GEDITBENCH V2: A HUMAN-ALIGNED BENCHMARK FOR GENERAL IMAGE EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in image editing have enabled models to handle complex instructions with impressive realism. However, existing evaluation frameworks lag behind: current benchmarks suffer from narrow task coverage, while standard metrics fail to adequately capture *visual consistency*, i.e., the preservation of identity, structure and semantic coherence between edited and original images. To address these limitations, we introduce **GEditBench v2**, a comprehensive benchmark with 1,200 real-world user queries spanning 23 tasks, including a dedicated *open-set* category for unconstrained, out-of-distribution editing instructions beyond predefined tasks. Furthermore, we propose **PVC-Judge**, an open-source pairwise assessment model for visual consistency, trained with two novel region-decoupled preference data synthesis pipelines. Besides, we construct **VCReward-Bench** using expert-annotated preference pairs to assess the alignment of PVC-Judge with human judgments on visual consistency evaluation. Experiments show that our PVC-Judge achieves state-of-the-art evaluation performance among open-source models and even surpasses GPT-5.1 on average. Finally, by benchmarking 16 frontier editing models, we show that GEditBench v2 enables more human-aligned evaluation, revealing critical limitations of current models, and providing a reliable foundation for advancing precise image editing.

1 INTRODUCTION

Instruction-based image editing models (Labs et al., 2025; Liu et al., 2025; Wu et al., 2025a;b; Z.ai, 2026; Team et al., 2025a;b) have rapidly evolved to execute complex visual modifications directly from natural language instructions. Recently, Nano Banana Pro (Team et al., 2023) emerged as a landmark model, demonstrating robust generalization across diverse instructions while maintaining exercise fine control for high-fidelity results. The success of Nano Banana Pro has shifted the community's attention from coarse-grained **instruction following** toward a more nuanced understanding of instruction boundaries (Yin et al., 2026) – the delicate line between instruction following (identifying what must be changed) and what we define as **visual consistency** (the imperative to preserve non-target elements). For instance, when tasked with "*replacing a subject's cotton shirt with a silk one,*" an excellent editing model must precisely render the new texture and sheen while strictly preserving the subject's identity, the background illumination, and the spatial geometry of the surrounding environment. Consequently, such precise control over the editing process has become a key indicator of high-quality image editing models. However, existing evaluation protocols remain inadequate for assessing their visual consistency capability.

To bridge the aforementioned evaluation gap, recent studies have adopted the VLM-as-a-Judge paradigm to assess visual consistency (Ye et al., 2026b;a; Luo et al., 2026). In general, these approaches design prompt templates based on predefined consistency criteria and then query advanced Vision-Language Models (VLMs), such as GPT-4.1 (OpenAI, 2025), to assign an absolute rating score for each edited image. Although straightforward and easy to implement, this evaluation protocol suffers from three key limitations. First, it typically relies on closed-source APIs, making results difficult to reproduce and potentially unstable as the underlying models evolve. Second, replacing these systems with open-source alternatives introduces an accuracy-cost trade-off: smaller models (e.g., 4B/8B) often lack sufficient priors for reliable judgment, whereas larger models incur substantial deployment cost for inference. Third, the pointwise scoring scheme is poorly aligned with human judgment, which favors pairwise comparison over absolute rating as evidenced in Fig. 2. Furthermore,

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

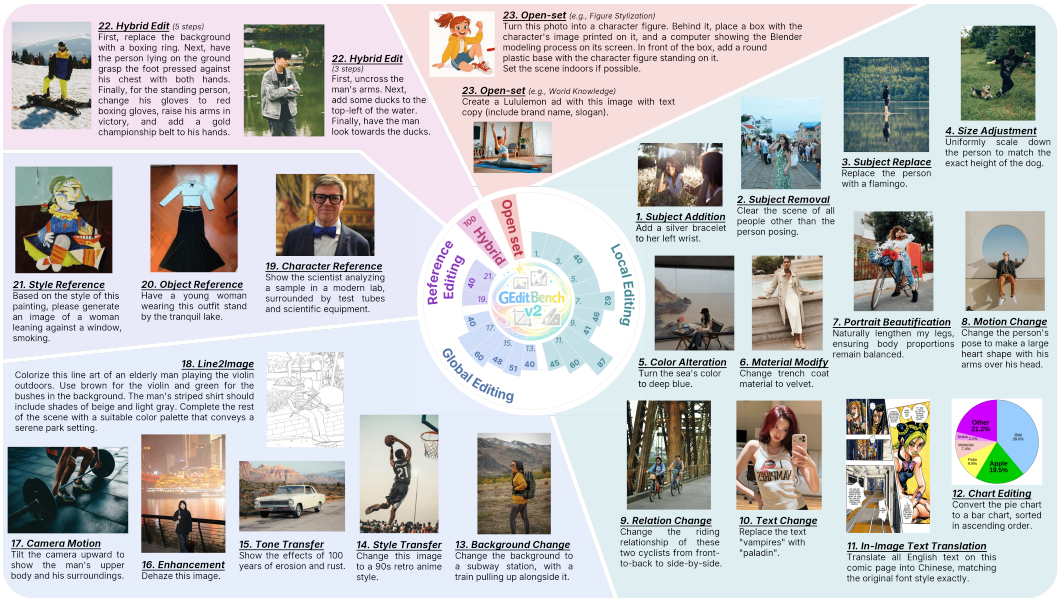


Figure 1: **GEditBench v2** spans 23 diverse image editing tasks, ranging from predefined edits to complex open-set real-world instructions, offering a comprehensive testbed for evaluating instruction-based image editing models. The central rose diagram visualizes the corresponding count distribution.

existing benchmarks (Labs et al., 2025; Yu et al., 2025; Ye et al., 2026b; Liu et al., 2025; Pan et al., 2025; Ye et al., 2026a) typically restrict task coverage to a closed set of predefined editing categories, limiting their ability to evaluate the generalization of editing models in open real-world scenarios.

In this work, we introduce a comprehensive evaluation protocol to address these issues in both model assessment and existing benchmarks. As shown in Fig. 1, we first propose **GEditBench v2**, with an *open-set* category that extends evaluation from standard edit tasks to out-of-distribution instructions, meeting the demands of real-world image editing. To reliably assess diverse edits and overcome the limitations of the pointwise scheme, we develop **PVC-Judge**, a human-aligned, **P**airwise assessment model dedicated to **V**isual **C**onsistency. To train the PVC-Judge, we design novel **object-** and **human-centric data curation pipelines** that robustly synthesize high-quality preference pairs at scale by decoupling edited from non-edited regions and ensembling traditional metrics. Furthermore, to validate the effectiveness of PVC-Judge, we introduce a meta-benchmark, **VCReward-Bench**, comprising 3,506 expert-annotated preference pairs across 21 predefined tasks, serving as a gold standard for quantifying models' human alignment in assessing visual consistency. Experimental results on VCReward-Bench show that PVC-Judge achieves the state-of-the-art performance for open-source assessment models, even outperforming GPT-5.1 with an average accuracy of 81.82 versus 76.89. In summary, our key contributions are as follows:

- We introduce GEditBench v2, a comprehensive benchmark comprising 22 predefined edit tasks and one dedicated open-set category to evaluate editing models in real-world scenarios.
- We propose PVC-Judge, a pairwise assessment model for visual consistency, trained with two novel region-decoupled preference synthesis pipelines, enabling human-aligned evaluation.
- We develop VCReward-Bench to evaluate assessment models for instruction-guided image editing in visual consistency, supported by 3,506 expert-annotated preference pairs.

2 Related Work

Image Editing Models. The field of instruction-based image editing has rapidly evolved from modular, text-guided pipelines (Li et al., 2026b; Wang et al., 2023) to unified, free-form generative architectures (Zhang et al., 2023; Zhao et al., 2024; Yu et al., 2025). Early models like InstructPix2Pix (Brooks et al., 2023) demonstrated the feasibility of diffusion-based editing with synthetic supervision, yet struggled with complex reasoning. Recent progress addresses this limitation by tightly coupling VLMs with diffusion backbones (Deng et al., 2025; Wu et al., 2025b; Liu et al.,

Table 1: **Comparison of general image editing benchmarks.** GEditBench v2 pioneers a necessary transition to complex open-set scenarios, spanning 23 diverse tasks curated from real-world queries to establish a truly comprehensive evaluation standard.

Benchmark	Size	Tasks	Complex Edit Support	Open-set Instruction	Evaluation	
					Unify Metrics	Human-Aligned
KontextBench (Labs et al., 2025)	1,026	5	✗	✗	✗	–
AnyEdit-Bench (Yu et al., 2025)	1,250	25	✗	✗	✗	Low
ImgEdit-Bench (Ye et al., 2026b)	811	14	✗	✗	✓	Mid
GEdit-Bench (Liu et al., 2025)	606	11	✗	✗	✓	Mid
ICE-Bench (Pan et al., 2025)	6,538	31	✗	✗	✗	Low
UnicBench (Ye et al., 2026a)	1,100	22	✓	✗	✓	Mid
GEditBench v2	1,200	23	✓	✓	✓	**High**

2025; Team et al., 2025a; Wu et al., 2025a). Generally, this integration follows two main paradigms: models such as BAGEL (Deng et al., 2025) and OmniGen2 (Wu et al., 2025b) jointly optimize multi-modal understanding and generation within a unified framework, whereas decoupled designs like Step1X-Edit (Liu et al., 2025) and Qwen-Image-Edit (Wu et al., 2025a) leverage VLMs as powerful multi-modal encoders to provide structured editing conditions for diffusion transformers. Concurrently, proprietary systems (e.g., GPT-Image-1.5 (OpenAI, 2025), Nano Banana Pro (Team et al., 2023), and Seedream4.5 (Team et al., 2025b)) further advance zero-shot, open-domain editing capabilities through large-scale multi-modal training and integrated chain-of-thought. Despite their impressive capabilities, current models still suffer from a fundamental limitation in understanding instruction boundaries, leading to degraded visual consistency. This gap highlights the need for a rigorous and reliable evaluation of visual consistency in image editing.

Benchmarking for Instruction-based Image Editing. Early benchmarking efforts, such as KontextBench (Labs et al., 2025), primarily relied on human evaluation, which is costly and difficult to scale. Later works like AnyEdit-Bench (Yu et al., 2025) and ICE-Bench (Pan et al., 2025) introduce automated metrics (e.g., L_1-norm, CLIP (Radford et al., 2021)/DINO (Oquab et al., 2023) scores) for each evaluation dimension, but combining these disparate metrics often leads to fragmented and inconsistent assessments. Motivated by the VLM-as-a-Judge paradigm (Ku et al., 2024), ImgEdit (Ye et al., 2026b), GEdit (Liu et al., 2025) and UnicBench (Ye et al., 2026a) benchmarks leverage powerful VLMs (e.g., GPT-4o) to unify evaluation. However, these approaches remain constrained by their reliance on opaque, closed-source APIs and the use of absolute rating schemes that inherently struggle to capture the relative nature of human preference. To this end, we develop an 8B assessment model fine-tuned for pairwise comparison, achieving strong human alignment. In addition, we extend evaluation beyond closed-set task definitions by incorporating open-set instructions derived from trending real-world edits that resist explicit task categorization, enabling a more realistic evaluation of image editing. A comparative analysis with prior image editing benchmarks is presented in Table 1.

3 GEditBench v2

3.1 Benchmark Construction

To ensure comprehensive task coverage, based on (Labs et al., 2025; Yu et al., 2025; Ye et al., 2026b; Liu et al., 2025), we first structure our benchmark into four main categories, encompassing 19 distinct tasks: **(1) Local Editing**, which includes edits within a restricted region like *Subject Addition*, *Subject Removal*, *Subject Replace*, *Size Adjustment*, *Color Alteration*, *Material Modification*, *Portrait Beautification*, *Motion Change*, *Relation Change*, and *Text Editing*; **(2) Global Editing**, covering holistic visual transformations such as *Background Change*, *Style Transfer*, *Tone Transfer*, *Camera Motion*, and *Line2Image*; **(3) Reference Editing**, which tests identity-driven generation such as *Character Reference*, *Object Reference*, and *Style Reference*; **(4) Hybrid Editing**, combining 3∼5 basic edits into a single complex instruction, termed *Hybrid*.

