GPT-IMAGE-EDIT-1.5M: A MILLION-SCALE EDITING DATASET AND WHAT ACTUALLY WORKS TO TRAIN STRONG EDITORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advancements in proprietary multimodal models such as GPT-Image-1 have set new standards for high fidelity, instruction guided image editing. However, their closed-source nature restricts open research and reproducibility. To bridge this gap, we introduce GPT-IMAGE-EDIT-1.5M, a publicly available dataset comprising over 1.5 million high-quality editing triplets systematically unified from OmniEdit, HQEdit, and UltraEdit. Our data curation pipeline leverages *output regeneration* and *instruction rewriting* to significantly enhance instruction following (IF) and perceptual quality (PQ), while intentionally preserving challenges in identity preservation (IP) typical of GPT-generated images. We benchmark three MMDiT diffusion architectures—SD3 InstructPix2Pix (channel-wise conditioning), Flux with SigLIP (token-wise conditioning), and FluxKontext (token-wise conditioning) to analyze their robustness against IP degradation. Our results indicate that token-wise conditioning methods consistently outperform channel-wise conditioning. To ensure evaluation transparency, we specify when results involve thinking-rewritten prompts to avoid potential ambiguity. Moreover, we examine text encoders within a common frozen-encoder scenario, demonstrating that T5 embeddings consistently meet or exceed multimodal large language model (MLLM) embeddings, particularly with lengthier prompts. Simple linear or query-based integration methods, however, offer limited improvements, indicating deeper cross-modal fusion methods may be necessary. Fine-tuning FluxKontext on GPT-IMAGE-EDIT-1.5M achieves open-source performance competitive with GPT-Image-1 (**7.66**@GEdit-EN and **3.90**@ImgEdit-Full, with thinking-rewritten prompts; **8.97**@Complex-Edit). Our findings highlight critical interactions among instruction complexity, semantic alignment, and identity preservation, informing future directions in open-source image editing.

1 INTRODUCTION

Instruction-guided image editing is a fundamental task for generative AI, spurring significant progress in diffusion-based models such as InstructPix2Pix (Brooks et al., 2023), Prompt-to-Prompt (Hertz et al., 2022), SDEdit (Meng et al., 2021), and Imagic (Kawar et al., 2023). Proprietary models, notably GPT-Image-1 (Hurst et al., 2024), currently set the highest standards in instruction-following (IF) and perceptual quality (PQ). However, their closed-source nature severely restricts open research and reproducibility, creating a persistent gap between proprietary and open-source methods (Shi et al., 2024; Wang et al., 2025b).

A critical obstacle for open-source methods is the lack of large-scale, diverse, and well-aligned datasets. Existing datasets such as OmniEdit (Wei et al., 2025a), HQEdit (Hui et al., 2025), and UltraEdit (Zhao et al., 2024) frequently provide overly simplistic instructions or suffer from weak alignment between instructions and images. Consequently, open-source models trained on these datasets typically fail to achieve performance comparable to proprietary solutions.

To overcome these limitations, we introduce GPT-IMAGE-EDIT-1.5M, a unified dataset comprising over 1.5 million high-quality editing triplets (Fig. 1). Our streamlined pipeline leverages GPT-Image-1 to significantly enhance IF and PQ through *output regeneration* and *instruction rewriting*.

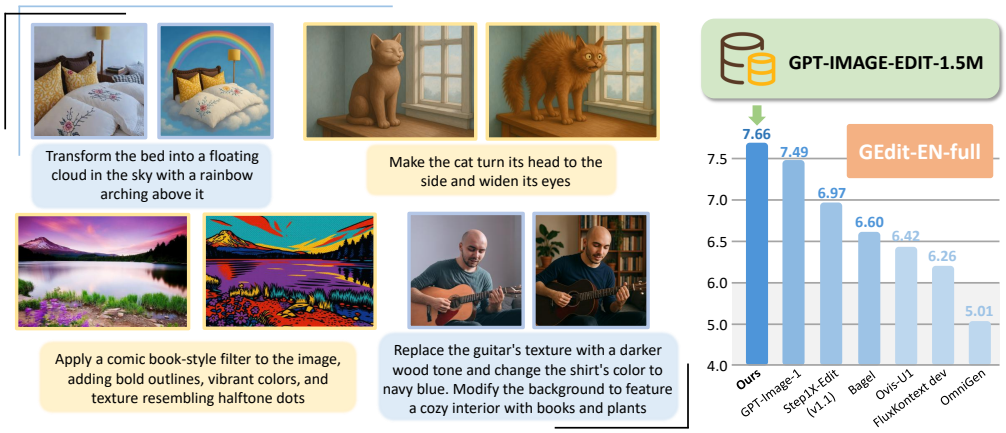


Figure 1: An overview of the GPT-IMAGE-EDIT-1.5M dataset. The figure presents qualitative examples showcasing diverse and complex instruction-guided edits. The bar chart demonstrates that a model fine-tuned on GPT-IMAGE-EDIT-1.5M achieves 7.66 on the GEdit-EN-full benchmark, surpassing existing open-source methods.

Unlike previous works that filter out challenging identity preservation (IP) cases, we deliberately retain them to reflect realistic complexities found in GPT-generated data (Labs et al., 2025; Chen et al., 2025b). This choice is motivated by recent findings that models trained solely on simplified data fail to generalize to practical, real-world editing tasks effectively.

Considering the inherent IP challenges, we systematically evaluate three diffusion architectures built upon MMDiT (Esser et al., 2024): SD3 InstructPix2Pix (Zhao et al., 2024) with channel-wise conditioning, and Flux with SigLIP (Lin et al., 2025) and FluxKontext (Labs et al., 2025), both employing token-wise conditioning. Our analyses consistently indicate that token-wise conditioning notably surpasses channel-wise methods across all evaluated metrics. This finding supports the idea that finer-grained token-level conditioning can more effectively manage semantic nuances and spatial alignment, essential for accurate instruction-guided edits.

Additionally, we examine text encoder strategies under a common practical constraint: frozen encoder parameters during fine-tuning. We observe that robust text-only encoders, such as T5, consistently match or exceed multimodal large language model (MLLM) embeddings, particularly with detailed, lengthy prompts. Furthermore, shallow integration methods like linear projections (Lin et al., 2025; Liu et al., 2025) or query-based connectors (Pan et al., 2025; Wei et al., 2025b) offer limited improvements, underscoring the need for deeper and more sophisticated cross-modal fusion methods (Tang et al., 2025; Deng et al., 2025; Xie et al., 2025).

Fine-tuning FluxKontext on GPT-IMAGE-EDIT-1.5M achieves open-source performance approaching proprietary GPT-Image-1, particularly when employing **7.66**@GEdit-EN, **3.90**@ImgEdit-Full with thinking-rewritten prompts, **8.97**@Complex-Edit. Rather than merely restating numerical results, our study provides nuanced insights into the relationships between instruction complexity, semantic alignment, and identity preservation, guiding future open-source advancements.

Contribution

- **Data:** We leverage GPT-Image-1 to build GPT-IMAGE-EDIT-1.5M, a unified dataset of over 1.5 million high-quality editing triplets, significantly enriching instruction diversity and alignment.
- **Conditioning Mechanism:** A systematic evaluation demonstrating the superiority of token-wise conditioning over channel-wise approaches for accurate and context-aware editing.
- **Text Encoder:** A detailed comparative analysis of text encoders under frozen encoder conditions, confirming the effectiveness of T5 embeddings and highlighting limitations in shallow MLLM integration methods.
- **Evaluation:** Comprehensive empirical evaluation across major benchmarks, clearly identifying strengths, weaknesses, and critical trade-offs necessary to advance open-source image editing.

2 Related Works

Instruction-Guided Image Editing. The task of instruction-guided image editing was established by pioneering works such as InstructPix2Pix (Brooks et al., 2023), Prompt-to-Prompt (Hertz et al., 2022), SDEdit (Meng et al., 2021), and Imagic (Kawar et al., 2023). InstructPix2Pix introduced a scalable, two-step approach: first leveraging GPT-3 (Brown et al., 2020) to generate synthetic instruction-image triplets, then using a diffusion model guided by Prompt-to-Prompt control (Hertz et al., 2022) to produce the corresponding image edits. Despite its foundational impact, the performance of these early models was constrained by the underlying diffusion architectures (U-Net-based latent diffusion models trained with CLIP (Radford et al., 2021)), limiting their photorealism and semantic precision (Rombach et al., 2022). This motivated subsequent research to pursue improvements in both dataset quality and architectural capability.

Data-Centric Advancements. Recognizing the critical role of data quality, recent approaches have prioritized sophisticated dataset curation. For instance, HQEdit (Hui et al., 2025) utilizes powerful proprietary models such as GPT-4V (Hurst et al., 2024) and DALL-E 3 (OpenAI, 2023) to generate more aligned and high-quality editing pairs. Concurrently, ShareGPT-4o-Image (Chen et al., 2025a) demonstrates effective direct distillation from proprietary models, creating high-quality datasets explicitly designed to transfer advanced editing capabilities to smaller, open-source models. Aligning with this strategy, our work systematically leverages GPT-Image-1 to refine and unify large-scale datasets, significantly enhancing data alignment and diversity without complex design.

