CV-ARENA: AN OPEN BENCHMARK FOR INSTRUCTIONAL COMPUTER VISION PROBLEM SOLVING WITH HUMAN-AI COLLABORATIVE PREFERENCES

Anonymous authors
Paper under double-blind review

ABSTRACT

Instruction-guided image editing is becoming a general interface for visual work, yet existing benchmarks provide uneven coverage of the diverse real-image tasks encountered in professional workflows. Here, we define instructional computer vision problem solving as a broader formulation of image editing: given a real input image and a natural-language instruction, a system must produce an edited output that realizes the requested transformation while satisfying explicit preservation, geometric, physical, and usability constraints. We introduce CV-Arena, an open benchmark designed to evaluate this capability in professional editing tasks. CV-Arena contains 12K high-resolution real-image instruction pairs spanning 16 instruction-based visual task types, constructed using CogRetriever, a dual-track retrieval-and-curation pipeline that combines targeted web search, agentic query refinement, verification, and traceability. To evaluate models at scale with selective expert supervision, we propose Active Elo, a human-AI collaborative preference protocol that leverages CV-Judge, a logic-gated, multi-dimensional VLM evaluator, to identify clear failures and resolve high-confidence comparisons, while routing close, high-quality comparisons to expert raters. Mixed human and AI supervision is then aggregated through reliability-weighted Elo updates. Our comprehensive evaluation of 21 systems, including proprietary, open-source, and agentic models, on CV-Arena reveals persistent gaps in instruction adherence, physical reasoning, structural control, and fine-grained detail preservation. We further develop CV-Agent, a simple agentic baseline that combines planning, editing, and verification. Our results suggest that closed-loop reasoning is a promising direction for professional-grade instruction-following visual editing.

1 INTRODUCTION

A long-standing problem in modern computer vision (CV) is to *modify an image according to human intent*. Instruction-guided image editing offers a natural interface for this goal: given an image and a natural-language instruction, a system is expected to change only what is requested while preserving the rest Zhang et al. (2023); Ye et al. (2025). Recent multimodal generative models, including vision-language models (VLMs) and unified generative models Wang et al. (2024); Zhang et al. (2024a); Guo et al. (2024); Wang et al. (2025c); Lu et al. (2024), have made this interface increasingly practical. Both proprietary systems OpenAI (2025b); Comanici et al. (2025) and open models Lin et al. (2025a); Ye et al. (2025); Wu et al. (2025a) are now being used for everyday editing and increasingly complex visual workflows Meta (2025); OpenAI (2025a); Lin et al. (2025b).

Existing work on instruction-guided image editing spans a broad range of transformations Hui et al. (2024); Zhao et al. (2024a); Mao et al. (2025). However, current benchmarks provide limited unified evaluation of professional tasks that require multiple constraints to be satisfied simultaneously on high-resolution real images Ye et al. (2025); Zhang et al. (2023); Qian et al. (2025); Chang et al. (2025a); Ku et al. (2023); Peng et al. (2024). We therefore define **instructional computer vision problem solving (iCVPS)** as a broader formulation of image editing that emphasizes realizing the requested transformation while jointly satisfying preservation, geometric, physical, and usability constraints. For instance, an instruction may require a system to restore degraded content, enhance low-light or hazy images, recover faded text, manipulate object pose or geometry, insert objects with

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.

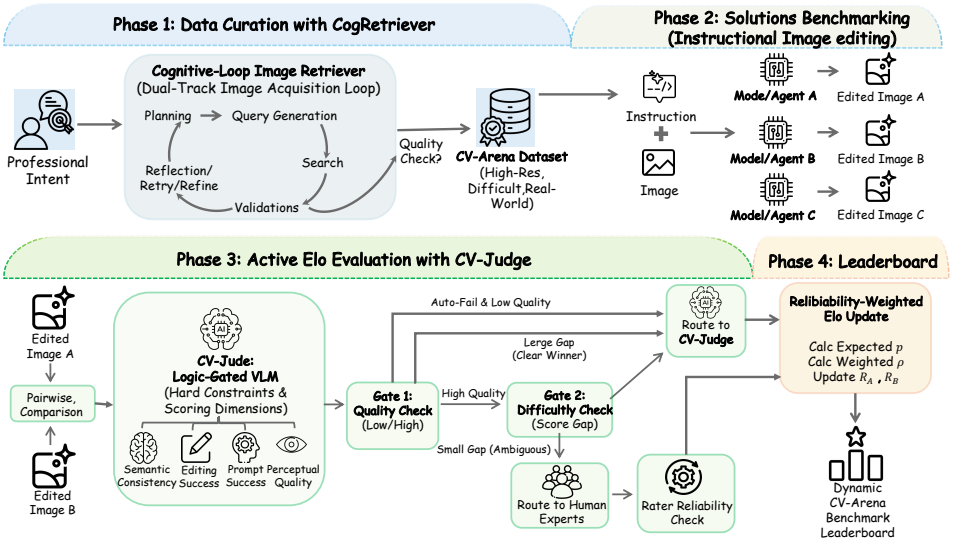

Figure 1: **The Overall Pipeline.** CogRetriever retrieves images with at most two retrieval iterations, routing cases with insufficient candidate pools to manual review. The curated images form CV-Arena, on which editing models and agents are evaluated. CV-Judge scores the edited outputs, and Active Elo applies a two-gate routing policy to send close, high-quality comparisons to human experts. Human and automatic preferences are then aggregated through reliability-weighted Elo updates.

physically consistent lighting, or perform structure-preserving outpainting Luo et al. (2025); Janjua et al. (2026); Wang et al. (2025a); Zhou et al. (2025); Gu et al. (2025b); Zhang et al. (2026); Cao et al. (2025); Song et al. (2026); Jia et al. (2025); Zhang et al. (2025b). This broader view exposes a professional gap that is not well captured by existing editing benchmarks. First, many current systems can produce visually plausible images but often re-synthesize the input rather than faithfully modifying it, leading to unintended content changes and constraint violations Wang et al. (2025b). Second, existing evaluation protocols can be unreliable for subtle, high-resolution comparisons, where small local artifacts, text errors, boundary inconsistencies, or geometric mistakes may determine whether an output is usable. Third, coverage of these task families remains uneven across benchmarks, making it difficult to assess performance across diverse professional workflows Naveed et al. (2025); Achiam et al. (2023); Team et al. (2023); PBC.

To close this gap, we introduce **CV-Arena**, an open benchmark that brings these task families together under a common image–instruction interface. Crucially, CV-Arena focuses on high-resolution, in-the-wild images whose content and quality resemble those encountered in real visual workflows, complementing existing editing benchmarks Ye et al. (2025); Zhang et al. (2023); Chang et al. (2025a); Ku et al. (2023); Peng et al. (2024). To construct the dataset at scale, we develop a text-initiated multimodal retrieval pipeline that converts professional editing intents into targeted web search, candidate discovery, verification, and traceable data records. We further combine this agentic acquisition process with manual search and expert curation to address rare, difficult scenarios and reduce redundancy, resulting in the CV-Arena Dataset, which contains 12K open-domain iCVPS image–instruction pairs across diverse high-resolution settings.

A second challenge is *scalable and reliable evaluation* for CV-Arena. Many of these instructions admit multiple valid outputs, motivating pairwise evaluation guided by task-specific criteria. Classical image quality metrics such as PSNR and SSIM Korhonen & You (2012); Wang et al. (2004) ignore instruction adherence and semantic preservation, while embedding-based metrics capture only partial signals for high-fidelity professional edits Radford et al. (2021). Recent benchmarks increasingly adopt VLM-as-a-judge for scalable scoring Ye et al. (2025); Wu et al. (2025b), but VLM judges can be brittle on subtle or near-tied comparisons, especially when correctness depends on fine local details. Arena-style human preference evaluation Chiang et al. (2024) is more faithful, but costly, difficult to scale, and vulnerable to low-quality voting in crowdsourced settings Zhao et al. (2025); Jiang et al. (2024). To address these limitations, we propose **Active Elo**, a human-AI collaborative preference protocol that combines automated judging with selective expert supervision. Our proposed **CV-Judge** first performs logic-gated, multi-dimensional evaluation to identify clear failures and high-confidence preferences; Active Elo then routes close, high-quality comparisons to expert raters

Table 1: **Comparison of Existing iCVPS Datasets and Benchmarks.** #Size and #Tasks represent the reported resource size and number of editing types; resource sizes include training corpora. Max Res. denotes the maximum resolution. We also mark important attributes such as Real Image, Physics, Reasoning, Low Level, and Complex to compare the dataset diversity. The last column lists the reported evaluation metrics and protocols.

Dataset	#Size	#Tasks	Max Res. (px)↑	Real Image	Physics	Reasoning	Low Level	Complex	Metrics
MagicBrush Zhang et al. (2023)	10K	5	500	✓	✗	✗	✗	✓	L1, L2, CLIP, DINO
InstructPix2Pix Brooks et al. (2022)	313K	4	512	✗	✗	✗	✗	✗	CLIP
HQ-Edit Hui et al. (2024)	197K	6	≥768	✗	✗	✗	✓	✗	GPT
SEED-Data-Edit Ge et al. (2024)	3.7M	6	768	✗	✗	✗	✗	✓	N/A
UltraEdit Zhao et al. (2024a)	4M	9	512	✗	✗	✗	✓	✗	L1, L2, CLIP, DINO
AnyEdit Yu et al. (2025)	2.5M	25	512	✓	✓	✓	✗	✓	L1, CLIP, DINO
ImgEdit Ye et al. (2025)	1.2M	13	≥ 1280	✓	✗	✓	✗	✗	GPT
CV-Arena	**12K**	**16**	**≥ 2048**	✓	✓	✓	✓	✓	**GPT + Human**

and aggregates both human and AI decisions through reliability-weighted Elo updates. This design focuses expert supervision on close, high-quality comparisons while retaining automatic judgments for the remaining pairs. A budget-controlled routing study examines human agreement and ranking stability under this protocol (Section 4.2).

Beyond benchmarking existing editors, we also study whether agentic reasoning can improve iCVPS. To this end, we build **CV-Agent**, a simple agentic baseline that decouples high-level instruction understanding, planning, and verification from low-level image manipulation. The agent uses a strong editor as a tool, and wraps it with a closed-loop reasoning process that refines the edit and checks whether the output satisfies the instruction and constraints. Using the same base editor, our module ablations show progressively higher Active Elo point estimates as instruction understanding, planning, and closed-loop refinement are added (Appendix O). In summary, our contributions are:

- **CV-Arena,** an open, professional-grade benchmark for iCVPS on real, high-resolution images, covering task families beyond appearance-centric editing while preserving native aspect ratios.
- **Active Elo System,** a scalable human-AI collaborative preference protocol that combines a logic-gated multi-dimensional VLM evaluator with selective expert annotation and reliability-weighted Elo aggregation under constrained human budgets.
- **CV-Agent,** a simple agentic baseline that decouples high-level planning and verification from low-level image manipulation, with same-editor module ablations showing incremental improvements under Active Elo.