Next, to better reflect real-world user needs within the established taxonomy, we introduce three novel tasks. Specifically, under the Local Editing category, we introduce *In-Image Text Translation*, which aims to reduce the costs of producing multilingual posters and advertisements, and *Chart Editing*, designed to support chart refinement and chart-type transformation. Furthermore, within the Global

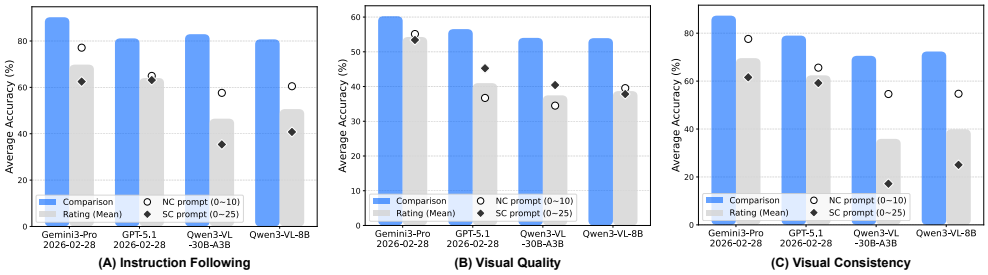


Figure 2: **Human preference agreement of pointwise and pairwise evaluation paradigms** across four VLMs in (A) instruction following, (B) visual quality, and (C) visual consistency dimensions. Pairwise evaluation consistently achieves higher agreement with human judgments, suggesting its superior human alignment over prior absolute scoring. NC and SC prompts are adopted from (Ye et al., 2026a) and (Luo et al., 2026), respectively.

Editing category, we elevate *Enhancement* to an independent task from tone transfer due to its critical practical utility, spanning nine low-level restoration tasks (i.e., blur, compression, moiré, low-light, noise, flare, reflection, haze, and rain), old photo restoration, and overexposed photo rescue.

Finally, to move beyond closed-set paradigms toward real-world, open-ended scenarios, we introduce the fifth category: **Open-Set Editing**. This category comprises 100 trending real-world instructions that cannot be explicitly assigned to predefined task taxonomies, enabling a more realistic evaluation of instruction generalization in the wild. Please refer to Appendix B.1 for detailed task definition.

For the above tasks, following (Liu et al., 2025), we collect real-world user editing requests from the Internet, e.g., Reddit and X, and manually filter instructions with similar intents by trained experts. To protect user privacy, we replace the original user-uploaded images with publicly available images, supplemented by a small portion generated using Nano Banana Pro (Team et al., 2023) or sourced from existing benchmarks (Wu et al., 2025b; Liu et al., 2025). This strategy preserves realistic editing contexts while ensuring privacy protection and reproducibility. Finally, GEditBench v2 comprises 1,200 testing examples covering 22 predefined tasks and one open-set editing task.

3.2 Evaluation Metrics

Following prior works (Liu et al., 2025; Ye et al., 2026a), we utilize the VLM-as-a-Judge paradigm to evaluate the instruction-based editing models in three dimensions. **(1) Instruction Following**, **IF**: Measures both instruction comprehension and understanding of the underlying concepts. **(2) Visual Quality**, **VQ**: Evaluates the perceptual quality of the generated image, focusing on overall realism, natural appearance, and the absence of noticeable artifacts. **(3) Visual Consistency**, **VC**: Assesses preservation of non-target regions, penalizing unintended changes outside the specified edits.

In VLM-as-a-Judge, there are typically two schemes for evaluating generative models: *pointwise rating* and *pairwise comparison* (Chen et al., 2024). The pointwise scheme prompts VLMs to assign an absolute score to each image, while the pairwise scheme requires VLMs to express a relative preference between two candidate images. Although pointwise evaluation is widely adopted in existing image editing benchmarks (Ye et al., 2026b;a; Liu et al., 2025; Luo et al., 2026), pairwise evaluation is motivated by two main advantages.

• *Stronger Human Alignment:* To empirically validate this, we evaluated four VLMs across IF, VQ, and VC dimensions on two EditReward-Bench from (Luo et al., 2026; Wu et al., 2025c), randomly swapping image positions with a 50% probability to mitigate position bias. The used pointwise prompt templates for visual consistency evaluation, i.e., NC and SC, were proposed in (Ye et al., 2026a) and (Luo et al., 2026), respectively. As shown in Fig. 2, pairwise comparison consistently achieves substantially higher agreement with human judgments across all dimensions, indicating that pairwise preference modeling better reflects human judgment than pointwise rating.

• *Ceiling Effect Mitigation:* From a training perspective, pointwise evaluators learn a rigid mapping to absolute scores, severely bottlenecking their cognitive upper bound to the training distribution. When evaluating out-of-distribution edits, they tend to produce similar scores, resulting in a performance ceiling. Conversely, pairwise training optimizes for relative preference, ensuring robust generalization to new models without losing discriminative power.

Therefore, we adopt pairwise comparison for all evaluation dimensions. Specifically, for IF, evaluation requires extensive world knowledge to handle diverse and flexible user editing instructions. Since open-source models are generally insufficient for this task, we rely on GPT-4o (OpenAI, 2025) to perform pairwise assessments. For VQ, evaluation is instruction-free and can leverage existing text-to-image generation assessment models (Wang et al., 2026; Wu et al., 2026; Wang et al., 2025). For simplicity, we also use GPT-4o in a pairwise manner for VQ evaluation. For VC, to address the limitations discussed in Sec. 1 – reproducibility issues, model size trade-off, and absolute scoring ceiling – we develop **PVC-Judge**, an open-source assessment model explicitly fine-tuned for pairwise evaluation of visual consistency in image editing. Detailed description of PVC-Judge is provided in Sec. 4, while the pairwise prompts used for evaluation are presented in the Appendix B.2.

4 PVC-JUDGE: PAIRWISE VISUAL CONSISTENCY JUDGE

In this section, we first introduce the candidate image generation process and two novel preference pair construction pipelines for training **PVC-Judge**. We then present **VCReward-Bench** to rigorously evaluate its performance. Training details can be found in Appendix C.4.

4.1 CANDIDATE IMAGE GENERATION PIPELINE

Before constructing preference pairs, we first build a diverse candidate pool of edited images that enables meaningful comparison. As shown in Fig. 3, our pipeline has two stages. In the **prompt curation** stage, we collect (Input Image, Instruction) pairs, denoted as $(I_{in}, Inst)$, aligned with GEditBench v2's taxonomy from three open-source datasets: Pico-Banana-400K (Qian et al., 2025), Nano-Consistency-150K (Ye et al., 2025), and UnicEdit-10M (Ye et al., 2026a). Due to task coverage mismatch, valid pairs are obtained for all tasks except In-Image Text Translation, Relation Change, Chart Editing, Line2image, and Hybrid. To ensure semantic diversity within each task, we embed $(I_{in}, Inst)$ into a joint space using Qwen3-VL-Embedding (Li et al., 2026a) and apply a K-center greedy selection strategy (Sener & Savarese, 2018) to choose N representative input pairs per task. In the **image generation** stage, for each $(I_{in}, Inst)$ pair, we generate edited outputs using 7 distinct editing models, including BAGEL (Deng et al., 2025), Kontext (Labs et al., 2025), two variants of Step1X-Edit1.2 (preview and standard) (Yin et al., 2026), and Qwen-Image-Edit series (Base, 2509, and 2511) (Wu et al., 2025a).

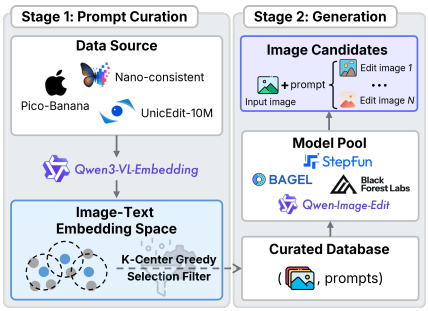


Figure 3: Two-stage candidates curation pipeline with prompt filtering.

Ablation Study of Pairs Number N per Task. Since a small N limits generalization, while a large N substantially increases the cost of preference construction and model training. To identify the most efficient scale, we vary N from 500 to 3,000 in increments of 500 across six representative editing tasks: Subject Addition, Subject Removal, Subject Replacement, Background Change, Style Transfer, and Tone Transfer. For each input pair, we randomly choose two candidate models for generation and leverage Gemini 3 Pro (Team et al., 2023) with the VC pairwise prompt to construct preference pair. As shown in Fig. 4, the fine-tuned Qwen3-VL-8B steadily improves on EditReward-Bench (Luo et al., 2026) as N increases to 1,500, after which the performance saturates. Therefore, we set N to 1,500 as a practical trade-off between coverage and efficiency.

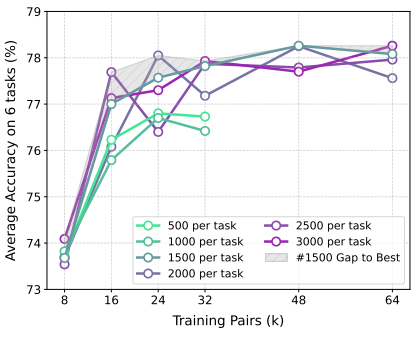


Figure 4: **Average accuracy of different image-instruction pairs per task** on six representative tasks for visual consistency. Performance improves steadily and saturates around 1,500.

Finally, we constructed ∼180k output images as the candidate pool for the following preference data construction.

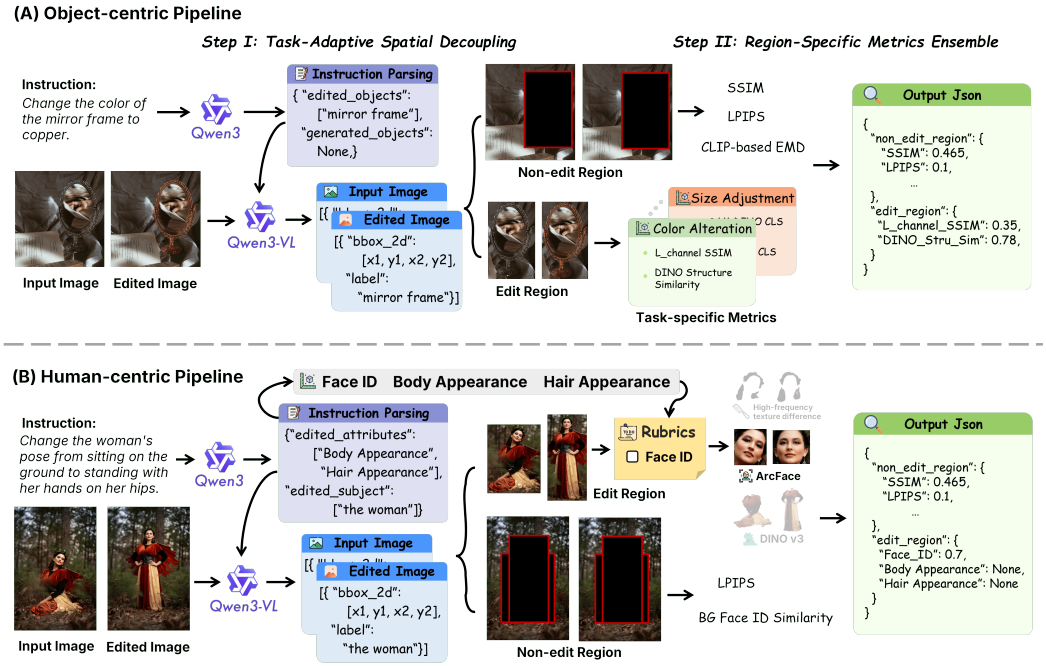


Figure 5: **Preference data construction pipelines.** (A) Object-centric pipeline: instructions are parsed to localize edited entities, partitioning the image into edit and non-edit regions. Region-specific metrics are then applied to enforce background fidelity while evaluating identity consistency within edited areas. (B) Human-centric pipeline: extends the object-centric pipeline by decomposing human attributes (face identity, body appearance, hair appearance) and dynamically excluding the edited attribute, enabling fine-grained consistency evaluation using specialized expert models.

4.2 Preference Pair Construction Protocol for Visual Consistency

We construct preference pairs via three specialized pipelines: object- and human-centric pipelines with a preference synthesis method for local editing, and a VLM-as-a-Judge pipeline for global tasks.

Object-centric Pipeline. As shown in Fig. 5(A), this pipeline evaluates instance identity for subject-level tasks (e.g., addition, removal, and attribute modification) through the following two steps.

• *Step I: Task-Adaptive Region Decoupling.* We first leverage Qwen3-4B-Instruct-2507 (Yang et al., 2025) to extract the editing target from the instruction. Since different editing operations affect the input image and output image asymmetrically, we then apply Qwen3-VL-8B-Instruct (Bai et al., 2025) for task-adaptive target grounding. For instance, the editing target is localized in the input image for removal, in the output image for addition, and in both for replacement or attribute modification. This process adaptively partitions each image into an 'Edit Region' (Ω_{edit}) covering the target regions and a 'Non-edit Region' (Ω_{non}) covering the remaining content.

