

FineEdit: Fine-Grained Image Edit with Bounding Box Guidance

Haohang Xu¹, Lin Liu² [†], Zhibo Zhang², Rong Cong²,
Xiaopeng Zhang² ^{*}, and Qi Tian² ^{*}

¹ East China Normal University, Shanghai, China

² Huawei, China

hhxu@ee.ecnu.edu.cn, liulin161@huawei.com,
zhangzhibo18@huawei.com, congrong7914@163.com,
zxphistory@gmail.com, tian.qi1@huawei.com

Abstract. Diffusion-based image editing models have achieved significant progress in real world applications. However, conventional models typically rely on natural language prompts, which often lack the precision required to localize target objects. Consequently, these models struggle to maintain background consistency due to their global image regeneration paradigm. Recognizing that visual cues provide an intuitive means for users to highlight specific areas of interest, we utilize bounding boxes as guidance to explicitly define the editing target. This approach ensures that the diffusion model can accurately localize the target while preserving background consistency. To achieve this, we propose **FineEdit**, a multi-level bounding box injection method that enables the model to utilize spatial conditions more effectively. To support this high precision guidance, we present **FineEdit-1.2M**, a large scale, fine-grained dataset comprising 1.2 million image editing pairs with precise bounding box annotations. Furthermore, we construct a comprehensive benchmark, termed **FineEdit-Bench**, which includes 1,000 images across 10 subjects to effectively evaluate region based editing capabilities. Evaluations on FineEdit-Bench demonstrate that our model significantly outperforms state-of-the-art open-source models (e.g., Qwen-Image-Edit and LongCat-Image-Edit) in instruction compliance and background preservation. Further assessments on open benchmarks (GEdit and ImgEdit Bench) confirm its superior generalization and robustness. The dataset and code will be publicly available through the project page at <https://laulampaul.github.io/FineEdit>.

Keywords: Diffusion Model · Image Editing · Visual Guidance

1 Introduction

Diffusion models have recently achieved substantial advancements across various generative applications, including image generation [15, 39, 40, 48], video generation [5, 47, 51], and their respective editing tasks. Fundamentally, these models

[†] Project Leader. ^{*} Corresponding Authors.



Fig. 1: Qualitative comparison between the SOTA closed source model Nano-Banana Pro [12], and our proposed FineEdit. The results (left: Input; middle: Nano-Banana Pro; right: FineEdit) demonstrate that even the Nano family models struggle with precise object localization in complex scenarios. In contrast, by leveraging visual bounding-box instructions, FineEdit achieves superior editing fidelity and localization accuracy.

transform random noise into the target data distribution via a progressive denoising process. State-of-the-art diffusion-based image editing models [24, 43, 56] now produce high quality results that rival professional human editing. However, conventional methods [7, 24, 36] typically rely exclusively on text-based instructions to specify the location and goal of an editing task. While effective in simple, object centric scenarios, these approaches often falter in complex, real world scenes containing multiple objects. As shown in Fig. 1, in complex multi-object scenarios, text alone is often insufficient to unambiguously specify the target. This limitation introduces two primary shortcomings: first, the model may fail to precisely localize the target object, resulting in inaccurate editing; and second, the absence of explicit spatial constraints leaves the background vulnerable to unintended changes, such as shifts in color scheme or layout.

Recent research has extensively explored the incorporation of visual priors into diffusion models to enhance controllability. For instance, ControlNet [64] utilizes side networks to inject structural guidance. However, it typically operates at a holistic level, lacking the precise localization capabilities required for complex, multi-object scenes. Alternatively, inpainting-based approaches leverage masks to isolate specific regions. Within this paradigm, training-free methods like RePaint [31] are constrained by the generative limits of the frozen base model, while training-based models such as BrushNet [20] are primarily optimized for content restoration (i.e., recovering missing pixels) rather than semantic modification. More recently, state-of-the-art editing models (e.g., Qwen-Image-Edit [56] and LongCat-Image-Edit [49]) have allowed users to define editing areas via bounding boxes. Nevertheless, these methods often yield unsatisfactory results. Typical issues include boundary leakage, where edits unintentionally spill into the surrounding background, and visual artifacts, such as residual traces of the bound-

ing boxes appearing in the final output. Consequently, existing approaches still struggle to achieve precise artifact-free editing within strictly defined regions.

To address these issues, we introduce **FineEdit**, a novel framework that leverages bounding-box visual priors to explicitly guide the image editing process. FineEdit resolves two critical challenges: First, we propose a multi-level visual instruction injection architecture that effectively utilizes spatial constraints to guide the denoising process. To further align the model with human perception, we introduce a decoupled post-training reinforcement learning strategy. This strategy independently evaluates foreground modifications and background consistency, overcoming the misaligned reward signals inherent in conventional global VLM scoring. Second, to support the high-precision training paradigm, we present **FineEdit-1.2M**, a rigorously filtered, high quality dataset comprising 1.2 million image editing pairs. Unlike previous editing datasets [11, 23, 53, 54, 62, 66], FineEdit-1.2M ensures high editing quality and strict background consistency. Extensive experiments demonstrate that FineEdit significantly improves spatial instruction following capabilities and background preservation, outperforming state-of-the-art editing models by a large margin.

The key contributions of this paper are summarized as follows:

- We propose FineEdit, a novel architecture utilizing multi-level visual instruction injection to guide the denoising process via bounding-box priors. Additionally, we introduce a decoupled post-training reinforcement learning strategy that independently evaluates foreground edits and background preservation, resolving the reward misalignment typical of global VLM scoring.
- We construct FineEdit-1.2M, a large scale dataset containing 1.2 million high quality image editing pairs with precise spatial annotations and strict background consistency.
- Evaluations demonstrate that FineEdit significantly surpasses existing editing models in both spatial instruction adherence and background preservation, establishing a new standard for fine-grained image editing.

2 Related Work

2.1 Diffusion-based Image Editing

Diffusion-based models [39] treat image editing as a conditional generation task, typically falling into two categories: training-free and training-based frameworks. Training-free methods [8, 14, 18, 22, 41, 50] manipulate the internal representations of a frozen model to preserve source structures (e.g., Prompt-to-Prompt [14], Plug-and-Play [50]). While bypassing fine-tuning costs, they are hyperparameter sensitive and struggle with complex structural modifications. Conversely, training-based models [7, 11, 46, 62, 63, 66] are fine-tuned to directly learn editing mappings, as pioneered by InstructPix2Pix [7]. However, most training-based methods rely solely on textual guidance, lacking the capacity to accept explicit visual spatial instructions (e.g., bounding boxes) for precise localized editing.

2.2 Visual Instruction for Image Editing

Recent works [17, 33, 37, 60, 64, 67] incorporate visual priors to enhance diffusion model controllability. Methods like ControlNet [64], T2I-Adapter [37], and GLIGEN [26] utilize side-networks or attention layers to accept structural maps or bounding boxes. However, these approaches are primarily optimized for conditional generation rather than instruction-based editing. Consequently, they exhibit two major limitations: First, they typically apply visual conditions globally, lacking the granularity to isolate specific objects for modification for background consistency. Second, they struggle to interpret edit instructions, often requiring complex engineering or additional inpainting masks to prevent unintended changes to the rest of the image.