2 Related Work

Real-world visual understanding has driven AI progress since MNIST and ImageNet LeCun (1998); Deng et al. (2009); Voulodimos et al. (2018); Szeliski (2022), but while recognition-oriented tasks have advanced substantially Elngar et al. (2021); Minaee et al. (2021); Zou et al. (2023), higher-tier objectives involving image realism, visual naturalness, and the plausibility of edits Theis (2024); Li et al. (2023); Ye et al. (2025); Chen et al. (2025b) remain far from solved. Existing instructional editing benchmarks reflect this gap: object-insertion datasets such as iHarmony4 Cong et al. (2019) and ObjectDrop Kim et al. (2025); Winter et al. (2024) reduce the task to appearance harmonization or static placement, ignoring dynamic interactions with deformable media; semantic editing benchmarks such as MagicBrush Zhang et al. (2023) and RefCOCO-Edit Liu et al. (2024) conflate insertion, replacement, and reconstruction, blurring evaluation signals; and geometry-related edits in MagicBrush, InstructPix2Pix, and AnyEdit Zhang et al. (2023); Brooks et al. (2022); Yu et al. (2025) are entangled with appearance changes, while ImgEdit Ye et al. (2025) includes spatial reasoning and multi-object editing alongside multi-turn evaluation. Typography and UI recovery Qu et al. (2023); Fang et al. (2025) are similarly under-served despite their importance in professional workflows. A more thorough discussion of each line of work is provided in Appendix A.

3 CV-Arena Dataset

Our dataset consists of 12K image-instruction pairs encompassing 16 task types. By integrating physical interaction and geometric constraints alongside traditional restoration tasks, CV-Arena covers tasks ranging from low-level restoration to high-level structural manipulation. The construction follows a definition-driven pipeline (Figure 1): (i) defining instructional CV problem solving and

deriving image selection criteria, (ii) designing a task taxonomy reflecting professional editing intents, and (iii) retrieving, filtering, and verifying real-world images for task suitability, quality, and traceability. Comparisons against concurrent datasets are summarized in Table 1.

3.1 Problem Definition

We formulate **instructional computer vision problem solving** (iCVPS) as a generalization of instruction-guided image editing. Given a real input image x and a natural-language instruction I, a system must produce an edited output $\hat{x} = \mathrm{Edit}(x, I; m)$ that realizes the requested transformation while preserving everything that should remain unchanged.

This formulation introduces a set of professional constraints that go beyond perceptual realism: the output should additionally satisfy instruction adherence, semantic preservation, physical plausibility, geometric consistency, and high-resolution usability Luo et al. (2025); Janjua et al. (2026); Wang et al. (2025a); Zhou et al. (2025); Gu et al. (2025b); Zhang et al. (2026); Cao et al. (2025); Song et al. (2026); Jia et al. (2025); Zhang et al. (2025b). These constraints in turn guide the image selection criteria: each pair must contain sufficient visual evidence for the task, a visible and unambiguous target region, and a clear success condition. We intentionally retain difficult real-world conditions (complex lighting, cluttered scenes, fine local structures, non-canonical viewpoints) so long as the source remains visually interpretable and the task intent unambiguous.

3.2 Task Design and Taxonomy

The task taxonomy guides image collection according to professional editing intents. Beyond classical restoration tasks (exposure correction, deblurring, super-resolution) curated to reflect realistic difficulty, CV-Arena deliberately incorporates underrepresented task families critical to professional workflows: physically grounded scene composition, semantic-aware content manipulation, geometry-driven structural transformation, and typography or UI restoration in natural images. Figure 2(a, b) summarizes the task composition and image-resolution distribution. Three task families particularly distinguish CV-Arena from prior benchmarks: *Scene Composition and Object Insertion* requires physically and semantically coherent integration across geometry, lighting, scale, and semantics; *Semantic-Aware Content Instruction* modifies intrinsic properties (pose, functional state, spatial configuration) without introducing or removing entities; and *Text-Based Geometric Warping and Structural Control* performs precise, logically consistent shape transformations driven purely by language, including pose changes, viewpoint shifts, and fine-grained expression mixtures. Detailed definitions of these three task families are provided in Appendix B.

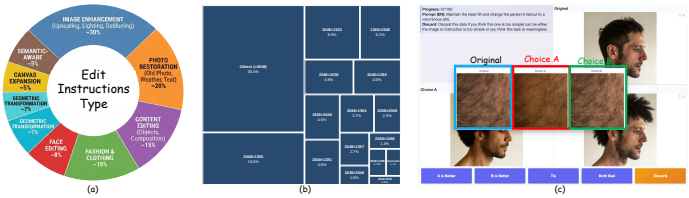

Figure 2: **Dataset Statistics and Human Evaluation Interface.** From left to right: (a) task composition; (b) image-resolution distribution; and (c) the *zoom-in* interface used to inspect local details during human evaluation.

3.3 Data Acquisition, Filtering, and Human Verification

To collect images satisfying the above criteria at scale, we develop **CogRetriever**, a text-initiated multimodal search pipeline with a dual-track strategy. The *Base Track* uses manual keyword search for high-precision acquisition in straightforward scenarios; the *Agentic Track* provides scalable coverage for complex professional intents through a closed-loop system that maintains a reflection memory m_t over at most $T = 2$ iterations and operates in three stages: ❶ **Planning**, where a planner decomposes the instruction $\mathbf{I_i}$ into a diverse query set $\mathcal{Q} = \{q_1, \ldots, q_K\}$ ($K = 5$); ❷ **Action & Perception**, where the system retrieves the top-N candidates per query ($N = 20$), validates them, and produces dense visual captions; and ❸ **Evaluation & Pool Construction**, where a VLM scores candidates with $s(\mathbf{x}; \mathbf{I_i}) \in [0, 1]$, retains those with $s \geq \tau$ ($\tau = 0.8$), and, if the first iteration yields fewer than $K_p = 3$ qualified candidates, writes a reflection m_{t+1} to refine the queries for the second iteration. Retrieval stops once the pool contains at least K_p qualified candidates; cases still below this target after the second iteration are sent to manual review. For a sufficient pool, the selected

candidate set is $\mathcal{X}^* = \mathrm{TopK}_{\mathbf{x}\in\mathcal{P}_t}\, s(\mathbf{x}; \mathbf{I_i})$, selecting the top-$K_p$ candidates by score. This refinement updates retrieval queries and the candidate pool; CogRetriever does not synthesize or modify the source images. Full algorithmic details and hyperparameters are in Appendix C.

During acquisition, we used the Creative Commons rights filter for the Base Track and `cc_publicdomain` and `cc_attribute` restrictions via the Google Custom Search API for the Agentic Track to screen candidates by their reported licenses. Retrieved images then underwent near-duplicate removal and low-quality filtering. Human curators verified the surviving image-instruction pairs for task-category match, target-region visibility, instruction feasibility, and a consistent success criterion, using the interactive *zoom-in* tool (Figure 2c) for detail-sensitive cases. Pairs failing these checks were repaired or discarded. Search-time license filtering is a candidate-screening step; per-image source/license verification and privacy and harmful-content review are separate release checks. Filtering, provenance records, and the human-verification protocol are described in Appendix D.

4 Evaluation: Active Elo with CV-Judge

We evaluate instructional computer vision problem-solving models through pairwise comparisons under identical input conditions. The bottom of Figure 1 summarizes the overall evaluation pipeline. Our evaluation stack consists of two components: **CV-Judge**, a multi-dimensional evaluator for instructional image editing, and **Active Elo**, a human-AI collaborative ranking framework that routes close, high-quality comparisons to experts and aggregates human and automatic preferences through reliability-weighted updates.

4.1 Preliminaries: Arena and Elo Ranking

Arena-style evaluation uses blinded pairwise preferences to compare open-ended generative systems Chiang et al. (2024). Two outputs produced from the same input image and instruction are compared according to task-specific criteria. An Elo-style system then aggregates these outcomes into a leaderboard: each model m maintains a rating R_m, and pairwise win probabilities are modeled through a Bradley-Terry-Luce link Bradley & Terry (1952). CV-Arena adopts this ranking view while accounting for expert annotation cost and the varying reliability of automatic judgments. The resulting ratings describe relative performance for the evaluated dataset, model pool, and protocol; they are not absolute correctness scores.

4.2 Active Elo with Human-AI Collaboration

Pairwise comparisons. Let $\{(\mathbf{x_i}, \mathbf{I_i})\}_{i=1}^{N}$ denote the image-instruction pairs in CV-Arena. For a model m, the edited output is $\hat{x}_{i,m} = \mathrm{Edit}(\mathbf{x_i}, \mathbf{I_i}; m)$. For any two models (A, B) on the same instance i, we compare their outputs $\hat{x}_{i,A}$ and $\hat{x}_{i,B}$ under identical input conditions. Let $s_{i,m}$ denote the final scalar component of the CV-Judge evaluation:

$$s_{i,m} := \text{CV-Judge}\left(\mathbf{x_i}, \mathbf{I_i}, \hat{x}_{i,m}\right). \tag{1}$$

The automatic preference is $\hat{z}^{\mathrm{AI}}_{i,A,B} = \mathbb{I}[s_{i,A} \geq s_{i,B}]$. The final binary outcome $z_{i,A,B} \in \{0, 1\}$ used by Active Elo is the human preference for a human-routed comparison and $\hat{z}^{\mathrm{AI}}_{i,A,B}$ otherwise. Here, $z_{i,A,B} = 1$ indicates a preference for A. Human-routed comparisons use the same blinded pairwise format; automatic scores do not choose or override their winner.

CV-Judge evaluation. Evaluating instructional image editing requires checking whether an edited image follows the instruction while preserving content that should remain unchanged. Given an input image x, an instruction I, and an edited image $\hat{x}$, CV-Judge returns a scalar score, a binary success flag, and four dimension scores: semantic consistency, editing success, prompt following, and perceptual quality.

Semantic consistency measures whether identities, key objects, and layout not intended to change are preserved. *Editing success* captures whether the core edit specified by the instruction is realized with sufficient strength. *Prompt following* evaluates adherence to detailed instruction constraints, including explicit restrictions. *Perceptual quality* assesses visual realism and usability, penalizing artifacts, unnatural blending, and structural distortions. These dimensions distinguish editing correctness from perceptual appearance.

We denote the four scores as S_{sem}, S_{edit}, S_{prompt}, and S_{perc}. Each is scored on $[0, 1000]$, and the initial overall score is $S_{\text{init}} = \omega_s S_{\text{sem}} + \omega_e S_{\text{edit}} + \omega_i S_{\text{prompt}} + \omega_p S_{\text{perc}}$. We use $(\omega_s, \omega_e, \omega_i, \omega_p) = (0.20, 0.30, 0.30, 0.20)$. Since all dimensions share the same range, these weights encode relative importance, placing greater weight on edit success and instruction following.

