changing requirements across spatial regions and denoising stages. They therefore continue to face a trade-off between editing flexibility and source image fidelity. Recent DiT, rectified-flow, and autoregressive formulations further broaden the design space for image editing (Peebles & Xie, 2023; Liu et al., 2023; Lipman et al., 2023; Ma et al., 2024; Esser et al., 2024; Mao et al., 2026).

2.2 Mixture-of-Experts and Dynamic Routing

Mixture-of-Experts (MoE) dynamically routes inputs to specialized experts, improving conditional modeling capacity under limited computation (Riquelme et al., 2021; Fedus et al., 2022; Zhou et al., 2022; Wu et al., 2024). In image editing and controllable generation, task- or condition-aware routing and modular control have been explored through AnyEdit, CARE-Edit, and related conditional control frameworks (Yu et al., 2025; Wang et al., 2026; Zhang et al., 2023b; Huang et al., 2023; Mou et al., 2024c).

However, existing approaches primarily define experts as functionally distinct network modules, with routing objectives centered on task categories or conditional modalities. They do not explicitly address dynamic regulation of memory states throughout generation. In contrast, our approach defines different **K/V memory sources** as mixable experts and dynamically controls the balance between source image memory and the target generation state, directly modeling the editing and fidelity trade-off.

3 Method

Given a source image, an editing mask, and an editing instruction, the proposed method first extracts the source image K/V memory through a frozen diffusion inversion process. During the target denoising process, it continuously computes discrepancy measures between the source image K/V and the current target K/V states. A lightweight routing network (Router) then predicts the source memory coefficient α based on the denoising progress, mask geometry, K/V conflict, textual semantics, and preservation budget, thereby dynamically controlling the relative contributions of source memory and the target state, as shown in Figure 2.

3.1 Problem Formulation and Dynamic K/V Mixing

Given a source image X_s, a binary editing mask $M \in \{0,1\}^{H\times W}$, and an editing instruction P, the goal is to generate an edited image Y such that the masked region undergoes the semantic changes required by the instruction, while the non-masked region preserves the content, structure, and local texture of the source image as faithfully as possible. To this end, we first perform frozen diffusion inversion on the source image and extract the corresponding source image K/V memory (K_s^t, V_s^t) at each denoising timestep t. During the forward denoising process for target editing, the current state produces target K/V representations (K_t^t, V_t^t).

The central learning objective is to learn a routing function that predicts the optimal source memory coefficient α_t for each routing unit. For the basic timestep-level model, let g_θ denote the trainable routing network, and let z_t be the input feature vector at timestep t. The source memory coefficient is defined as:

$$\alpha_t = g_\theta(z_t), \qquad z_t = \left[f_{\text{time}}^t, f_{\text{mask}}, f_{\text{conflict}}^t, f_{\text{text}}, \rho\right]. \tag{1}$$

Here, f_{time}^t denotes the current denoising progress and timestep information, f_{mask} characterizes the geometric properties of the editing region, f_{conflict}^t represents the source–target K/V conflict statistics, f_{text} is a compressed semantic representation obtained from a frozen text encoder, and

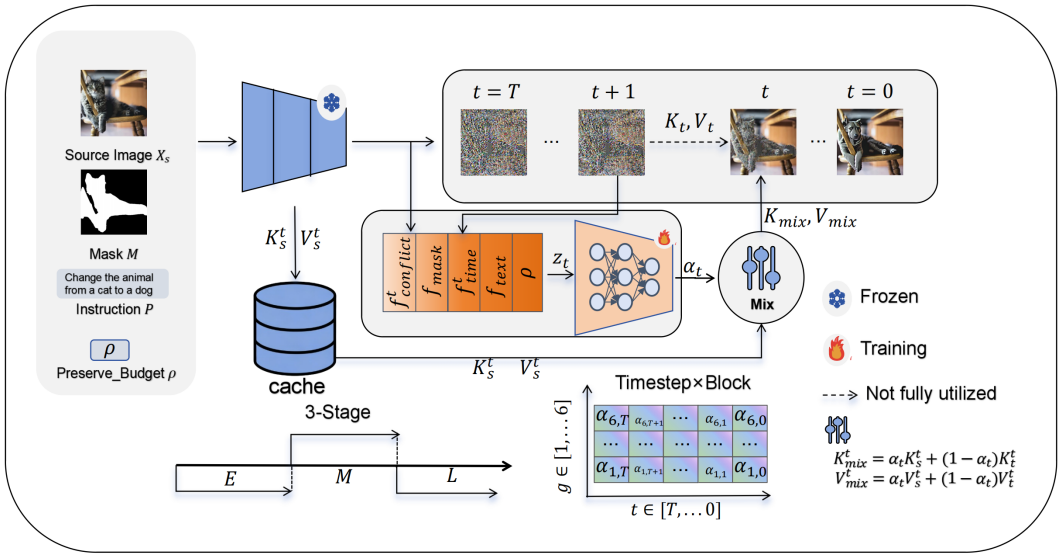

Figure 2: **The overall framework of the proposed dynamic K/V memory routing method.** The framework extracts source K/V memory (K_s^t, V_s^t) via frozen diffusion inversion. During the target denoising process, a lightweight Router network dynamically predicts a continuous source memory coefficient α_t. The prediction is based on a composite feature z_t, which encodes denoising progress (f_{time}^t), mask geometry (f_{mask}), real time source-target K/V conflict statistics (f_{conflict}^t), text semantics (f_{text}), and preservation budget (ρ). This coefficient directs the continuous mixing of source and target K/V representations for spatially and temporally adaptive editing.

$\rho \in \{0.25, 0.50, 0.75\}$ is the user-specified preservation budget. A larger ρ indicates a stronger preference for preserving the source image. The Router g_θ includes a final sigmoid activation, ensuring that its output coefficient lies in $[0, 1]$.

For an arbitrary routing granularity, the source K/V and target K/V are combined using the same continuous mixing rule:

$$K_{\text{mix}}^t = \alpha_t K_s^t + (1 - \alpha_t) K_t^t, V_{\text{mix}}^t = \alpha_t V_s^t + (1 - \alpha_t) V_t^t. \tag{2}$$

3.2 K/V Memory Provenance and the Continuous Expert Space

To characterize the roles of different K/V memory sources, we define source image K/V and target-state K/V as two endpoints of a continuous memory space: (K_s^t, V_s^t) represents the source endpoint, preserving structural, background, and local geometric information from the input image; (K_t^t, V_t^t) represents the target endpoint, providing the generation flexibility required for semantic editing.

These two endpoints are not discrete alternatives but form a continuous interpolation space:

$$\mathcal{E}(\alpha) = \alpha E_{\text{src}} + (1 - \alpha) E_{\text{tar}}, \qquad \alpha \in [0, 1]. \tag{3}$$

Here, E_{src} and E_{tar} denote the K/V provenance of the source image and target denoising state, respectively. Intermediate states with $0 < \alpha < 1$ correspond to the Bridge region rather than an additional neural expert.

This formulation decouples memory provenance from the routing coefficient: (1) provenance defines available memory sources; (2) α determines their contribution at the current denoising state.

Therefore, instead of predicting discrete expert weights with a three-way Softmax router, our method directly learns a continuous coefficient α. This design: (1) reduces the output space and training complexity; (2) maintains consistency between routing decisions and linear K/V mixing; (3) avoids additional expert parameters while preserving the lightweight nature of the framework.

3.3 Conflict Feature Modeling

The Router is guided by the source-target K/V conflict under the current denoising state rather than manually defined editing categories. The source K/V encodes information that should be preserved, while the target K/V represents the current editing direction. Large discrepancies indicate potential conflicts between preservation and editing, whereas consistent representations suggest that stronger source memory can reduce undesired drift.

Instead of retaining full K/V tensors, we extract low-dimensional conflict statistics from fixed Transformer block groups to reduce memory and computational overhead. Table B.2 provides a detailed list of the router's input features. For a region Ω, the Key cosine similarity is defined as:

$$\mathrm{Cos}_K^t(\Omega) = \frac{1}{|\Omega|} \sum_{i \in \Omega} \frac{K_{s,i}^t \cdot K_{t,i}^t}{\left\|K_{s,i}^t\right\|_2 \left\|K_{t,i}^t\right\|_2}. \tag{4}$$

The corresponding relative difference is:

$$\mathrm{RelDiff}_K^t(\Omega) = \frac{1}{|\Omega|} \sum_{i \in \Omega} \frac{\left\|K_{s,i}^t - K_{t,i}^t\right\|_1}{\left\|K_{s,i}^t\right\|_1 + \epsilon}. \tag{5}$$

The same statistics are computed for V. The resulting conflict representation is formulated as:

$$c_t = \Phi\left(K_s^t, V_s^t, K_t^t, V_t^t, M\right), \tag{6}$$

which is used as the Router input to predict the timestep-dependent fusion coefficient.