Alternatively, text-guided inpainting methods [2, 20, 24, 25, 35, 45, 59] enable localized generation within masked regions. However, these approaches are fundamentally designed for image reconstruction and completion rather than high level semantic editing. They prioritize seamless boundary transitions over complex structural transformations, often failing to follow intricate editing instructions. Furthermore, implementations relying on separately trained auxiliary modules (e.g., ControlNet-Inpaint [64]) to handle mask constraints frequently suffer from suboptimal feature alignment. Consequently, they exhibit weaker preservation of unmasked regions and produce logically inconsistent edits compared to end-to-end trained editing models.

3 Method

3.1 An Overview of FineEdit Framework

As illustrated in Fig. 2, FineEdit builds upon the Qwen-Image [56] architecture, employing Qwen2.5-VL [4] to encode conditional features from both the text prompt and the source image. The MM-DiT diffusion backbone accepts two primary inputs: noisy target image latents and clean source image latents. Both sets of latents are encoded via a frozen VAE [21] and subsequently concatenated along the token sequence dimension. The training process comprises two stages: *Pre-training stage* establishes multi-level spatial priors using early and deep fusion mechanisms. In early fusion, VAE-encoded masked source tokens are concatenated with noisy target tokens along the channel dimension. In deep fusion, features extracted via a side-adapter are uniformly injected into intermediate transformer layers to enhance the perception of both the input content and the masked regions. *Post-training stage* applies reinforcement learning with a novel decoupled reward function. By separating the conventional global reward into a foreground region-of-interest (ROI) score and a background preservation score, the framework achieves a highly precise assessment of localized editing quality. Each stage is detailed below.

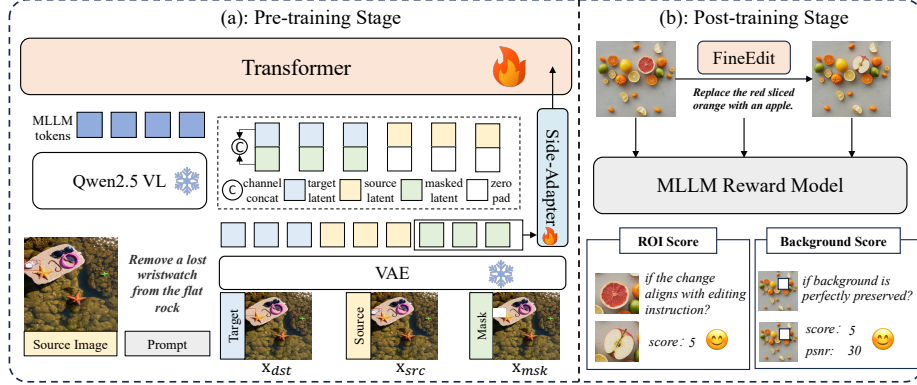


Fig. 2: Overview of FineEdit framework, which includes two training stages: (a) Pre-training stage establishes multi-level spatial priors using early and deep fusion. (b) Post-training stage applies reinforcement learning with a novel decoupled reward function.

3.2 Pre-training: Multi-level Visual Instruction Injection

Preliminaries. In the context of image editing, we are given a source image x_{src} , a target image x_{dst} , and a corresponding text prompt c . A pre-trained VAE first encodes x_{src} and x_{dst} into the latent space, yielding q_{src} and q_{dst} , respectively. Subsequently, the forward diffusion process perturbs the target latent by adding noise: $\hat{q}_{dst} = t\epsilon + (1-t)q_{dst}$. Here, $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ denotes random Gaussian noise, and $t \in [0, 1]$ serves as the timestep parameter controlling the noise magnitude.

Conventional image editing models [24, 56] typically concatenate the flattened source latent q_{src} and the noisy target latent $\hat{q}_{dst}$ along the token sequence dimension. The combined sequence is input to the diffusion transformer f . In a flow matching framework, the model predicts the denoising velocity $\mathbf{v}$ by:

$$\mathbf{v} = f([q_{src}, \hat{q}_{dst}] \mid t, c) \quad (1)$$

As aforementioned, such a pipeline lacks explicit instructions regarding spatial information, which yields two primary challenges: imprecise spatial localization and poor background consistency. To address these issues, we propose to fuse visual instructions via two distinct strategies: *early fusion*, achieved by concatenating visual instructions at the input side, and *deep fusion*, implemented by injecting visual instruction features into the deep transformer blocks.

Early Fusion. Visual instructions typically possess a sparse formulation, for instance, a bounding box is defined by only four parameters. To encode such instructions efficiently, we integrate them directly into the source image space. Specifically, given a bounding box $b = [x_1, y_1, x_2, y_2]$, we generate a binary mask $\mathbf{M} \in \{0, 1\}^{H \times W}$, where pixels outside the bounding box are set to 1 and those

within are set to 0. As illustrated in Fig. 2(a), we apply this mask to the source image $\mathbf{x}_{src}$ via element-wise multiplication, yielding a masked image $\mathbf{x}_{msk} = \mathbf{M} \odot \mathbf{x}_{src}$. Visually, $\mathbf{x}_{msk}$ preserves the visual information in the background while indicating the region of interest (ROI) through the zero-padded area.

$$\mathbf{v} = f([\mathbf{q}_{src} \parallel \mathbf{0}], [\hat{\mathbf{q}}_{dst} \parallel \mathbf{q}_{msk}] \mid t, c) \quad (2)$$

where $\mathbf{q}_{msk}$ denotes the latent representation of the masked image $\mathbf{x}_{msk}$. Here, the operators $[\cdot, \cdot]$ and $[\cdot \parallel \cdot]$ represent concatenation along the sequence and channel dimensions, respectively. The channel-wise fusion $[\hat{\mathbf{q}}_{dst} \parallel \mathbf{q}_{msk}]$ ensures that the denoising process is explicitly conditioned on both the preserved background context and the precise spatial localization of the editing area. For channel consistency, we concatenate the source latent $\mathbf{q}_{src}$ with a zero-padded tensor $\mathbf{0}$ to maintain identical channel dimensionality across inputs.

Deep Fusion. We employ a side adapter module to encode the masked latent into deep feature representations, which are subsequently injected into the intermediate layers of the diffusion transformer. Formally, for the i -th block, the output feature $\mathbf{h}_i$ is computed as:

$$\mathbf{h}_i = \mathcal{F}_i([\mathbf{q}_{src} \parallel \mathbf{0}], [\hat{\mathbf{q}}_{dst} \parallel \mathbf{q}_{msk}] \mid t, c) + \mathcal{A}(\mathbf{q}_{msk}) \quad (3)$$

where $\mathcal{F}_i$ denotes the backbone transformer block and $\mathcal{A}$ represents the side adapter. We utilize a 6-layer transformer as the side adapter, which introduces 10% additional parameters relative to the backbone.