• *Step II: Region-Specific Metrics Ensemble.* Building on the region partition, we apply a dual-strategy metric to assess consistency without penalizing the intended edit. In Ω_{non}, we enforce strict visual consistency using a combination of SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018), and CLIP-based Earth Mover's Distance (Rubner et al., 1998) (EMD) to ensure both low-level visual features and high-level semantic content remain unchanged. Conversely, within Ω_{edit}, we utilize task-specific metrics to decouple identity preservation from the editing effects. Take the color alteration task as an example, we compute SSIM exclusively on the lightness channel to assess structural integrity while allowing for the chromatic shifts required by the instruction. The mapping of these task-to-metric configurations is detailed in the Appendix C.1.

Human-centric Pipeline. For tasks involving human subjects, as shown in Fig. 5(B), we extend the object-centric pipeline by decomposing human visual properties into three orthogonal attributes: *Face IDentity* (Face ID), *Body Appearance*, and *Hair Appearance*. While inheriting the spatial decoupling logic described above, this specialization requires a more granular instruction-parsing phase: the language model identifies not only the target subjects but also the specific attribute slated

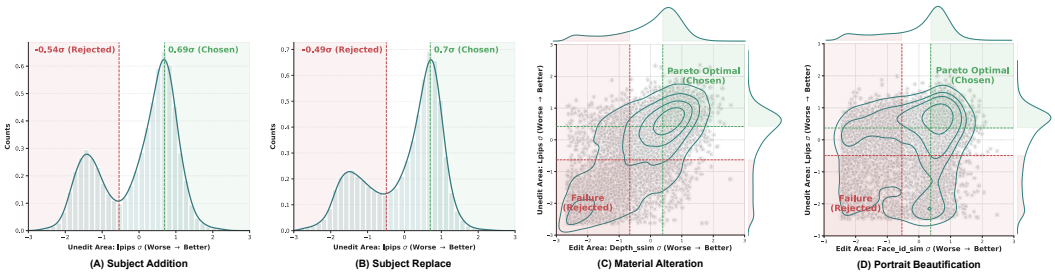


Figure 6: **Task-wise primary score distributions for preference pair synthesis.** (A-B) Shaded regions indicate the top/bottom 30% Z-score threshold used to identify Winners and Losers with a clear optimization margin for preference learning. (C-D) Combined with Pareto filtering, candidate pairs that only exhibit score separation across two primary metrics are retained.

for modification. This allows us to derive a dynamic, attribute-conditional rubric within Ω_{edit} by automatically excluding the modified property from the three attributes. The remaining stationary attributes are then quantified using specialized expert models, such as ArcFace (Deng et al., 2019) for Face ID and selfie segmenter for Body Appearance. While Ω_{non} remains governed by the similarity metrics defined above, this specialization generates reliable preference supervision that isolates intended human edits from unintended identity bleeding or anatomical distortions.

Score-based Robust Preference Synthesis. To transform raw scores into high-confidence preference pairs, we implement a synthesis approach based on statistical distribution and multi-metric consensus. For each task, we empirically select one primary metric in Ω_{non} and Ω_{edit}, respectively (e.g., LPIPS for Ω_{non} and Face ID for human Ω_{edit}), while treating all other metrics as auxiliary validators. The construction process begins with task-wise z-score normalization of primary metrics within each $(I_{in}, Inst)$ group to mitigate inter-group variance. These scores are then aggregated with candidates in the top and bottom 30% of the resulting distribution identified as Winners and Losers, respectively. The empirical score distributions for four representative tasks, as visualized in Fig. 6, further validate the efficacy of this thresholding in capturing clear optimization margins. To reconcile potentially conflicting regional requirements, we enforce a strict Pareto Dominance rule: a candidate pair (I_A, I_B) is used only if I_A is superior to I_B in at least one regional primary metric without being inferior in the other. Finally, the derived preference is cross-validated through majority voting across all auxiliary validators; any pair exhibiting a conflict between primary and auxiliary indicators is discarded. Such dual-layer filtering ensures that the resulting preference pairs exhibit a significant optimization margin and are grounded in consistent visual evidence across all evaluated dimensions.

VLM-as-a-Judge Annotation. For global editing tasks where localized metrics are inapplicable, we utilize Gemini 3 Pro (Team et al., 2023) for pairwise consistency assessment. To avoid the cost of exhaustively evaluating all $\binom{7}{2}$ candidate pairs for each $(I_{in}, Inst)$, we randomly sample 6 in-group pairs, balancing pair diversity and annotation cost. Ablation results are provided in Appendix C.2.

Using the above three pipelines, we construct ∼128K training preference pairs from ∼180K candidate images obtained in Sec. 4.1. Detailed data distributions are provided in Appendix C.3.

4.3 VCREWARD-BENCH

To evaluate the alignment of PVC-Judge with human judgment, we build VCReward-Bench, a high-quality meta-evaluation set that offers a more comprehensive taxonomy and larger scale than previous benchmarks like EditReward-Bench in (Luo et al., 2026) and (Wu et al., 2025c) (see Table 2). Adopting the 21 predefined tasks in Sec. 3.1, we collect image-instruction pairs by repurposing existing image editing benchmarks for most categories. For newly defined tasks such as Chart Editing, we harvest raw images from the web and employ trained experts to craft precise instructions. Candidate images for each input pair are synthesized using the 7 editing models adopted in Sec. 4.1 alongside Nano Banana Pro (Team et al., 2023), ensuring a broad spectrum of editing artifacts. Unlike the automated protocol used for training data, VCReward-Bench is entirely annotated by

Table 2: **Comparison of VCReward-Bench with existing reward benchmarks.**

Benchmark	Size	Tasks
EditScore (Luo et al., 2026)	3K	11
EditReward (Wu et al., 2025c)	1.5K	-
VCReward-Bench	**3.5K**	**21**

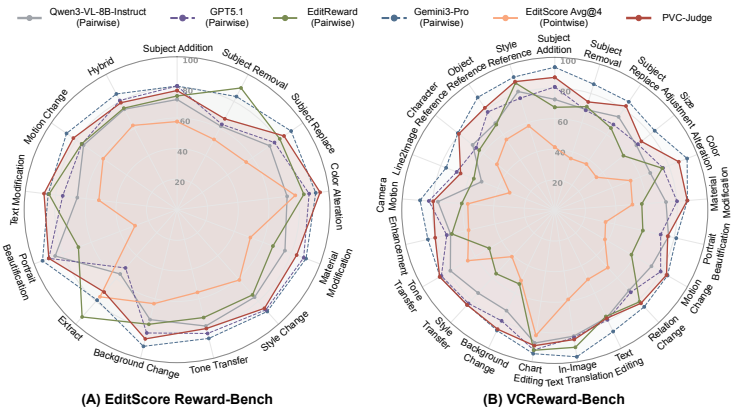


Figure 7: **Human alignment of assessment models for visual consistency** on (A) EditReward-Bench (Luo et al., 2026) and (B) VCReward-Bench. Notably, our model achieves state-of-the-art performance among open-source evaluators across nearly all tasks, performing on par with or even surpassing the proprietary GPT-5.1-2026-02-28.

human experts through rigorous pairwise comparisons (detailed annotation process is provided in Appendix D). Finally, VCReward-Bench contains 3,506 preference pairs across 21 tasks, serving as a robust testbed for evaluating the assessment models of image editing in visual consistency.

5 Experiments

In this section, we conduct a meta-evaluation of human agreement for our PVC-Judge and report the leaderboard on GEditBench v2 across the three evaluation dimensions. Additional qualitative analyses of challenging applications, including **open-set edits**, **spatial relation perception**, and **small-face consistency**, are provided in Appendix E.3.

5.1 Meta-Evaluation of PVC-Judge

Setup. To evaluate the effectiveness of our PVC-Judge, we compare it against Qwen-3-VL-8B-Instruct, GPT-5.1, Gemini 3 Pro, and two editing reward models, i.e., the EditScore (Luo et al., 2026) (Qwen-3-VL-8B version) and EditReward (Wu et al., 2025c) (MiMo-VL-7B-SFT-2508 version). For a fair comparison, the three VLMs employ the exact same pairwise comparison prompt as PVC-Judge. For EditScore and EditReward, their original author-provided prompts are adopted. Besides, we further enhance EditScore with its recommended self-ensembling strategy to establish a better baseline, denoted as EditScore-Avg@4. To assess human alignment, we evaluate the models on our VCReward-Bench and the visual consistency subset of EditReward-Bench (Luo et al., 2026).

Results and Discussion. Comparison results in Fig. 7 reveal three key observations: (1) PVC-Judge consistently outperforms its base model, Qwen3-VL-8B-Instruct, across all tasks on both benchmarks, validating the quality of our synthesized preference pairs and data curation pipeline. (2) Compared with existing reward models, our model achieves higher accuracy on most tasks, indicating that PVC-Judge provides more precise and human-aligned evaluation for visual consistency. (3) Despite its modest 8B size, our model achieves performance on par with leading proprietary VLMs and even surpasses GPT-5.1 on average. Appendix E.2 provides the detailed numerical results.

5.2 Main Results on GEditBench v2

Setup. We evaluate 16 representative image editing models on **GEditBench v2** to provide a comprehensive evaluation of current capabilities. These models include three leading closed-source models, GPT-Image-1.5 (OpenAI, 2025), Nano Banana Pro (Team et al., 2023), and Seedream4.5 (Team et al., 2025b), as well as thirteen open-source models: BAGEL (Deng et al., 2025), OmniGen2 (Wu et al., 2025b), Kontext (Labs et al., 2025), Step1x-Edit-v1p2 (Yin et al., 2026), three versions of Qwen-Image-Edit (Base, 2509, 2511) (Wu et al., 2025a), FLUX.2 [dev] (Labs, 2025), Flux.2 [dev] Turbo, LongCat-Image-Edit (Team et al., 2025a), GLM-Image (Z.ai, 2026), along with two efficiency-

Table 3: **GEditBench v2 leaderboard.** Models are ranked by Elo ratings derived from a pairwise comparison paradigm. Instruction Following and Visual Quality are assessed by GPT-4o (26-03-24), while Visual Consistency is exclusively evaluated by our proposed PVC-Judge. Overall Elo scores and their 95% Confidence Intervals (CI) are computed via 1,000 bootstrap iterations. ● means closed-source, ● means open-source. ⋆The Elo scores of Arena were recorded on March 26, 2026.

Model	#Samples	Instruction Following		Visual Quality		Visual Consistency		Overall		Arena⋆	
		ELO↑	95% CI	ELO↑	95% CI	ELO↑	95% CI	ELO↑	95% CI	ELO↑	Rank
● Nano Banana Pro (26-03-04)	1,156	1,126	-13/+15	1,066	-9/+10	1,108	-11/+11	1,096	-6/+6	1,251	2
● Seedream 4.5 (26-03-11)	1,190	1,111	-12/+12	1,142	-11/+11	1,030	-11/+12	1,089	-7/+7	1,196	3
● GPT Image 1.5 (26-03-04)	1,081	1,260	-13/+15	1,149	-12/+12	846	-13/+13	1,071	-7/+6	1,270	1
● FLUX.2 [klein] 9B	1,200	1,083	-13/+12	1,025	-11/+10	1,019	-10/+9	1,039	-6/+6	1,166	4
● Qwen-Image-Edit-2511	1,200	1,095	-10/+10	1,060	-11/+11	972	-9/+10	1,038	-6/+6	1,164	5
● FLUX.2 [klein] 4B	1,200	1,007	-12/+12	1,019	-10/+10	1,070	-10/+10	1,031	-6/+6	1,107	10
● FLUX.2 [dev] Turbo	1,200	1,068	-12/+12	936	-10/+10	1,064	-11/+10	1,021	-6/+6	1,153	6
● Qwen-Image-Edit-2509	1,200	1,033	-10/+11	1,062	-10/+12	955	-9/+9	1,014	-5/+6	1,142	7
● Qwen-Image-Edit	1,200	991	-10/+10	1,073	-11/+12	971	-11/+11	1,010	-6/+6	1,088	12
● FLUX.2 [dev]	1,200	1,037	-12/+13	965	-10/+10	1,018	-11/+11	1,006	-7/+7	1,137	8
● LongCat-Image-Edit	1,200	1,018	-10/+11	968	-10/+9	1,017	-10/+9	1,001	-6/+5	1,111	9
● Step1X-Edit-v1p2	1,200	909	-12/+12	1,007	-12/+11	1,067	-11/+11	996	-6/+7	1,093	11
● GLM-Image	1,200	787	-13/+14	1,023	-11/+11	1,109	-13/+14	979	-6/+6	930	14
● OmniGen V2	1,200	807	-13/+12	910	-12/+12	929	-13/+13	888	-7/+7	919	15
● FLUX.1 Kontext [dev]	1,200	849	-13/+13	900	-13/+14	840	-14/+13	869	-7/+8	1,017	13
● Bagel	1,200	820	-13/+13	694	-17/+16	987	-13/+14	851	-8/+8	915	16

oriented step-distilled variants, FLUX.2 [klein] 4B/9B (Labs, 2026). Default hyperparameter settings are used to ensure fairness. For our pairwise evaluation setting, we rank models using ELO scores derived from the Bradley-Terry model (Bradley & Terry, 1952). Details are provided in Appendix E.1.