3.4 Counterfactual Candidate Generation and Pareto-Based Supervision

Assigning a fixed "optimal α" for each training sample is challenging because editing quality and source image preservation are inherently competing objectives. To obtain learnable routing supervision, we adopt a counterfactual candidate generation strategy, as shown in Figure 3. Given the same source image inversion cache, multiple candidate outputs are generated by varying the K/V mixing trajectories to explore different editing–preservation trade-offs. The candidate routing strategies are constructed as described in Algorithm 1.

For each candidate output Y_j, we evaluate four aspects: (1) editing quality S_{edit}, measuring target consistency within the masked region; (2) preservation quality S_{preserve}, measuring source consistency outside the masked region; (3) boundary quality S_{boundary}, evaluating transition naturalness around editing boundaries; (4) computational cost C, accounting for additional inference latency and memory consumption. Given a preservation budget ρ, we define the overall reward of candidate strategy R_j as:

$$R_j(\rho) = (1-\rho) S_{\mathrm{edit}}(Y_j) + \rho S_{\mathrm{preserve}}(Y_j) + \lambda_b S_{\mathrm{boundary}}(Y_j) - \lambda_c C(Y_j). \tag{7}$$

Here, S_{edit}, S_{preserve}, and S_{boundary} are normalized quality scores, $C(Y_j)$ denotes the additional computational cost, and λ_b and λ_c are the corresponding weighting coefficients.

3.5 Multi-Granularity Spatiotemporal Dynamic Routing

The key to dynamic routing lies not in increasing the dimensionality of the Router output, but in determining the spatiotemporal granularity at which routing decisions are made. We adopt a two-level routing design from coarse to fine, balancing representational capacity and training stability. Training details can be seen in Algorithm 2 and Appendix B.1.

3.5.1 Three-Stage Temporal Routing

We divide the denoising trajectory into three consecutive stages: Early, Middle, and Late. For each temporal stage $s \in \{E, M, L\}$, the Router predicts a scalar source memory coefficient:

$$\alpha_s = g_\theta(z_s), \tag{8}$$

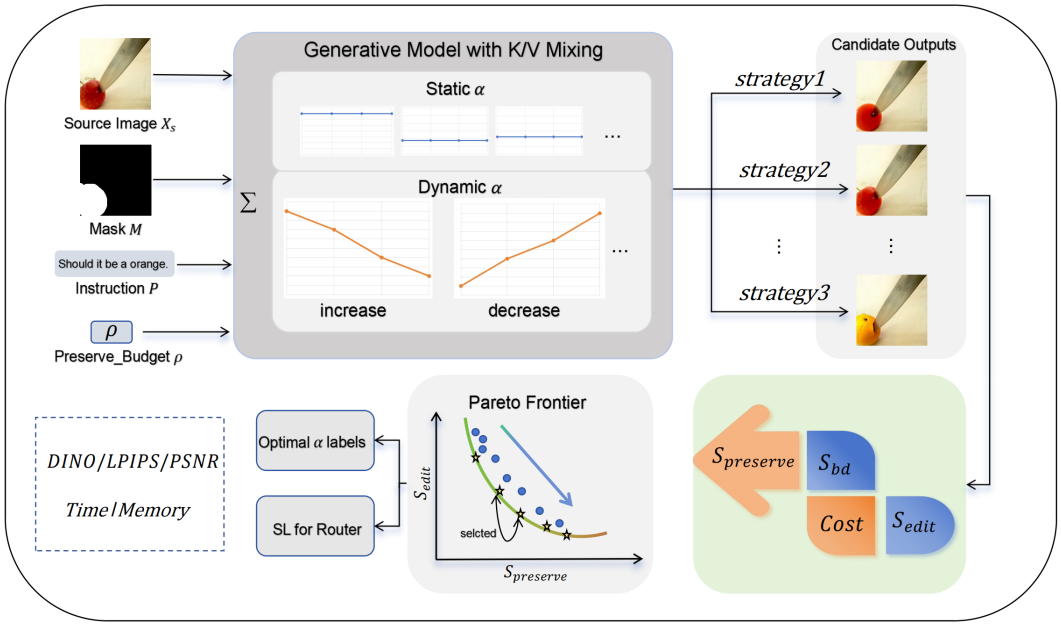

Figure 3: **Illustration of the counterfactual candidate generation and Pareto-based supervision.** Given a source image, editing mask, and instruction, multiple candidate outputs are generated by exploring various continuous K/V mixing trajectories (α). These candidates are comprehensively evaluated across editing quality (S_{edit}), preservation quality (S_{preserve}), boundary quality (S_{boundary}), and computational cost (C). Based on the user-specified preservation budget (ρ), a Pareto frontier is constructed to select the optimal routing labels (α) and ranking relationships for supervising the lightweight Router.

where z_s denotes the routing feature associated with stage s, and g_θ denotes the complete Router, including its final sigmoid activation. Therefore, g_θ directly outputs $\alpha_s \in [0, 1]$.

The three coefficients $\{\alpha_E, \alpha_M, \alpha_L\}$ serve as three temporal memory control points rather than expert weights. They are applied to the Early, Middle, and Late denoising stages, respectively. This stage-wise parameterization provides a balance between expressiveness and training stability. Compared with a single static α, it captures stage-dependent variations in the demand for source memory throughout the diffusion process. Compared with layer-wise or head-wise routing, it substantially reduces the label space, memory footprint, and inference overhead. Each stage still follows the K/V mixing rule defined in Eq. (2).

3.5.2 Fine-Grained Timestep × Block-Group Routing

Building upon the basic three-stage model, we further investigate joint spatiotemporal routing. For a T-step sampling process, each denoising timestep is treated as an independent temporal unit. The 19 double blocks of the Transformer are divided into 6 consecutive block groups, as detailed in Appendix A.1. The Router predicts an independent source memory coefficient for each timestep and block group:

$$\alpha_{t,g} = g_\theta(z_{t,g}), \qquad t \in \{1, \dots, T\}, \qquad g \in \{1, \dots, 6\}. \tag{9}$$

Therefore, the fine-grained model needs to predict at most $T \times 6$ routing coefficients. Each $\alpha_{t,g}$ controls the K/V mixing only within the corresponding timestep and block group, without modifying the parameters of the diffusion backbone.

To prevent fine-grained routing from exhibiting meaningless high frequency oscillations, the smoothness term in the objective is extended to impose joint constraints along both the temporal and block-group dimensions. Specifically, penalties are applied to changes in $\alpha_{t,g}$ between adja-

cent timesteps and adjacent block groups. Importantly, this fine-grained model still predicts a single continuous coefficient α.

3.5.3 UNIFIED TRAINING OBJECTIVE

To jointly constrain routing accuracy, ranking consistency, spatiotemporal smoothness, and preservation budget consistency, we optimize the following multi-component objective:

$$\mathcal{L} = \mathcal{L}_{\alpha} + 0.2\mathcal{L}_{\text{rank}} + 0.05\mathcal{L}_{\text{smooth}} + 0.05\mathcal{L}_{\text{budget}}. \tag{10}$$

Here, $\mathcal{L}_{\alpha}$ denotes the Huber regression loss between Router predictions and offline Pareto-optimal α labels. $\mathcal{L}_{\text{rank}}$ preserves the relative preference ordering among counterfactual candidates. $\mathcal{L}_{\text{smooth}}$ regularizes neighboring routing coefficients across temporal stages and block groups. $\mathcal{L}_{\text{budget}}$ enforces consistency between the preservation budget ρ and the source memory usage ratio. More details are in Appendix A.2.

For the timestep $\times$ block-group model, $\mathcal{L}_{\text{smooth}}$ is applied jointly along the temporal and block-group neighborhoods. In this way, the model retains fine-grained adaptive routing capability while avoiding excessively large and physically implausible differences between adjacent routing positions.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets: We train the lightweight Router on the official MagicBrush dataset (Zhang et al., 2023a), which contains approximately 10K instruction-guided image editing instances. The pretrained FLUX.1-dev backbone remains frozen throughout training, and only the Router parameters are optimized. The trained model is directly evaluated on PIE-Bench without additional fine-tuning.

Evaluation Benchmark: We adopt PIE-Bench (Ju et al., 2024) as the primary evaluation benchmark, containing 700 real-world images across 10 editing categories. Compared with MagicBrush, PIE-Bench provides more diverse image distributions and editing semantics, allowing us to evaluate generalization to unseen images and editing instructions.