We observe that this module not only boosts localization capacity but also significantly accelerates convergence. This efficiency stems from the injection of source derived features directly into deeper layers, effectively functioning as a skip connection.

3.3 Post-training: Reinforcement Learning with Decoupled Rewards

Conventional reinforcement learning post-training frameworks for image editing typically rely on a holistic reward function. These rewards, often derived from pre-trained VLMs such as GPT-4o, evaluate generated samples by computing global relative advantages. However, we contend that such coarse-grained signals are insufficient for fine-grained editing tasks, as they frequently fail to discern subtle semantic changes between the pre-edit and post-edit images.

To address this, we leverage our proposed FineEdit-1.2M dataset (detailed below), which provides dense bounding box annotations to decompose the original global reward into two complementary components: *local editing fidelity* and *background preservation*. Formally, the total reward $R(\mathbf{x}_{dst})$ is formulated as:

$$R(\mathbf{x}_{dst}) = R_{roi} + R_{bg} \quad (4)$$

Specifically, for the Region of Interest (ROI), we employ Qwen3-VL [3] to compute R_{roi} , focusing on the semantic accuracy of the local edit. For the background

reward R_{bg} , we integrate rule-based metrics with model-based scores to strike a balance between precise pixel-level preservation and perceptual naturalness:

$$R_{roi} = \Phi_{\text{VLM}}(\mathbf{x}_{src} \odot \mathbf{M}, \mathbf{x}_{dst} \odot \mathbf{M}, c_1) \quad (5)$$

$$R_{bg} = \Phi_{\text{VLM}}(\mathbf{x}_{src} \odot \bar{\mathbf{M}}, \mathbf{x}_{dst} \odot \bar{\mathbf{M}}, c_2) + \Psi_{\text{PSNR}}(\mathbf{x}_{src} \odot \bar{\mathbf{M}}, \mathbf{x}_{dst} \odot \bar{\mathbf{M}}) \quad (6)$$

where $\mathbf{M}$ denotes the binary mask and $\bar{\mathbf{M}} = \mathbf{1} - \mathbf{M}$ represents its complement. The terms c_1 and c_2 are the corresponding VLM scoring prompts. $\Phi_{\text{VLM}}(\cdot)$ denotes a normalization function that maps the raw VLM scores to a bounded interval of $[0, 1]$, and $\Psi_{\text{PSNR}}(\cdot)$ represents the PSNR metric, which is further normalized via a min-max scaling approach.

3.4 Unified Visual Instruction for Versatile Editing

Beyond enhancing localization precision, our visual instruction mechanism serves as a unified interface that generalizes seamlessly across diverse editing granularity. As shown in Fig. 3, by simply manipulating the spatial extent and quantity of the bounding boxes, FineEdit can adapt to Local, Global, or Hybrid editing scenarios without architectural modifications.

Local and Global Unification. The bounding box B acts as a flexible controller for the editing scope. For standard *Local Editing* tasks (e.g., Add, Replace, Remove), the user specifies a region $B_{local} \subset \mathcal{I}$ to constrain the generation within the object of interest. Conversely, for *Global Editing* tasks such as global style transfer or background replacement, the instruction is naturally extended by expanding the bounding box to cover the entire image canvas ($B_{global} \approx \mathcal{I}$). This allows the model to leverage the same attention-masking mechanism to perform holistic regeneration while adhering to the text prompt.

Hybrid Editing. Our framework inherently supports multi-turn or multi-focus editing within a single pass. By providing multiple bounding boxes $\{B_1, \dots, B_n\}$, users can perform *Hybrid Editing*, such as simultaneously replacing an object in the foreground while modifying a distant background element. The model attends to these disjoint regions independently within the latent space, ensuring that complex, multi-objective instructions are executed without interference.

Local Style Transfer. Most notably, our experimental validation highlights a distinct capability of FineEdit: *Local Style Transfer*. Existing diffusion-based editing models often struggle to disentangle style from content spatially, typically applying style changes (e.g., “oil painting style”, “cyberpunk theme”) globally to the entire image. In contrast, by combining a local bounding box B_{local} with a stylistic text prompt, our method effectively confines the style migration to the specific target object while preserving the semantic and stylistic integrity of the surrounding background. This granular control over style distribution represents a significant advancement over current state-of-the-art methods.

4 FineEdit Dataset and Benchmark

Conventional approaches for constructing image editing datasets typically treat data pair generation and quality screening as decoupled stages. However, with

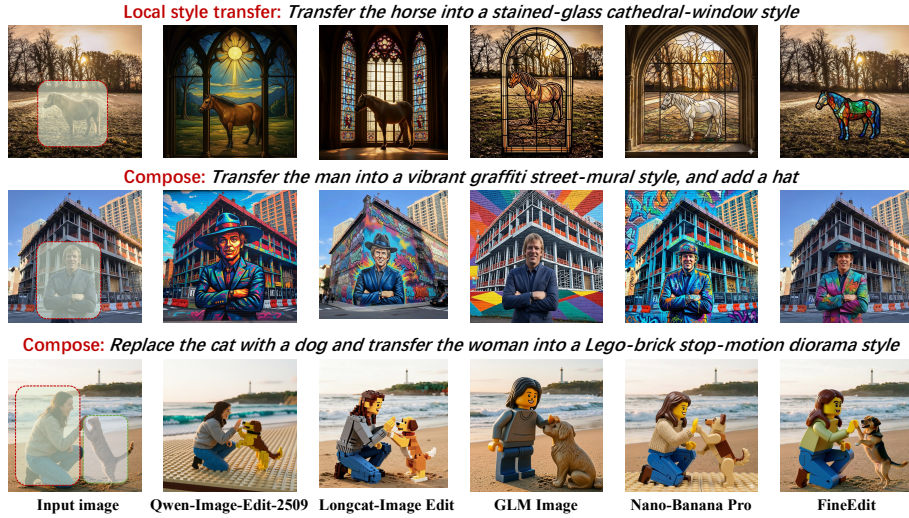


Fig. 3: Unified visual instructions for diverse editing settings. Three representative configurations are showcased: localized style transfer, single-bbox composite editing, and multi-bbox composite editing. FineEdit provides a flexible and consistent interface for both single-target and multi-target manipulation.

the introduction of bounding box (bbox) annotations to spatially localize editing regions, we observed that such a separated workflow leads to significant error accumulation across distinct steps. To address this challenge, we developed a unified pipeline (named U-GAF Pipeline) that integrates data generation, spatial bbox annotation, and rigorous data filtering into a cohesive framework. This synergistic approach ensures high alignment between the visual content and annotations, serving as the foundation for our FineEdit-1.2M dataset.