CV-Judge also applies failure conditions and score caps. If $S_{\text{edit}} < \omega_e \cdot 1000$, the final score is capped at $(\omega_e + \omega_i) \cdot 1000$ and the output is marked unsuccessful. If $S_{\text{sem}} < \omega_s \cdot 1000$ or $S_{\text{perc}} < \omega_p \cdot 1000$, the score is capped at $(\omega_p + \omega_s) \cdot 1000$ and the output is marked unsuccessful. These failure thresholds are distinct from the upper bounds imposed on the final score. We denote the score after capping by S; for output $\hat{x}_{i,m}$, this is the scalar $s_{i,m}$ used in routing and Elo aggregation.

CV-Judge uses GPT-4o as its backbone VLM. Weight and routing-threshold sensitivity results are reported in Appendix N; backbone choice and its limitations are discussed in Appendix J.

Human-AI routing. VLM judging scales well but can be unreliable on subtle comparisons. We use a two-gate policy to select comparisons for expert annotation. For a pair (A, B) on instance i, define the score gap $g_i(A, B) = |s_{i,A} - s_{i,B}|$. The pair is routed to humans iff

$$\min(s_{i,A}, s_{i,B}) \geq \tau \quad \text{and} \quad g_i(A, B) < \Delta. \tag{2}$$

The *quality gate* requires both outputs to reach the quality threshold. The *ambiguity gate* selects pairs with a small automatic score gap. All remaining pairs use the automatic preference. Appendix G explains the motivation for allocating expert effort to this region.

On the reported human validation comparisons, direct AI-human agreement increases across the score-gap bins, from 56.3% at $g < 50$ to 94.8% at $g \geq 200$ (Appendix K). This supports score gap as a practical indicator of automatic-judgment difficulty. These bin-wise results do not establish the same agreement for every automatically processed comparison. The reported routing statistics show that 33.7% of comparisons are routed to humans, with rates ranging from 46.8% for geometry-driven warping to 26.2% for restoration (Appendix L). These rates describe how human supervision is distributed across task families.

Reliability-weighted Elo update. Each model m maintains a rating R_m, updated after each pairwise comparison. For a match between A and B, we use a Bradley-Terry-Luce probability (Bradley & Terry, 1952): $p_{AB} = \sigma\big((R_A - R_B)/S_{AB}\big)$, where σ is the sigmoid function and $S_{AB} = (s_{i,A} + s_{i,B})/2$ is an instance-dependent scale computed from the two final CV-Judge scores. For human-routed pairs, the human label determines the winner, while these automatic scores still enter the scale S_{AB}.

Let $q \in [0, 1]$ denote the reliability input to the credibility-weighting rule. Given p_{AB} and the final outcome $z_{i,A,B}$, define

$$w = \begin{cases} p_{AB}, & z_{i,A,B} = 1, \\ 1 - p_{AB}, & z_{i,A,B} = 0, \end{cases} \quad \rho = \frac{q\, w}{q\, w + (1-q)\frac{1}{2}}. \tag{3}$$

For human labels, $q \approx 1$ is a modeling assumption. For automatic comparisons, $q = q_{\text{AI}}(g_i(A, B))$ is estimated from comparisons with human preferences, as described in Appendix F. Empirical agreement and a mixture-component probability are distinct quantities; Appendix H discusses the conditional mixture interpretation of the weighting formula.

The ratings are then updated by

$$R_A \leftarrow R_A + K_r\, \rho\, (z_{i,A,B} - p_{AB}), \quad R_B \leftarrow R_B - K_r\, \rho\, (z_{i,A,B} - p_{AB}), \tag{4}$$

where K_r depends on the rater type. Human matches use K_H, while automatic matches use the smaller step size $K_{AI} = \alpha K_H$. The credibility weights and rater-dependent step sizes control the contribution of each comparison to the ranking.

Validation of the routing policy. We compare routing strategies under a matched human annotation budget B_H, with CV-Judge-only evaluation as an automatic reference. Following an LMArena-style blinded pairwise protocol Chiang et al. (2024), the human validation set $\mathcal{H}_{\text{test}}$ includes four selected models, eight curated task categories, and 50 annotators. CV-Agent is not included in this set. The study therefore evaluates the protocol within this setting and does not independently validate CV-Agent's position among all 21 systems.

Table 2 compares two complementary measures: human agreement and bootstrap ranking stability. The two-gate setting has the highest reported values on both. Human agreement (Acc_H) measures how often the final Elo ordering predicts the human pairwise preference Ouyang et al. (2022); Zheng et al. (2023). Bootstrap Spearman correlation (ρ_S) measures the stability of resampled rankings relative to the full-comparison ranking under the same protocol Kendall (1938); Dubois et al. (2024); it is not a correlation with an independent human ranking. Protocol details are provided in Appendix I.

5 CV-AGENT: SIMPLE AGENTIC BASELINE

We introduce **CV-Agent**, a simple agentic baseline for studying whether instruction understanding, planning, and feedback can improve the performance of an off-the-shelf image editor on CV-Arena. Gemini 2.5 Pro Google (2025) serves as the VLM controller for understanding, planning, and internal verification, while nano banana DeepMind (2025a) performs image editing. The modular pipeline follows a ReAct-style loop Yao et al. (2022) without task-specific model fine-tuning and proceeds in three stages.

Table 2: **Validation of the Two-Gate Routing Policy.** Strategies using human labels are compared under a matched annotation budget B_H; CV-Judge only is the automatic reference. Acc_H measures agreement with human preferences, while ρ_S measures bootstrap ranking stability.

Method	$\mathrm{Acc}_H \uparrow$	$\rho_S \uparrow$
Budget-matched human-only	60.7%	0.68
CV-Judge only	51.4%	0.63
Quality-only gate	68.8%	0.81
Ambiguity-only gate	73.2%	0.75
Two-gate (Ours)	**82.6%**	**0.94**

Stage❶: Understanding. Given the input image $\mathbf{x}$ and instruction $\mathbf{I}$, the controller produces a compact task specification describing the requested visual changes and the constraints on content that should remain unchanged. This specification guides the subsequent planning and editing steps.

Stage❷: Planning. The VLM controller generates a structured editing plan from the task specification. It determines whether the request should be addressed in one step or multiple steps and allocates an editing-step budget of at most $T_{\text{agent}} = 3$.

Stage❸: Closed-loop editing. Starting from $\hat{x}_0 = \mathbf{x}$, nano banana edits the current image using a step-specific prompt to produce $\hat{x}_t$. The VLM controller acts as an *internal verifier*: it compares $\hat{x}_t$ with the original image and instruction $(\mathbf{x}, \mathbf{I})$, and returns an internal quality score v_t, a success indicator, and corrective feedback when needed. The loop stops when the internal verifier reports success or the allocated editing-step budget is exhausted. The agent returns the candidate with the highest internal quality score among the outputs generated before stopping.

The internal verifier supports feedback, stopping, and output selection within CV-Agent. The benchmark's GPT-4o-based CV-Judge (Section 4.2) evaluates the submitted final outputs under the evaluation protocol.

Table 3: **CV-Agent Module Ablation.** Active Elo point estimates for configurations using the same nano banana editor.

Configuration	Active Elo
nano banana (single pass)	1056
+ Stage 1 (Understanding)	1089
+ Stage 1 + Stage 2 (Planning)	1118
Full CV-Agent (Stages 1–3, closed-loop)	**1145**

Same-editor module ablation. Table 3 reports configurations that progressively add the three stages while keeping nano banana as the base editor. The Understanding-plus-Planning configuration executes a multi-step plan, while the full configuration adds closed-loop verification and feedback. The Active Elo point estimate increases from 1056 for single-pass editing to 1089 with Understanding, 1118 with Understanding and Planning, and 1145 for the full pipeline. The full configuration is 89 Elo points above the single-pass baseline. These results characterize the combined configurations under the reported Active Elo protocol; interpretation and configuration details are discussed in Appendix O.

6 EXPERIMENTS

We evaluate 21 instructional image editing systems on CV-Arena, including standalone editors and agentic solutions. We report Active Elo (Section 4.2) alongside budget-matched human-only Chiang et al. (2024); Jiang et al. (2024), CV-Judge-only, and EditReward-only Wu et al. (2025b) evaluation. These settings provide complementary views of model ordering under different evaluation protocols. We describe the evaluated systems and setup in Section 6.1, followed by rankings and supplementary diagnostics in Section 6.2.

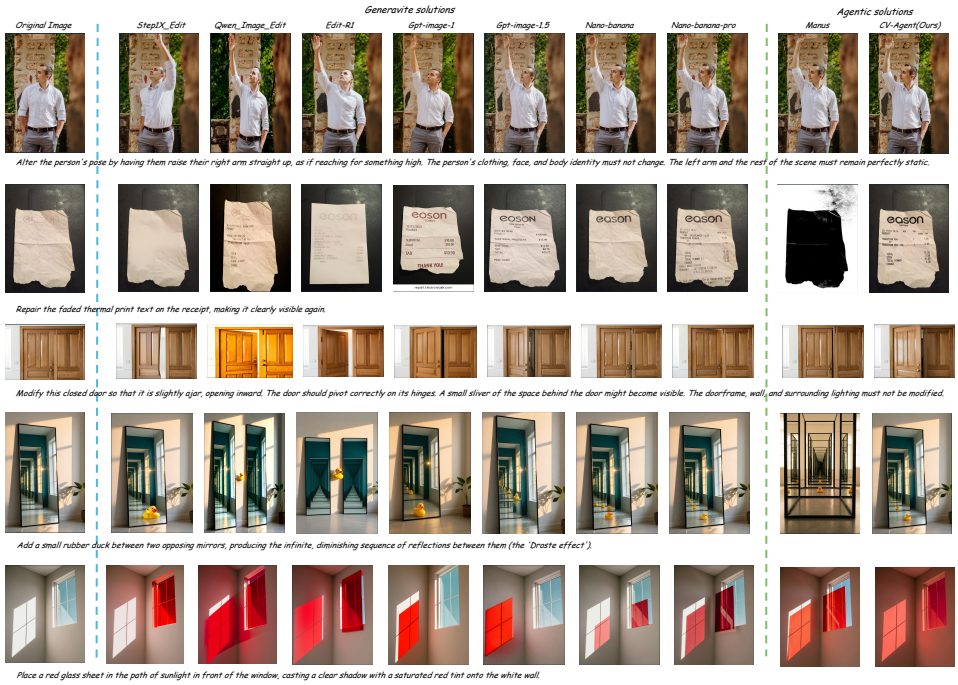

Figure 3: **Selected Qualitative Comparisons.** Outputs from single-pass editors and agentic systems on pose editing, text restoration, geometric manipulation, and physically constrained composition.