Leaderboard and Discussion. Table 3 presents the final evaluation results ranked by the Overall ELO score, including IF and VQ ELO scores obtained via GPT-4o pairwise comparisons, as well as VC ELO scores evaluated by our PVC-Judge. For a broader context, we include human-annotated Arena (Analysis, 2026) Elo scores as a reference. Although Arena scores derive from different test sets, our Overall Elo achieves a strong Spearman's rank correlation (ρ=0.929, p<2e-7) with the Arena rankings, validating that our automated evaluation ecosystem reliably aligns with human preferences.

At the apex of the rankings, proprietary models continue to set the performance ceiling. Nano Banana Pro secures the definitive first place, closely followed by Seedream 4.5 with only a marginal capability gap. Strikingly, the open-source community demonstrates remarkable competitiveness through architectural efficiency. FLUX.2 [klein] 9B, a 4-step distilled model, emerges as the open-source champion and successfully narrows the gap with proprietary giants. It maintains a narrow lead over formidable open-source alternatives like Qwen-Image-Edit-2511, highlighting the immense potential of lightweight models for high-quality image editing.

Deconstructing the Overall Elo into its constituent dimensions reveals the inherent trade-off between aggressive instruction execution and visual consistency. Models like GLM-Image and Bagel perfectly illustrate an "under-editing" trap. They achieve artificially inflated VC scores (1,109 and 987, respectively) precisely because their diminished IF capabilities (787 and 820) prevent them from meaningfully modifying the input image. This phenomenon strongly validates the necessity of this multi-dimensional ranking ecosystem. It suggests that evaluating visual consistency in isolation remains insufficient for assessing true editing proficiency. Conversely, top-tier models like Nano Banana Pro and the distilled FLUX.2 [klein] variants successfully navigate this complex trade-off. They secure top overall rankings by maintaining a delicate equilibrium between executing complex user prompts and preserving unintended elements seamlessly.

6 Conclusion

In this paper, we present a unified evaluation ecosystem to address the evaluative mismatch in instruction-based image editing. We introduce **GEditBench v2**, a 23-task benchmark that moves beyond constrained settings toward complex, open-set real-world scenarios. To enable reliable assessment for visual consistency, we propose **PVC-Judge**, an open-source pairwise evaluation model. Powered by two novel region-decoupled preference data synthesis pipelines, our model achieves strong human alignment in visual consistency. We further establish **VCReward-Bench**, designed to evaluate assessment models of image editing in visual consistency. In future work, we plan to integrate PVC-Judge into the training loop as a reward model for precise image editing.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data sets and to assist with translation. Specifically, generative AI tools were used as part of constructing the preference pair training data for PVC-Judge, as described in Sec. 4.2, and to assist with translation during manuscript preparation. We have not used generative AI tools to design or provide feedback on research methodology or experiments, implement methods, clean or reformat datasets, support qualitative or thematic data analysis, or interpret results, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to polish the paper to improve readability. We have reviewed all AI-assisted work. All AI-assisted text and translations were reviewed by 3 authors, and the quality and effectiveness of the constructed training data were further evaluated using an independent human-annotated meta-benchmark (Sec. 4.3). We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Artificial Analysis. Artificial Analysis Image Editing Leaderboard, 2026. https://artificialanalysis.ai/image/leaderboard/editing.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*, 2025.

Ralph Allan Bradley and Milton E Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3/4):324–345, 1952.

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.

Dongping Chen, Ruoxi Chen, Shilin Zhang, Yaochen Wang, Yinuo Liu, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. MLLM-as-a-Judge: Assessing Multimodal LLM-as-a-Judge with Vision-Language Benchmark. In *Proceedings of the International Conference on Machine Learning*, pp. 6562–6595, 2024.

DeepSeek-AI. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *arXiv preprint arXiv:2501.12948*, 2025.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. Emerging Properties in Unified Multimodal Pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4690–4699, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*, 2022.

Zhangqi Jiang, Junkai Chen, Beier Zhu, Tingjin Luo, Yankun Shen, and Xu Yang. Devils in Middle Layers of Large Vision-Language Models: Interpreting, Detecting and Mitigating Object Hallucinations via Attention Lens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25004–25014, 2025.

Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. VIEScore: Towards Explainable Metrics for Conditional Image Synthesis Evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12268–12290, 2024.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence, 2025. https://bfl.ai/blog/flux-2.

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

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space. *arXiv preprint arXiv:2506.15742*, 2025.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Qwen3-VL-Embedding and Qwen3-VL-Reranker: A Unified Framework for State-of-the-Art Multimodal Retrieval and Ranking. *arXiv preprint arXiv:2601.04720*, 2026a.

Yaowei Li, Yuxuan Bian, Xuan Ju, Zhaoyang Zhang, Junhao Zhuang, Ying Shan, Yuexian Zou, and Qiang Xu. BrushEdit: All-In-One Image Inpainting and Editing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026b.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, Guopeng Li, Yuang Peng, Quan Sun, Jingwei Wu, Yan Cai, Zheng Ge, Ranchen Ming, Lei Xia, Xianfang Zeng, Yibo Zhu, Binxing Jiao, Xiangyu Zhang, Gang Yu, and Daxin Jiang. Step1X-Edit: A Practical Framework for General Image Editing. *arXiv preprint arXiv:2504.17761*, 2025.

Xin Luo, Jiahao Wang, Chenyuan Wu, Shitao Xiao, Xiyan Jiang, Defu Lian, Jiajun Zhang, Dong Liu, and Zheng Liu. EditScore: Unlocking Online RL for Image Editing via High-Fidelity Reward Modeling. In *International Conference on Learning Representations*, pp. 33027–33056, 2026.

OpenAI. Introducing 4o Image Generation, 2025. URL https://openai.com/index/introducing-4o-image-generation/.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning Robust Visual Features without Supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Yulin Pan, Xiangteng He, Chaojie Mao, Zhen Han, Zeyinzi Jiang, Jingfeng Zhang, and Yu Liu. ICE-Bench: A Unified and Comprehensive Benchmark for Image Creating and Editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16586–16596, 2025.

Yusu Qian, Eli Bocek-Rivele, Liangchen Song, Jialing Tong, Yinfei Yang, Jiasen Lu, Wenze Hu, and Zhe Gan. Pico-Banana-400K: A Large-Scale Dataset for Text-Guided Image Editing. *arXiv preprint arXiv:2510.19808*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the International Conference on Machine Learning*, pp. 8748–8763, 2021.

Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. A Metric for Distributions with Applications to Image Databases. In *Proceedings of the International Conference on Computer Vision*, pp. 59–66, 1998.

ByteDance Seed. Official Release of Seed1.8: A Generalized Agentic Model, 2026. https://seed.bytedance.com/en/blog/official-release-of-seed1-8-a-generalized-agentic-model.

Ozan Sener and Silvio Savarese. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *International Conference on Learning Representations*, 2018.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: A Family of Highly Capable Multimodal Models. *arXiv preprint arXiv:2312.11805*, 2023.

Meituan LongCat Team, Hanghang Ma, Haoxian Tan, Jiale Huang, Junqiang Wu, Jun-Yan He, Lishuai Gao, Songlin Xiao, Xiaoming Wei, Xiaoqi Ma, Xunliang Cai, Yayong Guan, and Jie Hu. LongCat-Image Technical Report. *arXiv preprint arXiv:2512.07584*, 2025a.

Seedream Team, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, Xiaowen Jian, Huafeng Kuang, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, Wei Liu, Yanzuo Lu, Zhengxiong Luo, Tongtong Ou, Guang Shi, Yichun Shi, Shiqi Sun, Yu Tian, Zhi Tian, Peng Wang, Rui Wang, Xun Wang, Ye Wang, Guofeng Wu, Jie Wu, Wenxu Wu, Yonghui Wu, Xin Xia, Xuefeng Xiao, Shuang Xu, Xin Yan, Ceyuan Yang, Jianchao Yang, Zhonghua Zhai, Chenlin Zhang, Heng Zhang, Qi Zhang, Xinyu Zhang, Yuwei Zhang, Shijia Zhao, Wenliang Zhao, and Wenjia Zhu. Seedream 4.0: Toward Next-generation Multimodal Image Generation. *arXiv preprint arXiv:2509.20427*, 2025b.

Qian Wang, Biao Zhang, Michael Birsak, and Peter Wonka. InstructEdit: Improving Automatic Masks for Diffusion-based Image Editing With User Instructions. *arXiv preprint arXiv:2305.18047*, 2023.

Yibin Wang, Zhimin Li, Yuhang Zang, Yujie Zhou, Jiazi Bu, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. Pref-GRPO: Pairwise Preference Reward-based GRPO for Stable Text-to-Image Reinforcement Learning. *arXiv preprint arXiv:2508.20751*, 2025.

Yibin Wang, Zhimin Li, Yuhang Zang, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. Unified Multimodal Chain-of-Thought Reward Model through Reinforcement Fine-Tuning. *Advances in Neural Information Processing Systems*, 38:159130–159157, 2026.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. 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, Ze Liu, Ziyi Xia, Chaofan Li, Haoge Deng, Jiahao Wang, Kun Luo, Bo Zhang, Defu Lian, Xinlong Wang, Zhongyuan Wang, Tiejun Huang, and Zheng Liu. OmniGen2: Exploration to Advanced Multimodal Generation. *arXiv preprint arXiv:2506.18871*, 2025b.

Keming Wu, Sicong Jiang, Max Ku, Ping Nie, Minghao Liu, and Wenhu Chen. EditReward: A Human-Aligned Reward Model for Instruction-Guided Image Editing. *arXiv preprint arXiv:2509.26346*, 2025c.

Tianhe Wu, Jian Zou, Jie Liang, Lei Zhang, and Kede Ma. VisualQuality-R1: Reasoning-Induced Image Quality Assessment via Reinforcement Learning to Rank. *Advances in Neural Information Processing Systems*, 38:88167–88190, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, 2025.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth Anything V2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024.

Junyan Ye, Dongzhi Jiang, Zihao Wang, Leqi Zhu, Zhenghao Hu, Zilong Huang, Jun He, Zhiyuan Yan, Jinghua Yu, Hongsheng Li, Conghui He, and Weijia Li. Echo-4o: Harnessing the Power of GPT-4o Synthetic Images for Improved Image Generation. *arXiv preprint arXiv:2508.09987*, 2025.

Keming Ye, Zhipeng Huang, Canmiao Fu, Qingyang Liu, Jiani Cai, Zheqi Lv, Chen Li, Jing Lyu, Zhou Zhao, and Shengyu Zhang. UnicEdit-10M: A Dataset and Benchmark Breaking the Scale-Quality Barrier via Unified Verification for Reasoning-Enriched Edits. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 37279–37289, 2026a.

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. *Advances in Neural Information Processing Systems*, 38, 2026b.

Fukun Yin, Shiyu Liu, Yucheng Han, Zhibo Wang, Peng Xing, Rui Wang, Wei Cheng, Yingming Wang, Aojie Li, Zixin Yin, Pengtao Chen, Xiangyu Zhang, Daxin Jiang, Xianfang Zeng, and Gang Yu. ReasonEdit: Towards Reasoning-Enhanced Image Editing Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23283–23293, 2026.

Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. AnyEdit: Mastering Unified High-Quality Image Editing for Any Idea. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26125–26135, 2025.

Z.ai. GLM-Image: Auto-regressive for Dense-knowledge and High-fidelity Image Generation, 2026. `https://z.ai/blog/glm-image`.