4.1 U-GAF Pipeline: Unified Generation, Annotation, and Filtering

The proposed U-GAF Pipeline consists of four stages: 1) **Data Curation**, 2) **Data Annotation**: generation of bounding box locations and edit prompt, 3) **Edit Generation**: generation of edit pairs, and 4) **Data Refinement**: generation of final image pairs. The flowchart of the Pipeline is illustrated in Fig. 4.

Data Curation. Our training data is curated from multiple sources, including Megalith-10M [6], LAION-Aesthetic-6M [42], and 5M internal images. To mitigate the bias towards simple scenes with sparse objects and synthetic generation datasets, we include diverse perception oriented datasets, including LVIS [13], COCO [28], and CC3M [44]. Following the filtering pipeline [56] that excludes samples with suboptimal saturation, clarity, or low RGB entropy, we select a subset of 6M high-quality images for subsequent processing.

Data Annotation. We use Co-DETR [71] for object localization, and filter out roughly 20% of the initial samples by excluding boxes with low confidence

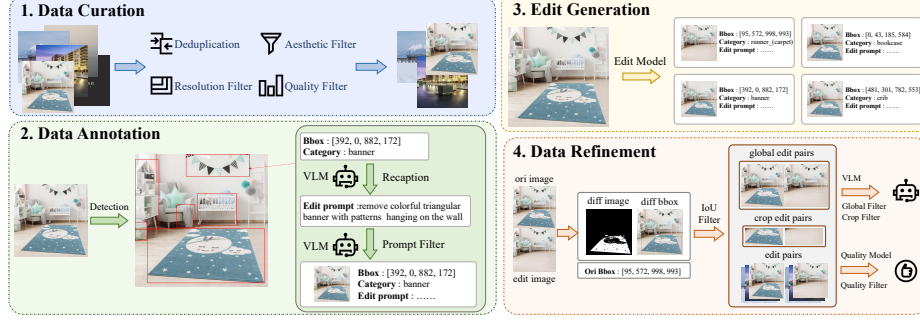


Fig. 4: The U-GAF Pipeline, which consists of four synergistic stages for high-quality data synthesis: (1) **Data Curation**, (2) **Data Annotation**, (3) **Edit Generation**, and (4) **Data Refinement**.

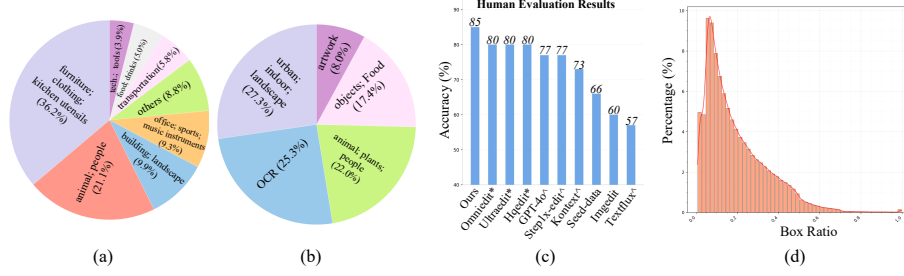


Fig. 5: Information of our proposed FineEdit dataset. (a) Training dataset distribution. (b) Evaluation dataset distribution. (c) Human evaluation accuracy on different datasets. (d) Distribution of bounding box area ratios.

score (threshold 0.7) and extremely large box ratios. InternVL3-38B [70] is deployed to generate fine-grained editing instructions based on the refined visual inputs. Then we use Qwen3-32B-thinking to filter the editing instructions. The screening criteria aim to exclude the unreasonable image editing instructions, including inappropriate human-centric manipulations and content that violates safety guidelines (e.g., NSFW material).

Edit Generation. To generate diverse and high-quality edited samples, we incorporate multiple diffusion-based editing models as our synthetic engines, primarily Flux1-kontext-dev [24] and Qwen-Image-Edit [56].

Data Refinement. To guarantee the high fidelity and precision of the FineEdit-1.2M dataset, we implement a rigorous four-stage sequential filtering pipeline, where each sample must successfully pass through every consecutive phase. We first enforce spatial consistency by calculating the Intersection over Union (IoU) between the detected change mask (difference of original and edited images) and the ground-truth bounding box (generated by Co-DETR), discarding edits that drift significantly outside the target region. Subsequently, we employ an RGB

Entropy (RGBE) metric to assess information density, effectively filtering out trivial samples with low visual complexity (e.g., simple objects on monochromatic backgrounds). The pipeline then proceeds to semantic verification using Qwen3-VL-32B, which operates on two scales: a global pass to ensure overall visual coherence with the instruction, and a fine-grained local crop pass that specifically validates the editing quality within the designated bounding box. This coarse-to-fine strategy ensures that the final dataset maintains both global naturalness and precise local alignment with user prompts. Finally, we employ an image editing quality assessment model EditScore [32], to filter out artifacts and enhance the realism of the final dataset.

Fig. 5(a) and (d) provide a detailed breakdown of our FineEdit-1.2M dataset. To assess the quality of FineEdit-1.2M, we conduct a comprehensive evaluation involving both human annotation and quantitative comparisons with existing open-source image editing datasets, such as filtered Omniedit, Ultraedit, and Hqedit selected from GPT-Image-Edit-1.5M [19, 53, 54, 66] (denoted by * in Fig. 5(c)), and filtered Kontext, Step1X-Edit, TextFlux, and GPT-4o selected from X2Edit Dataset [24, 30, 34] (denoted by ^ in Fig. 5(c)). As illustrated in the results, our dataset demonstrates superior annotation accuracy and broader stylistic diversity compared to prevailing alternatives. Specifically, human evaluators favor FineEdit-1.2M for its precise alignment between textual instructions and spatial bounding-box cues, validating its effectiveness for training high-fidelity controllable editing models.

4.2 FineEdit-1k Evaluation Bench

To build diversity in the evaluation dataset, we refer to the category settings used in the data collection of Qwen-Image [56]. We have set up data for ten categories: Objects, Urban Scenes, Indoors, Landscapes, Food, Plants, Animals, People, Text, and Artwork. Fig. 5(b) shows the details of our FineEdit-1k Evaluation Bench. To ensure the high quality of our evaluation bench, we constructed three types of evaluation tasks: addition, removal, and replacement, based on the data from NHR-Edit [23] and MARIO-10M [9] (Text data). Specifically, we compute the pixel-wise difference between the pre- and post-edit images. Subsequently, morphological operations, including filtering, erosion, and dilation, are applied to generate a binary mask. Finally, the bounding box is derived from the largest rectangular region within this mask. To obtain high-quality prompts, we regenerate the ‘remove’ prompts using Qwen2.5-7B and the ‘replace’ prompts using Qwen2.5-VL-32B. Finally, we conducted manual quality checks on all evaluation data pairs to ensure that the images were consistent with their categories, the prompts were appropriate, and the bounding box positions were correct.