Baselines: We compare CATD-MoE with representative attention-control methods (P2P (Hertz et al., 2023), PnP (Tumanyan et al., 2023), and MasaCtrl (Cao et al., 2023)), inversion-based editing pipelines (Null-text Inversion (Mokady et al., 2023) + P2P and Direct Inversion (Ju et al., 2024) + PnP), flow-based editing methods (RF-Inversion (Rout et al., 2025), FlowEdit (Kulikov et al., 2025), and ReFlex (Kim et al., 2025)), and K/V memory-based editing method KV-Edit (Zhu et al., 2025). All baseline results are reproduced using publicly available implementations under the PIE-Bench evaluation protocol. We follow official configurations whenever available and keep the input images, editing instructions, masks, and resolutions consistent whenever supported. Detailed implementation settings are provided in Appendix B.3.

Metrics: We evaluate structural consistency, source content preservation, and semantic editing quality. Structure Distance measures structural consistency (Tumanyan et al., 2022), while PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018) are computed exclusively over non-edited regions to evaluate source image preservation. Global CLIP measures overall semantic similarity, and Edit CLIP evaluates semantic alignment within the edited content (Radford et al., 2021).

4.2 QUANTITATIVE COMPARISON

Table 1 presents the quantitative comparison on the PIE-Bench benchmark. Existing instruction-guided image editing methods usually face a trade-off between editing effectiveness and source image preservation. Our method achieves a better balance between these two objectives by dynamically adjusting the information flow between source image memory and target denoising states.

CATD-MoE obtains the best numerical results on PSNR (34.71) and SSIM (96.13), together with the second best LPIPS (53.74), indicating improved preservation of non-edited regions under this evaluation protocol. Compared with KV-Edit, our method improves PSNR from 33.12 to 34.71 and

Table 1: **Quantitative comparison on the PIE-Bench benchmark.** Our method achieves a better balance between editing effectiveness and source image preservation. The best and second best results are highlighted in bold and underline, respectively. Structure Distance and LPIPS are scaled by 10^3, while SSIM is scaled by 10^2 following the PIE-Bench evaluation protocol. **All baseline results are obtained from our own reproduction using publicly available implementations.**

Method	Structure$_{10^3}$ ↓	PSNR↑	SSIM$_{10^2}$ ↑	LPIPS$_{10^3}$ ↓	CLIP↑	Edit CLIP↑
P2P (Hertz et al., 2023)	69.35	17.72	71.13	208.56	24.96	22.19
PnP (Tumanyan et al., 2023)	28.21	22.26	78.98	113.67	25.39	22.53
MasaCtrl (Cao et al., 2023)	28.29	22.74	79.64	106.82	23.94	21.09
NT + P2P (Mokady et al., 2023)	**13.56**	26.87	84.67	61.68	24.53	21.71
DI + PnP (Ju et al., 2024)	<u>24.27</u>	22.43	76.68	106.03	25.39	22.56
RF-Inversion (Rout et al., 2025)	43.51	20.19	75.47	183.64	24.93	22.58
FlowEdit (Kulikov et al., 2025)	27.53	22.01	84.25	107.36	26.05	22.69
ReFlex (Kim et al., 2025)	35.76	27.89	91.97	58.03	**26.29**	**23.04**
KV-Edit (Zhu et al., 2025)	37.62	33.12	96.01	**52.68**	25.67	22.53
Ours	36.52	**34.71**	**96.13**	<u>53.74</u>	<u>25.89</u>	<u>23.01</u>

achieves a comparable SSIM score (96.13 vs. 96.01) while maintaining competitive LPIPS. For editing effectiveness, CATD-MoE achieves an Edit CLIP score of 23.01, closely matching ReFlex (23.04), which obtains the highest score. Although individual baselines perform better on specific metrics, the proposed method combines strong non-edited region preservation with competitive semantic editing performance. This result is consistent with our objective of dynamically balancing source memory preservation and editing flexibility rather than maximizing either objective independently. The **user study** in Appendix C provides complementary perceptual evaluation, where CATD-MoE obtains the highest average scores on Source Fidelity and Editing Effectiveness while remaining competitive in Background Preservation.

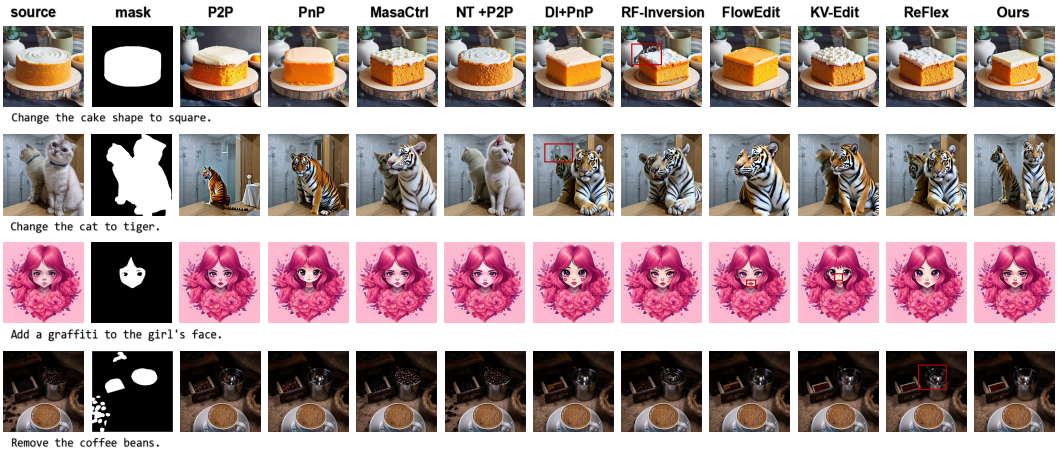

Figure 4: **Qualitative comparison on PIE-Bench.** We compare CATD-MoE with representative instruction-guided image editing methods under different editing scenarios, including object modification, object replacement, local attribute editing, and object removal. CATD-MoE achieves more accurate instruction following while better preserving non-edited regions and maintaining structural consistency with the source image.

4.3 Qualitative comparison

Figure 4 presents qualitative comparisons between our method and representative baselines across different editing tasks with **additional results provided in Appendix D**. For the cake shape mod-

ification task, P2P, PnP, and MasaCtrl introduce noticeable background changes, while Null-text Inversion (NT+P2P) and Direct Inversion(DI+PnP) provide limited editing flexibility. In contrast, our method achieves accurate shape transformation while preserving texture details and background structure. For object replacement and local attribute editing, existing methods may suffer from local artifacts, structural mismatches around the edited regions, insufficient semantic changes, or unintended modifications in non-edited regions. Our method better follows the editing mask and achieves natural editing while maintaining source image structures. Overall, the qualitative results demonstrate that our method effectively balances source image preservation and editing flexibility, enabling more controllable local image editing.

4.4 Ablation study

We conduct ablation studies on PIE-Bench to evaluate the key components of CATD-MoE. As shown in Table 2, removing either the conflict feature or temporal routing decreases both PSNR and Edit CLIP, demonstrating the importance of conflict information and temporal adaptivity for K/V allocation. As the routing granularity progresses from global routing and three-stage temporal routing to timestep $\times$ block-group routing, Structure Distance, PSNR, and Edit CLIP consistently improve, validating the effectiveness of fine-grained dynamic routing. Removing Pareto supervision further degrades the overall performance, supporting its role in learning effective routing decisions. Notably, the fixed K/V ratio and the variant without preservation budget conditioning achieve slightly lower LPIPS but substantially lower Edit CLIP, indicating that stronger static source constraints may improve non-edited region fidelity at the expense of semantic editability. Overall, the full model provides a better balance between source preservation and editing effectiveness.

Table 2: **Ablation study of the key components in CATD-MoE on PIE-Bench.** PSNR and LPIPS are evaluated over non-edited regions. 3-stage denotes coarse temporal routing over early, middle, and late denoising stages, while the full model performs timestep $\times$ block-group routing.

Variant	Structure$_{10^3}$ $\downarrow$	PSNR$\uparrow$	LPIPS$_{10^3}$ $\downarrow$	Edit CLIP$\uparrow$
Full Model	**36.52**	**34.71**	53.74	**23.01**
w/o conflict feature	38.06	32.97	54.13	22.58
w/o temporal routing	38.12	32.64	54.36	22.50
Fixed K/V ratio	38.05	33.09	52.84	22.16
Global routing	37.16	33.05	53.94	22.61
3-stage temporal routing	37.02	33.74	53.97	22.85
w/o Pareto supervision	37.86	32.84	54.79	21.05
w/o preservation budget	38.42	34.67	52.81	21.64

5 Conclusion

This paper presents CATD-MoE, a conflict-aware timestep-dynamic K/V routing method for addressing the trade-off between editing effectiveness and source image preservation in instruction-guided local image editing. With the diffusion backbone fully frozen, our method models the conflict between source image K/V memory and the current target denoising state and employs a lightweight Router to dynamically predict the K/V fusion coefficient. This enables adaptive control of source information and generation flexibility throughout the denoising process. Experimental results show that our method improves instruction-following capability while effectively preserving the structural and background information of the source image, achieving competitive quantitative and perceptual performance on PIE-Bench. Overall, CATD-MoE provides a lightweight and effective solution for mitigating the conflict between preservation and editing in local image editing.