6.1 Benchmark Details

Solutions. We evaluate a diverse suite of instructional image editing solutions (*21 solutions* in total). Closed-source systems include gpt-image-1 OpenAI (2025b), gpt-image-1.5 OpenAI (2025c), nano banana DeepMind (2025a), nano banana pro DeepMind (2025b), Flux2 Labs (2025), Seedream 4.5 Seedream et al. (2025), and wan 2.5 preview Wan et al. (2025). Open-source baselines include Edit-R1 Lin et al. (2025a); Ye et al. (2025), VAREdit Mao et al. (2025), ICEdit Zhang et al. (2025a), AnyEdit Yu et al. (2025), Instruct-CLIP Chen et al. (2025a), Step1X-Edit Liu et al. (2025), MagicBrush Zhang et al. (2023), UniWorld Lin et al. (2025a), Qwen Image Edit Wu et al. (2025a), ByteMorph Chang et al. (2025b), and SuperEdit Li et al. (2025); Gu et al. (2025a). We also evaluate three agentic systems: Manus 1.6 Meta (2025), JarvisEvo Lin et al. (2025b), and our CV-Agent.

Inference protocol. Editing is performed at each model's native output resolution. Unless otherwise stated, we retain generated outputs without additional image post-processing. The results describe the evaluated system configurations, including standalone editors and multi-step agents. **Human evaluation interface.** The interface uses uniformly resized renderings for display only; this display processing changes neither the stored model outputs nor the inputs supplied to CV-Judge. An interactive zoom-in interface supports inspection of local details such as text, boundaries, and object parts (Figure 2c). These controls standardize presentation and support detailed visual inspection.

Evaluation settings. We report four evaluation settings: (i) **Active Elo**, which combines automatic and human preferences through the routing and aggregation procedure in Section 4.2; (ii) **budget-matched human-only**, which uses human comparisons within the annotation budget Chiang et al. (2024); Jiang et al. (2024); (iii) **CV-Judge only**, which uses automatic CV-Judge preferences; and (iv) **EditReward only**, which uses the EditReward evaluator Wu et al. (2025b). The human-only setting is a limited-budget reference rather than a complete human ground-truth ranking of all systems.

6.2 Benchmark Results

Leaderboard. Table 4 shows the top five systems under each protocol; Table 5 provides the full 21-system rankings. Under Active Elo, CV-Agent has the highest point estimate (1145), followed by nano banana pro (1127) and Manus (1109). Model ordering varies across protocols: CV-Agent is fifth under budget-matched human-only evaluation and sixth under CV-Judge-only evaluation, while Qwen Image Edit is first under budget-matched human-only and eleventh under Active Elo.

These differences describe the dependence of the reported ranking on the evaluation protocol. A supplementary visualization of the reported average win rates is provided in Appendix E.

Empirical support for the routing policy comes from the human-agreement study in Section 4.2. That study covers four selected models and eight task categories, with 50 annotators, and does not include CV-Agent. Its evidence should therefore be interpreted within that validation setting rather than as independent confirmation of the full 21-system ordering.

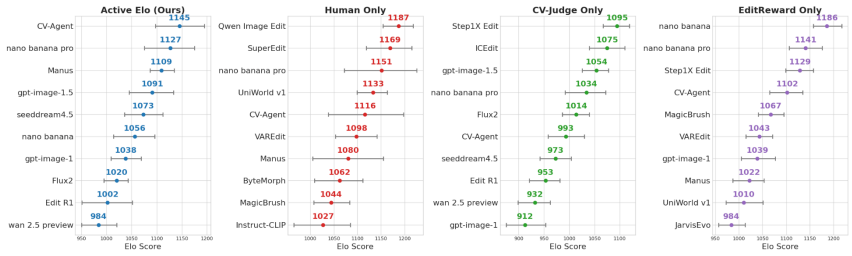

Figure 4: Bootstrap Elo estimates for the ten highest-rated systems under each of four evaluation settings, using 1000 resampling rounds. From left to right: Active Elo (ours), budget-matched human-only, CV-Judge only, and EditReward only.

Ranking uncertainty. Figure 4 reports variability in Elo estimates under comparison resampling. Narrower intervals indicate lower estimated variability under the corresponding resampling procedure; they do not by themselves establish greater agreement with human preferences. The reported intervals for CV-Agent, nano banana pro, and Manus overlap, and the current results do not establish a statistically unique lead. The same-editor comparison in Table 3 provides a complementary observation: the full CV-Agent configuration is 89 Active Elo points above its single-pass nano banana baseline.

Table 4: **Top-5 Leaderboards.** Each block reports the five highest-rated systems under its evaluation protocol, with Elo point estimates and 95% confidence intervals. Ratings describe relative performance within each protocol.

Active Elo (Ours)			**Budget-matched human-only**		
Model	Elo	95% CI	Model	Elo	95% CI
CV-Agent	1145	+50/-48	Qwen Image Edit	1187	+31/-33
nano banana pro	1127	+48/-52	SuperEdit	1169	+46/-50
Manus	1109	+26/-22	nano banana pro	1151	+75/-79
gpt-image-1.5	1091	+42/-46	UniWorld v1	1133	+30/-34
Seedream 4.5	1073	+39/-37	CV-Agent	1116	+82/-78
CV-Judge Only			**EditReward Only**		
Model	Elo	95% CI	Model	Elo	95% CI
Step1X Edit	1095	+24/-28	nano banana	1186	+31/-29
ICEdit	1075	+35/-35	nano banana pro	1141	+35/-35
gpt-image-1.5	1054	+24/-28	Step1X Edit	1129	+28/-30
nano banana pro	1034	+38/-42	CV-Agent	1102	+33/-37
Flux2	1014	+26/-28	MagicBrush	1067	+28/-26

Per-dimension analysis. Among the five systems reported in Appendix L, CV-Agent and Manus have higher mean CV-Judge editing-success and prompt-following scores than the three single-pass editors, while nano banana pro, gpt-image-1.5, and Seedream 4.5 have higher mean perceptual-quality scores. These are dimension-specific observations for the reported systems under CV-Judge. Selected qualitative comparisons are shown in Figure 3, with additional low-level tasks and failure cases in Figures 6 and 7.

Comparison with traditional metrics. We evaluate CLIP-I Radford et al. (2021), DINO Caron et al. (2021), and CLIPScore Hessel et al. (2021) on approximately 1K randomly sampled input cases for five selected competitive systems, yielding approximately 5K edited outputs. The CLIP-I and DINO mean scores span 0.034 and 0.033, respectively. Their model rankings each have Spearman correlation $\rho = 0.50$ with the Active Elo ordering restricted to these five systems, while CLIPScore has $\rho = 0.90$. These correlations are computed across the five systems using Active Elo as the reference; they do not measure agreement with an independent human ranking. Paired-bootstrap tests identify differences between several model pairs on CLIP-I and DINO, but metric-level differences do not by themselves establish which ordering better reflects instruction satisfaction. We present this analysis as a supplementary diagnostic limited to the sampled cases and selected systems. Scores and testing details are provided in Appendix M.

7 Conclusion

We introduce **CV-Arena**, a benchmark for instructional computer vision problem solving on real, high-resolution images, together with **Active Elo**, a human-AI collaborative evaluation protocol. Within the reported validation setting, two-gate routing yields higher human agreement and bootstrap ranking stability. Our simple **CV-Agent** baseline shows progressively higher Active Elo point estimates in same-editor ablations with added understanding, planning, and closed-loop refinement, motivating further study of agentic image editing.

AI USE STATEMENT

We used LLMs and VLMs for query generation, visual captioning, and candidate assessment in CogRetriever, automated scoring in CV-Judge, and instruction understanding, planning, and verification in CV-Agent. Generative image editors produced the outputs evaluated in the benchmark; the benchmark's source images were retrieved real images. We also used LLM assistance for English-language editing, LaTeX formatting, and feedback on methodological descriptions, mathematical explanations, and interpretations of existing results. The authors take responsibility for the final manuscript, including all methods, claims, results, and artifacts produced with AI assistance.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

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. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18392–18402, 2022. URL https://api.semanticscholar.org/CorpusID:253581213.

Mingdeng Cao, Xuaner Zhang, Yinqiang Zheng, and Zhihao Xia. Instruction-based image manipulation by watching how things move. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2704–2713, 2025.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9650–9660, 2021. URL https://arxiv.org/abs/2104.14294.

Di Chang, Mingdeng Cao, Yichun Shi, Bo Liu, Shengqu Cai, Shijie Zhou, Weilin Huang, Gordon Wetzstein, Mohammad Soleymani, and Peng Wang. Bytemorph: Benchmarking instruction-guided image editing with non-rigid motions. *arXiv preprint arXiv:2506.03107*, 2025a.

Di Chang, Mingdeng Cao, Yichun Shi, Bo Liu, Shengqu Cai, Shijie Zhou, Weilin Huang, Gordon Wetzstein, Mohammad Soleymani, and Peng Wang. Bytemorph: Benchmarking instruction-guided image editing with non-rigid motions. *arXiv preprint arXiv:2506.03107*, 2025b.

Sherry X. Chen, Misha Sra, and Pradeep Sen. Instruct-clip: Improving instruction-guided image editing with automated data refinement using contrastive learning, 2025a. URL https://arxiv.org/abs/2503.18406.

Zheng Chen, Yulun Zhang, Ding Liu, Jinjin Gu, Linghe Kong, Xin Yuan, et al. Hierarchical integration diffusion model for realistic image deblurring. *Advances in neural information processing systems*, 36:29114–29125, 2023a.

Zhihong Chen, Xuehai Bai, Yang Shi, Chaoyou Fu, Huanyu Zhang, Haotian Wang, Xiaoyan Sun, Zhang Zhang, Liang Wang, Yuanxing Zhang, et al. Opengpt-4o-image: A comprehensive dataset for advanced image generation and editing. *arXiv preprint arXiv:2509.24900*, 2025b.

Zijian Chen, Wei Sun, Haoning Wu, Zicheng Zhang, Jun Jia, Zhongpeng Ji, Fengyu Sun, Shangling Jui, Xiongkuo Min, Guangtao Zhai, et al. Exploring the naturalness of ai-generated images. *arXiv preprint arXiv:2312.05476*, 2023b.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. *arXiv preprint arXiv:2403.04132*, 2024.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Wenyan Cong, Jianfu Zhang, Li Niu, Liu Liu, Zhixin Ling, Weiyuan Li, and Liqing Zhang. Image harmonization dataset iharmony4: Hcoco, hadobe5k, hflickr, and hday2night. *arXiv preprint arXiv:1908.10526*, 2019.

Google DeepMind. Introducing gemini 2.5 flash image, our state-of-the-art image model, 2025a. URL https://developers.googleblog.com/introducing-gemini-2-5-flash-image/.

Google DeepMind. Introducing nano banana pro, 2025b. URL https://blog.google/technology/ai/nano-banana-pro/.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*, 2024.

Ahmed A Elngar, Mohamed Arafa, Amar Fathy, Basma Moustafa, Omar Mahmoud, Mohamed Shaban, and Nehal Fawzy. Image classification based on cnn: a survey. *Journal of Cybersecurity and Information Management*, 6(1):18–50, 2021.