Architectural Evolution: Diffusion and Flow Matching. Generative model architectures have evolved considerably, transitioning from U-Net-based diffusion models (Rombach et al., 2022) to more scalable Transformer-based Diffusion Transformers (DiT) (Peebles & Xie, 2023). More recently, flow matching (FM) models (Lipman et al., 2022) have emerged as efficient alternatives, predicting continuous velocity fields to directly model complex distributions. Specifically, FLUX.1 Kontext (Labs et al., 2025) exemplifies a state-of-the-art FM-based architecture, efficiently unifying generation and editing through token-wise conditioning, demonstrating robust semantic and perceptual capabilities. We leverage this architecture due to its proven effectiveness and efficiency, particularly suited to instruction-guided editing tasks.

Semantic Enhancement via Token-Wise Conditioning. An essential improvement in multimodal generative models has been the advancement in conditioning strategies—particularly token-wise versus channel-wise integration. State-of-the-art open-source models such as Step1X-Edit (Liu et al., 2025) and UniWorld-V1 (Lin et al., 2025) leverage token-wise conditioning schemes: Step1X-Edit utilizes Kontext-based token fusion, while UniWorld-V1 employs SigLIP-based token-wise integration, each conditioned on powerful multimodal large language models (MLLMs) like Qwen-VL (Bai et al., 2025) or LLaVA (Liu et al., 2023). These approaches significantly enhance semantic alignment and editing precision compared to earlier channel-wise methods. Our systematic exploration of these paradigms demonstrates clear advantages of token-wise conditioning in robustness and semantic fidelity, especially under realistic identity preservation (IP) challenges.

Evaluation Benchmarks. We evaluate on comprehensive benchmarks capturing diverse editing scenarios: *GEdit-Bench-EN (Full)* covers 11 distinct editing tasks with MLLM-based scoring (Liu et al., 2025); *ImgEdit (Full)* assesses across 9 task families using a unified pipeline (Ye et al., 2025); *Complex-Edit* evaluates compositional reasoning through chained edits (Yang et al., 2025); These benchmarks ensure rigorous evaluation across multiple dimensions (IF, IP, PQ), guiding the reliable assessment and comparison of editing model architectures.

3 Data & Method

Our primary goal is to construct a large-scale, high-quality dataset to facilitate robust open-source instruction-guided image editing. To this end, we introduce a unified, minimalist pipeline for data curation as shown in Fig. 2, designed to produce well-aligned instruction-image pairs while intentionally preserving challenging identity preservation (IP) scenarios typical in GPT-generated content. Given the IP challenges inherent to our dataset, we further investigate how different conditioning mechanisms and text encoder choices within MMDiT architectures influence editing quality and

Figure 2: An overview of GPT-IMAGE-EDIT-1.5M data curation pipeline. We applied multiple methods to collect high-quality image-editing data. we re-write 10% of OmniEdit instructions to better match regenerated images, and the input images originally generated by DALL-E in HQEdit were re-synthesized by GPT-Image-1 for higher alignment.

robustness. Below, we first describe our dataset curation process in detail, followed by an exploration of these key architectural decisions.

3.1 UNIFIED DATA CURATION AND EVALUATION PIPELINE

Our dataset curation process strategically integrates multiple methods to enhance the alignment, complexity, and quality of instruction-guided image editing data. We employ GPT-Image-1 to regenerate output images from existing instruction-image pairs, substantially improving visual fidelity and instruction-following accuracy. To address potential semantic drift from regeneration, we further utilize GPT-4o to selectively rewrite approximately 10% of OmniEdit instructions, ensuring that textual prompts precisely reflect the visual modifications in regenerated images.

Additionally, we introduce composite editing instructions of moderate complexity (three-step atomic edits, C3-level) (Yang et al., 2025) to approximately half of the OmniEdit dataset, enriching the dataset's realism and complexity. To upgrade the input quality of the HQ-Edit dataset, originally synthesized by DALL-E 3, we regenerate all inputs using GPT-Image-1, thereby enhancing visual quality and ensuring stronger alignment between instructions and images.

To maintain dataset consistency across varying aspect ratios, we implement a robust pad-and-crop procedure that standardizes images to three fixed ratios (1:1, 3:2, 2:3) without distortion. Post-generation, images are precisely cropped, with automated filtering mechanisms eliminating images containing artifacts or residual padding. Further details are presented in the Appendix B.

During evaluation, recognizing ambiguity in conventional short-text instructions, we employ GPT-5 at inference time to systematically rewrite raw benchmark prompts into clearly structured instructions. This rewriting step clarifies the intended edits by explicitly defining input conditions, desired edits, and expected outputs, while preserving the original images and evaluation scoring systems, thereby maintaining transparency and evaluation integrity. Comprehensive procedural details and examples are included in the Appendix C.

3.2 CONDITIONING PARADIGMS: CHANNEL-WISE VS. TOKEN-WISE

An MMDiT-based image editing model typically conditions the generation process on both image and text inputs. We explore three distinct conditioning paradigms within the broader MMDiT architecture family (Fig. 3), each varying by conditioning granularity, token fusion strategy, and computational complexity:

SD3 InstructPix2Pix (Channel-wise Conditioning). This method concatenates conditioning information directly along the channel dimension at input embedding layers, effectively increasing the embedding dimensionality. Post-processing within MMDiT embedding layers subsequently compresses these concatenated channels back to the original token dimension. While straightforward,

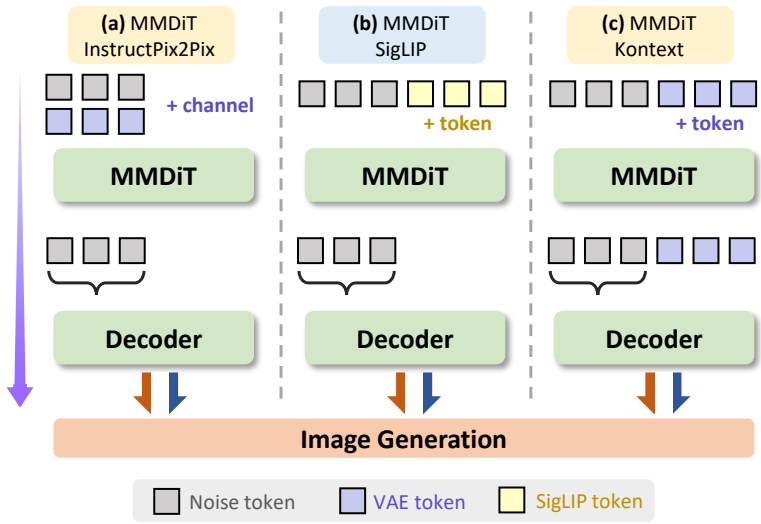


Figure 3: Conditioning paradigms comparison within the MMDiT architecture. (a) Channel-wise conditioning as in SD3 InstructPix2Pix concatenates conditioning information along input channels, subsequently compressing dimensionality within the model. (b) Flux with SigLIP employs token-wise merging via SigLIP visual features into textual embedding space, maintaining simplicity and strong semantic alignment. (c) FluxKontext leverages a robust dual-stream token-wise conditioning method, embedding visual and noise prediction tokens separately for enhanced precision and identity preservation, albeit at higher computational cost.

this approach may suffer from redundancy and higher complexity in handling channel-wise concatenation, potentially limiting its robustness to spatial misalignments and minor semantic discrepancies.

Flux with SigLIP (Token-wise Conditioning). In the Flux architecture, extracted SigLIP visual tokens are first projected into the textual embedding space, then merged token-wise with text embeddings via Flux's distinctive hybrid dual-to-single stream strategy. Unlike traditional dual-stream approaches, Flux merges visual and textual tokens within its single-stream stage. Consequently, only noise prediction tokens are active in the final decoding layers, significantly simplifying the conditioning mechanism and promoting more robust semantic alignment and identity preservation.

FluxKontext (Token-wise Conditioning). FluxKontext adopts a comprehensive dual-stream conditioning framework, embedding noise prediction tokens alongside latent visual tokens (derived from a VAE) as separate image branches. These branches remain distinct yet are processed in parallel by MMDiT layers, effectively doubling computational demands compared to single-stream approaches. Despite increased complexity, FluxKontext consistently achieves strong performance across multiple open-source benchmarks, reflecting its precise and robust conditioning capabilities.

3.3 Text Encoders and Fusion Strategies

Frozen-encoder setting. In all our experiments, we keep the text encoders (T5 (Raffel et al., 2020) and Qwen2.5-VL-7B Bai et al. (2025)) frozen. Both encoder-decoder (T5) and decoder-only (MLLM) models are used purely as *encoders*, performing a single forward pass without autoregressive decoding. We only fine-tune the lightweight projection and fusion layers, ensuring fair comparison.