5 Experiments

5.1 Experimental Setup

Implementation Details. The training of FineEdit follows a two-stage paradigm: pre-training followed by reinforcement learning-based post-training. In pre-training

stage, we train FineEdit on the FineEdit-1.2M dataset using 64 NVIDIA H100 GPUs. The per-device batch size is set to 1, with gradient accumulation over 8 steps to achieve an effective global batch size of 512. We employ the AdamW optimizer with a learning rate of 5×10^{-5} . All training images are resized to a fixed resolution of 1024×1024 , and the total training duration is 15,000 steps. In post-training stage, following standard RL post-training protocols [68], we incorporate LoRA [16] layers with a rank of $r = 32$ and an alpha of $\alpha = 64$, while keeping the transformer backbone frozen. The learning rate for this stage is 3×10^{-4} . The RL optimization is organized into 24 groups per epoch, with a group size of 16 for relative advantage calculation.

Evaluation Benchmarks. We evaluate the performance of our method on the FineEdit-1k Benchmark, a curated test set designed to assess fine-grained editing capabilities. To further demonstrate the generalization of FineEdit, we also report results on the Single-turn subset of ImgEdit-Bench [61] and GEdit-Bench-EN [30], respectively. These two test sets encompass a broader range of editing categories beyond the standard addition, removal, and replacement tasks, providing a robust test for versatile editing scenarios.

Evaluation Metrics. To comprehensively assess performance, we employ a multi-dimensional evaluation protocol covering reconstruction quality, semantic alignment, and perceptual fidelity:

- **Low-level Metrics:** We report PSNR, SSIM, and LPIPS calculated specifically on regions outside the boxes to quantify background preservation.
- **Semantic Alignment:** We utilize CLIP scores to measure the correspondence between edited results and text instructions within the bbox.
- **VLM Evaluation:** We use VLM to evaluate on four dimensions [61]: Prompt Compliance (PC), Visual Naturalness (VN), Physical & Detail Integrity (PDI), and Out-of-Box Retention (OBR). Notably, OBR assesses the semantic consistency of non-edited regions to complement pixel-level metrics.

5.2 Quantitative Results

We evaluate FineEdit against SOTA methods on three benchmarks: FineEdit-1k bench, ImgEdit-bench and GEdit-bench. All inference results are generated using a classifier-free guidance (CFG) scale of 4 and 30 sampling steps.

Performance on FineEdit-1k Bench. In Table 1, we categorize the baseline models into two groups based on their input modalities. The upper section comprises Flux-based models [24], which typically generate edited results via a mask and a descriptive prompt. Since their prompt requirements differ from instruction-based methods (e.g., add/remove...), we manually refined the prompts for these baselines to better align with their architectural priors. This adaptation yielded a significant performance boost, ensuring a fair and rigorous comparison. The lower section presents recent specialized editing methods that utilize direct bounding box inputs for spatial localization. As evidenced by the

Table 1: Quantitative Comparison with State-of-the-Art methods on FineEdit-1k Evaluation Bench. PC, VN, PDI, and OBR mean Prompt Compliance, Visual Naturalness, Physical & Detail Integrity, and Out-of-Box Retention, respectively.

Model	Background Preservation				BBox Editing Accuracy			
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	OBR $\uparrow$	CLIP $\uparrow$	PC $\uparrow$	VN $\uparrow$	PDI $\uparrow$
Flux-fill-dev [24]	34.2	0.90	0.08	4.99	0.26	3.56	4.03	3.98
BrushNet [20]	29.6	0.83	0.13	4.99	0.25	3.28	3.38	3.34
BrushEdit [25]	30.9	0.84	0.13	4.95	0.25	3.27	3.45	3.41
PrefPaint [29]	27.1	0.77	0.21	4.83	0.23	2.74	3.01	2.96
Asuka-flux [52]	34.4	0.88	0.16	5.00	0.22	2.20	2.92	2.85
FluPA [45]	32.7	0.89	0.09	4.96	0.26	3.68	4.05	4.01
GLM-Image [69]	27.4	0.85	0.14	4.48	0.25	3.66	3.98	3.95
Z-Image-turbo-Control [1]	28.7	0.86	0.10	4.99	0.24	3.18	3.51	3.48
LongCat-Image-Edit [49]	27.3	0.83	0.12	4.38	0.26	4.59	4.52	4.56
Qwen-Image-Edit [56]	31.4	0.88	0.11	4.23	0.25	3.94	4.21	4.19
Qwen-Image-Edit-2509 [56]	27.1	0.80	0.13	4.15	0.26	4.56	4.66	4.65
FineEdit	34.4	0.91	0.08	4.80	0.27	4.67	4.71	4.69
FineEdit-r1	34.7	0.91	0.09	4.89	0.27	4.80	4.83	4.81

Table 2: Quantitative Comparison with SOTA methods on ImgEdit-Bench.

Model	Add	Adjust	Extract	Replace	Remove	Background	Style	Hybrid	Action	Overall
OmniGen [58]	3.47	3.04	1.71	2.94	2.43	3.21	4.19	2.24	3.38	2.96
ICEdit [65]	3.58	3.39	1.73	3.15	2.93	3.08	3.84	2.04	3.68	3.05
Step1X-Edit [30]	3.88	3.14	1.76	3.40	2.41	3.16	4.63	2.64	2.52	3.06
BAGEL [10]	3.56	3.31	1.70	3.30	2.62	3.24	4.49	2.38	4.17	3.20
UniWorld-V1 [27]	3.82	3.64	2.27	3.47	3.24	2.99	4.21	2.96	2.74	3.26
OmniGen2 [57]	3.57	3.06	1.77	3.74	3.20	3.57	4.81	2.52	4.68	3.44
Kontext-pro [24]	4.25	4.15	2.35	4.56	3.57	4.26	4.57	3.68	4.63	4.00
Kontext-dev [24]	4.12	3.80	2.04	4.22	3.09	3.97	4.51	3.35	4.25	3.71
GPT-4o-Image [38]	4.61	4.33	2.90	4.35	3.66	4.57	4.93	3.96	4.89	4.20
Qwen-Image-Edit [56]	4.38	4.16	3.43	4.66	4.14	4.38	4.81	3.82	4.69	4.27
Qwen-Image-Edit-2509 [56]	4.32	4.36	4.04	4.64	4.52	4.37	4.84	3.39	4.71	4.35
FineEdit	4.53	4.57	4.03	4.83	4.65	4.53	4.68	3.85	4.86	4.50

results, FineEdit consistently outperforms all competing methods. Notably, FineEdit achieves superior scores in both region-of-interest (ROI) editing fidelity and background preservation, demonstrating its robust localization and high-fidelity synthesis capabilities. Furthermore, our model variant subjected to RL post-training, denoted as FineEdit-r1, yields further performance gains.

Performance on ImgEdit and GEdit Bench. In Table 2 and Table 3, we further demonstrate the versatility of our approach. Beyond the standard *add*, *remove*, and *replace* tasks, FineEdit maintains robust performance across a diverse range of editing scenarios by simply adjusting the bounding box format. This underscores the flexibility of our unified visual instruction in handling complex, multi-target, and out-of-distribution editing tasks. For fair comparisons, we report results obtained solely after the pre-training stage, without the task specific RL fine-tuning across both evaluation benchmarks.