AI USE STATEMENT

Following the ICLR 2027 guidelines on the use of generative AI tools, we disclose that Large Language Models (LLMs) were used as assistive tools during the preparation of this manuscript. Their use included refining wording, improving sentence-level readability, checking linguistic correctness, enhancing coherence across sections, and assisting with the retrieval and discovery of research relevant to this work. Any literature identified or suggested through LLM-assisted retrieval was independently verified by the authors against the original papers and official publication records to ensure its existence, relevance, and citation accuracy. LLMs were not used to generate the research ideas, design the proposed methodology, conduct experiments, or derive scientific conclusions. All technical contributions, experimental results, references, and final interpretations were reviewed and validated by the authors, who assume full responsibility for the manuscript.

ETHICS STATEMENT

This work investigates instruction-guided image editing techniques with the goal of improving the controllability, fidelity, and reliability of generative image models. In addition to experiments on publicly available benchmark datasets, we conducted a small-scale human evaluation study to assess the perceptual quality of edited results. The human evaluation only collected anonymous subjective ratings and does not involve the collection of personally identifiable information, private user data, or sensitive personal content. All benchmark datasets used in this work are publicly available and are utilized according to their corresponding licenses and usage policies.

The human evaluation was conducted following relevant ethical guidelines. Before participation, all participants were informed about the purpose of the study and voluntarily agreed to participate. No personally identifiable information was collected, and all evaluation responses were stored anonymously and used solely for research purposes.

We acknowledge that advanced image editing and generation technologies may introduce potential ethical risks, including the creation of misleading visual content, unauthorized manipulation of personal images, or misuse in sensitive scenarios. Although our work focuses on improving source image preservation, instruction following, and controllability of image editing systems, these capabilities should be accompanied by responsible deployment practices. We encourage future applications of such technologies to follow appropriate ethical standards, legal requirements, and content governance policies.

Our method is designed to improve the reliability and controllability of image editing systems by explicitly balancing source information preservation and target content generation. We do not intend to develop systems for deceptive purposes or applications that may harm individuals or communities. We have carefully considered the potential societal impacts of this research and believe that improving the transparency and predictability of generative models can contribute to the development of more trustworthy AI systems.

REPRODUCIBILITY STATEMENT

We have taken several steps to ensure the reproducibility of our work. First, Appendix A provides detailed descriptions of the six-group routing mechanism and the exact Router training objective in Appendix A.1 and A.2, respectively. The complete procedures for counterfactual label construction, continuous Router training, and dynamic inference are further specified in Algorithms 1–3. Second, Appendix B.1 reports the computing infrastructure, frozen and trainable components, Router architecture and input features, inference hyperparameters, numerical precision, image resolution, and the three-seed evaluation protocol; the principal implementation details are summarized in Tables B.1 and B.2. We also document the datasets, evaluation metrics, baseline configurations, and public implementation sources, with baseline reproduction details provided in Appendix B.3 and Table B.3. Finally, anonymized source code and configuration files are included in the **supplementary material**, and the complete implementation will be released publicly upon publication.

REFERENCES

Manuel Brack, Felix Friedrich, Katharia Kornmeier, Linoy Tsaban, Patrick Schramowski, Kristian Kersting, and Apolinário Passos. LEDITS++: Limitless image editing using text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.00846.

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. URL https://arxiv.org/abs/2211.09800.

Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. MasaCtrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. URL https://arxiv.org/abs/2304.08465.

Sherry X. Chen, Misha Sra, and Pradeep Sen. Instruct-CLIP: Improving instruction-guided image editing with automated data refinement using contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://ieeexplore.ieee.org/document/11093248.

Wenkai Dong, Song Xue, Xiaoyue Duan, and Shumin Han. Prompt tuning inversion for text-driven image editing using diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. URL https://doi.ieeecomputersociety.org/10.1109/ICCV51070.2023.00683.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. URL https://dl.acm.org/doi/10.5555/3692070.3692573.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 2022. URL https://www.jmlr.org/papers/volume23/21-0998/21-0998.pdf.

Tsu-Jui Fu, Wenze Hu, Xianzhi Du, William Yang Wang, Yinfei Yang, and Zhe Gan. Guiding instruction-based image editing via multimodal large language models. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=S1RKWSyZ2Y.

Zigang Geng, Binxin Yang, Tiankai Hang, Chen Li, Shuyang Gu, Ting Zhang, Jianmin Bao, Zheng Zhang, Houqiang Li, Han Hu, Dong Chen, and Baining Guo. InstructDiffusion: A generalist modeling interface for vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.01208.

Yiming Gong, Zhen Zhu, and Minjia Zhang. InstantEdit: Text-guided few-step image editing with piecewise rectified flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/papers/Gong_InstantEdit_Text-Guided_Few-Step_Image_Editing_with_Piecewise_Rectified_Flow_ICCV_2025_paper.pdf.

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. URL https://arxiv.org/abs/2208.01626.

Lianghua Huang, Di Chen, Yu Liu, Yujun Shen, Deli Zhao, and Jingren Zhou. Composer: Creative and controllable image synthesis with composable conditions. In *Proceedings of the 40th International Conference on Machine Learning*, 2023. URL https://proceedings.mlr.press/v202/huang23b.html.

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. URL https://arxiv.org/abs/2312.06739.

Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly DDPM noise space: Inversion and manipulations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://ieeexplore.ieee.org/document/10657449.

Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. PnP Inversion: Boosting diffusion-based editing with 3 lines of code. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/661caac7729aa7d8c6b8ac0d39ccbc6a-Paper-Conference.pdf.

Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2210.09276.

Jimyeong Kim, Jungwon Park, Yeji Song, Nojun Kwak, and Wonjong Rhee. ReFlex: Text-guided editing of real images in rectified flow via mid-step feature extraction and attention adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/papers/Kim_ReFlex_Text-Guided_Editing_of_Real_Images_in_Rectified_Flow_via_ICCV_2025_paper.pdf.

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. URL https://ieeexplore.ieee.org/document/11445781.

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*, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/papers/Li_SuperEdit_Rectifying_and_Facilitating_Supervision_for_Instruction-Based_Image_Editing_ICCV_2025_paper.pdf.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, 2024. URL https://doi.org/10.1007/978-3-031-72980-5_2.

Qingyang Mao, Qi Cai, Yehao Li, Yingwei Pan, Mingyue Cheng, Ting Yao, Qi Liu, and Tao Mei. Visual autoregressive modeling for instruction-guided image editing. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/8a0501a7e57e3669849e0ecf6f1a6295-Paper-Conference.pdf.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=aBsCjcPu_tE.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. NULL-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2211.09794.

Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. DiffEditor: Boosting accuracy and flexibility on diffusion-based image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024a. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.00811.

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*, 2024b. URL https://openreview.net/forum?id=OEL4FJMg1b.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2I-Adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024c. URL https://ojs.aaai.org/index.php/AAAI/article/view/28226.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. URL https://arxiv.org/abs/2212.09748.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning*, 2021. URL https://arxiv.org/abs/2103.00020.

Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. In *Advances in Neural Information Processing Systems*, 2021. URL https://arxiv.org/abs/2106.05974.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://arxiv.org/abs/2112.10752.

Litu Rout, Yujia Chen, Nataniel Ruiz, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Semantic image inversion and editing using rectified stochastic differential equations. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/c07bc8b520aa87326b015b0da6a70b48-Abstract-Conference.html.

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. URL https://openaccess.thecvf.com/content/CVPR2024//papers/Sheynin_Emu_Edit_Precise_Image_Editing_via_Recognition_and_Generation_Tasks_CVPR_2024_paper.pdf.

Yujun Shi, Chuhui Xue, Jun Hao Liew, Jiachun Pan, Hanshu Yan, Wenqing Zhang, Vincent Y. F. Tan, and Song Bai. DragDiffusion: Harnessing diffusion models for interactive point-based image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.00844.

Enis Simsar, Alessio Tonioni, Yongqin Xian, Thomas Hofmann, and Federico Tombari. UIP2P: Unsupervised instruction-based image editing via edit reversibility constraint. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://arxiv.org/abs/2412.15216.

Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing ViT features for semantic appearance transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52688.2022.01048.

Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2211.12572.

Bram Wallace, Akash Gokul, and Nikhil Naik. EDICT: Exact diffusion inversion via coupled transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.02158.

Yucheng Wang, Zedong Wang, Yuetong Wu, Yue Ma, and Dan Xu. CARE-Edit: Condition-aware routing of experts for contextual image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2603.08589.

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*, 2004. URL https://doi.org/10.1109/TIP.2003.819861.

Xun Wu, Shaohan Huang, and Furu Wei. Mixture of LoRA experts. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2404.13628.

Sihan Xu, Yidong Huang, Jiayi Pan, Ziqiao Ma, and Joyce Chai. Inversion-free image editing with language-guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.00903.

Yingjing Xu, Jie Kong, Jiazhi Wang, Xiao Pan, Bo Lin, and Qiang Liu. InsightEdit: Towards better instruction following for image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/papers/Xu_InsightEdit_Towards_Better_Instruction_Following_for_Image_Editing_CVPR_2025_paper.pdf.

Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. AnyEdit: Mastering unified high-quality image editing for any idea. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://cvpr.thecvf.com/virtual/2025/poster/34767.

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*, 2023a. URL https://arxiv.org/abs/2306.10012.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023b. URL https://arxiv.org/abs/2302.05543.

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/CVF Conference on Computer Vision and Pattern Recognition*, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/papers/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.pdf.

Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Huan Wang, Silvio Savarese, Stefano Ermon, Caiming Xiong, and Ran Xu. HIVE: Harnessing human feedback for instructional visual editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.00862.

Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris N. Metaxas, and Jian Ren. SINE: SINgle image editing with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023c. URL https://arxiv.org/abs/2212.04489.

Haozhe Zhao, Xiaojian Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. UltraEdit: Instruction-based fine-grained image editing at scale. In *Advances in Neural Information Processing Systems*, 2024. URL https://dl.acm.org/doi/10.5555/3737916.3738016.

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*, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/papers/Zhou_FireEdit_Fine-grained_Instruction-based_Image_Editing_via_Region-aware_Vision_Language_Model_CVPR_2025_paper.pdf.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y. Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V. Le, and James Laudon. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2202.09368.

Tianrui Zhu, Shiyi Zhang, Jiawei Shao, and Yansong Tang. KV-Edit: Training-free image editing for precise background preservation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://ieeexplore.ieee.org/document/11445132.

APPENDIX

A METHOD DETAILS

A.1 ROUTING DETAILS

The Double Stream Transformer in FLUX.1-dev contains 19 Double Blocks, indexed from 0 to 18 in the implementation. Instead of predicting an independent routing coefficient for every individual block, the current implementation partitions the 19 Double Blocks into six contiguous routing groups according to their normalized depth positions. Specifically, Group 0 contains blocks 0, 1, and 2, corresponding to the earliest depth range; Group 1 contains blocks 3, 4, and 5; Group 2 contains blocks 6, 7, and 8; Group 3 contains blocks 9, 10, and 11; Group 4 contains blocks 12, 13, and 14; and Group 5 contains blocks 15, 16, 17, and 18, including the final endpoint block. Therefore, the actual implementation uses a contiguous 3+3+3+3+3+4 partition. The block index is mapped into routing groups according to:

$$p(l) = \frac{l}{L-1}, \qquad g(l) = \min\left(G-1, \lfloor G p(l) \rfloor\right), \qquad l \in \{0, \ldots, 18\}. \tag{A.1}$$

Here l denotes the Double Block index, L=19, G=6, $p(l)$ represents the normalized block depth position, and $g(l)$ denotes the corresponding routing group index.

During target denoising, the Router is applied only to the 19 Double Blocks. For each denoising timestep and routing group, the model computes $\alpha_{t,g}$ once and caches it for all Double Blocks in the same group. Compared with per-layer, per-head, or per-token routing, this timestep-group routing strategy substantially reduces the label space, cache state, and inference overhead, while still allowing different depth ranges to express different source memory requirements across denoising stages.

The 38 Single Blocks are not dynamically routed, but this does not mean that they discard K/V memory. We keep the original fixed rule: during source inversion, the source K/V of Single Blocks is cached; during target denoising, the masked region uses the current target image K/V, while the non-masked region continues to use the cached source K/V. In other words, Single Blocks do not call the Router and do not receive additional dynamic α values. This design concentrates dynamic conflict adjustment in the Double Blocks, where image and text streams coexist and where conflicts between source and target K/V are most directly expressed. At the same time, keeping Single Blocks fixed avoids introducing per-step and per-layer routing states for 38 additional modules, and reduces memory consumption and trace-generation overhead.

A.2 TRAINING LOSS

Given a mini-batch of training states $\mathcal{B}$, each state $i \in \mathcal{B}$ is associated with an image sample, a source preservation budget ρ_i, a denoising timestep t_i, and a routing group g_i.

Let y_i denote the piecewise-linearly interpolated value of the highest-reward candidate curve at the current denoising progress, and let y_i^{alt} denote the corresponding value of the second-ranked candidate curve. We use m_i to denote the reward gap between the two candidates and w_i to denote the confidence weight of the label. The current implementation assigns $w_i = 1.0$ to confident labels and $w_i = 0.25$ to low-confidence labels, with a minimum weight of 0.05.

Confidence-Weighted Regression Loss L_α : The basic regression term is a confidence-weighted Smooth L1 (Huber) loss:

$$L_\alpha = \frac{\sum_{i \in \mathcal{B}} w_i \, \text{Huber}(a_i - y_i)}{\sum_{i \in \mathcal{B}} w_i}. \tag{A.2}$$

This term directly encourages the continuous Router prediction to match the pseudo-label supplied by the winning candidate curve. The Huber penalty is quadratic for small residuals while reducing the influence of noisy or outlying pseudo-labels for larger residuals.

Ranking Regularization L_{rank} : To exploit the relative preference between the two highest-ranked candidates, we compute the absolute distances from the prediction to the winning and second-ranked candidate curves:

$$d_i^+ = |a_i - y_i|, \qquad d_i^{\text{alt}} = |a_i - y_i^{\text{alt}}|. \tag{A.3}$$

The reward gap is clipped to a bounded interval:

$$\widetilde{m}_i = \mathrm{clip}(m_i, 0.01, 0.50). \tag{A.4}$$

The per-state ranking loss is defined as

$$\ell_{\mathrm{rank},i} = \mathrm{softplus}\left(d_i^{+} - d_i^{\mathrm{alt}} + \widetilde{m}_i\right), \tag{A.5}$$

where $\mathrm{softplus}(x) = \log(1 + \exp(x))$. The confidence-weighted ranking regularizer is

$$L_{\mathrm{rank}} = \frac{\sum_{i \in \mathcal{B}} w_i \ell_{\mathrm{rank},i}}{\sum_{i \in \mathcal{B}} w_i}. \tag{A.6}$$

This loss does not directly recompute or compare the image rewards. Instead, it encourages a_i to be closer to the winning curve than to the second-ranked curve in the routing-coefficient space. The clipped reward gap $\widetilde{m}_i$ controls the required relative separation. A more decisive reward advantage requires the Router to distinguish the two candidate curves more clearly.

Temporal and Group-Wise Smoothness L_{smooth} : We introduce a spatiotemporal smoothness regularizer to suppress unnecessary variations across adjacent timesteps and routing groups. Let $\mathcal{E}_t$ contain pairs of adjacent states after sorting by timestep for the same sample, preservation budget, and routing group. Similarly, let $\mathcal{E}_g$ contain pairs of adjacent states after sorting by routing-group index for the same sample, preservation budget, and timestep.

The temporal smoothness term is

$$L_t = \begin{cases} \dfrac{1}{|\mathcal{E}_t|} \displaystyle\sum_{(i,j) \in \mathcal{E}_t} |a_i - a_j|, & |\mathcal{E}_t| > 0, \\ 0, & |\mathcal{E}_t| = 0, \end{cases} \tag{A.7}$$

and the group-wise smoothness term is

$$L_g = \begin{cases} \dfrac{1}{|\mathcal{E}_g|} \displaystyle\sum_{(i,j) \in \mathcal{E}_g} |a_i - a_j|, & |\mathcal{E}_g| > 0, \\ 0, & |\mathcal{E}_g| = 0. \end{cases} \tag{A.8}$$

The combined smoothness regularizer is

$$L_{\mathrm{smooth}} = \frac{1}{2}\left(L_t + L_g\right). \tag{A.9}$$

The implementation penalizes absolute L1 variations rather than squared differences. Because neighboring pairs are constructed only from states available in the current mini-batch, this term is a batch-wise approximation of the complete temporal-and-depth adjacency regularizer. If either edge set is empty, its corresponding contribution is defined as zero.