Let $E_{\mathrm{t5}} \in \mathbb{R}^{L_{\mathrm{t5}} \times d}$ denote embeddings from T5, and $E_{\mathrm{mllm}} \in \mathbb{R}^{L_{\mathrm{m}} \times d_{\mathrm{m}}}$ from Qwen2.5-VL-7B. We use linear layer $W \in \mathbb{R}^{d_{\mathrm{m}} \times d}$ to align MLLM features to the dimension of T5.

T5-only (baseline). We directly feed E_{t5} into the editor as the instruction representation and fine-tune the editor exclusively using these embeddings. This approach serves as both our baseline and primary fine-tuning model, especially effective for handling complex instructions.

Table 1: Comparison on the GEdit-EN-full benchmark; (†): inference with thinking-rewritten prompts.

Model	BG Change	Color Alt.	Mat. Mod.	Motion	Portrait	Style	Add	Remove	Replace	Text	Tone	Avg
Open-Sourced Models												
AnyEdit (Yu et al., 2024)	4.31	4.25	2.64	0.67	1.90	1.95	3.72	3.75	3.23	0.77	4.21	2.85
MagicBrush (Zhang et al., 2023)	6.17	5.41	4.75	1.55	2.90	4.10	5.53	4.13	5.10	1.33	5.07	4.19
Instruct-Pix2Pix (Brooks et al., 2023)	3.94	5.40	3.52	1.27	2.62	4.39	3.07	1.50	3.48	1.13	5.10	3.22
OmniGen (Xiao et al., 2024)	5.23	5.93	5.44	3.12	3.17	4.88	6.33	6.35	5.34	4.31	4.96	5.01
Step1X-Edit (Liu et al., 2025)	7.03	6.26	6.46	3.66	5.23	7.24	7.17	6.42	7.39	7.40	6.62	6.44
Bagel (Deng et al., 2025)	7.44	6.99	6.26	5.09	4.82	6.04	7.94	7.37	7.31	7.16	6.17	6.60
Bagel-thinking (Deng et al., 2025)	7.22	7.24	6.69	7.12	6.03	6.17	7.93	7.44	7.45	3.61	6.36	6.66
Ovis-U1 (Wang et al., 2025a)	7.49	6.88	6.21	4.79	5.98	6.46	7.49	7.25	7.27	4.48	6.31	6.42
OmniGen2 (Wu et al., 2025b)	-	-	-	-	-	-	-	-	-	-	-	6.42
Step1X-Edit(v1.1) (Liu et al., 2025)	7.45	7.38	6.95	4.73	4.70	7.11	8.20	7.59	7.80	7.91	6.85	6.97
FluxKontext dev (Labs et al., 2025)	7.06	7.03	5.52	5.62	4.68	5.55	6.95	6.76	6.13	6.10	7.48	6.26
Qwen-Image (Wu et al., 2025a)	-	-	-	-	-	-	-	-	-	-	-	7.56
Proprietary Models												
Gemini	7.11	7.14	6.47	5.67	3.99	4.95	8.12	6.89	7.41	6.85	7.01	6.51
Doubao	8.07	7.36	7.20	5.38	6.28	7.20	8.05	7.71	7.87	4.01	7.67	6.98
GPT-Image-1	6.96	6.85	7.10	5.41	6.74	7.44	7.51	8.73	8.55	8.45	8.69	7.49
Ours	7.39	7.43	7.07	6.29	6.91	6.62	7.84	7.36	7.17	6.22	8.04	7.12
Ours†	7.87	8.02	7.02	7.73	7.53	7.05	8.56	7.78	8.42	6.21	8.02	7.66

MLLM projection. We encode the instruction once using Qwen2.5-VL-7B and project its embeddings to match T5 dimensions: $\hat{E}_{\text{mllm}} = E_{\text{mllm}} \times W \in \mathbb{R}^{L_{\text{m}} \times d}$. These projected tokens replace T5 embeddings to test the standalone encoding capability of MLLM.

MLLM projection + T5 concatenation. We concatenate T5 and projected MLLM tokens along the token dimension: $\tilde{E} = [E_{\text{t5}}; \hat{E}_{\text{mllm}}] \in \mathbb{R}^{(L_{\text{t5}}+L_{\text{m}}) \times d}$, and add a small learned type embedding to differentiate their sources. This evaluates if T5 and MLLM embeddings complement each other.

MLLM MetaQuery projection. Following the MetaQuery approach (Pan et al., 2025; Wei et al., 2025b), we append $N = 256$ special query tokens to the instruction and run a single forward pass through Qwen2.5-VL-7B. We retain only the embeddings corresponding to these query tokens, project them to dimension d via W, and use the resulting compact representation $E_{\text{mq}} \in \mathbb{R}^{N \times d}$ for conditioning. This approach summarizes instructions into fixed-length embeddings independent of their original lengths.

4 Experiments

4.1 Experimental Setup

Models. Our primary model (referred to as *Ours*) is built upon the state-of-the-art *FluxKontext dev* (Labs et al., 2025), utilizing token-wise conditioning for enhanced semantic alignment and editing robustness. For comparative ablation studies, we evaluate two additional architectures: the SD3 InstructPix2Pix model, based on *SD3-Medium* (Esser et al., 2024), which employs channel-wise conditioning, and the Flux with SigLIP model, based on *Flux 1.0 dev* (Labs, 2024), leveraging token-wise control using SigLIP features (Zhai et al., 2023). Details shown in the Appendix E.

Benchmarks. We conduct comprehensive evaluations using multiple benchmarks designed to measure diverse editing capabilities. Specifically, we assess general editing performance using *GEdit-EN-full* (Liu et al., 2025) and *ImgEdit-Full* (Ye et al., 2025), and examine compositional understanding with the *Complex-Edit* benchmark (Yang et al., 2025). Full descriptions of these benchmarks are provided in the Appendix D.

4.2 Main Results

As demonstrated in Tables 1, 2, and 3, our model, trained on the GPT-Image-Edit-1.5M dataset, achieves competitive performance among open-source methods and is highly competitive with leading proprietary models such as GPT-Image-1.

Table 2: Comparison on the ImgEdit-Full benchmark; (†): inference with thinking-rewritten prompts.

Model	Add	Adjust	Extract	Replace	Remove	Background	Style	Hybrid	Action	Overall
MagicBrush (Zhang et al., 2024)	2.84	1.58	1.51	1.97	1.58	1.75	2.38	1.62	1.22	1.90
Instruct-Pix2Pix (Brooks et al., 2023)	2.45	1.83	1.44	2.01	1.50	1.44	3.55	1.20	1.46	1.88
AnyEdit (Yu et al., 2024)	3.18	2.95	1.88	2.47	2.23	2.24	2.85	1.56	2.65	2.45
UltraEdit (Zhao et al., 2024)	3.44	2.81	2.13	2.96	1.45	2.83	3.76	1.91	2.98	2.70
OmniGen (Xiao et al., 2024)	3.47	3.04	1.71	2.94	2.43	3.21	4.19	2.24	3.38	2.96
Step1X-Edit (Liu et al., 2025)	3.88	3.14	1.76	3.40	2.41	3.16	4.63	2.64	2.52	3.06
ICEdit (Zhang et al., 2025)	3.58	3.39	1.73	3.15	2.93	3.08	3.84	2.04	3.68	3.05
BAGEL (Deng et al., 2025)	3.56	3.31	1.70	3.30	2.62	3.24	4.49	2.38	4.17	3.20
UniWorld-V1 (Lin et al., 2025)	3.82	3.64	2.27	3.47	3.24	2.99	4.21	2.96	2.74	3.26
OmniGen2 (Wu et al., 2025b)	3.57	3.06	1.77	3.74	3.20	3.57	4.81	2.52	4.68	3.44
Ovis-U1 (Wang et al., 2025a)	4.13	3.62	2.98	4.45	4.06	4.22	4.69	3.45	4.61	4.00
FluxKontext dev (Labs et al., 2025)	3.76	3.45	2.15	3.98	2.94	3.78	4.38	2.96	4.26	3.52
Qwen-Image (Wu et al., 2025a)	4.38	4.16	3.43	4.66	4.14	4.38	4.81	3.82	4.69	4.27
GPT-Image-1	4.61	4.33	2.90	4.35	3.66	4.57	4.93	3.96	4.89	4.20
Ours	4.19	3.79	2.09	4.22	3.96	3.90	4.76	3.23	4.49	3.85
Ours†	4.07	3.77	2.75	4.32	4.04	3.92	4.79	3.23	4.23	3.90

Table 3: Comparison on the Complex-Edit benchmark.