Zhengyao Fang, Pengyuan Lyu, Jingjing Wu, Chengquan Zhang, Jun Yu, Guangming Lu, and Wenjie Pei. Recognition-synergistic scene text editing. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 13104–13113, 2025.

David A Forsyth and Jean Ponce. *Computer vision: a modern approach*. prentice hall professional technical reference, 2002.

Yuying Ge, Sijie Zhao, Chen Li, Yixiao Ge, and Ying Shan. Seed-data-edit technical report: A hybrid dataset for instructional image editing. *arXiv preprint arXiv:2405.04007*, 2024.

Google. Gemini 2.5 pro model card, 2025. URL https://modelcards.withgoogle.com/assets/documents/gemini-2.5-pro.pdf. Accessed: 2026-01-11.

Xin Gu, Ming Li, Libo Zhang, Fan Chen, Longyin Wen, Tiejian Luo, and Sijie Zhu. Multi-reward as condition for instruction-based image editing. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Yuxuan Gu, Haoxuan Wang, Pengyang Ling, Zhixiang Wei, Huaian Chen, Yi Jin, and Enhong Chen. Improving visual and downstream performance of low-light enhancer with vision foundation models collaboration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16071–16080, 2025b.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. *arXiv preprint arXiv:2402.01680*, 2024.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 7514–7528, 2021.

Mude Hui, Siwei Yang, Bingchen Zhao, Yichun Shi, Heng Wang, Peng Wang, Yuyin Zhou, and Cihang Xie. Hq-edit: A high-quality dataset for instruction-based image editing. *arXiv preprint arXiv:2404.09990*, 2024.

Muhammad Kamran Janjua, Amirhosein Ghasemabadi, Kunlin Zhang, Mohammad Salameh, Chao Gao, and Di Niu. Grounding degradations in natural language for all-in-one video restoration. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5734–5743, 2026.

Bohan Jia, Wenxuan Huang, Yuntian Tang, Junbo Qiao, Jincheng Liao, Shaosheng Cao, Fei Zhao, Zhaopeng Feng, Zhouhong Gu, Zhenfei Yin, et al. Compbench: Benchmarking complex instruction-guided image editing. *arXiv preprint arXiv:2505.12200*, 2025.

Dongfu Jiang, Max Ku, Tianle Li, Yuansheng Ni, Shizhuo Sun, Rongqi Fan, and Wenhu Chen. Genai arena: An open evaluation platform for generative models. *Advances in Neural Information Processing Systems*, 37:79889–79908, 2024.

Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93, 1938.

Jinwoo Kim, Sangmin Han, Jinho Jeong, Jiwoo Choi, Dongyeoung Kim, and Seon Joo Kim. Orida: Object-centric real-world image composition dataset. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 3051–3060, 2025.

Jari Korhonen and Junyong You. Peak signal-to-noise ratio revisited: Is simple beautiful? In *2012 Fourth International Workshop on Quality of Multimedia Experience*, pp. 37–38, 2012. doi: 10.1109/QoMEX.2012.6263880.

Max Ku, Tianle Li, Kai Zhang, Yujie Lu, Xingyu Fu, Wenwen Zhuang, and Wenhu Chen. Imagenhub: Standardizing the evaluation of conditional image generation models. *arXiv preprint arXiv:2310.01596*, 2023.

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

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Ming Li, Xin Gu, Fan Chen, Xiaoying Xing, Longyin Wen, Chen Chen, and Sijie Zhu. Superedit: Rectifying and facilitating supervision for instruction-based image editing. 2025. URL `https://arxiv.org/abs/2505.02370`.

Simin Li, Shuning Zhang, Gujun Chen, Dong Wang, Pu Feng, Jiakai Wang, Aishan Liu, Xin Yi, and Xianglong Liu. Towards benchmarking and assessing visual naturalness of physical world adversarial attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12324–12333, 2023.

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

Yunlong Lin, Linqing Wang, Kunjie Lin, Zixu Lin, Kaixiong Gong, Wenbo Li, Bin Lin, Zhenxi Li, Shiyi Zhang, Yuyang Peng, et al. Jarvisevo: Towards a self-evolving photo editing agent with synergistic editor-evaluator optimization. *arXiv preprint arXiv:2511.23002*, 2025b.

Chang Liu, Xiangtai Li, and Henghui Ding. Referring image editing: Object-level image editing via referring expressions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13128–13138, 2024.

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.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, et al. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*, 2024.

Wenyang Luo, Haina Qin, Zewen Chen, Libin Wang, Dandan Zheng, Yuming Li, Yufan Liu, Bing Li, and Weiming Hu. Visual-instructed degradation diffusion for all-in-one image restoration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12764–12777, 2025.

Qingyang Mao, Qi Cai, Yehao Li, Yingwei Pan, Mingyue Cheng, Ting Yao, Qi Liu, and Tao Mei. Visual autoregressive modeling for instruction-guided image editing. *arXiv preprint*, 2025.

Manus Meta. Introducing manus 1.6: Max performance, mobile dev, and design view, 2025. URL https://manus.im/blog/manus-max-release.

Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3523–3542, 2021.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *ACM Transactions on Intelligent Systems and Technology*, 16(5):1–72, 2025.

OpenAI. Introducing chatgpt agent: bridging research and action, 2025a. URL https://openai.com/index/introducing-chatgpt-agent/. Accessed: 2026-01-26.

OpenAI. Gpt image 1, 2025b. URL https://platform.openai.com/docs/models/gpt-image-1.

OpenAI. The new chatgpt images is here, 2025c. URL https://openai.com/index/new-chatgpt-images-is-here/.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Anthropic PBC. The claude 3 model family: Opus, sonnet, haiku. URL https://api.semanticscholar.org/CorpusID:268232499.

Yuang Peng, Yuxin Cui, Haomiao Tang, Zekun Qi, Runpei Dong, Jing Bai, Chunrui Han, Zheng Ge, Xiangyu Zhang, and Shu-Tao Xia. Dreambench++: A human-aligned benchmark for personalized image generation. *arXiv preprint arXiv:2406.16855*, 2024.

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.

Yadong Qu, Qingfeng Tan, Hongtao Xie, Jianjun Xu, Yuxin Wang, and Yongdong Zhang. Exploring stroke-level modifications for scene text editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 2119–2127, 2023.

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

Team Seedream, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, et al. Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427*, 2025.

Wensong Song, Hong Jiang, Zongxin Yang, Zheqiao Cheng, Ruijie Quan, and Yi Yang. Insert anything: Image insertion via in-context editing in dit. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 9097–9105, 2026.

Richard Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022.

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.

Yoad Tewel, Rinon Gal, Dvir Samuel, Yuval Atzmon, Lior Wolf, and Gal Chechik. Add-it: Training-free object insertion in images with pretrained diffusion models. *arXiv preprint arXiv:2411.07232*, 2024.

Lucas Theis. What makes an image realistic? *arXiv preprint arXiv:2403.04493*, 2024.

Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018(1):7068349, 2018.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Chao Wang, Hehe Fan, Huichen Yang, Sarvnaz Karimi, Lina Yao, and Yi Yang. Adapting text-to-image generation with feature difference instruction for generic image restoration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 23539–23550, 2025a.

Chenglin Wang, Yucheng Zhou, Qianning Wang, Zhe Wang, and Kai Zhang. Complexbench-edit: Benchmarking complex instruction-driven image editing via compositional dependencies. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 13391–13397, 2025b.

Qinsi Wang, Bo Liu, Tianyi Zhou, Jing Shi, Yueqian Lin, Yiran Chen, Hai Helen Li, Kun Wan, and Wentian Zhao. Vision-zero: Scalable vlm self-improvement via strategic gamified self-play. *arXiv preprint arXiv:2509.25541*, 2025c.

Yufei Wang, Zhanyi Sun, Jesse Zhang, Zhou Xian, Erdem Biyik, David Held, and Zackory Erickson. Rl-vlm-f: Reinforcement learning from vision language foundation model feedback. *arXiv preprint arXiv:2402.03681*, 2024.

Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

Daniel Winter, Matan Cohen, Shlomi Fruchter, Yael Pritch, Alex Rav-Acha, and Yedid Hoshen. Objectdrop: Bootstrapping counterfactuals for photorealistic object removal and insertion. In *European Conference on Computer Vision*, pp. 112–129. Springer, 2024.

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, 2025a. URL https://arxiv.org/abs/2508.02324.

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*, 2025b.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

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

Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. Anyedit: Mastering unified high-quality image editing for any idea. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26125–26135, 2025.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5625–5644, 2024a.

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.

Yafei Zhang, Shuaitian Song, Huafeng Li, Shujuan Wang, and Yu Liu. Adaptive dynamic dehazing via instruction-driven and task-feedback closed-loop optimization for diverse downstream task adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 12888–12896, 2026.

Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. In-context edit: Enabling instructional image editing with in-context generation in large-scale diffusion transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025a. arXiv:2504.20690.

Ziming Zhang, Yuping Shao, Yiqing Zhang, Fangzhou Lin, Haichong Zhang, and Elke Rundensteiner. Deep loss convexification for learning iterative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024b.

Zongyan Zhang, CL Philip Chen, Haohan Weng, and Tong Zhang. Self-prompt guided image outpainting model for captions absence in social scenes. *IEEE Transactions on Computational Social Systems*, 2025b.

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, 2024a.

Wenting Zhao, Alexander M Rush, and Tanya Goyal. Challenges in trustworthy human evaluation of chatbots. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 3359–3365, 2025.

Xia Zhao, Limin Wang, Yufei Zhang, Xuming Han, Muhammet Deveci, and Milan Parmar. A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*, 57(4):99, 2024b.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Han Zhou, Wei Dong, Xiaohong Liu, Yulun Zhang, Guangtao Zhai, and Jun Chen. Low-light image enhancement via generative perceptual priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10752–10760, 2025.

Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023.

A Extended Related Work

This appendix expands the related-work discussion summarized in Section 2, covering datasets and benchmarks for real-world visual understanding in greater detail.

A.1 Datasets for Real-World Visual Understanding

Since the early breakthroughs brought by the MNIST and ImageNet datasets LeCun (1998); Deng et al. (2009), real-world visual understanding has remained one of the most fundamental tasks in AI and has continuously driven the development of the entire AI community Voulodimos et al. (2018); Forsyth & Ponce (2002); Szeliski (2022); Zhang et al. (2024b); Zhao et al. (2024b). With the rapid progress of modern architectures and large-scale training, recognition tasks such as image classification, semantic segmentation, and object detection have improved substantially Elngar et al. (2021); Minaee et al. (2021); Zou et al. (2023). However, tasks at a higher semantic and perceptual tier, involving notions such as image realism Theis (2024); Chen et al. (2023a), visual naturalness Li et al. (2023); Chen et al. (2023b), and the plausibility of edits Ye et al. (2025); Chen et al. (2025b), are still far from being solved. Unlike recognition-oriented benchmarks, these tasks require modeling not only what is visible in the image, but also whether the visual content makes sense, remains natural, and aligns with implicit world knowledge and commonsense constraints.