Table 3: Quantitative Comparison with SOTA methods on GEdit-Bench-EN.

Model	G_SC	G_PQ	G_O	Model	G_SC	G_PQ	G_O
Instruct-P2P [7]	3.58	5.49	3.68	GPT-Image-1 [High]	7.85	7.62	7.53
AnyEdit [62]	3.18	5.82	3.21	Gemini 2.0	6.73	6.61	6.32
MagicBrush [63]	4.68	5.66	4.52	Kontext-dev [24]	6.52	7.38	6.00
UniWorld-V1 [27]	4.93	7.43	4.85	Kontext-pro [24]	7.02	7.60	6.56
OmniGen [58]	5.96	5.89	5.06	UniPic2 [55]	-	-	7.10
OmniGen2 [57]	7.16	6.77	6.41	Qwen-Image-Edit [56]	8.00	7.86	7.56
BAGEL [10]	7.36	6.83	6.52	Qwen-Image-Edit-2509 [56]	8.15	7.86	7.54
Step1X-Edit [30]	7.66	7.35	6.97	FineEdit	8.71	7.47	7.97

Table 4: Ablation study on fusion strategies and reward functions.

Ablation Module		Background Preservation				Bbox Editing Accuracy			
		PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	OBR $\uparrow$	CLIP $\uparrow$	PC $\uparrow$	VN $\uparrow$	PDI $\uparrow$
Fusion	Baseline [56]	27.1	0.80	0.13	4.15	0.26	4.56	4.66	4.65
	Early	15.9	0.48	0.40	2.43	0.21	2.45	1.85	1.85
	Deep	33.1	0.88	0.10	4.61	0.26	4.52	4.51	4.61
	Early + Deep	34.4	0.91	0.08	4.80	0.27	4.67	4.71	4.69
Reward	Pre-RL	34.4	0.91	0.08	4.80	0.27	4.67	4.71	4.69
	Global	32.7	0.90	0.10	4.76	0.27	4.74	4.76	4.73
	Decoupled	34.7	0.91	0.09	4.89	0.27	4.80	4.83	4.81

5.3 Qualitative Results

In Fig. 6, we present a visual comparison of our method with other state-of-the-art methods. The red boxes in the first column indicate the regions to be edited. The second example is a case of complex prompt coupled with fine-grained object replacement. Other methods produced editing errors (Asuka-flux removed the object but did not add the certificate; BrushNet added the wrong object; Qwen-Image-Edit added objects outside the red box; and GLM-image incorrectly extracted the object). The last example demonstrates the text editing capability. Our method successfully modified the word while matching the color and font style with the original input image. Fig. 3 demonstrates that our method exhibits superior local editing capabilities (as seen in the first and second samples) and hybrid editing performance (third sample) compared to existing approaches. Furthermore, we analyze the model’s editing performance across a wide range of bbox scales. As illustrated in Fig. 7, FineEdit demonstrates consistent performance across diverse bbox scales and ratios, maintaining high-quality editing results from expansive regions to extremely localized areas.

5.4 Ablation Study

Impact of Early and Deep Fusion. We incorporate bounding box conditions into FineEdit using both early and deep fusion mechanisms. To evaluate their

Replace vivid teal sun-hat with a delicate white parasol adorned with gold trim, positioned gracefully beside the couple*Replace stethoscope hanging on the wall with a framed certificate or diploma showcasing veterinary credential**Remove a small waterproof speaker playing music by the waters edge**Replace the word BREELAN with CREELAN*

Input image

Asuka-flux

BrushEdit

GLM-image

Qwen-Image Edit

Qwen-Image-Edit-2509

FineEdit

Fig. 6: Comparison on FineEdit-1k Evaluation Bench.

individual contributions, we conduct an ablation study isolating each strategy. As shown at the top of Table 4, combining both methods yields optimal performance. Specifically, using early fusion alone causes convergence issues, despite initialization with pre-trained Qwen-Image-Edit weights. Conversely, deep fusion alone avoids these issues by preserving the original input manifold at the initial layer, yet it still underperforms the full model. This demonstrates that early fusion is essential for providing strong spatial priors, while deep fusion ensures robust multi-level feature alignment.

Impact of Decoupled Reward for RL Post-training. We investigate the influence of our proposed decoupled reward function during the post-training stage. As summarized in the bottom of Table 4, we compare its performance against a conventional global reward mechanism. Our results indicate that while the global reward yields only marginal improvements in localized editing accuracy, it significantly compromises background preservation. In contrast, by utilizing the decoupled reward function, both editing precision within the bounding box and background fidelity are consistently enhanced. Notably, we observe a substantial boost in region-specific editing accuracy, validating the effectiveness of our strategy in balancing local manipulation with global context retention.

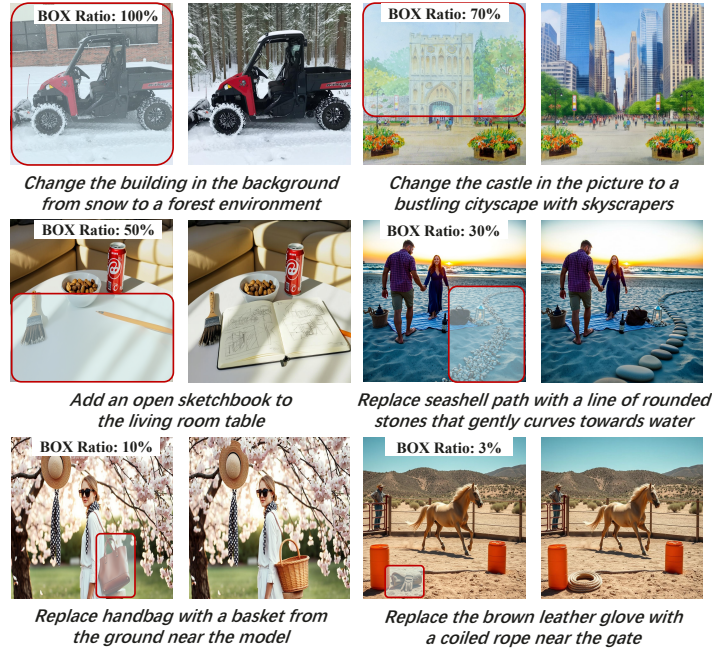


Fig. 7: Robustness to Various Bounding Box Scales and Ratios.

6 Conclusion

This paper presents FineEdit, a framework designed to address the challenges of precise spatial localization and background consistency in diffusion-based image editing. By shifting from text-only prompts to multi-level bounding-box guidance, FineEdit enables intuitive, high precision control over editing targets. To facilitate the training and evaluation of these capabilities, we introduce FineEdit-1.2M, a dataset comprising 1.2 million annotated pairs, alongside FineEdit-Bench, a comprehensive benchmark for region-based editing. Extensive experiments demonstrate that our approach significantly outperforms state-of-the-art models in instruction compliance and perceptual fidelity. Furthermore, FineEdit generalizes robustly across diverse edit benchmarks. This unified visual instruction paradigm, together with the accompanying dataset, provides a strong foundation for future research in high-fidelity, controllable image editing.