Method	IF	IP	PQ	O
AnyEdit (Yu et al., 2024)	1.60	8.15	7.25	5.67
UltraEdit (Zhao et al., 2024)	6.56	5.93	7.29	6.59
OmniGen (Xiao et al., 2024)	6.25	6.42	7.54	6.74
FluxKontext dev (Labs et al., 2025)	8.56	8.39	8.51	8.49
Imagen3 (Baldridge et al., 2024)	7.56	6.55	7.67	7.26
SeedEdit (Shi et al., 2024)	8.49	6.91	8.74	8.04
GPT-Image-1	9.29	7.51	9.47	8.76
Ours	9.20	8.57	9.14	8.97

GEdit-EN-full. Our model achieves an average score of **7.12**, surpassing all open-source models except Qwen-Image (Wu et al., 2025a). Notably, our model comprises only 12 billion parameters, representing less than half the parameter size of Qwen-Image (20+7B). When applying *thinking-rewritten* prompts—a structured clarification of ambiguous instructions using GPT-5 at inference without altering image content or evaluation metrics (examples detailed in Appendix C)—the performance further improves significantly to **7.66**, matching proprietary methods like GPT-Image-1. The improvements under rewritten prompts are particularly prominent in categories like *Motion* (6.29 $\rightarrow$ 7.73, +1.44) and *Remove* (7.17 $\rightarrow$ 8.42, +1.25), highlighting the model's capacity for nuanced semantic understanding and precision.

ImgEdit-Full. On this benchmark, our method obtains an overall score of **3.85**, outperforming existing open-source methods except Qwen-Image, again despite having significantly fewer parameters. Enabling thinking-rewritten prompts further boosts performance to **3.90**, closely rivaling proprietary systems such as GPT-Image-1 (4.20). Performance gains with rewritten prompts are observed broadly across tasks including *Add*, *Replace*, *Remove*, and *Style*, underscoring the robustness of our dataset and token-wise conditioning strategy.

Complex-Edit (C8). On the challenging Complex-Edit benchmark, characterized by lengthy and detailed instructions without any additional rewriting, our model demonstrates strong overall performance at **8.97**, showing an impressive balance between Instruction Following (IF: 9.20), Identity Preservation (IP: 8.57), and Perceptual Quality (PQ: 9.14). This balanced performance significantly exceeds that of GPT-Image-1 in IP (+1.06), while closely matching it in IF and PQ metrics. The effectiveness of our conditioning strategy in preserving identity under complex, multi-step instructions is particularly evident here.

4.3 Ablation Studies

We conduct a series of ablation studies to dissect the sources of performance improvements, isolating the effects of our data curation strategies and model architecture choices.

Table 4: Complex-Edit metrics results on GEdit-EN and ImgEdit; (*): indicates using pretrained FluxKontext weights.

Conditioning Mechanism	GEdit-EN				ImgEdit			
	IP	IF	PQ	Overall	IP	IF	PQ	Overall
SD3 InstructPix2Pix	8.25	5.92	7.91	7.36	7.73	5.70	5.99	6.48
Ours(SD3 InstructPix2Pix)	5.58	7.12	7.42	6.71	6.33	8.01	7.91	7.42
Flux with SigLIP	5.46	5.76	7.70	6.30	6.49	6.30	8.69	7.16
Ours(Flux SigLIP)	7.40	6.08	8.94	7.47	7.73	6.95	9.20	7.96
FluxKontext*	9.38	7.77	8.19	8.45	9.14	7.79	8.00	8.31
Ours(FluxKontext)	8.97	8.28	8.41	8.56	8.99	8.52	8.48	8.66

Table 5: Text encoder ablation (Complex-Edit metrics); Top block: original prompts; bottom block: thinking-rewritten prompts (†).

Text Encoder	GEdit-EN				ImgEdit			
	IP	IF	PQ	Overall	IP	IF	PQ	Overall
Baseline	**9.38**	7.77	8.19	8.45	**9.14**	7.79	8.00	8.31
Qwen2.5-7B-VL-Instruct Metaquery	8.34	8.11	8.17	8.21	8.64	7.99	8.39	8.34
Qwen2.5-7B-VL-Instruct	8.99	6.63	**8.43**	8.02	9.02	7.50	**8.56**	8.36
Qwen2.5-7B-VL-Instruct+T5	8.98	**8.28**	8.42	**8.56**	8.97	8.37	8.53	8.62
T5	8.99	**8.28**	8.42	**8.56**	8.99	**8.52**	8.48	**8.66**
Baseline†	**9.37**	7.68	8.20	8.42	**9.29**	8.58	7.87	8.58
Qwen2.5-7B-VL-Instruct Metaquery†	8.32	8.10	8.13	8.18	8.76	8.75	8.41	8.64
Qwen2.5-7B-VL-Instruct†	8.85	7.54	**8.40**	8.26	9.08	8.88	8.58	**8.85**
MLLM+T5†	9.01	8.77	8.26	8.68	8.92	7.91	**8.60**	8.48
T5†	9.05	**8.93**	8.38	**8.79**	9.03	**9.01**	8.47	8.84

Conditioning Paradigm (Channel-wise vs. Token-wise) Our curated dataset intentionally preserves realistic identity preservation (IP) challenges and mild spatial misalignments commonly found in practical editing scenarios. These conditions can significantly penalize methods employing coarse conditioning fusion mechanisms, such as channel-wise concatenation.

As detailed in Table 4, the conditioning mechanism significantly impacts Identity Preservation (IP). When finetuning SD3 InstructPix2Pix (Brooks et al., 2023; Esser et al., 2024), which utilizes channel-wise conditioning, on our curated dataset, we observe a marked decrease in IP and overall editing quality (e.g., GEdit overall declines from 7.36 to 6.71). Conversely, methods employing token-wise conditioning, such as Flux with SigLIP (Lin et al., 2025) and FluxKontext (Labs et al., 2025), consistently gain across Identity Preservation (IP), Instruction Following (IF), and Perceptual Quality (PQ). Specifically, Flux with SigLIP's GEdit overall improves from 6.30 to 7.47, while FluxKontext rises from 8.45 to 8.56. These results indicate that token-wise fusion is robust to spatial misalignments and enhances semantic precision. Official benchmark scores shown in Appendix F.

Impact of Text Encoder Table 5 shows the impact of different text encoders under frozen-encoder settings. T5 (Raffel et al., 2020) consistently achieves superior or competitive performance across Identity Preservation (IP), Instruction Following (IF), and Perceptual Quality (PQ) compared to Qwen2.5-7B-VL-Instruct embeddings, achieving top or near-top overall scores (GEdit: 8.56, ImgEdit: 8.66). Direct replacement with Qwen2.5-7B-VL-Instruct notably reduces performance, particularly impacting IF (GEdit IF drops from 8.28 to 6.63). Similarly, compact MetaQuery-based embeddings yield lower overall performance. Concatenating Qwen and T5 embeddings offers minimal and inconsistent benefits, suggesting shallow fusion inadequately captures complementary enough signals.

Thinking-rewritten prompts significantly enhance T5 performance (GEdit: 8.56 to 8.79, ImgEdit: 8.66 to 8.84), highlighting T5's strength in handling detailed instructions. Detailed official metric results are in the Appendix F. Overall, T5 emerges as the most reliable text encoder under frozen conditions, balancing IF, IP, and PQ. Limited gains from shallow MLLM integrations highlight the need for deeper cross-modal fusion strategies to fully leverage multimodal embeddings.

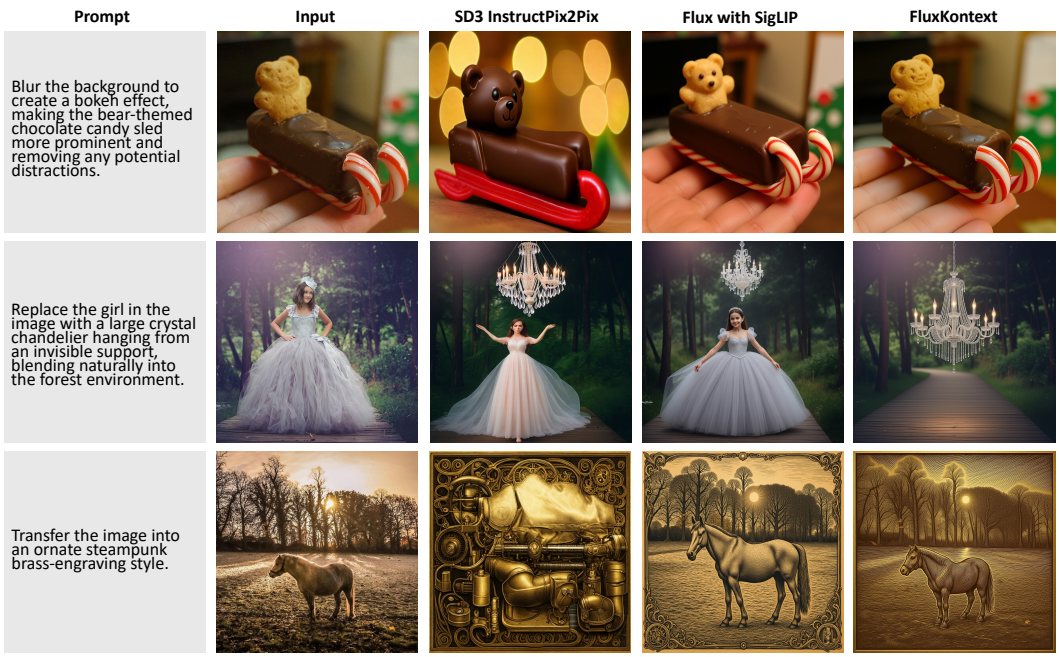


Figure 4: Qualitative comparison of editing performance across models.