Budget-Monotonicity Regularization L_{budget} : The preservation budget ρ represents the requested degree of source content preservation. To ensure a consistent and interpretable response to this control variable, we introduce a budget-monotonicity regularizer.

Let $\mathcal{E}_\rho$ contain adjacent states after sorting by ρ for the same sample, timestep, and routing group. Each pair $(i, j) \in \mathcal{E}_\rho$ satisfies $\rho_i < \rho_j$. The corresponding regularizer is:

$$L_{\mathrm{budget}} = \begin{cases} \dfrac{1}{|\mathcal{E}_\rho|} \displaystyle\sum_{(i,j) \in \mathcal{E}_\rho} \max(0, a_i - a_j), & |\mathcal{E}_\rho| > 0, \\ 0, & |\mathcal{E}_\rho| = 0. \end{cases} \tag{A.10}$$

No penalty is introduced when the prediction associated with the larger budget is no smaller than that associated with the lower budget, i.e., $a_j \geq a_i$. A linear penalty is applied whenever this monotonic relationship is violated. This term is not confidence-weighted and is set to zero when the mini-batch contains no multiple-budget states sharing the same sample, timestep, and routing group.

A.3 ALGORITHM

This section presents three algorithms that summarize the complete implementation pipeline of CATD-MoE. Algorithm 1 describes offline counterfactual candidate generation, evaluation, and supervision-label construction. Algorithm 2 specifies the training procedure for the lightweight Router using the offline labels and the joint multi-term objective. Algorithm 3 details the dynamic routing process at inference time, where a continuous mixing coefficient α is predicted and cached for each denoising-timestep and Double-Block routing group to control the relative contributions of source image memory and the current target state. Together, the three algorithms cover supervision construction, Router optimization, and closed-loop inference, thereby providing an end-to-end description of the proposed method.

Algorithm 1 Counterfactual Label Construction

Require: Frozen editing model $\mathcal{M}$, training records $\mathcal{D} = \{(X_s, M, I, X_t)\}$, candidate routing trajectories $\mathcal{A} = \{\alpha_j(\tau)\}_{j=1}^{N}$, preservation budgets $\mathcal{R} = \{0.25, 0.50, 0.75\}$

for each training sample $(X_s, M, I, X_t) \in \mathcal{D}$ **do**
  for each candidate trajectory $\alpha_j(\tau) \in \mathcal{A}$ **do**
    Generate candidate output:

$$Y_j \leftarrow \mathcal{M}(X_s, M, I; \alpha_j(\tau))$$

    Compute evaluation metrics:

$$\{S_{\text{edit}}(Y_j), S_{\text{preserve}}(Y_j), S_{\text{boundary}}(Y_j), C(Y_j)\}$$

  end for
  Normalize candidate metrics within current sample
  Construct Pareto frontier:

$$\mathcal{P} \leftarrow \text{ParetoFrontier}(S_{\text{edit}}, S_{\text{preserve}}, S_{\text{boundary}}, -C)$$

  for each preservation budget $\rho \in \mathcal{R}$ **do**
    Compute candidate reward:

$$R_j(\rho) = (1-\rho)S_{\text{edit}}(Y_j) + \rho S_{\text{preserve}}(Y_j) + \lambda_b S_{\text{boundary}}(Y_j) - \lambda_c C(Y_j)$$

    Select optimal routing trajectory:

$$j^\star \leftarrow \arg\max_j R_j(\rho)$$

    Compute reward margin:

$$\Delta R = R_{j^\star}(\rho) - R_{j(2)}(\rho)$$

    Store supervision label:

$$(\rho, \alpha_{j^\star}(\tau), \Delta R, \mathcal{P}, \text{Rank})$$

  end for
end for
return Counterfactual routing supervision labels

B ADDITIONAL EXPERIMENTAL DETAILS

B.1 EXPERIMENTAL CONFIGURATION

Compute setup: Table B.1 summarizes the compute setup, frozen modules, and base inference environment used in our experiments. In general, all experiments are built on a frozen FLUX.1-dev editing pipeline, with only the lightweight CATD Router being trained. Candidate generation,

Algorithm 2 Continuous α Router Training

Require: Labels generated by Algorithm 1 , winning routing trajectories $\alpha_{j^\star}(\tau)$, candidate rankings, preservation budgets ρ

for each sample-budget label $(\rho, \alpha_{j^\star}(\tau), \Delta R, \mathcal{P}, \mathrm{Rank})$ **do**
  Select conflict traces of the corresponding winning candidate
  for each routing state (t, g) **do**
    Construct Router input:

$$z_{t,g} = [c_t, t, g, \rho]$$

    Obtain routing target:

$$y_{t,g} = \mathrm{PiecewiseAlpha}(\alpha_{j^\star}(\tau), t)$$

    Obtain ranking reference:

$$y^{\mathrm{alt}}_{t,g} = \mathrm{PiecewiseAlpha}(\alpha_{j(2)}(\tau), t)$$

    Update Router parameters θ by minimizing:

$$\mathcal{L} = \mathcal{L}_\alpha + 0.2\mathcal{L}_{\mathrm{rank}} + 0.05\mathcal{L}_{\mathrm{smooth}} + 0.05\mathcal{L}_{\mathrm{budget}}$$

  end for
end for
Save Router parameters θ and feature normalization statistics (mean/std)
return Trained Router g_θ

Router training, and closed-loop evaluation are conducted under the same local-checkpoint, text-encoding, image-codec, and numerical-precision settings.

The Router architecture: The current implementation uses a lightweight two-layer MLP that predicts one continuous source memory coefficient α for each timestep-group. The input is a 21-dimensional scalar feature vector normalized by the training-set mean and standard deviation, including the source image, mask, instruction, current K/V statistics, and the preservation budget ρ, as shown in Table B.2. The first hidden layer maps the 21-dimensional input to 128 dimensions and applies LayerNorm, GELU, and 0.1 dropout, projecting heterogeneous temporal, spatial, semantic, and conflict-related features into a shared hidden space. The second hidden layer keeps the 128-dimensional representation and applies GELU with 0.1 dropout, further modeling interactions among timestep progress, routing-group position, mask geometry, text semantics, and source-target K/V differences. The Router ends with a $128 \rightarrow 1$ linear projection followed by a sigmoid activation and directly outputs the continuous coefficient $\alpha \in [0, 1]$. With affine LayerNorm, the Router contains 19,713 trainable parameters, excluding all frozen FLUX backbone and encoding/decoding modules.

Inference and statistical protocol: All inference experiments use a unified text-encoding setup, scheduler configuration, random-seed protocol, and numerical precision to ensure comparability across the main experiments, ablation studies, and closed-loop evaluation. Text encoding is performed by frozen T5 and CLIP encoders. The maximum T5 length is set to 512 to produce the text sequence txt used by both Double Blocks and Single Blocks, with a context dimension of 4096. The maximum CLIP length is set to 77 to produce the 768-dimensional pooled embedding vector.

For sampling, the guidance scale is set to 1.5 during source image inversion and 5.5 during target denoising. The total number of scheduler steps is set to 28, and the first four schedule nodes are removed before denoising, corresponding to skip step = 4. This setting keeps the inversion and target-editing stages consistent while reducing the influence of the earliest high-noise schedule nodes. The default seed for a single job is 42, and both PyTorch CPU and CUDA seeds are set at the beginning of each job to reduce unintended randomness.

The quantitative results in the main tables and ablation tables are reported as the arithmetic mean over three random seeds: 42, 43, and 45. The FLUX text-encoding path and VAE inference path use bfloat16 precision by default. The maximum input resolution is limited to 1366×768, and spatial dimensions are cropped to multiples of 16 to match the latent downsampling and Transformer tokenization requirements of FLUX.