Limitations and Future Work. Although FineEdit excels in spatial control and background preservation, it relies primarily on bounding boxes for visual guidance. While effective for defining object boundaries, bounding boxes lack the flexibility needed for fine-grained tasks like irregular shape modifications or precise path-based editing. Future work will extend FineEdit to support interactive prompts such as points, scribbles, and polygons, enabling complex geometric transformations and more granular control over generation.

References

1. Alibaba Cloud PAI Team: Z-image-turbo-fun-controlnet-union-2.1. <https://huggingface.co/alibaba-pai/Z-Image-Turbo-Fun-Controlnet-Union-2.1> (2025), accessed: 2026-01-12
2. Avrahami, O., Fried, O., Lischinski, D.: Blended latent diffusion. *ACM transactions on graphics (TOG)* **42**(4), 1–11 (2023)
3. Bai, S., Cai, Y., Chen, R., Chen, K., Chen, X., Cheng, Z., Deng, L., Ding, W., Gao, C., Ge, C., et al.: Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631* (2025)
4. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., et al.: Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923* (2025)
5. Blattmann, A., Dockhorn, T., Kulal, S., Mendelevitch, D., Kilian, M., Lorenz, D., Levi, Y., English, Z., Voleti, V., Letts, A., et al.: Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127* (2023)
6. Boer Bohan, O.: Megalith-10m. <https://huggingface.co/datasets/madebyollin/megalith-10m> (2024), accessed: 2024-10-07
7. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 18392–18402 (2023)
8. Cao, M., Wang, X., Qi, Z., Shan, Y., Qie, X., Zheng, Y.: Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 22560–22570 (2023)
9. Chen, J., Huang, Y., Lv, T., Cui, L., Chen, Q., Wei, F.: Textdiffuser: Diffusion models as text painters. *Advances in Neural Information Processing Systems* **36**, 9353–9387 (2023)
10. Deng, C., Zhu, D., Li, K., Gou, C., Li, F., Wang, Z., Zhong, S., Yu, W., Nie, X., Song, Z., et al.: Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683* (2025)
11. Ge, Y., Zhao, S., Li, C., Ge, Y., Shan, Y.: Seed-data-edit technical report: A hybrid dataset for instructional image editing. *arXiv preprint arXiv:2405.04007* (2024)
12. Google DeepMind: Introducing nano banana pro. <https://blog.google/innovation-and-ai/products/nano-banana-pro/> (2025), accessed: 2026-02-27
13. Gupta, A., Dollar, P., Girshick, R.: Lvis: A dataset for large vocabulary instance segmentation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 5356–5364 (2019)
14. Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626* (2022)
15. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. *Advances in neural information processing systems* **33**, 6840–6851 (2020)
16. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al.: Lora: Low-rank adaptation of large language models. *ICLR* (2022)
17. Huang, L., Chen, D., Liu, Y., Shen, Y., Zhao, D., Zhou, J.: Composer: creative and controllable image synthesis with composable conditions. In: *Proceedings of the 40th International Conference on Machine Learning*. pp. 13753–13773 (2023)

18. Huberman-Spiegelglas, I., Kulikov, V., Michaeli, T.: An edit friendly ddpm noise space: Inversion and manipulations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12469–12478 (2024)
19. Hui, M., Yang, S., Zhao, B., Shi, Y., Wang, H., Wang, P., Zhou, Y., Xie, C.: Hq-edit: A high-quality dataset for instruction-based image editing. arXiv preprint arXiv:2404.09990 (2024)
20. Ju, X., Liu, X., Wang, X., Bian, Y., Shan, Y., Xu, Q.: Brushnet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In: European Conference on Computer Vision. pp. 150–168. Springer (2024)
21. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
22. Kulikov, V., Kleiner, M., Huberman-Spiegelglas, I., Michaeli, T.: Flowedit: Inversion-free text-based editing using pre-trained flow models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19721–19730 (2025)
23. Kuprashevich, M., Alekseenko, G., Tolstykh, I., Fedorov, G., Suleimanov, B., Dokholyan, V., Gordeev, A.: Nohumansrequired: Autonomous high-quality image editing triplet mining. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 6059–6068 (2026)
24. Labs, B.F., Batifol, S., Blattmann, A., Boesel, F., Consul, S., Diagne, C., Dockhorn, T., English, J., English, Z., Esser, P., et al.: Flux. 1 kontekst: Flow matching for in-context image generation and editing in latent space. arXiv preprint arXiv:2506.15742 (2025)
25. Li, Y., Bian, Y., Ju, X., Zhang, Z., Zhuang, J., Shan, Y., Zou, Y., Xu, Q.: Brushedit: All-in-one image inpainting and editing. IEEE Transactions on Pattern Analysis and Machine Intelligence (2026)
26. Li, Y., Liu, H., Wu, Q., Mu, F., Yang, J., Gao, J., Li, C., Lee, Y.J.: Gligen: Open-set grounded text-to-image generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22511–22521 (2023)
27. Lin, B., Li, Z., Cheng, X., Niu, Y., Ye, Y., He, X., Yuan, S., Yu, W., Wang, S., Ge, Y., et al.: Uniworld-v1: High-resolution semantic encoders for unified visual understanding and generation. arXiv preprint arXiv:2506.03147 (2025)
28. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
29. Liu, K., Zhu, Z., Li, C., Liu, H., Zeng, H., Hou, J.: Prefpaint: Aligning image inpainting diffusion model with human preference. Advances in Neural Information Processing Systems **37**, 30554–30589 (2024)
30. Liu, S., Han, Y., Xing, P., Yin, F., Wang, R., Cheng, W., Liao, J., Wang, Y., Fu, H., Han, C., et al.: Step1x-edit: A practical framework for general image editing. arXiv preprint arXiv:2504.17761 (2025)
31. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L.: Repaint: Inpainting using denoising diffusion probabilistic models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11461–11471 (2022)
32. Luo, X., Wang, J., Wu, C., Xiao, S., Jiang, X., Lian, D., Zhang, J., Liu, D., et al.: Editscore: Unlocking online rl for image editing via high-fidelity reward modeling. arXiv preprint arXiv:2509.23909 (2025)
33. Ma, J., Liang, J., Chen, C., Lu, H.: Subject-diffusion: Open domain personalized text-to-image generation without test-time fine-tuning. In: ACM SIGGRAPH 2024 Conference Papers. pp. 1–12 (2024)