4.4 Qualitative Results

Fig. 4 presents qualitative editing examples produced by our FluxKontext model fine-tuned on GPT-Image-Edit-1.5M across various editing scenarios. These results clearly illustrate the model's strong capability to interpret and follow editing instructions, generating realistic outputs while effectively preserving non-target image content. Additional qualitative examples across diverse categories are provided in Appendix G.

5 Limitation

Benchmarks partially rely on MLLM-based scoring, which can be sensitive to variations in style and phrasing. Therefore, we present both original and thinking-rewritten prompt scores side-by-side to maintain transparency. Although thinking-rewritten prompts improve semantic alignment, text rendering and fine-grained facial identity preservation (IP) continue to pose significant challenges. Additionally, our experiments under a frozen-encoder setup reveal that shallow fusion approaches are insufficient, indicating a clear need for deeper cross-modal fusion techniques in future research.

6 Conclusion

In this work, we introduced GPT-Image-Edit-1.5M, a unified dataset of over 1.5 million instruction-based editing samples systematically refined from existing sources using GPT-Image-1. Our approach significantly enhances instruction-following and perceptual quality while intentionally preserving realistic identity-preservation (IP) challenges. Experiments confirmed the effectiveness of our dataset and conditioning strategies, highlighting the superiority of token-wise conditioning and the robustness of T5 embeddings under frozen-encoder conditions. By releasing GPT-Image-Edit-1.5M and corresponding models, we provide a valuable resource to accelerate future research. Future work includes exploring datasets with improved IP quality (e.g., Nano-Banana) and deeper multimodal fusion between MLLMs and MMDiT architectures.

7 Ethics Statement

All authors of this work have read and commit to adhering to the ICLR Code of Ethics.

8 Reproducibility

To ensure reproducibility, we provide full code in the *Supplementary Material*.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Jason Baldridge, Jakob Bauer, Mukul Bhutani, Nicole Brichtova, Andrew Bunner, Lluis Castrejon, Kelvin Chan, Yichang Chen, Sander Dieleman, Yuqing Du, et al. Imagen 3. *arXiv preprint arXiv:2408.07009*, 2024.

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*, pp. 18392–18402, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Junying Chen, Zhenyang Cai, Pengcheng Chen, Shunian Chen, Ke Ji, Xidong Wang, Yunjin Yang, and Benyou Wang. Sharegpt-4o-image: Aligning multimodal models with gpt-4o-level image generation. *arXiv preprint arXiv:2506.18095*, 2025a.

Tianyu Chen, Yasi Zhang, Zhi Zhang, Peiyu Yu, Shu Wang, Zhendong Wang, Kevin Lin, Xiaofei Wang, Zhengyuan Yang, Linjie Li, et al. Edival-agent: An object-centric framework for automated, scalable, fine-grained evaluation of multi-turn editing. *arXiv preprint arXiv:2509.13399*, 2025b.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Mude Hui, Siwei Yang, Bingchen Zhao, Yichun Shi, Heng Wang, Peng Wang, Cihang Xie, and Yuyin Zhou. HQ-edit: A high-quality dataset for instruction-based image editing. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=mZptYYttFj.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

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*, pp. 6007–6017, 2023.

Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. Uniworld: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1x-edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025.

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. *arXiv preprint arXiv:2108.01073*, 2021.

OpenAI. GPT-4v System Card. https://openai.com/research/dall-e-3-system-card, 2023.

Xichen Pan, Satya Narayan Shukla, Aashu Singh, Zhuokai Zhao, Shlok Kumar Mishra, Jialiang Wang, Zhiyang Xu, Jiuhai Chen, Kunpeng Li, Felix Juefei-Xu, et al. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

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*, pp. 10684–10695, 2022.

Yichun Shi, Peng Wang, and Weilin Huang. Seededit: Align image re-generation to image editing. *arXiv preprint arXiv:2411.06686*, 2024.

Bingda Tang, Boyang Zheng, Sayak Paul, and Saining Xie. Exploring the deep fusion of large language models and diffusion transformers for text-to-image synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28586–28595, 2025.

Guo-Hua Wang, Shanshan Zhao, Xinjie Zhang, Liangfu Cao, Pengxin Zhan, Lunhao Duan, Shiyin Lu, Minghao Fu, Xiaohao Chen, Jianshan Zhao, et al. Ovis-u1 technical report. *arXiv preprint arXiv:2506.23044*, 2025a.

Peng Wang, Yichun Shi, Xiaochen Lian, Zhonghua Zhai, Xin Xia, Xuefeng Xiao, Weilin Huang, and Jianchao Yang. Seededit 3.0: Fast and high-quality generative image editing. *arXiv preprint arXiv:2506.05083*, 2025b.

Cong Wei, Zheyang Xiong, Weiming Ren, Xeron Du, Ge Zhang, and Wenhu Chen. Omniedit: Building image editing generalist models through specialist supervision. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=Hlm0cga0sv.

Hongyang Wei, Baixin Xu, Hongbo Liu, Cyrus Wu, Jie Liu, Yi Peng, Peiyu Wang, Zexiang Liu, Jingwen He, Yidan Xietian, et al. Skywork unipic 2.0: Building kontext model with online rl for unified multimodal model. *arXiv preprint arXiv:2509.04548*, 2025b.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025b.

Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. *arXiv preprint arXiv:2409.11340*, 2024.

Ji Xie, Trevor Darrell, Luke Zettlemoyer, and XuDong Wang. Reconstruction alignment improves unified multimodal models. *arXiv preprint arXiv:2509.07295*, 2025.

Siwei Yang, Mude Hui, Bingchen Zhao, Yuyin Zhou, Nataniel Ruiz, and Cihang Xie. Complex-edit: Cot-like instruction generation for complexity-controllable image editing benchmark. *arXiv preprint arXiv:2504.13143*, 2025.

Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. *arXiv preprint arXiv:2505.20275*, 2025.

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. *arXiv preprint arXiv:2411.15738*, 2024.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 11975–11986, 2023.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36:31428–31449, 2023.

Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Huan Wang, Silvio Savarese, Stefano Ermon, et al. Hive: Harnessing human feedback for instructional visual editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9026–9036, 2024.

Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. In-context edit: Enabling instructional image editing with in-context generation in large scale diffusion transformer, 2025. URL https://arxiv.org/abs/2504.20690.

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 *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=9ZDdlgH6O8.

A THE USE OF LARGE LANGUAGE MODELS (LLM)

During manuscript preparation, we used OpenAI GPT-4.1 for minor language refinement and writing polish. Additionally, GPT-4.1 was utilized for evaluating benchmark results. The primary dataset was generated using GPT-Image-1, with a subset of instruction prompts rewritten by GPT-4o and GPT-5. These uses of LLMs are clearly described and marked throughout the manuscript to ensure transparency.

B DATASET-SPECIFIC PROCESSING DETAILS

Given the heterogeneity of aspect ratios across the original datasets and the restriction that our generative model supports only three predefined ratios (1:1, 3:2, and 2:3), we adopted a standardized padding and cropping approach. Rather than directly resizing, which could distort the original content, we applied padding to each input image to match the nearest supported aspect ratio, conducted the image generation, and subsequently cropped out the padding. This process preserved the original geometry and pixel density, ensuring consistency and comparability across all datasets.

B.1 ULTRAEDIT DATASET PROCESSING

The UltraEdit dataset originally provides images at a 512×512 resolution. To enhance visual fidelity and maintain benchmark compatibility, we regenerated these images at a higher resolution of 1024×1024. Afterward, bicubic interpolation was employed to downscale the regenerated images back to the original size of 512×512. This method retains high-frequency details that might otherwise be lost through direct generation at a lower resolution.

B.2 OMNIEDIT DATASET ENHANCEMENT

For OmniEdit, additional refinements were introduced to improve data quality. Following the standard padding and cropping procedure, we systematically regenerated the output images to enhance both their visual quality and their alignment with the associated textual instructions. Recognizing semantic inconsistencies, approximately 10% of the original textual prompts underwent careful manual rewriting to better reflect the corresponding images. Furthermore, we augmented a substantial portion of this dataset by introducing compositional edits involving multiple sequential editing instructions, thus significantly enriching the dataset's instructional complexity.