Algorithm 3 CATD-MoE Inference with Dynamic Preservation Budget

Require: Source image X_s, editing mask M, instruction I, trained Router g_θ, optional preservation budget ρ

Obtain source K/V memory cache:

$$(K_s^t, V_s^t) \leftarrow \text{Inversion}(X_s)$$

if ρ is not specified **then**
  Set ρ to default preservation budget
end if
for each target denoising timestep $t = 1, \ldots, T$ **do**
  Clear routing cache
  for each Double Block l in depth order **do**
    Determine block group:

$$g \leftarrow \text{Group}(l)$$

    if $\alpha_{t,g}$ is not cached **then**
      Extract source-target conflict:

$$c_t = \Phi(K_s^t, V_s^t, K_t^t, V_t^t, M)$$

      Construct Router input:

$$z_{t,g} = [c_t, t, g, \rho]$$

      Compute dynamic routing coefficient:

$$\alpha_{t,g} = g_\theta(z_{t,g})$$

      Cache $\alpha_{t,g}$
    end if
    Mix masked K/V representations:

$$(K, V) = \alpha_{t,g}(K_s^t, V_s^t) + (1 - \alpha_{t,g})(K_t^t, V_t^t)$$

    Execute attention and frozen Double Block l
  end for
  Execute Single Blocks with original masked target K/V replacement
end for
Decode final latent using frozen autoencoder
return Edited image Y

Table B.1: details

Item	Setting	Role
Compute	8 × NVIDIA RTX A6000	/
Backbone	FLUX.1-dev	Contains 19 Double Blocks and 38 Single Blocks.
Text encoding	T5 and CLIP	T5 provides token sequences and CLIP provides the pooled embedding.
Image En/decoder	FLUX autoencoder VAE	Maps between RGB and latent spaces and remains frozen.
Trainable parameters	CATD MLP Router only	FLUX, T5, CLIP, VAE, and the base editing path remain frozen.
Precision	bfloat16	Used by the FLUX text-encoding and VAE inference paths.

B.2 Dataset Details

MagicBrush: MagicBrush is a large-scale human-annotated dataset for instruction-guided image editing, containing approximately 10K editing instances. Each sample consists of a source image, an editing instruction, and the corresponding edited result.

Table B.2: Input features of the CATD-MoE Router.

No.	Feature	Role
1	progress	Normalized denoising progress u_t
2	timestep	Current scheduler time coordinate
3	global_block_position	Normalized Double-Block depth position
4	mask_ratio	Fraction of image tokens inside the edit mask
5	mask_bbox_ratio	Relative area of the mask bounding box
6	mask_boundary_ratio	Mask-boundary complexity or boundary-ratio statistic
7	mask_center_x	Normalized horizontal mask center
8	mask_center_y	Normalized vertical mask center
9	cosine_key	Source-target Key cosine similarity over masked tokens
10	relative_difference_key	Relative L1 Key difference over masked tokens
11	cosine_value	Source-target Value cosine similarity over masked tokens
12	relative_difference_value	Relative L1 Value difference over masked tokens
13	source_background_gap	Gap between source-region and source-background K/V statistics
14	source_key_norm	Source Key norm statistic
15	target_key_norm	Target Key norm statistic
16	source_value_norm	Source Value norm statistic
17	target_value_norm	Target Value norm statistic
18	text_mean	Mean of frozen text features
19	text_std	Standard deviation of frozen text features
20	text_norm	Norm of frozen text features
21	preservation_budget	User-specified source preservation budget ρ

PIE-Bench: PIE-Bench is specifically designed to evaluate text-driven image editing methods under realistic scenarios. It contains 700 real-world images covering 10 editing categories, including object replacement, attribute modification, object removal, background manipulation, and global semantic editing. Each evaluation sample provides a source image, an editing instruction, a target prompt, and an editing-region mask.

B.3 Baseline Details

The selected baselines cover representative paradigms in instruction-guided image editing, including attention control, inversion-based editing pipelines, flow-based editing, and memory-augmented editing. All baseline results are reproduced using publicly available implementations under the PIE-Bench evaluation protocol. The detailed implementation sources and reproduction settings are summarized in Table B.3.

Attention-control Methods: Prompt-to-Prompt (P2P) performs instruction-guided editing by manipulating cross-attention maps to control the correspondence between textual concepts and image regions, enabling localized semantic modifications while preserving the original image structure. Plug-and-Play (PnP) improves structural consistency by injecting intermediate diffusion features, including spatial features and self-attention information, from the source image during the denoising process. MasaCtrl extends attention-based editing through mutual self-attention control, allowing better utilization of source image information to preserve object layouts and local details.

Inversion-based Editing Pipelines: Inversion-based approaches aim to recover source image representations in the latent space before applying editing operations. Since inversion techniques mainly provide source reconstruction or initialization strategies rather than complete editing frameworks, we evaluate them together with corresponding editing controllers. Specifically, Null-text Inversion is combined with P2P (NT+P2P), where optimized unconditional text embeddings improve inversion fidelity while maintaining prompt-based editing capability. Direct Inversion is combined with PnP (DI+PnP), where an improved inversion process enhances source reconstruction quality and editing consistency.

Flow-based Editing Methods: RF-Inversion, FlowEdit, and ReFlex represent recent editing approaches developed for rectified-flow-based generative models. RF-Inversion constructs a controlled trajectory between source images and noise representations to enable semantic editing. FlowEdit avoids explicit inversion by constructing a direct transport path between source and target distribu-

Table B.3: **Implementation sources and reproduction settings of baseline methods.** All baseline results are obtained from our own reproduction using publicly available implementations.

Method	Source	Repository	Result Source
P2P	Official	GitHub	Our reproduction
PnP	Official	GitHub	Our reproduction
MasaCtrl	Official	GitHub	Our reproduction
NT + P2P	Official + Official	GitHub	Our reproduction
DI + PnP	Official + Official	GitHub	Our reproduction
RF-Inversion	Public	GitHub	Our reproduction
FlowEdit	Official	GitHub	Our reproduction
KV-Edit	Official	GitHub	Our reproduction
ReFlex	Public	GitHub	Our reproduction

tions in pretrained flow models. ReFlex extends feature-based editing to Rectified Flow models by exploiting FLUX MM-DiT representations, extracting source features from mid-step latents, and introducing attention adaptation strategies to balance source structure preservation and target prompt alignment.

Memory-augmented Editing Methods: KV-Edit improves source content preservation by reusing source image K/V memory during the generation process. Specifically, it stores background token key-value information during inversion and injects these representations into target generation to maintain consistency between edited and non-edited regions. This approach is closely related to our motivation of dynamically balancing source information preservation and editing flexibility.

B.4 Metric Details

Structural Preservation: Structure Distance measures the structural discrepancy between the edited image and the source image, with lower values indicating better preservation of the original spatial structure. This metric evaluates whether an editing method can maintain the geometric arrangement, object contours, and global structural information of the source image after modification. SSIM evaluates structural similarity by jointly considering luminance, contrast, and local structural information between the generated image and the source image. In our evaluation, Structure Distance and SSIM are mainly used to measure the ability of a method to preserve non-edited regions and maintain visual consistency with the original image.

Visual Fidelity: PSNR measures pixel-level similarity between generated results and source images, where higher values indicate smaller reconstruction errors and stronger preservation of original image details. Following the PIE-Bench evaluation protocol, PSNR is computed on non-edited regions to evaluate the preservation of unchanged content, including background structures, textures, and local appearance. LPIPS measures perceptual differences based on deep feature representations extracted from pretrained networks. Compared with pixel-level metrics, LPIPS better captures high-level visual similarity and is used to evaluate perceptual consistency between generated results and source images.

Semantic Alignment: Global CLIP Similarity measures the overall semantic correspondence between the generated image and the target editing concept by computing CLIP-based image-text similarity. This metric reflects whether the generated image globally follows the given instruction. Edited-region CLIP Similarity focuses only on the edited area and measures the alignment between the modified content and the target instruction. By reducing the influence of unchanged regions, this metric provides a more direct evaluation of editing effectiveness and semantic correctness.

C User study Details

To further evaluate the perceptual quality of different image editing methods, we conduct a user study on the PIE-Bench test set. We randomly select 50 editing samples covering five representative editing categories, including object replacement, object removal, attribute modification, background editing, and local appearance adjustment. These samples are selected to provide diversity in terms of editing types and difficulty levels.

A total of 50 participants with normal vision and no involvement in model development are recruited for the evaluation. To reduce evaluator fatigue, participants are randomly assigned subsets of evaluation samples rather than being required to assess all samples. For each sample, participants are provided with the source image, editing instruction, and generated results from different methods. The method names are removed, and all generated results are presented in a randomly shuffled order to avoid potential bias caused by prior knowledge of specific methods.

Participants independently evaluate each generated result from three perspectives: source image fidelity, background preservation, and editing effectiveness. All criteria are rated on a five-point Likert scale, where higher scores indicate better performance. The final score for each criterion is calculated by averaging the ratings from all participants.

The evaluation criteria are defined as follows:

(1) Source Fidelity evaluates the overall visual consistency between the generated image and the original source image. A score of 5 indicates that the source image content and structure are almost completely preserved; 4 indicates minor changes while maintaining overall consistency; 3 indicates noticeable changes; 2 indicates significant structural or content deviations; and 1 indicates severe deviation from the source image.

(2) Background Preservation focuses on the structural integrity of non-edited regions, preservation of unchanged objects, local texture consistency, and lighting consistency. A score of 5 indicates complete preservation of non-edited areas; 4 indicates only minor variations; 3 indicates visible disturbances; 2 indicates obvious background changes; and 1 indicates severe structural drift.