34. Ma, J., Zhu, X., Pan, Z., Peng, Q., Guo, X., Chen, C., Lu, H.: X2edit: Revisiting arbitrary-instruction image editing through self-constructed data and task-aware representation learning. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 40, pp. 7764–7772 (2026)
35. Manukyan, H., Sargsyan, A., Atanyan, B., Wang, Z., Navasardyan, S., Shi, H.: Hd-painter: high-resolution and prompt-faithful text-guided image inpainting with diffusion models. In: *International Conference on Learning Representations*. vol. 2025, pp. 96301–96330 (2025)
36. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.Y., Ermon, S.: Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073* (2021)
37. Mou, C., Wang, X., Xie, L., Wu, Y., Zhang, J., Qi, Z., Shan, Y.: 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*. vol. 38, pp. 4296–4304 (2024)
38. OpenAI: Image generation api. <https://openai.com/index/image-generation-api> (2025), accessed: 2026-02-27
39. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: 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)
40. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems* **35**, 36479–36494 (2022)
41. Samuel, D., Meiri, B., Maron, H., Tewel, Y., Darshan, N., Avidan, S., Chechik, G., Ben-Ari, R.: Lightning-fast image inversion and editing for text-to-image diffusion models. In: *International Conference on Learning Representations*. vol. 2025, pp. 38384–38404 (2025)
42. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al.: Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in neural information processing systems* **35**, 25278–25294 (2022)
43. Seedream, T., Chen, Y., Gao, Y., Gong, L., Guo, M., Guo, Q., Guo, Z., Hou, X., Huang, W., Huang, Y., et al.: Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427* (2025)
44. Sharma, P., Ding, N., Goodman, S., Soricut, R.: Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. pp. 2556–2565 (2018)
45. Shen, Y., Yuan, J., Aonishi, T., Nakayama, H., Ma, Y.: Follow-your-preference: Towards preference-aligned image inpainting. *arXiv preprint arXiv:2509.23082* (2025)
46. Sheynin, S., Polyak, A., Singer, U., Kirstain, Y., Zohar, A., Ashual, O., Parikh, D., Taigman, Y.: Emu edit: Precise image editing via recognition and generation tasks. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 8871–8879 (2024)
47. Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al.: Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792* (2022)
48. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456* (2020)

49. Team, M.L., Ma, H., Tan, H., Huang, J., Wu, J., He, J.Y., Gao, L., Xiao, S., Wei, X., Ma, X., et al.: Longcat-image technical report. arXiv preprint arXiv:2512.07584 (2025)
50. Tumanyan, N., Geyer, M., Bagon, S., Dekel, T.: 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. pp. 1921–1930 (2023)
51. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
52. Wang, Y., Cao, C., Yu, J., Fan, K., Xue, X., Fu, Y.: Towards enhanced image inpainting: Mitigating unwanted object insertion and preserving color consistency. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 23237–23248 (2025)
53. Wang, Y., Yang, S., Zhao, B., Zhang, L., Liu, Q., Zhou, Y., Xie, C.: Gpt-image-edit-1.5 m: A million-scale, gpt-generated image dataset. arXiv preprint arXiv:2507.21033 (2025)
54. Wei, C., Xiong, Z., Ren, W., Du, X., Zhang, G., Chen, W.: Omniedit: Building image editing generalist models through specialist supervision. In: International Conference on Learning Representations. vol. 2025, pp. 259–271 (2025)
55. Wei, H., Xu, B., Liu, H., Wu, S., Liu, J., Peng, Y., Wang, P., Liu, Z., He, J., Xietian, Y., et al.: Skywork unipic 2.0: Building kontekst model with online rl for unified multimodal model. arXiv preprint arXiv:2509.04548 (2025)
56. Wu, C., Li, J., Zhou, J., Lin, J., Gao, K., Yan, K., Yin, S.m., Bai, S., Xu, X., Chen, Y., et al.: Qwen-image technical report. arXiv preprint arXiv:2508.02324 (2025)
57. Wu, C., Zheng, P., Yan, R., Xiao, S., Luo, X., Wang, Y., Li, W., Jiang, X., Liu, Y., Zhou, J., et al.: Omnigen2: Exploration to advanced multimodal generation. arXiv preprint arXiv:2506.18871 (2025)
58. Xiao, S., Wang, Y., Zhou, J., Yuan, H., Xing, X., Yan, R., Li, C., Wang, S., Huang, T., Liu, Z.: Omnigen: Unified image generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13294–13304 (2025)
59. Xie, S., Zhang, Z., Lin, Z., Hinz, T., Zhang, K.: Smartbrush: Text and shape guided object inpainting with diffusion model. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22428–22437 (2023)
60. Ye, H., Zhang, J., Liu, S., Han, X., Yang, W.: Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. arXiv preprint arXiv:2308.06721 (2023)
61. Ye, Y., He, X., Li, Z., Yuan, S., Yan, Z., Hou, B., Yuan, L., et al.: Imgedit: A unified image editing dataset and benchmark. *Advances in Neural Information Processing Systems* **38** (2026)
62. Yu, Q., Chow, W., Yue, Z., Pan, K., Wu, Y., Wan, X., Li, J., Tang, S., Zhang, H., Zhuang, Y.: Anyedit: Mastering unified high-quality image editing for any idea. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26125–26135 (2025)
63. Zhang, K., Mo, L., Chen, W., Sun, H., Su, Y.: Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems* **36**, 31428–31449 (2023)
64. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3836–3847 (2023)

65. Zhang, Z., Xie, J., Lu, Y., Yang, Z., Tang, Y.: In-context edit: Enabling instructional image editing with in-context generation in large scale diffusion transformer. arxiv 2025. arXiv preprint arXiv:2504.20690
66. Zhao, H., Ma, X., Chen, L., Si, S., Wu, R., An, K., Yu, P., Zhang, M., Li, Q., Chang, B.: Ultraedit: Instruction-based fine-grained image editing at scale. *Advances in Neural Information Processing Systems* **37**, 3058–3093 (2024)
67. Zhao, S., Chen, D., Chen, Y.C., Bao, J., Hao, S., Yuan, L., Wong, K.Y.K.: Uni-controlnet: All-in-one control to text-to-image diffusion models. *Advances in neural information processing systems* **36**, 11127–11150 (2023)
68. Zheng, K., Chen, H., Ye, H., Wang, H., Zhang, Q., Jiang, K., Su, H., Ermon, S., Zhu, J., Liu, M.Y.: Diffusionnft: Online diffusion reinforcement with forward process. arXiv preprint arXiv:2509.16117 (2025)
69. Zhipu AI Team: Glm-image: Auto-regressive for dense-knowledge and high-fidelity image generation. <https://z.ai/blog/glm-image> (2026), accessed: 2026-01-14
70. Zhu, J., Wang, W., Chen, Z., Liu, Z., Ye, S., Gu, L., Tian, H., Duan, Y., Su, W., Shao, J., et al.: Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. arXiv preprint arXiv:2504.10479 (2025)
71. Zong, Z., Song, G., Liu, Y.: Detsr with collaborative hybrid assignments training. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 6748–6758 (2023)