B.3 COMPLEX-EDIT SUBSET CONSTRUCTION

The Complex-Edit subset specifically emphasizes the dataset's compositional complexity and rigorously tests instruction-following capabilities. After applying the standard geometric preprocessing, we crafted multi-step editing instructions leveraging GPT-4o's generative capabilities. These detailed instructions, often involving two to three distinct editing operations, were subsequently used to guide GPT-Image-1 image generation. Post-generation, a rigorous filtering process eliminated any samples with detectable padding artifacts to ensure the dataset's integrity and quality.

B.4 HQEDIT DATASET: DUAL-SPLIT STRATEGY

The HQEdit dataset was strategically divided into two complementary subsets to maximize utility and diversity:

Edit Split: Existing input-instruction pairs underwent standard padding and cropping before image generation. This preserved the fidelity of original pairs while ensuring aspect ratio consistency.

Generate Split: For this subset, entirely new reference input images were synthesized directly from textual prompts. These generated inputs subsequently underwent editing based on the original textual instructions. To further diversify this split, aspect ratios were randomly selected from the three available presets (1:1, 3:2, 2:3), promoting variety within the generated data.

B.5 Quality Control and Filtering Mechanisms

To uphold dataset quality, strict filtering criteria were applied after the cropping step. Samples exhibiting residual padding exceeding 0.5% of the image boundary, indicative of processing inaccuracies, were automatically excluded. Additionally, any mismatch between recorded padding masks and actual cropped regions resulted in immediate rejection, thus effectively mitigating common preprocessing errors and maintaining consistent image quality standards.

B.6 Comprehensive Metadata Schema

Each dataset sample is accompanied by comprehensive JSON metadata, providing explicit documentation of data provenance and preprocessing details. This metadata encompasses source identification, original and adjusted dimensions, exact padding and cropping parameters, original and rewritten prompts (when applicable), and generation resolution specifics. Such detailed metadata supports reproducibility and offers extensive flexibility for downstream research and analysis.

C Evaluation-Time Prompt Rewriting with GPT-5

As described in Section 3.1, we implement an inference-time rewriting procedure leveraging GPT-5 to address ambiguity in short, underspecified evaluation instructions. The primary motivation is to convert vague prompts into explicit, actionable editing descriptions while preserving the original images and evaluation metrics. Specifically, GPT-5 rewrites each original benchmark prompt into a structured three-field specification: an `input` (briefly describing visible image content), an `edit` (clearly specifying the editing task), and an `output` (summarizing the final expected appearance). During evaluation, only the rewritten `edit` prompts are used as replacements for the original instructions, with results explicitly marked by a dagger (†) in tables.

This rewriting is guided by a detailed task prompt (*Visual Edit Prompt Rewriter*), instructing GPT-5 to generate concise, image-editing-oriented English outputs, strictly based on visible elements in reference images. To minimize subjective inference, the system is designed to adopt conservative assumptions wherever ambiguities remain unresolved.

Below, we illustrate two concrete examples demonstrating the clarity and specificity gained from this rewriting step:

Example A (Background Change—Fire Hydrant). *Original prompt:* "Change the background to a city street."

Rewritten prompt: "Replace the entire background beyond the fire hydrant with an urban city street featuring multi-story commercial buildings, asphalt roadway, and sidewalks, viewed from a sidewalk-level perspective; remove the trees, houses, and parked car; maintain the hydrant, soft overcast lighting, and shallow depth of field."

Example B (Background Change—Bear by River). *Original prompt:* "Change the background to a forest."

Rewritten prompt: "Replace the blurred riverside background beyond the water with a dense forest scene featuring green trees, trunks, and foliage; keep the bear and river foreground unchanged; match the soft daylight from the upper-left, preserve shallow depth-of-field, and blend the forest edges naturally around the bear's fur and water splashes."

This structured rewriting enhances clarity and reproducibility without altering evaluation fairness, as the original images and scoring remain unchanged. All rewritten prompts are made available for full transparency and reproducibility.

D Benchmarks and Evaluation Protocols

We comprehensively evaluate our approach using three established instruction-guided image editing benchmarks—*GEdit-EN (Full)*, *ImgEdit (Full)*, and *Complex-Edit*—selected for their complementary assessment of general editing quality, diverse editing operations, and compositional reasoning.

GEdit-EN (Full). The GEdit-EN (Full) benchmark encompasses 11 distinct editing categories: Background Change, Color Alteration, Material Modification, Motion, Portrait, Style, Add, Remove, Replace, Text, and Tone. This categorization provides broad coverage of common editing tasks and enables detailed, category-level analysis. Following the benchmark's standard evaluation protocol, we report scores for Semantic Consistency (SC), Perceptual Quality (PQ), and an aggregated Overall metric. Additionally, where explicitly noted with a dagger (†), results include minimal inference-time prompt rewriting to clarify ambiguous instructions, while strictly maintaining original images and official scoring procedures.

ImgEdit (Full). ImgEdit (Full) comprises nine task families—Add, Adjust, Extract, Replace, Remove, Background, Style, Hybrid, and Action—designed to evaluate model performance on a range of atomic, localized editing tasks. The official evaluation protocol is consistently followed, with results reported per task family as well as an aggregate Overall score. Similar to GEdit-EN, daggered (†) results indicate the use of inference-time prompt rewriting to resolve ambiguities without altering images or the official evaluation criteria.

Complex-Edit. The Complex-Edit benchmark specifically targets compositional editing through multi-step or constraint-rich instructions, emphasizing a model's ability to follow complex, structured prompts. Performance evaluation includes metrics for Instruction Following (IF), Identity Preservation (IP), Perceptual Quality (PQ), and an Overall aggregate score. Notably, evaluations on Complex-Edit strictly utilize the original, detailed instructions without any inference-time rewriting.

Reproducibility. To ensure full reproducibility, we adhere rigorously to official benchmark evaluation pipelines and release all evaluation scripts and configuration details.

E Implementation and Training Details

Model Variants. We evaluate three distinct multimodal diffusion transformer (MMDiT)-based editing paradigms: (i) **SD3 InstructPix2Pix**, employing channel-wise conditioning built upon *SD3-Medium*; (ii) **Flux w/ SigLIP**, leveraging token-wise conditioning with *Flux 1.0 dev* guided by SigLIP features; and (iii) our primary model **FluxKontext dev**, utilizing token-wise conditioning. Unless explicitly stated, all text encoders remain *frozen* throughout training. Fine-tuning involves updating only lightweight projection/fusion layers and the editor backbone during Stage 2. Benchmark protocols, data handling, and inference-time instruction rewriting are consistent with the main paper.

SD3 InstructPix2Pix (channel-wise conditioning). This model is trained following the original *UltraEdit* training recipe and hyperparameters (optimizer, learning rate schedule, regularization techniques) with the sole exception of utilizing our curated dataset. The training runs for **10 epochs**, thereby maintaining fidelity to the original SD3 configuration for a controlled comparative evaluation against token-wise architectures.

Flux w/ SigLIP (token-wise conditioning, two-stage training). We adopt a two-stage training protocol aligned with the *UniWorld* configuration:

- Stage 1: MLLM Connector Pretraining. We first pretrain a connector mapping Qwen2.5-VL embeddings into the SigLIP textual embedding space using the **Prodigy** optimizer for **100k steps**. Only the connector parameters are updated in this phase.
- Stage 2: Joint Connector and Flux Fine-tuning. We subsequently fine-tune both the connector and Flux using the **AdamW** optimizer at a learning rate of **1e-6** for **50k steps**, following the *UniWorld* fine-tuning strategy (data augmentation, batch packing, evaluation intervals), with the text encoder parameters remaining frozen.

FluxKontext dev (token-wise conditioning, two-stage training). For FluxKontext dev, we reuse the pretrained Stage 1 connector obtained from the Flux w/ SigLIP pipeline without further modification, and subsequently:

- Stage 2: Joint Connector and FluxKontext Fine-tuning. Both the reused connector and FluxKontext are jointly fine-tuned for **50k steps**, mirroring the Flux w/ SigLIP fine-tuning setup (AdamW optimizer, learning rate **1e-6**) to enable direct, controlled comparisons.

MetaQuery Connector (Qwen2.5-VL, two-stage training). For the MetaQuery variant, training follows a similar two-stage strategy:

- Stage 1: Connector Pretraining. We pretrain the MetaQuery connector, which compresses Qwen2.5-VL embeddings into $N = 256$ query tokens, summarizing and projecting them into the editor's embedding space.
- Stage 2: Joint Connector and FluxKontext Fine-tuning. The pretrained MetaQuery connector and FluxKontext are then jointly fine-tuned under identical conditions to FluxKontext Stage 2, isolating connector design effects (refer to Sec. 3.3 for tokenization specifics).

Hardware and Precision Settings. All experiments utilize **8×A100 (80GB)** GPUs in a distributed data parallel setup, employing mixed-precision training when feasible. Batch sizes and gradient accumulation steps are adjusted per architecture to fully utilize GPU memory capacity, ensuring comparable training throughput across all variants.