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.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

Haozhe Zhao, Xiaojian Shawn 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. *Advances in Neural Information Processing Systems*, 37:3058–3093, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

APPENDIX OVERVIEW

In this supplemental material, we provide additional details omitted from the main text:

A Limitations 15

B GEditBench v2 15

B.1 Detailed Task Definition 15

B.2 Pairwise Evaluation Prompts 18

C Pairwise Visual Consistency Judge 19

C.1 `(task, Pipeline, Region-Specific Metrics)` Mapping 19

C.2 VLM-as-a-Judge Pipeline of Preference Construction for Global Editing Tasks 20

C.3 Preference Data Distribution 21

C.4 Training Details 22

D Annotation Protocol for VCReward-Bench 22

E Ranking Methodology and Additional Results 23

E.1 Ranking Methodology 23

E.2 Full Numerical Results of Meta-Evaluation 24

E.3 Qualitative Analysis 24

A LIMITATIONS

While GEditBench v2 establishes a general image editing benchmark built on real-world instructions and develops an assessment model for visual consistency, it may still exhibit several limitations.

First, evaluating large-scale image editing models requires substantial computational resources and long inference times. To maintain practicality, we limit the current benchmark size, which may reduce the diversity of test samples within individual editing tasks. Future work will expand the dataset while exploring more efficient evaluation and sampling strategies.

Second, our automated object- and human-centric pipelines for constructing preference pairs rely on several pre-trained foundation models, e.g., SAM, CLIP, and DINOv3, to extract regional features. While these models enable scalable data construction, they may also introduce potential biases into the resulting preference dataset. Mitigating such inherited biases and improving the robustness of the underlying feature extractors will be an important direction for future iterations.

B GEDITBENCH V2

B.1 DETAILED TASK DEFINITION

GEditBench v2 establishes a comprehensive evaluation taxonomy comprising 23 distinct editing tasks. We distribute these tasks across five fundamental categories: (1) Local, (2) Global, (3) Reference, (4) Hybrid, and a new (5) Open-set category. This multi-level design strictly challenges both basic manipulation edits and advanced real-world instruction understanding. Below, we comprehensively explain each task.

Local Editing. This category evaluates spatially restricted modifications that target specific image regions or objects across the following 12 tasks.

1. *Subject Addition*: Seamlessly inserts a newly designated entity into a specific location within the original scene.
2. *Subject Removal*: Erases a specified target object and plausibly reconstructs the newly exposed underlying region.
3. *Subject Replace*: Swaps an existing object with a completely new target entity while strictly preserving the original spatial layout.
4. *Size Adjustment*: Scales a specific object up or down without distorting its inherent structural integrity.
5. *Color Alteration*: Modifies the hue of a targeted region while retaining its original physical texture and lighting.
6. *Material Modification*: Transforms the surface properties of a specific object to reflect a completely different physical material.
7. *Portrait Beautification*: Applies targeted aesthetic enhancements to human subjects without altering their underlying recognizable identity.
8. *Motion Change*: Alters the physical posture or dynamic action of a specified subject within the visual environment.
9. *Relation Change*: Modifies the spatial positioning or physical interaction dynamics between multiple existing entities. More examples are provided in Fig. 8.
10. *Text Editing*: Executes the fundamental addition, deletion, or semantic modification of text elements alongside targeted changes to their typographic styles and spatial layouts. Some cases are shown in Fig. 9.
11. *In-Image Text Translation*: Accurately translates embedded textual elements into a target language while seamlessly preserving the original typographic aesthetics and background context.
12. *Chart Editing*: Transforms data visualizations through targeted aesthetic enhancements or structural chart type conversions without compromising the underlying informational integrity (e.g., numerical values and geometric proportions). Fig. 10 visualizes representative examples.



Change the stacking order so the Bitcoin is behind the other two coins.

Place the purple chick figurine back into the packaging box with its face facing outward.

Insert the cut slice back into the cake's missing section, keeping the cut seams visible.

Rotate the white teapot so that its spout faces the white porcelain bowl.

Change their relationship from facing each other to standing back-to-back.

Figure 8: `(Input Image, Instruction)` examples for the *Relation Change* task.



Change the text 'Fairgrounds' to italics.

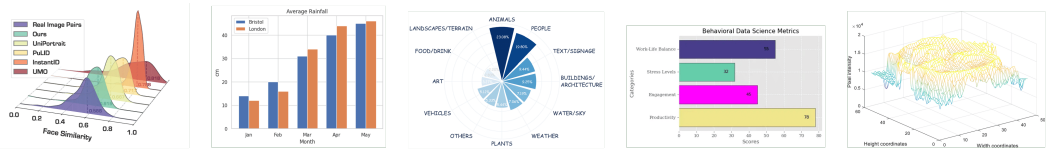
Modify the font of 'ESTATE TACHEN' in the picture to a handwritten font.

Swap the positions of 'Welcome to' and 'ONTARIO'.

Change the font of the text "Spring in the Air" in the poster to 'New Roman MT'.

Replace the text "生活" with "工作".

Figure 9: `(Input Image, Instruction)` examples for the *Text Editing* task. Our benchmark comprehensively evaluates fundamental semantic modifications alongside highly demanding typographic layout changes and precise font alterations.



Swap the stacking order of the PuLID and UMO distributions.

Change this to a line chart to better visualize the trend over time.

Change the color scheme from a blue gradient to "warm tones".

Add a vertical red dashed line at the score of 52.5 and include a legend labeling it as "Average Score".

Convert the 3D surface from a wireframe mesh rendering to a solid surface rendering.

Figure 10: `(Input Image, Instruction)` examples for the *Chart Editing* task.

Global Editing. This editing paradigm evaluates holistic transformations that alter the entire visual atmosphere or structural layout of the image across the following 6 tasks.

13. *Background Change*: Replaces the contextual environment surrounding the main subjects with diverse natural and cultural settings across both indoor and outdoor scenarios.
14. *Style Transfer*: Imposes 17 distinct artistic and aesthetic styles across the entire image while rigorously maintaining the underlying geometric layout.
15. *Tone Transfer*: Shifts the global lighting conditions, color palettes, seasonal atmospheres, and weather dynamics to evoke a completely different environmental mood.
16. *Enhancement*: Restores visual fidelity by eradicating 9 specific low-level degradations, including blur, compression, moiré, low-light, noise, flare, reflection, haze, and rain alongside complex old photo restoration and rigorous overexposure repair.



Pan the camera to the right, revealing the person in gray clothing at the right edge of the frame.

Tilt the camera upward so that the tower structure appears near the bottom of the frame.

Dolly in through the corridor, focusing on the person outside.

Switch the camera angle from a **low-angle** upward shot to a **high-angle** downward shot.

Rotate the camera 90 degrees to the left using the woman as the pivot point.

Figure 11: `(Input Image, Instruction)` examples for the *Camera Motion* task.



Have a model wear these shoes with white socks and give a close-up of the shoes.

Have a young girl, dressed in the outfit from the reference image, stand in front of a full-length mirror and take a selfie, facing the mirror.

Have a professionally dressed woman hold these two books and pose as if she's presenting or pitching them directly to the camera.

Generate an image of a knight in medieval armor driving this car for a ride along the coast.

A side-profile close-up shot of a woman wearing the crystal earring from this image.

Figure 12: `(Input Image, Instruction)` examples for the *Object Reference* task.

17. *Camera Motion*: Simulates physical camera movements encompassing typical zoom operations alongside complex pan, rotate, tilt, and special view switches (see Fig. 11).
18. *Line2Image*: Transforms sparse structural sketches or edge maps into fully rendered outputs with coherent global textures and lighting.

Reference Editing. This category rigidly evaluates the capacity of editing models to extract and faithfully transfer specific visual identities from an external guiding image across the following 3 tasks.

19. *Character Reference*: Synthesizes a specific person from a reference image across novel scenes and states while perfectly preserving their exact identity and characteristics from the reference image.
20. *Object Reference*: Synthesizes a specific object from the reference image across novel scenes and states while perfectly preserving its exact physical details. Fig. 12 visualizes representative examples.
21. *Style Reference*: Applies the visual style of the reference image to generate the target scene without copying its actual objects or content.

Hybrid Editing. This task (*Hybrid*) evaluates compositional editing instructions that require multiple operations within a single query. Each prompt combines $3 \sim 5$ predefined tasks from earlier categories into one unified instruction. This setting examines whether models can reliably execute multiple edits on the same image without omitting or conflating distinct semantic constraints.

Open-Set Editing. This open-set category is designed to evaluate model behavior under more general and flexible editing instructions that cannot be easily categorized into predefined task types. While the previous categories cover a broad range of structured editing operations, real-world queries often involve mixed intents or loosely specified objectives. These instructions do not strictly follow a fixed task taxonomy and, therefore, provide a useful test of model generalization beyond predefined settings. To construct this set, we manually curated 100 diverse prompts collected from public online

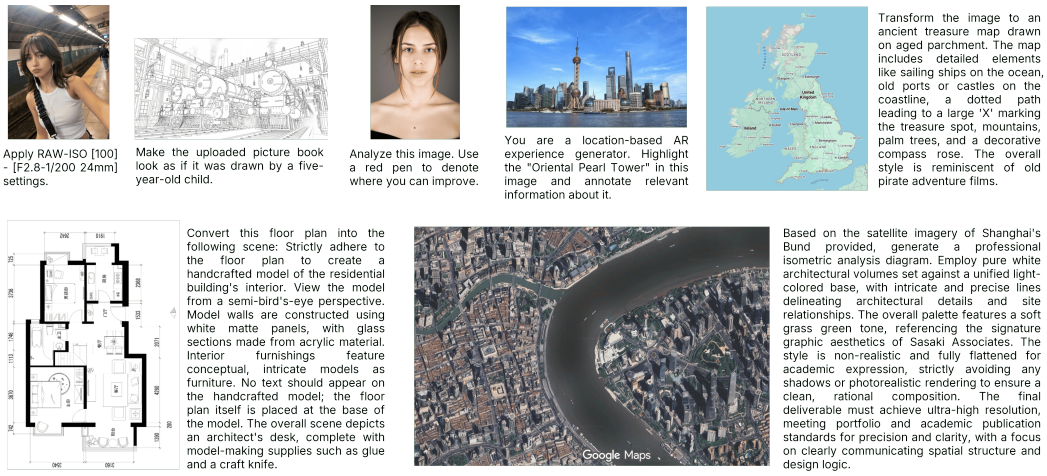


Figure 13: `(Input Image, Instruction)` examples for the *Open-Set* task.

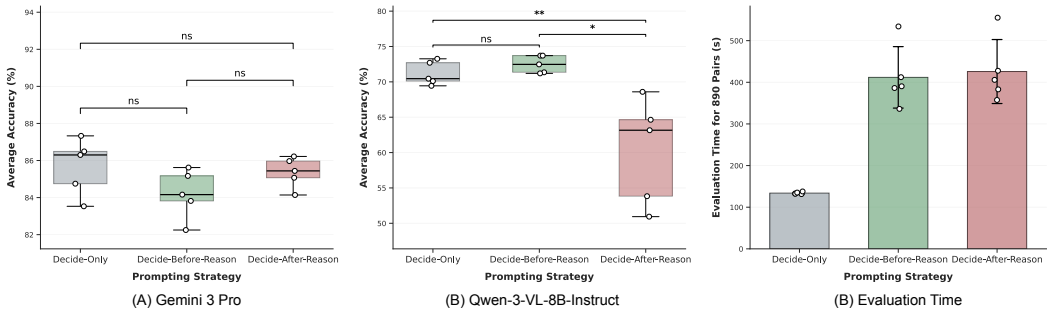


Figure 14: **Comparison results of three different prompting strategies** for visual consistency evaluation on EditReward-Bench (Luo et al., 2026). (A) and (B) show the results of Gemini 3 Pro and Qwen3-VL-8B-Instruct, respectively. Pairwise statistical significance is evaluated using the two–sided Mann–Whitney U test ($* * p < 0.01$, $*p < 0.05$, ns: $p \geq 0.05$). (C) reports the average evaluation time over 890 testing pairs measured with Qwen3-VL-8B-Instruct.

sources, including platforms such as X, Reddit, and community-curated GitHub repositories. These prompts were selected to reflect instruction patterns that are difficult to assign to a single editing category, often combining multiple intents or expressing goals at a higher semantic level (see Fig. 13). As a result, the open-set subset provides a complementary evaluation setting that better reflects the flexibility required in practical editing scenarios.

B.2 Pairwise Evaluation Prompts

This subsection introduces the pairwise evaluation prompts utilized by GEditBench v2 to rigorously evaluate the editing models from the Instruction Following, Visual Quality, and Visual Consistency dimensions. In general, the evaluation prompts can be categorized into three distinct strategies based on their output format: (1) **Decide-Only** template forces the model to output the winning candidate immediately without any explanation; (2) **Decide-After-Reason** template (Luo et al., 2026) requires the model to generate a detailed textual analysis before declaring the final winner; and (3) **Decide-Before-Reason** template (Ye et al., 2026a) instructs the model to state the winner upfront and subsequently append a textual justification. For accurate evaluation, we conducted the following comparative experiments on the visual consistency subset of EditReward-Bench (Luo et al., 2026).