A.2 Benchmarks for Real-World Visual Understanding

Existing benchmarks exhibit significant limitations in evaluating this capability. Early datasets, such as iHarmony4 Cong et al. (2019), reduce object insertion to appearance harmonization, focusing primarily on color correction while ignoring essential physical cues such as shadows, reflections, and contact interactions. More recent counterfactual datasets Kim et al. (2025); Winter et al. (2024) improve realism by capturing static object presence on rigid surfaces, but still neglect dynamic interactions with deformable or non-solid media such as water, sand, or soft furniture. Benchmarks emphasizing placement plausibility Tewel et al. (2024) further narrow the scope by evaluating only semantic appropriateness, overlooking physical consequences, aesthetic composition, and narrative coherence.

In CV-Arena, we shift the focus from static *presence* to dynamic *interaction*. We introduce scenarios probing interaction with deformable surfaces (e.g., ripples in water or footprints in sand), complex optical effects (e.g., distorted reflections or light caustics), and precise functional interactions (e.g., a key fitting into a lock).

Current benchmarks frequently conflate this task with simpler creation or erasure operations. Datasets such as MagicBrush Zhang et al. (2023) and RefCOCO-Edit Liu et al. (2024) mix reconstruction, replacement, and insertion tasks, blurring evaluation signals and obscuring true semantic understanding. CV-Arena isolates semantic manipulation as a first-priority task and emphasizes pose/state transition, spatial rearrangement, and component-level adjustments grounded in real images.

Existing benchmarks have partially touched upon geometry-related editing scenarios Zhang et al. (2023); Brooks et al. (2022); Yu et al. (2025); however, such tasks are often not treated as a distinct category, or are represented by only a limited number of simplified cases. As a result, geometric transformation is frequently entangled with general appearance editing, making it difficult to isolate and evaluate a model's structural reasoning capability. Some recent efforts, such as ImgEdit Ye et al. (2025), explicitly introduce *single-turn* and *multi-turn* editing formulations to better support complex editing behaviors, partially addressing the limitations of one-shot editing protocols. ImgEdit also includes an understanding–grounding–editing suite with spatial reasoning and multi-object targets, extending its coverage beyond interaction length.

In contrast, CV-Arena adopts a geometry-centric task design that formulates editing scenarios based on the intended structural transformation itself, rather than explicitly categorizing tasks by the number of editing turns. The required editing process is implicitly determined by the geometric and structural complexity of the instruction, allowing tasks to more naturally reflect real-world professional workflows. This design choice not only evaluates a model's image generation capability, but also probes its ability to accurately interpret and execute geometry-driven instructions, aligning more closely with the core objective of instruction-guided image editing.

Table 5: Full leaderboard comparison across four evaluation settings (21 systems). Human Only denotes the budget-matched human-only reference.

Active Elo (Ours)			**Human Only**			**CV-Judge only**			**EditReward Only**		
Model	Elo	95% CI	Model	Elo	95% CI	Model	Elo	95% CI	Model	Elo	95% CI
CV-Agent	1145	+50/-48	Qwen Image Edit	1187	+31/-33	Step1X Edit	1095	+24/-28	nano banana	1186	+31/-29
nano banana pro	1127	+48/-52	SuperEdit	1169	+46/-50	ICEdit	1075	+35/-35	nano banana pro	1141	+35/-35
Manus	1109	+26/-22	nano banana pro	1151	+75/-79	gpt-image-1.5	1054	+24/-28	Step1X Edit	1129	+28/-30
gpt-image-1.5	1091	+42/-46	UniWorld v1	1133	+30/-34	nano banana pro	1034	+38/-42	CV-Agent	1102	+33/-37
Seedream 4.5	1073	+39/-37	CV-Agent	1116	+82/-78	Flux2	1014	+26/-28	MagicBrush	1067	+28/-26
nano banana	1056	+40/-42	VAREdit	1098	+43/-45	CV-Agent	993	+37/-35	VAREdit	1043	+28/-28
gpt-image-1	1038	+31/-29	Manus	1080	+75/-75	Seedream 4.5	973	+31/-31	gpt-image-1	1039	+38/-34
Flux2	1020	+23/-25	ByteMorph	1062	+49/-53	Edit R1	953	+28/-32	Manus	1022	+31/-35
Edit R1	1002	+49/-51	MagicBrush	1044	+39/-37	wan 2.5 preview	932	+30/-34	UniWorld v1	1010	+41/-37
wan 2.5 preview	984	+36/-34	Instruct-CLIP	1027	+58/-62	gpt-image-1	912	+41/-37	JarvisEvo	984	+29/-27
Qwen Image Edit	965	+36/-34	Step1X Edit	1009	+59/-61	nano banana	892	+35/-39	Anysd	969	+29/-25
VAREdit	948	+33/-37	JarvisEvo	991	+68/-70	Manus	874	+34/-32	gpt-image-1.5	952	+36/-32
Step1X Edit	932	+30/-26	gpt-image-1.5	974	+45/-41	SuperEdit	856	+27/-23	Edit R1	931	+38/-34
UniWorld v1	915	+38/-42	ICEdit	956	+42/-38	JarvisEvo	839	+42/-38	Instruct-CLIP	909	+29/-33
MagicBrush	897	+31/-27	Flux2	938	+29/-31	Qwen Image Edit	821	+21/-25	SuperEdit	894	+25/-27
SuperEdit	880	+37/-37	Seedream 4.5	920	+70/-70	VAREdit	804	+18/-22	Seedream 4.5	873	+32/-36
ByteMorph	864	+30/-28	Edit R1	902	+75/-73	Anysd	787	+37/-33	ByteMorph	845	+30/-34
ICEdit	847	+41/-39	gpt-image-1	884	+78/-74	UniWorld v1	770	+26/-28	Flux2	823	+37/-37
Instruct-CLIP	831	+19/-23	nano banana	866	+74/-76	MagicBrush	754	+25/-27	ICEdit	794	+39/-39
Anysd	815	+35/-39	Anysd	848	+64/-64	ByteMorph	738	+32/-28	Qwen Image Edit	771	+30/-30
JarvisEvo	799	+32/-32	wan 2.5 preview	830	+43/-43	Instruct-CLIP	722	+36/-32	wan 2.5 preview	756	+35/-37

Restoration of typography and UI elements targets the correction, reconstruction, or removal of textual and graphical elements embedded in real-world images Qu et al. (2023); Fang et al. (2025). Real-world cases are substantially harder than synthetic overlays: restoring degraded signage, correcting distorted storefront typography, or removing watermarks/UI elements from faces or finely textured fabrics. These tasks require character accuracy, layout consistency, and seamless background integration.

CV-Arena explicitly incorporates typography- and UI-centric tasks (text in-painting/correction, watermark and complex graphic removal, layout-consistent restoration), reflecting professional standards beyond purely aesthetic outcomes.

B Appendix: Task Taxonomy Details

This appendix provides detailed definitions of the three task families highlighted in the main text.

Scene Composition and Object Insertion. This category requires models to move beyond naive object pasting and perform physically and semantically coherent scene composition. Successful object insertion demands consistent integration across geometry, lighting, scale, and semantics, ensuring that inserted objects obey physical plausibility and visual harmony with the surrounding environment. Representative instructions include placing an object onto a surface with correct shadow casting, inserting a reflective object that respects the existing illumination, and composing multiple objects whose spatial arrangement must remain physically stable.

Semantic-Aware Content Instruction. Semantic-aware content instruction challenges a model to modify intrinsic properties of existing objects, such as pose, functional state, or spatial configuration, strictly without introducing or removing entities. Unlike object addition or deletion, these edits require fine-grained manipulation grounded in physical common sense and part-whole relationships. Representative instructions include changing the pose of an articulated object while preserving its identity, switching the functional state of a device (e.g., open versus closed), or rearranging spatial relationships among existing entities without altering the inventory of the scene.

Text-Based Geometric Warping and Structural Control. Text-based geometric warping requires models to perform precise, logically consistent shape and structure transformations driven purely by language. Representative instructions include pose transformations, viewpoint changes, and fine-grained expression control specifying continuous mixtures (e.g., "slightly more surprised, less neutral") rather than discrete categories. These tasks stress a model's ability to translate symbolic linguistic descriptions into geometrically faithful structural edits while preserving identity and surrounding context.

The remaining task families (restoration, enhancement, computational photography operations, typography and UI recovery, etc.) follow standard formulations from the literature but are curated at high resolution under realistic difficulty conditions.

C Appendix: CogRetriever Implementation Details

Stage 1: Planning. Given a professional instruction $\mathbf{I_i}$, the planner decomposes $\mathbf{I_i}$ into searchable visual attributes and generates a diverse query set $\mathcal{Q} = \{q_1, \ldots, q_K\}$, where $K = 5$ is chosen to encourage coverage of complementary visual aspects (e.g., subject, scene, style, lighting, viewpoint).

Stage 2: Action & Perception. For each query, the system searches and downloads the top-N candidate images ($N = 20$), applies multifaceted checks (file validity, minimum size, format normalization), and produces dense visual captions $c(x)$ that describe both semantic content and appearance attributes such as atmosphere, style, and composition. Captions are used both for downstream scoring and for filtering near-duplicates by content rather than by raw pixel similarity alone.

Stage 3: Evaluation & Pool Construction. A vision-language model scores candidates with $s(\mathbf{x}; \mathbf{I_i}) \in [0, 1]$, identifying a pool $\mathcal{P}_t = \{x \mid s(\mathbf{x}; \mathbf{I_i}) \geq \tau\}$. If the first iteration produces $|\mathcal{P}_t| < K_p$ (where $K_p = 3$, $\tau = 0.8$), the agent writes a reflection in memory m_{t+1} identifying missing or unexpected visual attributes, which guides query refinement in the second iteration. Once a sufficient pool is reached, the final candidate set is taken as the top-K_p scoring elements:

$$\mathcal{X}^* = \mathrm{TopK}_{\mathbf{x} \in \mathcal{P}_t} \, s(\mathbf{x}; \mathbf{I_i}). \tag{5}$$

Iteration cap and termination. The maximum iteration budget is $T = 2$, comprising an initial retrieval and, if needed, one reflection-guided refinement. The loop stops once $|\mathcal{P}_t| \geq K_p$. If the pool remains insufficient after the second iteration, the case is flagged for manual review. Refinement updates the search queries and candidate pool; CogRetriever does not synthesize or modify the source images.

D Appendix: Filtering, Traceability, and Human Verification

Comprehensive logging and traceability. To maintain dataset integrity and facilitate downstream auditability, the system records logs including the query set $\mathcal{Q}_t$, the evaluation scores s_t, and the corresponding agentic reflections at each iteration. The pipeline generates two outputs: (i) the finalized image pool selected for dataset inclusion, and (ii) a complete candidate set preserved with metadata for auditing or re-scoring. This logging design ensures reproducibility and supports rigorous offline analysis or re-evaluation under updated criteria.