(3) Editing Effectiveness measures the degree to which the generated result satisfies the editing instruction, considering target correctness, attribute modification accuracy, and naturalness of the edited region. A score of 5 indicates complete compliance with the instruction; 4 indicates overall satisfaction with minor detail errors; 3 indicates partial fulfillment; 2 indicates editing failure; and 1 indicates that the target edit is not achieved.

Table C.1: User study results on the PIE-Bench benchmark.

Method	Source Fidelity↑	Background Preservation↑	Editing Effectiveness↑
P2P	2.91	2.73	3.18
PnP	3.24	3.37	3.26
MasaCtrl	3.31	3.45	3.14
Null-text Inversion	3.72	3.81	3.05
Direct Inversion	3.54	3.62	3.32
RF-Inversion	3.18	3.21	3.41
FlowEdit	3.46	3.52	3.63
KV-Edit	4.21	**4.35**	3.58
ReFlex	4.05	4.18	<u>3.82</u>
Ours	**4.24**	<u>4.32</u>	**3.90**

Table C.1 presents the user study results. Source-constrained methods (KV-Edit and ReFlex) achieve better Source Fidelity and Background Preservation, indicating that source image guidance effectively reduces undesired changes in non-edited regions. KV-Edit obtains the highest Background Preservation score (4.35), while its lower Editing Effectiveness suggests that fixed K/V fusion limits its editing flexibility. In contrast, CATD-MoE achieves the best Source Fidelity (4.24) and Editing Effectiveness (3.90), improving editing quality while maintaining source image consistency. These results demonstrate that dynamic K/V routing effectively balances source preservation and editing flexibility.

D Additional results

This part presents additional qualitative results of CATD-MoE on PIE-Bench, covering nine editing categories: object replacement (*change object*), object addition (*add object*), object deletion (*delete object*), content modification (*change attribute content*), pose modification (*change attribute pose*),

color modification (*change attribute color*), material modification (*change attribute material*), background modification (*change attribute background*), and style transfer (*change style*). These examples further illustrate the ability of CATD-MoE to balance instruction compliance and source image preservation across diverse editing requirements. Overall, the method performs reliably on object-level edits and localized attribute modifications while preserving the structure and appearance of non-edited regions. Its performance is comparatively weaker on background modification, which requires large-scale scene reconstruction, and some results exhibit less stable instruction alignment and global visual coherence. This observation suggests that the current dynamic routing mechanism is more effective for localized or object-centered editing, while large-scale background replacement remains an important direction for further improvement.

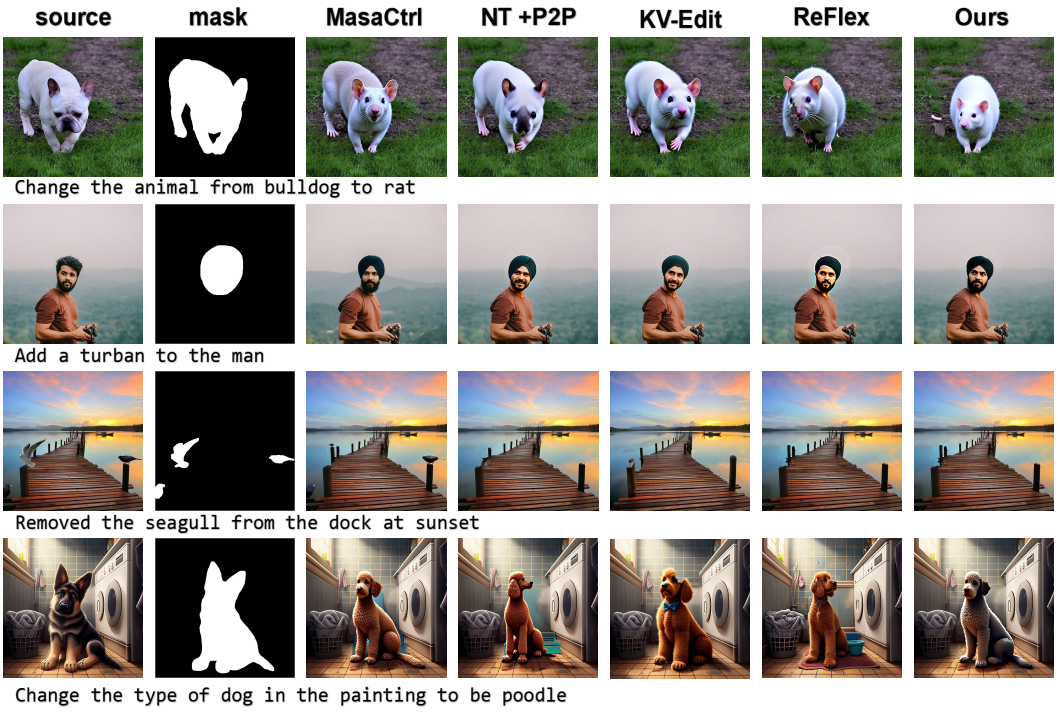

Figure D.1: Additional qualitative comparisons on PIE-Bench for object-level editing. From top to bottom, the examples show object replacement, object addition, object deletion, and object-content modification. The columns from left to right present the source image, editing mask, and results produced by MasaCtrl, NT+P2P, KV-Edit, ReFlex, and CATD-MoE (Ours), respectively. CATD-MoE performs the requested edits while largely preserving the structure and appearance of the unmasked regions.

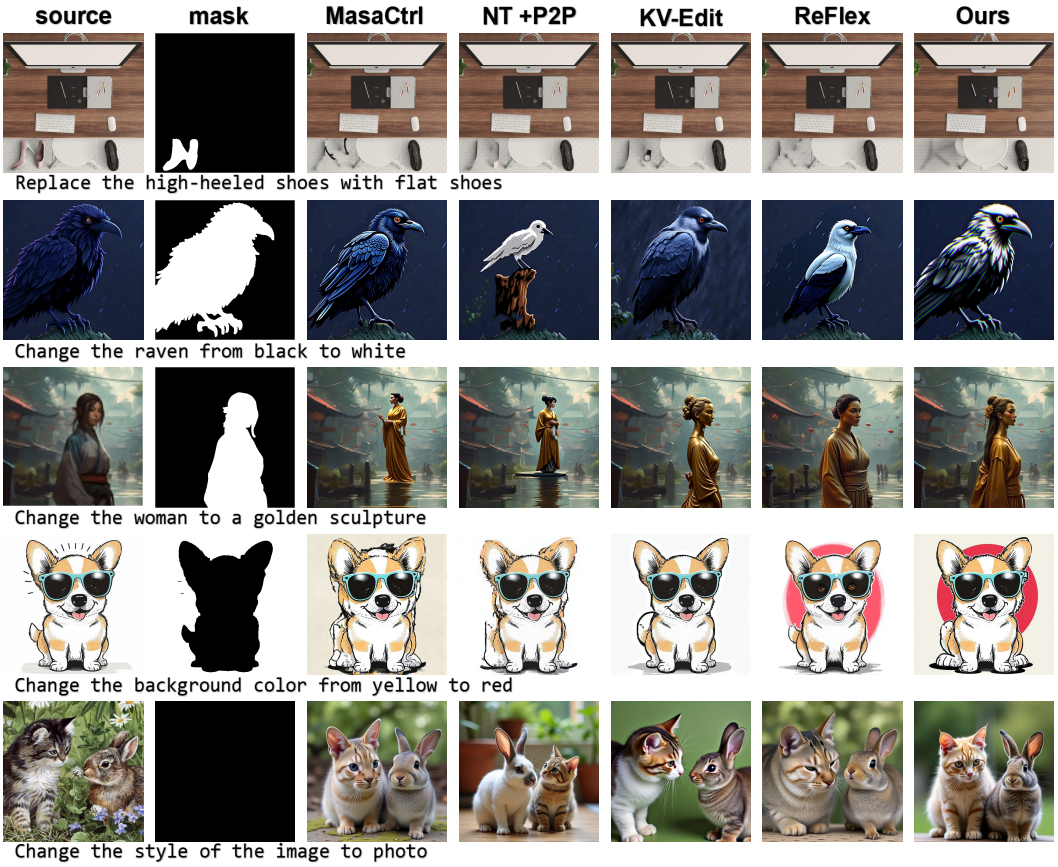

Figure D.2: Additional qualitative comparisons on PIE-Bench for content, attribute, and style editing. From top to bottom, the examples involve replacing high-heeled shoes with flat shoes, changing a raven from black to white, transforming a woman into a golden sculpture, changing the background color from yellow to red, and converting an illustration into a photographic style.