F Expanded Ablation Studies

Conditioning and official metrics. Table 7 groups official scores and shows that training on GPT-Image-Edit-1.5M consistently improves each backbone; gains are largest for token-wise models (FluxKontext, Flux+SigLIP), supporting the benefit of token-level fusion for real-world edits. **Text encoders.** With all encoders frozen, T5 is the most reliable choice overall (Table 8; detailed per-category trends in Tables 12–13). Qwen-VL alone underperforms on text-heavy instructions (e.g., the *Text* category), while concatenating Qwen-VL with T5 recovers most categories on GEdit-EN and remains competitive on ImgEdit. **Data curation.** On 100k-subset studies, regenerating outputs and then aligning instructions yields sizable, additive gains across both SD3 and Flux backbones (Table 9). **Complex-Edit inclusion.** Adding the Complex-Edit subset provides modest but consistent improvements on GEdit-EN (7.03 → 7.24) and ImgEdit (3.71 → 3.80) averages (Tables 10–11), mainly via *Motion/Hybrid/Action* categories, while *Text/Tone* remain challenging. **OmniContext.** Results on OmniContext *SINGLE* (Table 6) show balanced PF/SC and narrow the gap to proprietary systems, corroborating the qualitative trends in Fig. 7. Unless marked with †, all numbers use original benchmark prompts; † denotes inference-time "thinking-rewrites" that clarify under-specified prompts without altering images or scoring protocols.

G Additional Qualitative Results

We provide extended visualizations spanning four representative settings: *GEdit-EN* (Fig. 5), *ImgEdit* (Fig. 6), *OmniContext* (Fig. 7), and *Complex-Edit* (Fig. 8). Across categories, the model performs localized edits while preserving non-edited content, with improved semantic alignment for motion/background/style changes and multi-step compositions. Typical failure modes include minor facial drift and imperfect text replacement, mirroring the IP and text-handling challenges discussed in the main paper.

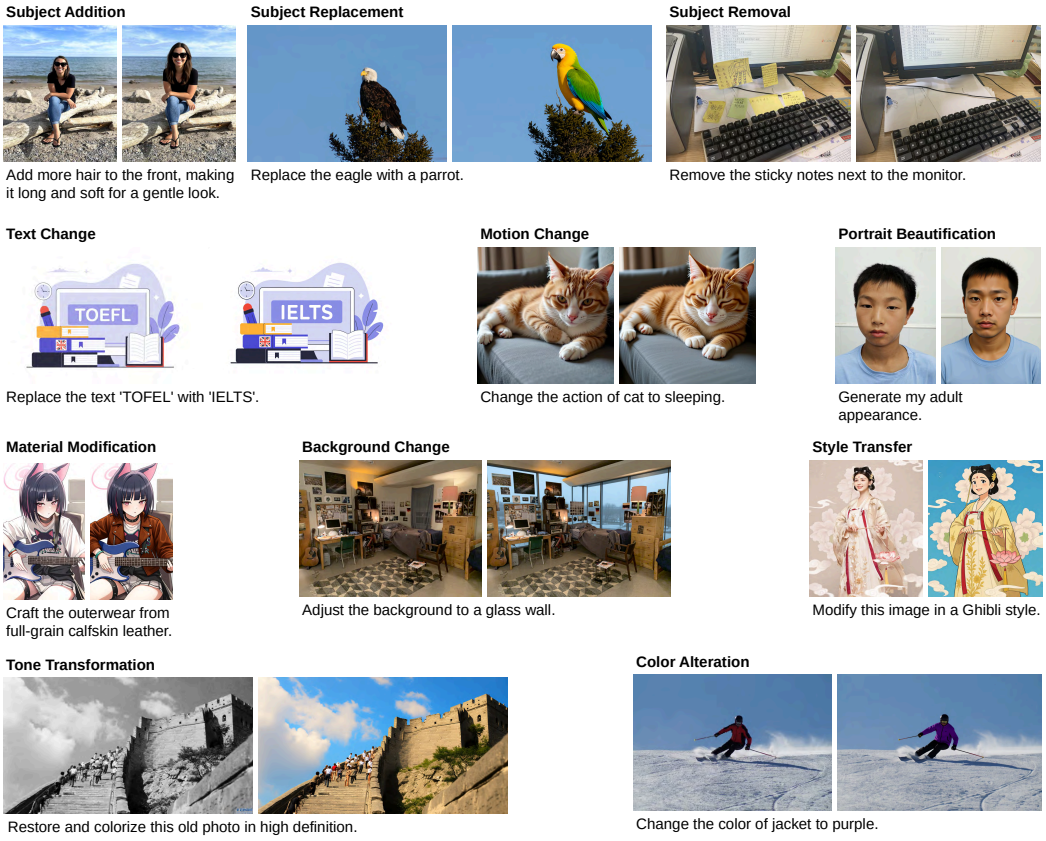


Figure 5: The qualitative results of our method on GEdit-Benchmark-EN.

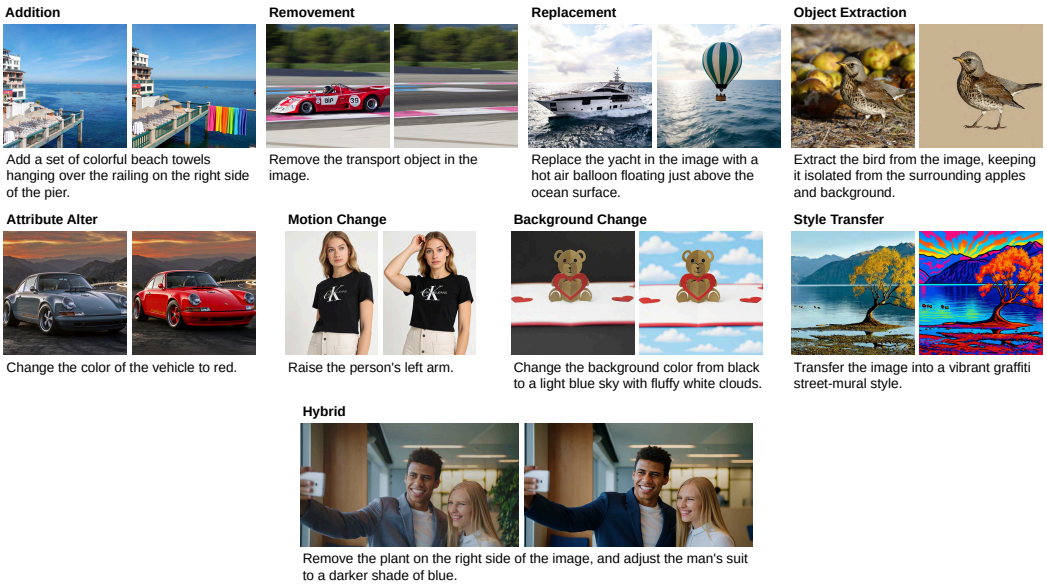


Figure 6: The qualitative results of our method on Img-Edit.

In a cozy studio, the person sits in a chair and speaks into a microphone.

A woman is playing with colorful beads and small cars on the red carpet.

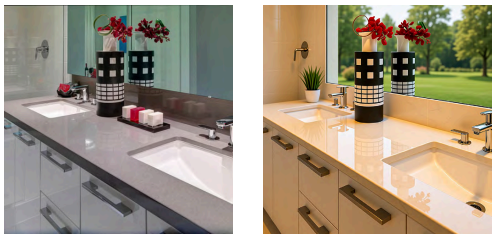
Display the intricate mandala artwork on the wall above the marble fireplace in the elegant living room, enhancing the serene atmosphere as the fire crackles softly.

Show the man in a suit embracing his partner in a lavender field, both smiling and holding bouquets.

Dance on the stage beneath the red curtain.

In the cozy corner of a rustic tropical restaurant, a Steller's sea eagle perches gracefully on a wicker chair, its sharp eyes scanning the warm ambiance filled with greenery and soft hanging lights.

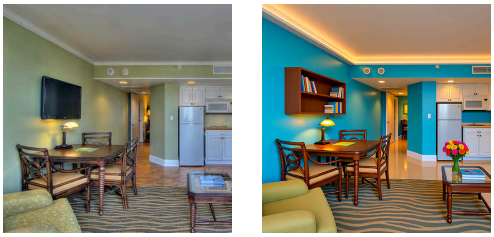
Figure 7: The qualitative results of our method on OmniContext.



Modify the mirror reflections to display a window with a garden view and increase the ambient lighting. Replace the black faucet with a stainless-steel one, and change the countertop texture to a white marble finish. Remove the soapdish near the right sink. Resize the ornamental vase to make it slimmer, and add a small potted plant next to the left sink. Apply a warm filter to the image.