License filtering and source verification. For manual acquisition in the Base Track, we used the `Creative Commons` rights filter in Google Images. The Agentic Track used the Google Custom Search API with parameters restricted to `cc_publicdomain` and `cc_attribute` content. These filters were used for candidate selection. Source URLs and available license and attribution information were recorded for provenance tracking. For the license-verified subset, the original source pages and stated licenses have been checked; search-time filters do not replace these per-image checks.

Privacy and harmful-content screening. The curation process included privacy and harmful-content screening, separately from license filtering. Release review covers both source images and editing instructions, including visible personal information, explicit or graphically violent content, and unsafe editing requests. These release checks are distinct from the task-feasibility verification described below.

Near-duplicate and low-quality removal. We additionally apply a filtering protocol that eliminates near-duplicates (using both perceptual hashing and caption similarity) and rejects low-quality samples (e.g., extreme compression artifacts, unrelated content). This stage further excludes ambiguous

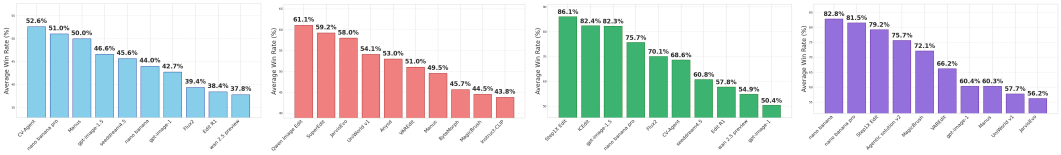

Figure 5: **Supplementary Win-Rate Visualization.** The panels show the originally reported average win rates under four evaluation settings, from left to right: Active Elo, budget-matched human-only, CV-Judge only, and EditReward only. Model labels and comparison-pool definitions require reconciliation with the underlying evaluation records.

sources that might impede consistent evaluation. The filter is specifically designed to enhance the signal quality of the benchmark without oversimplifying the underlying tasks: we prioritize retention of challenging real-world conditions provided the source imagery remains visually interpretable and the associated task intent is unambiguous.

Human verification protocol. After automatic filtering, human curators verified the remaining image-instruction pairs using four criteria: (i) task-category match, (ii) target-region visibility, (iii) instruction feasibility, and (iv) whether the requested edit had a consistent, assessable success criterion. For detail-sensitive cases, curators used the *zoom-in* function shown in Figure 2c to inspect text, boundaries, object parts, and fine structural details. Pairs failing any criterion were sent for manual repair or discarded. This verification supported the task suitability and visual interpretability of the 12K evaluation corpus; it is distinct from source-license, privacy, and harmful-content checks for public release.

E Appendix: Supplementary Win-Rate Visualization

F Appendix: Calibrating AI Reliability from Score Gap

Our hybrid protocol uses an instance-dependent reliability input for AI-resolved comparisons, denoted by $q_{\mathrm{AI}}(g) \in [0,1]$, estimated from empirical agreement with human preferences. The score gap

$$g_i(A,B) = |s_{i,A} - s_{i,B}| \tag{6}$$

serves as a practical proxy for comparison ambiguity.

F.1 Calibration set construction

We construct a small calibration set of paired comparisons by sampling instances i and model pairs (A,B). For each sampled pair, we collect:

- CV-Judge scores $(s_{i,A}, s_{i,B})$ and the induced AI preference
$$\hat{z}^{\mathrm{AI}}_{i,A,B} = \mathbb{I}[s_{i,A} \geq s_{i,B}], \tag{7}$$
- a human preference label $z^{\mathrm{H}}_{i,A,B} \in \{0,1\}$ under the same display protocol as the main benchmark.

We then define the agreement indicator

$$a_{i,A,B} = \mathbb{I}\big[\hat{z}^{\mathrm{AI}}_{i,A,B} = z^{\mathrm{H}}_{i,A,B}\big] \in \{0,1\}. \tag{8}$$

F.2 Binned estimation and monotone fitting

We partition the observed gaps $\{g_i(A,B)\}$ into J bins $\{\mathcal{B}_j\}_{j=1}^{J}$ and compute the empirical AI-human agreement in bin j as

$$\hat{q}_j = \frac{1}{|\mathcal{B}_j|} \sum_{(i,A,B)\in\mathcal{B}_j} a_{i,A,B}. \tag{9}$$

We model agreement as non-decreasing with g. Two possible monotone mappings are:

- **Piecewise-constant monotone projection:** apply isotonic regression on $(\bar{g}_j, \hat{q}_j)$;
- **Smooth parametric form:** fit a logistic mapping

$$q_{\mathrm{AI}}(g) = \sigma\big(\beta(g - g_0)\big) \tag{10}$$

where $\beta > 0$ controls the sharpness of the transition and g_0 is the midpoint of the logistic mapping.

In our implementation, we default to isotonic regression for its nonparametric stability.

F.3 Final reliability map used in ranking

The fitted agreement function $q_{\mathrm{AI}}(g)$ supplies the reliability input to the credibility weight ρ in the Elo updates (Section 4.2). For AI-resolved comparisons we set

$$q = q_{\mathrm{AI}}\big(g_i(A, B)\big), \tag{11}$$

while for human-labeled comparisons we use $q \approx 1$. Empirical AI-human agreement is not automatically the probability of the credible component in a mixture model. Appendix H therefore uses the mixture model to interpret the form of the weight, without treating agreement calibration as proof of a calibrated mixture-component probability.

G Appendix: Motivation for Two-Gate Routing

The two-gate policy allocates human comparisons to candidates that both meet a quality threshold and receive similar CV-Judge scores. The quality gate $\min(s_{i,A}, s_{i,B}) \geq \tau$ focuses annotation on pairs where both outputs reach the chosen score threshold. The ambiguity gate $g_i(A, B) = |s_{i,A} - s_{i,B}| < \Delta$ selects close automatic judgments within that region. Pairs that fail either gate are resolved automatically.

The score gap is an empirical proxy for judge ambiguity, supported by the agreement breakdown in Appendix K. It is distinct from the Elo-predicted win probability:

$$p_{AB} = \sigma\left(\frac{R_A - R_B}{S}\right), \tag{12}$$

where σ is the sigmoid function and $S = \frac{s_{i,A}+s_{i,B}}{2}$ is the instance-dependent scale used in the main text. Here S denotes the pairwise scale, not a single output's final score. The routing rule is a practical design choice evaluated by the routing comparison in Table 2; we do not claim that it maximizes an information-per-cost objective.

H Appendix: Interpretation of Reliability-Weighted Elo

We explain the form of the credibility weight using a conceptual mixture model. This interpretation does not establish that the empirical agreement input q is a calibrated mixture-component probability, nor that the implemented Elo procedure is an exact online-EM algorithm.

H.1 Weight form and a conceptual mixture

Let $z \in \{0, 1\}$ be the final observed preference after routing and define

$$w = \begin{cases} p_{AB}, & z = 1, \\ 1 - p_{AB}, & z = 0. \end{cases} \tag{13}$$

Consider a conceptual label model with probability λ of drawing from a credible Bradley–Terry component and probability $1 - \lambda$ of drawing a fair random guess. The observed-label likelihood is w in the credible component and $1/2$ in the random component, so Bayes' rule gives posterior credibility $\lambda w/(\lambda w + (1 - \lambda)/2)$. The credibility weight used in our Elo update has the same functional form:

$$\rho = \frac{q\, w}{q\, w + (1 - q)\frac{1}{2}}. \tag{14}$$

Here q is the reliability input specified in Appendix F. Matching the functional form does not make its empirical agreement estimate equivalent to the conceptual mixture prior λ.

H.2 Relation to a weighted likelihood

For a single comparison, the Bradley–Terry log-likelihood is

$$\ell(R_A, R_B) = z \log p_{AB} + (1 - z) \log(1 - p_{AB}). \tag{15}$$

Using the scale in Eq. equation 12, its derivative is

$$\frac{\partial \ell}{\partial (R_A - R_B)} = \frac{1}{S} (z - p_{AB}). \tag{16}$$

The Elo update, writing $\eta = K_r$ for the relevant rater type, remains

$$R_A \leftarrow R_A + \eta \, \rho \, (z - p_{AB}), \qquad R_B \leftarrow R_B - \eta \, \rho \, (z - p_{AB}). \tag{17}$$

For this algebraic comparison, treating ρ as fixed, the displayed update corresponds to a weighted-gradient step with effective learning rate ηS, which cancels the $1/S$ factor. Since S varies across comparisons, this is not a fixed-learning-rate gradient equivalence. The relation explains the update's form; it does not introduce a new optimization procedure or establish exact EM inference. The actual ranking rule remains the Elo update in Section 4.2.

I Appendix: Two-Gate Routing Policy Ablations

We compare supervision and routing strategies under a fixed human-comparison budget to assess human agreement and ranking stability. This comparison concerns which pairs receive human labels; it does not separately isolate reliability calibration or the credibility-weighted update. Notation follows the main text: CV-Judge scores $s_{i,m}$, gap $g_i(A, B) = |s_{i,A} - s_{i,B}|$, final binary preference $z \in \{0, 1\}$, Elo ratings R_m, and

$$p_{AB} = \sigma\left(\frac{R_A - R_B}{S}\right), \qquad \rho = \frac{q\, w}{q\, w + (1 - q)\frac{1}{2}}, \;\; w = \begin{cases} p_{AB}, & z = 1 \\ 1 - p_{AB}, & z = 0. \end{cases} \tag{18}$$

The Elo update uses binary preference outcomes.

I.1 Ablation Settings

Human Validation Set. We use a human validation set $\mathcal{H}_{\text{test}}$ comprising four selected models and eight task categories, with blinded pairwise preferences collected from 50 annotators under identical display conditions Jiang et al. (2024). CV-Agent is not among these four models. The results therefore assess the routing protocol in this validation setting and do not directly validate CV-Agent's position among all 21 systems.

Routing (human budget allocation). Under a fixed human budget B_H for strategies using human labels, we compare:

- **CV-Judge only:** $z = \mathbb{I}[s_{i,A} \geq s_{i,B}]$ for all pairs;
- **Human-only (budget-matched):** rank using only B_H human comparisons;
- **Quality-only:** human if $\min(s_{i,A}, s_{i,B}) \geq \tau$ (sample within-region to match B_H);
- **Ambiguity-only:** human if $g_i(A, B) < \Delta$;
- **Two-gate (ours):** human iff $\min(s_{i,A}, s_{i,B}) \geq \tau$ and $g_i(A, B) < \Delta$.

We report metrics that directly reflect (i) *faithfulness to human preference* and (ii) *leaderboard stability*. We avoid reporting raw Elo values, which are scale-dependent and less interpretable.