Which Output Format Performs Better? We initially drafted a basic prompt for each of these three prompting strategies. Then, we leverage Gemini (Team et al., 2023), GPT (OpenAI, 2025), DeepSeek (DeepSeek-AI, 2025), and Doubao (Seed, 2026) to refine these prompts to obtain the

Table 4: **(Sub-Task, Pipeline, Region-Specific Metrics) mapping for preference pair automated annotation.** Primary metrics are marked in **bold**.

Sub-Task	Pipeline Type	Grounding Image	Region-Specific Metrics	
			Edit Region (Ω_{edit})	Non-edit Region (Ω_{non})
Subject Addition	object-centric	output	–	SSIM, **LPIPS**, EMD
Subject Removal	object-centric	input	–	SSIM, **LPIPS**, EMD
Subject Replace	object-centric	input,output	–	SSIM, **LPIPS**, EMD
Size Adjustment	object-centric	input,output	SAM-based CLIP `[CLS]` similarity, **SAM-based DINO `[CLS]` similarity**	**LPIPS**, EMD
Color Alteration	object-centric	input,output	**L-channel SSIM**, DINO structure similarity	**LPIPS**, EMD
Material Modification	object-centric	input,output	**depth SSIM**, DINO structure similarity	**LPIPS**, EMD
Portrait Beautification	human-centric	input,output	**Face ID**, Hair Appearance, Body Appearance	**LPIPS**, BG Face ID similarity
Motion Change	human-centric	input,output	**Face ID**, Hair Appearance, Body Appearance	**LPIPS**, BG Face ID similarity
Text Editing	object-centric	input,output	–	SSIM, **LPIPS**, EMD
Character Reference	human-centric	input	**max match Face ID**	–
Object Reference	object-centric	input,output	SAM-based CLIP `[CLS]` similarity, **SAM-based DINO `[CLS]` similarity**	–

other four unique prompt variations for each strategy. Fig. 14 (a) and (b) show our experimental results across Gemini 3 Pro (Team et al., 2023) and Qwen3-VL-8B-Instruct (Bai et al., 2025). These results demonstrate marginal accuracy deviations across all three paradigms for the closed-source Gemini 3 Pro. We attribute this robust performance to its superior multi-modal understanding and long context window. Conversely, forcing the Qwen3-VL-8B-Instruct model to generate a rationale first triggers a significant performance degradation. We attribute this failure to hallucinations in vision-language models, which arise because generating lengthy prior text dilutes core visual attention and corrupts the final structural judgment (Jiang et al., 2025). Since the strategies "Decide-Only" and "Decide-Before-Reason" exhibit nearly identical accuracy for Qwen3-VL-8B-Instruct, we further record their evaluation time for a total of 890 testing pairs (see Fig. 14 (c)). The result shows that the "Decide-Only" strategy substantially reduces the average evaluation time compared to the "Decide-Before-Reason" approach (133.93s vs 411.84s).

Therefore, we adopt the efficient "Decide-Only" prompting strategy for all three dimensions evaluation, and the prompts are detailed in Fig. 22, Fig. 23, and Fig. 24, respectively.

C Pairwise Visual Consistency Judge

C.1 (task, Pipeline, Region-Specific Metrics) Mapping

We develop object- and human-centric automated pipelines to efficiently construct high-quality preference pairs. Our method dynamically partitions both input and output images into edit and non-edit regions, and applies dedicated metrics to each region to assess overall visual consistency. Table 4 provides a comprehensive mapping between tasks, pipeline types, grounding image, and the corresponding regional metrics. Beyond SSIM (Wang et al., 2004) and LPIPS (Zhang et al., 2018), we further incorporate the following metrics:

- **EMD**: We first extract patch-level embeddings from the non-edit regions using the CLIP vision encoder (Radford et al., 2021), and then compute the Earth Mover's Distance (Rubner et al., 1998) between the input and output feature distributions to quantify semantic preservation.
- **SAM-based CLIP `[CLS]` similarity**: We first isolate the target foreground object using the Segment Anything Model to eliminate background interference. The global `[CLS]` token is then extracted from the CLIP vision encoder to compute a category-level cosine similarity between input and output, which helps mitigate consistency artifacts caused by viewpoint changes or size variations.
- **SAM-based DINO `[CLS]` similarity**: Similar to SAM-based CLIP `[CLS]` similarity, except that the global `[CLS]` token is extracted from the DINOv3 (Siméoni et al., 2025) to compute the cosine similarity.

- **L-channel SSIM**: Designed for the delicate color alteration task. To prevent legitimate color edits from artificially lowering the consistency score, we decouple structural integrity from chromatic variations. Specifically, we extract only the L (Lightness) channel from the edit region and compute the SSIM on this illumination map, enabling an unbiased evaluation of structural consistency.
- **DINO Structure Similarity**: We extract patch embeddings from the edit region using DINOv3, and construct spatial self-similarity matrices for both the input and output images. The L_2 distance between these matrices is then computed to measure structural consistency, while remaining robust to surface-level texture variations.
- **depth SSIM**: Since material changes often alter surface illumination and distort the lightness channel, we instead focus on geometric consistency. We use the Depth Anything V2 model (Yang et al., 2024) to extract depth maps from both the input and output images, and compute the SSIM score between them within the edit region. This geometry-based metric evaluates shape preservation independently of material appearance.
- **Face ID**: We use the ArcFace (Deng et al., 2019) network to extract identity embeddings from the localized target face in both the input and output images, and compute their cosine similarity to verify that the generated result preserves the subject's facial identity.
- **Hair Appearance**: A dedicated hair segmenter is first applied within the edit region to obtain binary masks for both the input and output images. We then extract high-frequency texture maps by subtracting a Gaussian-blurred version from the masked crop:$H{=}I_{hair}{-}(G_\sigma{*}I_{hair})$, where I_{hair} denotes the isolated hair region and G_σ is a Gaussian kernel with standard deviation σ. The absolute difference between the paired high-frequency maps is finally computed to measure the microscopic structural consistency of hair strands.
- **Body Appearance**: Specifically evaluates clothing and pose consistency by isolating the human torso and limbs. A selfie segmenter is first applied to obtain the full human silhouette and remove background interference. The previously detected hair mask and facial bounding box are then subtracted to produce a mask containing only the body region. Dense patch embeddings are extracted within this mask using the DINOv3 and spatially averaged to form a unified semantic representation. The cosine similarity between the averaged embeddings of the input and output images is finally computed to measure body fidelity.
- **BG Face ID similarity**: Specifically targets attribute leakage during human-centric edits, where instructions may spill over to non-target background individuals and degrade global consistency. Face detectors are first applied within the non-edit region to extract the background faces' bounding boxes in both the input and output images. Intersection over Union (IoU) is used to establish correspondences between these paired faces. For each matched pair, identity embeddings are extracted using the ArcFace network, and cosine similarity is computed. The final score is obtained by averaging the similarities across all background faces to measure identity preservation for non-target individuals.
- **max match Face ID**: Specifically targets the unconstrained character reference task, where large spatial transformations between the input reference and the synthesized output make positional tracking unreliable. We first extract the target subject's identity embedding by the ArcFace network from the edit region of the input image. Face detectors are then applied to the output image to obtain identity embeddings for all generated faces. Cosine similarity is computed between the reference embedding and each candidate, and the maximum score is returned. This global matching strategy enables reliable identity verification under large positional or scale variations.

C.2 VLM-AS-A-JUDGE PIPELINE OF PREFERENCE CONSTRUCTION FOR GLOBAL EDITING TASKS

For global editing tasks where localized masking and region-specific metrics are inapplicable, we shift from bottom-up pixel-wise measures to top-down semantic reasoning. Specifically, for each candidate pair, we use Gemini 3 Pro to perform pairwise consistency assessments. However, for each $(I_{in}, Inst)$, evaluating the full set of $\binom{7}{2}$ possible candidate pairs involves prohibitive annotative costs and significant computational overhead. To balance *in-group pair diversity* and *efficiency*, we

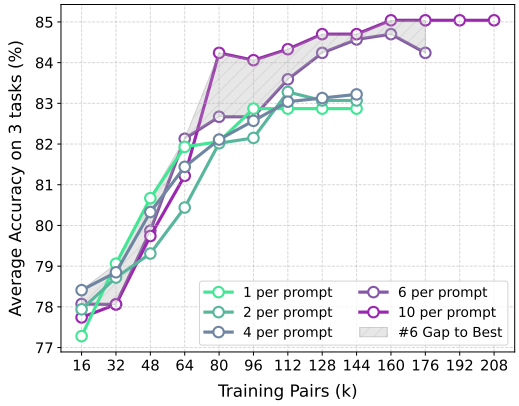


Figure 15: **Average accuracy on three tasks of different in-group pairs** (P) for visual consistency. P=6, chosen as the optimal balance between supervision density and computation cost.

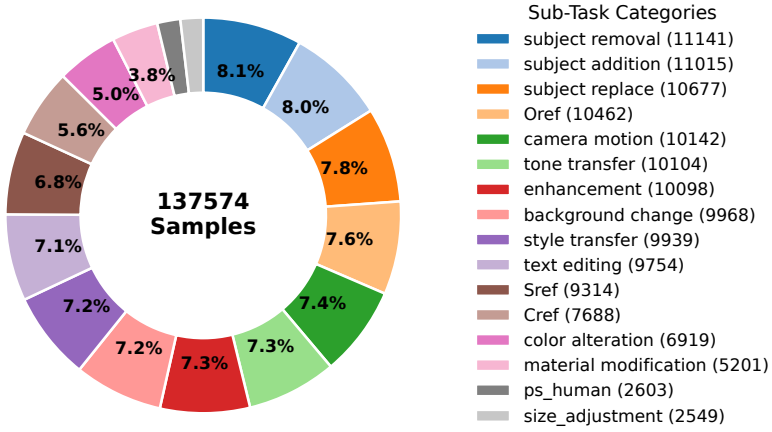


Figure 16: **Distribution of the preference dataset.** The donut chart shows the detailed allocation across 16 editing tasks, providing reliable signals for training the PVC-Judge model.

conduct an ablation study on three representative global tasks: Background Change, Style Transfer, and Tone Transfer. Fixing the sample size at $N{=}1,500$, we vary the number of sampled in-group pairs P within $\{1, 2, 4, 6, 10\}$. The evolution of average accuracy on EditReward-Bench (Luo et al., 2026) relative to training steps is shown in Fig. 15. We find that performance remains stagnant for $P{\leq}\ 4$ but exhibits a substantial leap at $P{=}6$, beyond which marginal gains become negligible. We attribute this improvement to the enriched intra-group diversity, which potentially provides more fine-grained preference signals conducive to robust discriminative learning. Thus, we fix P to 6 as the optimal trade-off between supervisorial density and computational cost.

C.3 Preference Data Distribution

The statistical distribution of the constructed preference dataset is shown in Fig. 16. This balanced distribution provides robust and high-quality preference signals for training our PVC-Judge model.

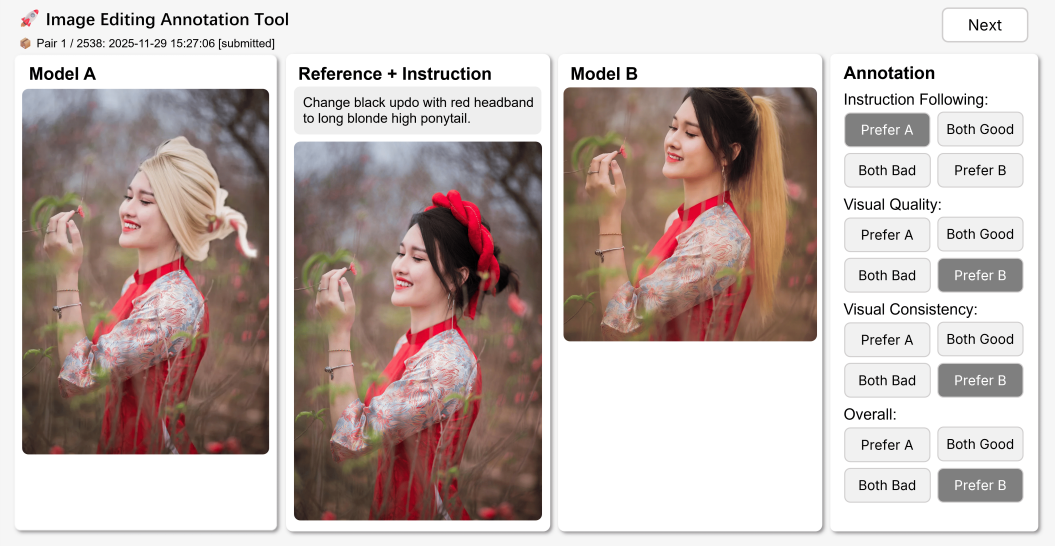


Figure 17: **Screenshots of our custom-built annotation interface for VCReward-Bench.**

C.4 Training Details

PVC-Judge is obtained by fine-tuning Qwen3-VL-8B-Instruct (Bai et al., 2025) with LoRA (Hu et al., 2022) on the preference dataset constructed in Sec. 4.2. The model is trained with the standard next-token prediction loss using the prompt template shown in Fig. 24. During inference, PVC-Judge directly outputs a JSON response, "{"winner": A/B}", without intermediate reasoning steps.