Transform the scene into a moody, rainy vintage depiction by replacing the clear sky with a cloudy, stormy one, colorizing with muted vintage tones, and adding falling raindrops. Introduce a vintage car near the curb, apply a wet, reflective texture to the road, and dim the overall lighting while adding subtle fog at ground level. Conclude with a sepia-tone filter to enhance the vintage atmosphere.

Replace the wall with an artistic mural and change the wooden bunk bed frame to white. Add a hammock hanging above the bunk beds diagonally, and modify the bedding to have floral patterns. Remove the chair near the flowers. Illuminate the scene with warm golden light, add a glowing aura to the bouquet, and finish with a vintage-style filter.



Change the wall color to a vibrant sky blue and the floor to a polished marble texture. Replace the TV with a wooden shelf with books and move the coffee table slightly closer to the couch. Add a vase with colorful flowers on the dining table, increase the brightness of the lamp, and apply a warm sepia tone across the image. Introduce softly glowing light trails along the ceiling corners.

Figure 8: The qualitative results of our method on Complex-Edit.

Table 6: Results on the OmniContext *SINGLE* benchmark.

Method	Character			Object			Average		
	PF	SC	Overall	PF	SC	Overall	PF	SC	Overall
InfiniteYou	7.81	5.15	6.05	—	—	—	—	—	—
UNO	7.56	6.48	6.60	7.78	6.65	6.83	7.67	6.56	6.72
BAGEL	7.72	4.86	5.48	8.56	6.06	7.03	8.14	5.46	6.25
OmniGen	7.12	7.58	7.21	7.66	5.04	5.71	7.39	6.31	6.46
OmniGen2	8.04	8.34	8.05	8.44	7.26	7.58	8.24	7.80	7.81
Flux.1 Kontext (dev)	7.70	8.72	8.07	8.76	8.22	8.33	8.23	8.47	8.20
Flux.1 Kontext (max)	7.98	9.24	8.48	8.78	8.76	8.68	8.38	9.00	8.58
Gemini-2.0-Flash	5.54	5.98	5.06	6.17	5.89	5.17	5.86	5.93	5.11
GPT-Image-1	8.89	9.03	8.90	9.40	8.74	9.01	9.14	8.88	8.95
Ours	8.10	8.36	8.11	8.50	7.68	7.87	8.30	8.02	7.99

Table 7: Official benchmark metrics (GEdit-EN: SC/PQ/Overall; ImgEdit: Overall). Training on GPT-IMAGE-EDIT-1.5M yields consistent gains across all backbones; improvements are largest for token-wise models (FluxKontext, Flux+SigLIP), highlighting the advantage of token-level fusion for real-world edits.

Model Arch.	GEdit-EN			ImgEdit
	SC	PQ	Overall	Overall
SD3 InstructPix2Pix	4.34	6.14	3.92	2.54
Ours	4.96	6.46	4.91	3.32
Flux with SigLIP	4.48	5.51	4.75	3.00
Ours	5.57	8.00	5.81	3.49
FluxKontext*	6.98	7.20	6.26	3.52
Ours	7.63	7.69	7.12	3.85

Table 8: Text-encoder ablation under *official* metrics with all encoders frozen. T5 alone is strong on both GEdit-EN and ImgEdit; concatenating Qwen2.5-VL with T5 is competitive on GEdit-EN and close on ImgEdit. Compact MetaQuery features underperform. † indicates inference-time prompt rewriting.

Text Encoder	GEdit-EN			ImgEdit
	SC	PQ	Overall	Overall
Baseline	6.98	7.20	6.26	3.52
Qwen2.5-7B-VL-Instruct Metaquery	7.28	7.69	7.05	3.64
Qwen2.5-7B-VL-Instruct	6.08	7.84	5.89	3.60
Qwen2.5-7B-VL-Instruct+T5	7.91	7.52	7.24	3.80
T5	7.63	7.69	7.12	3.85
Baseline	7.01	7.15	6.28	3.64
Qwen2.5-7B-VL-Instruct Metaquery	7.40	7.58	7.06	3.69
Qwen2.5-7B-VL-Instruct	7.25	7.81	6.87	3.61
Qwen2.5-7B-VL-Instruct+T5	8.08	7.57	7.45	3.89
T5	8.23	7.75	7.66	3.90

Table 9: Ablations on data curation (100k-subset runs). Regenerating outputs first (`-gpt/-output-regen`) provides large gains; aligning instructions (`-gpt-rewrite/-pair-regen`) adds further improvements. Trends hold for both channel-wise (SD3 InstructPix2Pix) and token-wise (Flux with SigLIP) backbones, supporting the two-step refinement pipeline.

Method	Dataset Variant	ImgEdit	GEdit-EN
	OmniEdit Ablations		
SD3 InstructPix2Pix	omniedit100k-base	2.54	3.92
SD3 InstructPix2Pix	omniedit100k-gpt	3.13	4.91
SD3 InstructPix2Pix	omniedit100k-gpt-rewrite	3.32	4.89
Flux with SigLIP	omniedit100k-base	2.94	4.93
Flux with SigLIP	omniedit100k-gpt	3.24	5.98
Flux with SigLIP	omniedit100k-gpt-rewrite	3.40	5.88
	HQEdit Ablations		
SD3 InstructPix2Pix	hqedit100k-base	2.19	2.00
SD3 InstructPix2Pix	hqedit100k-output-regen	3.02	4.45
SD3 InstructPix2Pix	hqedit100k-pair-regen	3.08	4.75
Flux with SigLIP	hqedit100k-base	3.12	4.34
Flux with SigLIP	hqedit100k-output-regen	3.44	5.67
Flux with SigLIP	hqedit100k-pair-regen	3.45	5.73
	Complex-Edit Instruction Ablation		
Flux with SigLIP	Complex-Edit	2.89	5.39

Table 10: Effect of including the *Complex-Edit* subset on GEdit-EN (per-category). Inclusion yields a consistent average gain (+0.21), with the largest improvements in *Motion*, *Add*, and *Replace*; small trade-offs appear in *Tone* and *Text*, reflecting the challenge of long compositional instructions.

Dataset	BG Change	Color Alt.	Mat. Mod.	Motion	Portrait	Style	Add	Remove	Replace	Text	Tone	Avg
Fluxkontext mllm+T5 w/o complex	7.62	7.55	6.77	7.08	6.74	6.74	7.68	7.74	6.82	5.36	7.23	7.03
Fluxkontext mllm+T5 (full)	7.80	7.54	7.12	7.75	7.09	6.74	8.04	7.95	7.17	5.45	6.95	7.24

Table 11: Effect of including the *Complex-Edit* subset on ImgEdit (per-family). Overall improves from 3.71 to 3.80, driven by gains in *Hybrid* and *Action*, while other categories remain stable.

Dataset	Add	Adjust	Extract	Replace	Remove	Background	Style	Hybrid	Action	Overall
Fluxkontext mllm+T5 w/o complex	4.07	3.69	1.94	4.17	3.93	3.73	4.74	2.91	4.19	3.71
Fluxkontext mllm+T5 (full)	4.07	3.79	2.04	4.13	3.89	3.90	4.84	3.04	4.52	3.80

Table 12: Text-encoder configurations on GEdit-EN (per-category; encoders frozen). T5 is a reliable default; Qwen-VL alone weakens *Text* (1.20) and *Replace*, while concatenating Qwen-VL with T5 restores most categories and attains the best average (7.24).

Text Encoder	BG Change	Color Alt.	Mat. Mod.	Motion	Portrait	Style	Add	Remove	Replace	Text	Tone	Avg
FluxKontext dev (T5)	7.06	7.03	5.52	5.62	4.68	5.55	6.95	6.76	6.13	6.10	7.48	6.26
Finetuned with T5	7.39	7.43	7.07	6.29	6.91	6.62	7.84	7.36	7.17	6.22	8.04	7.12
Finetuned with QwenVL	6.45	7.27	5.04	6.53	7.26	5.88	7.03	7.20	4.31	1.20	6.64	5.89
QwenVL + T5 (Ours)	7.80	7.54	7.12	7.75	7.09	6.74	8.04	7.95	7.17	5.45	6.95	7.24

Table 13: Text-encoder configurations on ImgEdit (per-family; encoders frozen). T5 attains the best overall (3.85); Qwen-VL+T5 is close (3.80) and strongest on *Style* and *Action*; Qwen-VL alone lags on *Extract* and *Replace*.

Text Encoder	Add	Adjust	Extract	Replace	Remove	Background	Style	Hybrid	Action	Overall
FluxKontext dev (T5)	3.76	3.45	2.15	3.98	2.94	3.78	4.38	2.96	4.26	3.52
Finetuned with T5	4.19	3.79	2.09	4.22	3.96	3.90	4.76	3.23	4.49	3.85
Finetuned with QwenVL	3.92	3.58	1.95	3.62	3.89	3.72	4.64	3.22	3.82	3.60
QwenVL + T5 (Ours)	4.07	3.79	2.04	4.13	3.89	3.90	4.84	3.04	4.52	3.80