We adopt the AdamW optimizer with a learning rate of 2.0×10^{-6}, together with a cosine learning rate scheduler and a warmup ratio of 0.05. The model is trained for 3 epochs using an effective batch size of 16, achieved via a per-device batch size of 2 across 8 NVIDIA L40S GPUs. For parameter-efficient adaptation, we set the LoRA rank (r) to 64, balancing adaptation capacity and training efficiency. Table 5 summarizes the hyper-parameters for the LoRA model trained during our experiments.

Table 5: **LoRA Training Parameters.**

Hyper-parameter	**Value**
optimizer	AdamW
num_train_epochs	3
per_device_train_batch_size	2
gradient_accumulation_steps	1
learning_rate	2.0e-6
warmup_ratio	0.05
lr_scheduler_type	cosine
weight_decay	0.1
lora_rank	64
lora_alpha	128
lora_dropout	0.05
lora_namespan_exclude	['lm_head', 'embed_tokens']
image_min_pixels	$256 * 32 * 32$
image_max_pixels	$1280 * 32 * 32$

D Annotation Protocol for VCReward-Bench

To guarantee the reliability, diversity, and challenge level of VCReward-Bench, we designed a strict multi-model generation and filtering pipeline. Initially, to capture a comprehensive distribution

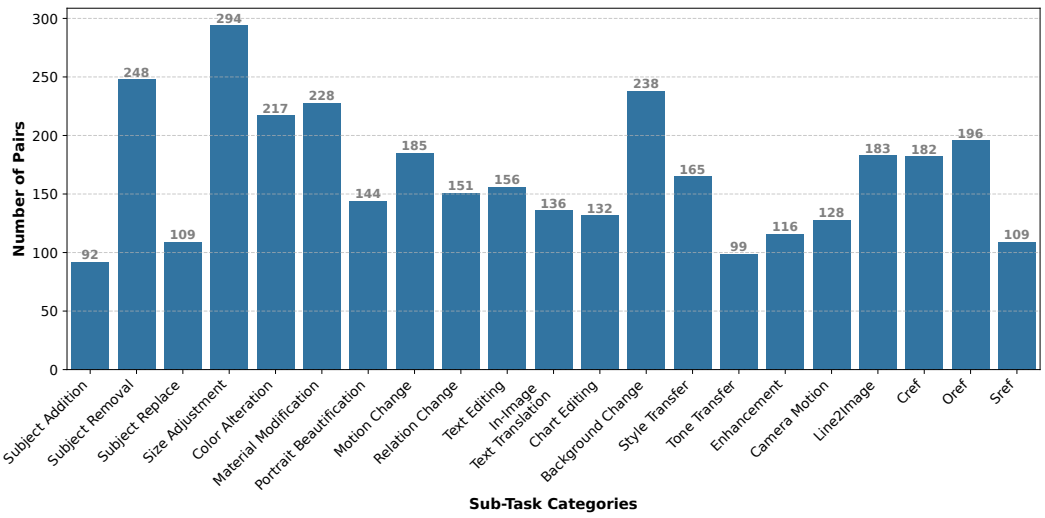


Figure 18: **Distribution of VCReward-Bench instances by tasks.**

of editing behaviors, artifacts, and failure modes, we utilized a diverse ensemble of both leading open-source (Yin et al., 2026; Wu et al., 2025a; Labs et al., 2025) and proprietary models (OpenAI, 2025; Team et al., 2023) to generate a vast pool of candidate images for each editing prompt.

These generated candidates were then formulated into pairwise comparisons. Expert annotators evaluated each pair across three orthogonal dimensions and one global preference: *Instruction Following* (IF), *Visual Quality* (VQ), *Visual Consistency* (VC), and *Overall*. The main interface is presented in Fig. 17. This interactive interface independently isolates the decision process for every single evaluation dimension, ensuring a rigorous multi-dimensional assessment paradigm. Furthermore, the system restricts the selection space to four options: "Prefer A", "Both Good", "Both Bad", and "Prefer B". The inclusion of explicit tie options helps absorb inevitable subjective variation among annotators, improving the statistical robustness of the evaluation.

For examining a model's ability to assess visual consistency rather than detect trivial prompt mismatches, we adopt a Pareto-style filtering strategy over multiple evaluation dimensions. Let $\succ_d$ denote the preference relation under evaluation dimension d. For two edited images I_A and I_B, the relation $I_A \succ_d I_B$ indicates that I_A is preferred to I_B with respect to dimension d. A candidate pair (I_A, I_B) is included in VCReward-Bench only if the following condition holds:

$$I_A \succ_{\mathrm{VC}} I_B, \quad I_A \succeq_d I_B, \ \forall d \in \{\mathrm{IF}, \mathrm{VQ}, \mathrm{Overall}\}.$$

That is, I_A must be strictly preferred to I_B on the visual consistency dimension, while being non-inferior across all remaining evaluation dimensions. This filtering procedure removes trivial pairs and ensures that retained pairs primarily differ in visual consistency, providing more informative signals for meta-evaluation.

Finally, this process resulted in a large-scale, high-quality dataset of 3,508 testing preference pairs spanning 21 tasks, with the detailed distribution shown in Fig. 18.

E Ranking Methodology and Additional Results

E.1 Ranking Methodology

For ranking methodology, rather than relying on simple win-rates, following (Zheng et al., 2023), we first utilize the Bradley-Terry (BT) model (Bradley & Terry, 1952) to estimate the underlying capability score for each editing model based on the aggregated win/tie/loss matrix. These capability scores are then transformed into standard Elo ratings to provide an intuitive and globally comparable measure of relative performance. Furthermore, to account for statistical variance and sampling noise, we compute the 95% Confidence Intervals (CI) for each model's ELO rating via 1,000 bootstrapping

iterations over the evaluation samples, ensuring that our final rankings reflect statistically significant performance differences. We construct the Overall metric by fitting a shared BT model on the aggregated pairwise comparisons across all evaluation dimensions.

E.2 Full Numerical Results of Meta-Evaluation

Tables 6 and 7 report the numerical meta-evaluation performance of various assessment models on the visual consistency dimension for EditReward-Bench (Luo et al., 2026) and VCReward-Bench. The best-performing open-source model is highlighted in bold for easy identification. Our proposed PVC-Judge consistently leads the open-source models, achieving state-of-the-art results across most complex tasks. Moreover, it demonstrates assessment capabilities comparable to large closed-source proprietary models.

Table 6: **Numerical results of assessment models for visual consistency on EditReward-Bench** (Luo et al., 2026). **Bold** marks the best-performing open-source model.

Sub-Tasks	Gemini 3 Pro	GPT 5.1	EditReward (Wu et al., 2025c)	EditScore Avg@4 (Luo et al., 2026)	Qwen3-VL-8B-Instruct	PVC-Judge
Subject Addition	80.88	80.88	74.38	57.65	72.06	**77.94**
Subject Removal	83.56	63.01	**89.90**	51.91	61.64	67.12
Subject Replace	90.57	77.36	81.55	54.85	73.58	**84.91**
Color Alteration	91.30	86.96	83.49	77.83	72.46	**94.20**
Material Modification	90.16	88.52	67.00	51.23	75.41	**83.61**
Style Change	88.89	87.50	74.39	61.38	76.39	**86.11**
Tone Transfer	86.67	83.33	72.61	55.65	78.33	**80.00**
Background Change	92.00	83.00	77.15	63.25	74.00	**87.00**
Extract	78.95	50.88	**93.66**	76.06	56.14	71.93
Portrait Beautification	94.12	89.71	69.02	29.41	85.29	**89.71**
Text Modification	84.93	75.34	84.38	51.56	65.75	**87.67**
Motion Change	87.84	75.68	75.83	58.75	74.32	**82.43**
Hybrid	85.48	80.65	75.00	62.26	74.19	**79.03**
Avg. Accuracy	87.33	78.68	78.34	57.83	72.27	**82.44**

Table 7: **Numerical results of assessment models for visual consistency on our VCReward-Bench. Bold** marks the best-performing open-source model.

Sub-Tasks	Gemini 3 Pro	GPT 5.1	EditReward (Wu et al., 2025c)	EditScore Avg@4 (Luo et al., 2026)	Qwen3-VL-8B-Instruct	PVC-Judge
Subject Addition	93.59	80.68	67.39	41.30	72.53	**86.96**
Subject Removal	86.32	68.49	70.97	35.48	69.01	**74.19**
Subject Replace	85.88	67.96	65.14	36.70	74.00	**82.57**
Size Adjustment	83.52	69.45	57.34	34.81	68.22	**72.35**
Color Alteration	92.90	75.96	75.46	53.24	67.01	**87.04**
Material Modification	86.91	80.09	57.02	50.88	72.89	**86.84**
Portrait Beautification	81.42	70.99	59.03	34.03	**76.23**	75.69
Motion Change	85.06	79.89	57.84	37.84	72.93	**84.32**
Relation Change	85.94	72.97	81.46	50.99	71.33	**82.78**
Text Editing	87.83	79.33	76.92	50.00	77.87	**78.21**
In-Image Text Translation	96.72	83.97	**90.44**	58.82	83.09	85.29
Chart Editing	94.66	92.25	**92.42**	82.58	87.60	89.39
Background Change	87.07	80.08	53.36	50.84	72.69	**86.13**
Style Transfer	84.08	82.91	56.36	41.21	73.62	**84.24**
Tone Transfer	86.60	85.57	49.49	65.66	78.79	**86.87**
Enhancement	85.05	72.41	68.97	57.76	69.83	**80.17**
Camera Motion	88.28	82.54	60.94	57.03	76.56	**79.69**
Line2Image	76.27	68.85	56.83	31.69	51.37	**66.12**
Cref	81.01	65.68	63.19	48.35	68.68	**80.22**
Oref	89.42	77.86	67.86	56.12	68.97	**81.12**
Sref	91.18	76.74	87.16	57.80	81.31	**88.07**
Avg. Accuracy	87.13	76.89	67.41	49.20	73.07	**81.82**

E.3 Qualitative Analysis

Open-Set Task. Fig. 19 comprehensively visualizes diverse cases across six representative editing models. One could observe that current open-source models exhibit difficulty in interpreting implicit user intentions, often overlooking subtle textual constraints contained in unpredictable prompts. These qualitative observations highlight a key limitation: models trained on predefined categories may not generalize reliably to open-set instructions. Furthermore, errors frequently manifest as missing or

misaligned edits, inconsistent object attributes, or partial adherence to the user prompt, suggesting that current models still struggle to robustly fuse complex textual cues with visual generation. This analysis underscores the importance of evaluating visual editing performance beyond narrowly predefined tasks and demonstrates the utility of open-set benchmarks in revealing real-world limitations of generative editing models.

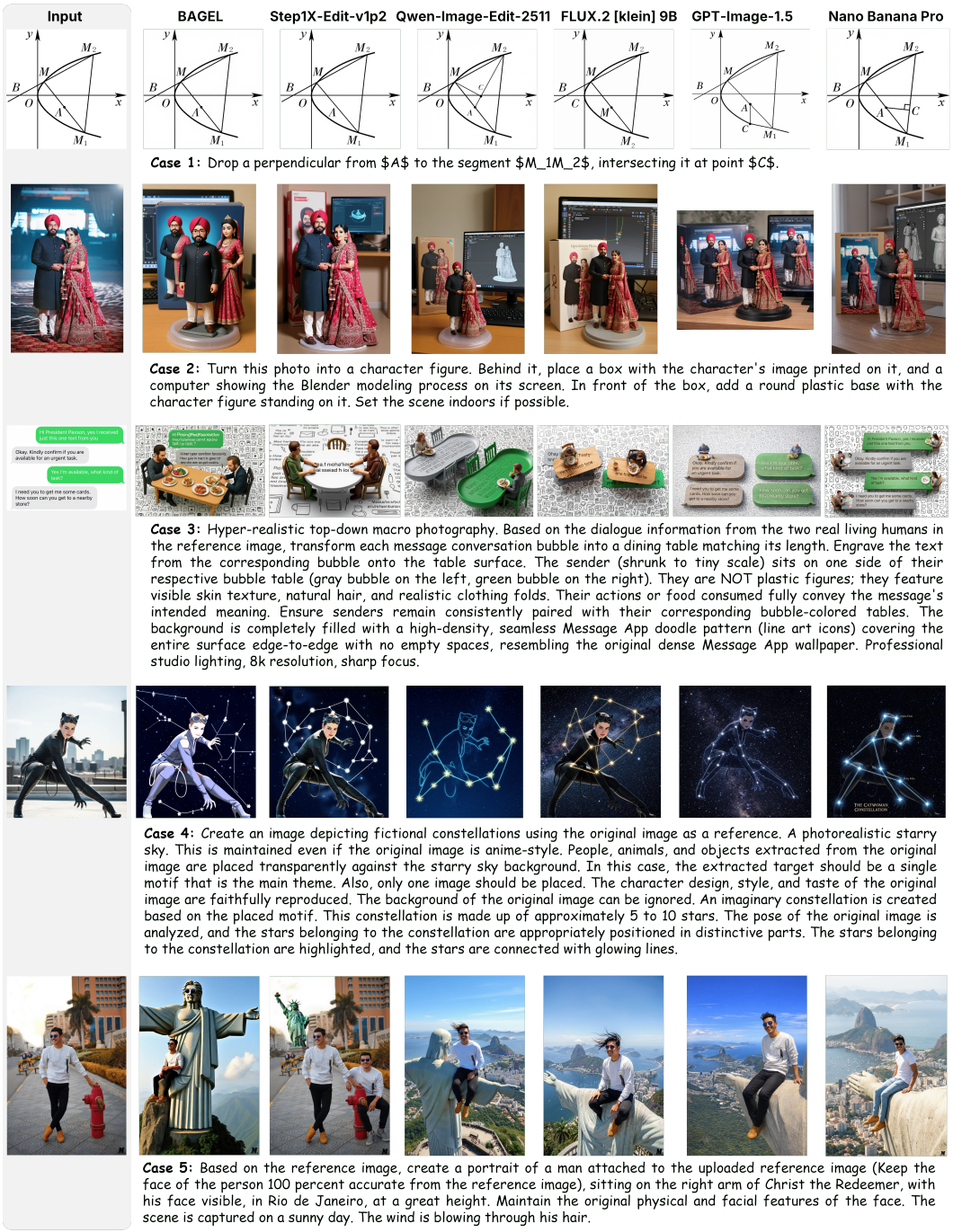


Figure 19: **Open-set editing examples across six representative models.** Many open-source models fail to fully capture implicit user instructions, leading to partial or inconsistent edits. This highlights the need for open-set evaluation to reveal limitations of current generative editing models.

Weak Perception of Inter-Object Relations. Fig. 20 presents generative results of representative models on the relation change task. Visual comparisons reveal clear gaps between current open-source models and stronger closed-source counterparts. Open-source models often struggle to accurately capture spatial relationships between objects, reducing complex relational edits to simple operations such as disjointed object addition or removal. In some cases, models partially or entirely ignore the provided spatial instructions, highlighting limitations in handling structured inter-object dependencies.

Struggle with Small Faces. Fig. 21 shows the results of editing models in generating fine-grained details. We observe that open-source models often produce distorted structures on small or background subjects, with facial regions being particularly challenging. Closed-source models alleviate these structural issues but struggle to maintain identity consistency on small faces (e.g., GPT-Image-1.5). These observations reveal a persistent gap in detail fidelity, which remains a key challenge for the practical deployment of generative editing models.



Figure 20: **Visualization results of relation change task.** Open-source models often struggle to capture complex spatial relationships, leading to partial or simplified relational edits, while closed-source models better preserve inter-object dependencies.

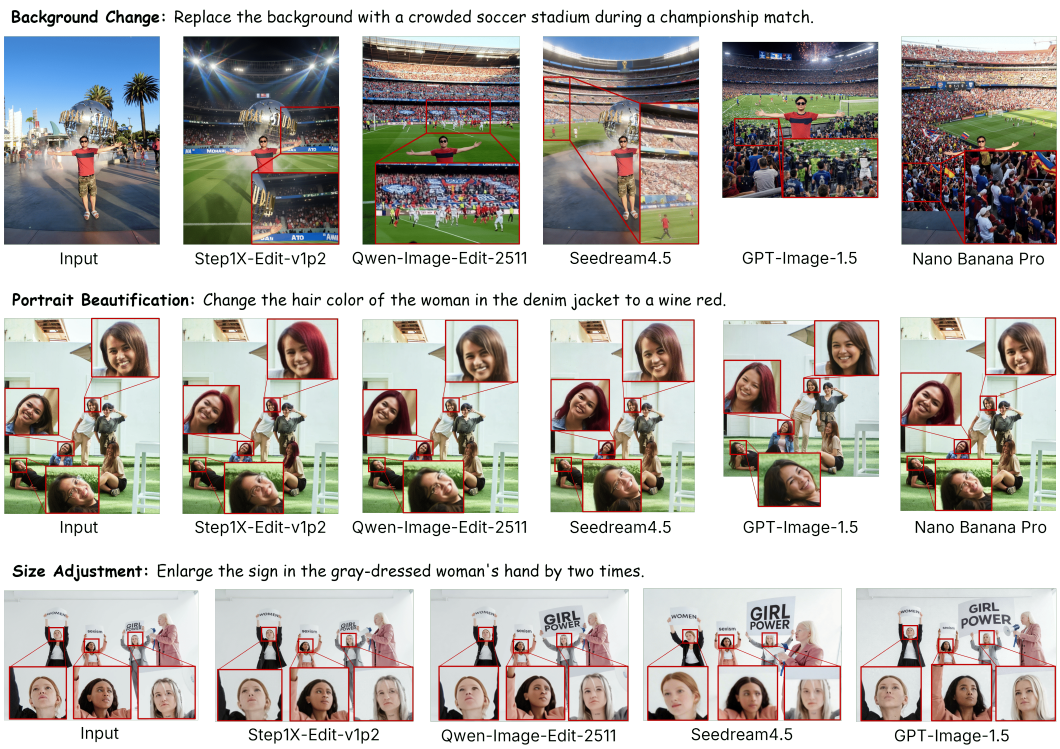


Figure 21: **Fine-grained detail generation across multiple tasks.** Open-source models frequently distort small entities and background subjects, whereas closed-source models better maintain structure, with some difficulty preserving identity on small faces.

Dimension I: Instruction Following

[SYSTEM PROMPT]

You are a meticulous instruction compliance analyst. Your sole purpose is to evaluate images based on how strictly they adhere to a given text instruction. You must be objective and focus exclusively on the fulfillment of the user's command. Output your answer in a structured JSON format.

[USER PROMPT]

Task: Evaluate Instruction Adherence ONLY.
You must determine which of the two generated images, Image A or Image B, more accurately and completely follows the editing instruction, based on the original image.

Your Single Focus:
- Did the image edit the correct object or region?
- Did the image apply the correct style, color, or content as requested?
- Were all parts of the instruction addressed?

Your Thought Process:
1. Analyze Image A's adherence to the instruction.
2. Analyze Image B's adherence to the instruction.
3. Compare them and declare a winner based *only* on adherence.
4. Provide a winner in the specified JSON format.

Required Output JSON Schema:
```json
{
   "winner": " Choose one from: 'Image A', 'Image B', or 'Tie' "
}
```

Editing Instruction: "<instruction>"
Original Image: <image_1>

--- Generated Images to Compare ---
Image A: <image_2>
Image B: <image_3>

Figure 22: The pairwise evaluation prompt for the **Instruction Following** dimension.

Dimension II: Visual Quality

[SYSTEM PROMPT]

You are a strict and highly critical image quality comparator. Your sole responsibility is to determine which image has better intrinsic visual quality. You must evaluate only what is visually observable in the two images.

Your judgment must be based exclusively on:
1) Objective visual defects and logical plausibility.
2) Visual aesthetics and artistic quality.

You must prioritize the absence of defects and logical errors over artistic appeal. Be decisive. Be critical. Avoid being overly generous.

Output only the required JSON. Do not output explanations.

[USER PROMPT]

Task: Decide which image has better overall visual quality.
You are given two generated images:
- Image A
- Image B
You must compare the two images purely based on their intrinsic quality.

Evaluation Dimension I: Artifacts & Logical Flaws (Higher Priority)

Carefully inspect both images for objective defects:
• Anatomical abnormalities (extra fingers, distorted limbs, fused body parts)
• Physically impossible structures
• Broken shadows or inconsistent lighting direction
• Incorrect reflections in mirrors or water
• Perspective inconsistencies
• Floating or unsupported objects
• Garbled or unreadable text
• Background warping or structural distortion
• Texture melting or unnatural blending
• Over-smoothing (plastic-like skin) or unnatural high-frequency noise
• Local blur patches or incoherent fine details

If one image contains more severe or more frequent defects, the cleaner image should win.

Major structural or logical flaws outweigh aesthetic advantages.

Evaluation Dimension II: Visual Aesthetics & Artistry

If both images are similarly free of major defects, compare their artistic and perceptual quality:
• Composition balance and clarity of subject
• Visual hierarchy and focal emphasis
• Color harmony and tonal coherence
• Lighting depth and dimensionality
• Realism of materials and textures
• Sharpness and clarity of details

Decision Rule
---
1. Prefer the image with fewer and less severe defects.
2. If both are similarly clean, prefer the image with superior aesthetics.
3. If both images are extremely similar in overall quality, return "Tie".

Required Output JSON

```json
{
  "winner": " Choose one from: 'Image A', 'Image B', or 'Tie' "
}
```

--- Generated Images to Compare ---
Image A: <image_1>
Image B: <image_2>

Figure 23: The pairwise evaluation prompt for the **Visual Quality** dimension.

Dimension III: Visual Consistency

[SYSTEM PROMPT]

You are a meticulous Image Integrity Analyst. Your function is to detect unintended modifications by comparing a generated image to its original source. Your sole purpose is to assess whether regions *outside the explicit scope of the editing instruction* have been unintentionally altered. You must first deduce the intended target of the edit, then ignore that target area and focus exclusively on the preservation of all other regions. Output your answer in a structured JSON format.

[USER PROMPT]

Task: Evaluate Preservation of Non-Target Regions ONLY.

You will be given an **Original Image**, an **Editing Instruction**, and two generated images: **Image A** and **Image B**. Your job is to determine which generated image did a better job of keeping the areas *not targeted by the instruction* identical to the **Original Image**.

Your Single Focus:
- **Scope Analysis:** First, what is the specific object, area, or concept the instruction intended to change?
- **Collateral Impact on Style:** In areas outside the instruction's direct scope, did the overall artistic style, mood, or filter change compared to the original?
- **Color & Lighting Spillover:** Did the color grading, saturation, or lighting shift in regions that should not have been affected by the edit? Compare them strictly against the Original Image.
- **Integrity of Untouched Elements:** Were any objects, patterns, or structures that were *not supposed to be edited* distorted, added, or removed?
- **Texture & Detail Fidelity:** In untouched parts of the image, was the original texture, sharpness, and fine detail preserved, or did it become blurry, artificially smooth, or noisy?

Your Thought Process:
1. **Deconstruct the Instruction:** First, analyze the editing instruction to clearly define the intended target area(s) of the modification.
2. **Analyze Image A:** Compare the regions *outside* of this target area in Image A against the Original Image, looking for any unintended changes.
3. **Analyze Image B:** Compare the regions *outside* of this target area in Image B against the Original Image, looking for any unintended changes.
4. **Declare a Winner:** Decide which image better preserved the non-target regions with higher fidelity to the Original Image. The image with fewer or less severe unintended changes wins. Provide your winner in the specified JSON format.

Required Output JSON Schema:
```json
{
  "winner": " Choose one from: 'Image A', 'Image B', or 'Tie' "
}
```

Editing Instruction: "<instruction>"

Original Image: <image_1>

--- Generated Images to Compare ---
Image A: <image_2>
Image B: <image_3>

Figure 24: The pairwise evaluation prompt for the **Visual Consistency** dimension.