(1) Acc_H **(Human-consistency / Agreement with Humans).** Given the final Elo ratings, we predict the preferred model in each validation comparison (A_k, B_k) as $\mathbb{I}[R_{A_k} > R_{B_k}]$ and compute agreement with human labels Ouyang et al. (2022); Zheng et al. (2023):

$$\mathrm{Acc}_H = \frac{1}{|\mathcal{H}_{\text{test}}|} \sum_{k \in \mathcal{H}_{\text{test}}} \mathbb{I}\big[\big(R_{A_k} > R_{B_k}\big) \iff \big(z_k^{\mathrm{H}} = 1\big)\big]. \tag{19}$$

Higher Acc_H indicates that the learned ranking better matches human pairwise preferences on these validation comparisons. This ranking-induced agreement differs from the direct CV-Judge–human agreement reported by score gap in Appendix K.

(2) Spearman correlation (Rank correlation). To quantify ranking consistency under resampling, we perform bootstrap resampling of comparisons (with replacement), recompute Elo for each bootstrap replicate b, and obtain a ranking $\pi^{(b)}$. Let $\pi^{(\mathrm{full})}$ denote the ranking from the full comparison set under the same protocol. We report the average Spearman correlation between bootstrap and full-data ranks Kendall (1938); Dubois et al. (2024):

$$\rho_S = \frac{1}{B}\sum_{b=1}^{B} \mathrm{Spearman}\Big(\pi^{(b)},\, \pi^{(\mathrm{full})}\Big), \tag{20}$$

where B is the number of bootstrap replicates. Larger ρ_S indicates that the relative ordering of models is stable under resampling; this metric compares bootstrap and full-data rankings, not model rankings against a separate human ranking.

In Table 2, two-gate routing achieves the highest reported Acc_H and bootstrap ρ_S among the compared supervision and routing strategies in this validation setting.

J Appendix: CV-Judge Backbone Choice and Limitations

CV-Judge uses GPT-4o as its backbone VLM. The choice of backbone can affect dimension scores, score gaps, routing decisions, and the reliability estimates used in ranking. Human routing and credibility weighting provide mechanisms for combining automatic and human judgments, but they do not by themselves establish that the leaderboard is invariant to the judge backbone. The reported results should therefore be interpreted under the specified GPT-4o judging protocol.

K Appendix: AI-Human Agreement Stratified by Score Gap

We stratify direct CV-Judge–human agreement by the score gap $g_i(A,B) = |s_{i,A} - s_{i,B}|$ on the human validation set $\mathcal{H}_{\text{test}}$.

Table 6: AI-Human agreement and pair fraction by score gap.

Score gap g	AI-Human agreement	Fraction of pairs
$g < 50$	56.3%	18.2%
$50 \le g < 100$	69.1%	24.6%
$100 \le g < 200$	83.5%	31.4%
$g \ge 200$	94.8%	25.8%

Agreement increases across the reported gap bins, supporting the use of the score gap as a proxy for automatic-judge ambiguity. The 94.8% agreement applies specifically to the $g \ge 200$ bin. AI-resolved comparisons also include pairs that fail the quality gate, including those with small gaps; the table therefore does not establish agreement above 90% for all automatically resolved pairs. This monotone pattern motivates gap-dependent reliability inputs but does not, by itself, establish calibration of $q_{\mathrm{AI}}(g)$.

L Appendix: Per-Dimension Score Breakdown and Task-Level Deferral Rates

L.1 Per-Dimension Breakdown for Top Solutions

We report mean CV-Judge scores (on $[0, 1000]$) along each evaluation dimension for the top-5 solutions under Active Elo.

Among these five reported systems, CV-Agent and Manus have higher mean S_{edit} and S_{prompt} than the three single-pass editors, while nano banana pro, gpt-image-1.5, and Seedream 4.5 have higher

Table 7: Per-dimension mean scores for top-5 Active Elo solutions.

Model	S_{sem}	S_{edit}	S_{prompt}	S_{perc}
CV-Agent	**782**	**798**	**785**	751
nano banana pro	756	741	732	**784**
Manus	771	758	749	738
gpt-image-1.5	748	726	731	769
Seedream 4.5	734	711	703	761

mean S_{perc} than the two agentic systems. CV-Agent has the highest reported mean S_{sem}, S_{edit}, and S_{prompt}, and nano banana pro has the highest mean S_{perc}. These are descriptive comparisons of CV-Judge dimension scores for the displayed systems; they do not isolate module effects or establish an advantage for either class of methods beyond this comparison.

L.2 Task-Level Deferral Rates

Approximately 33.7% of pairwise comparisons satisfy both gates ($\min(s_{i,A}, s_{i,B}) \geq \tau$ and $g_i < \Delta$) and are routed to human annotators. The deferral rate, however, varies substantially across task families.

Table 8: Task-level human deferral rates.

Task family	Deferral rate
Geometry-driven warping	46.8%
Physically grounded composition	44.6%
Semantic content manipulation	33.5%
Computational photography	28.7%
Restoration / Enhancement	26.2%

The reported human deferral rate is higher for geometry-driven warping and physically grounded composition than for restoration/enhancement. These rates describe how the two-gate policy distributes human comparisons across the reported task families. They do not alone establish the reliability of automatic judgments within a family or the amount of information gained from each human comparison.

M Appendix: Comparison with Traditional Metrics

We evaluate CLIP-I (input-output image similarity), DINO (visual feature similarity), and CLIPScore (text-image similarity) on approximately 1K randomly sampled image-instruction cases. The five systems shown in Table 9 were selected for this analysis, yielding approximately 5K edited outputs across the shared cases. Random sampling applies to the cases; the system selection is deliberate. This analysis covers the reported subset and these five systems.

CV-Arena has no ground-truth edited references. CLIP-I and DINO therefore compare edited outputs with their input images, providing measures of input-output similarity. The desired amount of similarity depends on the instruction: object insertion changes image content, whereas exposure correction generally preserves scene content.

M.1 Aggregate Scores

CLIP-I ranges from 0.857 to 0.891, an absolute span of 0.034, and DINO ranges from 0.809 to 0.842, an absolute span of 0.033. These spans describe the five reported means; they do not by themselves determine statistical distinguishability.

Table 9: Traditional metric scores on the ∼1K subset, with Active Elo rank for reference.

Model	CLIP-I ↑	DINO ↑	CLIPScore ↑	Active Elo
CV-Agent	0.891	0.842	0.287	1
Manus	0.876	0.831	0.281	3
nano banana pro	0.864	0.817	0.272	2
nano banana	0.869	0.823	0.256	6
gpt-image-1.5	0.857	0.809	0.265	4

M.2 Paired Tests and Ranking Agreement

The reported paired bootstrap tests ($B = 10000$) yield $7/10$ significant system pairs for CLIP-I and $6/10$ for DINO. These tests concern differences in the respective metric scores, rather than agreement with human preferences or significance of differences from the Active Elo ordering.

CLIP-I and DINO induce exactly the same ordering among these five systems: CV-Agent $>$ Manus $>$ nano banana $>$ nano banana pro $>$ gpt-image-1.5. Their Spearman correlation with the Active Elo ordering restricted to these five systems is $\rho = 0.50$ ($p = 0.39$). This describes agreement between two rankings over five systems, not a direct comparison with a human ground-truth ranking. The differences in ordering do not alone identify the cause of metric disagreement.

M.3 CLIPScore Agreement with Active Elo

For the same five systems, CLIPScore has a higher descriptive correlation with Active Elo ($\rho = 0.90$). The top and bottom systems agree, and the only ordering difference is an adjacent swap of Manus and nano banana pro. This observation concerns the selected systems and does not establish a statistically significant difference between the two reported correlations. The aggregate analysis does not separately assess sensitivity to local perceptual artifacts or explicit instruction-constraint violations. We therefore use these metrics as supplementary diagnostics alongside the reported preference-based evaluation, without treating rank correlation alone as evidence of human-level judgment or complete constraint coverage.

N Appendix: Hyperparameter Sensitivity

N.1 Routing Thresholds τ and Δ

Alongside the routing comparison in Table 2, we report two joint perturbations around the default operating point $(\tau^\star, \Delta^\star)$.

Table 10: Sensitivity of Active Elo to the routing thresholds.

Configuration	Acc_H ↑	Spearman ρ_S ↑
Default $(\tau^\star, \Delta^\star)$	**82.6%**	**0.94**
Loose $(\tau^\star - 50, \Delta^\star + 50)$	80.1%	0.91
Tight $(\tau^\star + 50, \Delta^\star - 50)$	81.4%	0.92

Both tested perturbations yield slightly lower human agreement and bootstrap rank correlation than the default. These results characterize sensitivity at the reported operating points.

N.2 Dimension Weights $(\omega_s, \omega_e, \omega_i, \omega_p)$

The dimension weights determine the weighted score and the hard-failure thresholds and score caps in CV-Judge. The default weights are $(\omega_s, \omega_e, \omega_i, \omega_p) = (0.20, 0.30, 0.30, 0.20)$, assigning larger weights to editing success and prompt following than to each of the other dimensions.

Among the three reported settings, the default achieves the highest human agreement and bootstrap rank correlation. Under the scoring rules in Section 4.2, the edit-heavy setting changes the edit-failure

Table 11: Sensitivity to CV-Judge dimension weights.

Configuration	$\mathrm{Acc}_H \uparrow$	Spearman $\rho_S \uparrow$
Uniform $(0.25, 0.25, 0.25, 0.25)$	78.1%	0.87
Default (ours) $(0.20, 0.30, 0.30, 0.20)$	**82.6%**	**0.94**
Edit-heavy $(0.15, 0.40, 0.30, 0.15)$	79.8%	0.91

threshold from $\omega_e \times 1000 = 300$ to 400, and the corresponding final-score cap from $(\omega_e + \omega_i) \times 1000 = 600$ to 700. The higher threshold flags a wider range of edit scores as failures, while the higher cap permits a larger final score for those failures. Because the weights, thresholds, and caps change together, the table does not isolate the contribution of any one change.

O Appendix: CV-Agent Module Ablation

All compared configurations use nano banana as their editing backend. The agent uses Gemini 2.5 Pro for planning and internal verification. The Understanding-plus-Planning configuration executes a multi-step plan, while the full agent runs for at most three execution–verification rounds and returns the best-so-far candidate, following the procedure in Section 5. Table 3 compares cumulative configurations under Active Elo: the single-pass editor scores 1056, adding Understanding yields 1089, adding Planning yields 1118, and the full configuration with closed-loop execution yields 1145. The corresponding increases in point estimates are $+33$, $+29$, and $+27$, with a total difference of $+89$ between the full configuration and its single-pass base editor.

These results describe the configurations as a whole. They do not separately quantify the effects of additional editing calls, verification feedback, or candidate selection, and therefore do not establish an independent causal contribution for each module. No paired confidence intervals for the reported score differences are provided here; the point estimates alone do not establish statistically significant stage-wise improvements.

The same-editor difference of $1145 - 1056 = 89$ is distinct from the cross-system difference of $1145 - 1127 = 18$ between CV-Agent and nano banana pro. Both are differences in Active Elo point estimates. Neither comparison, as reported here, establishes a statistically separated performance tier; the marginal confidence intervals of individual systems do not substitute for an interval or test of the corresponding paired difference.

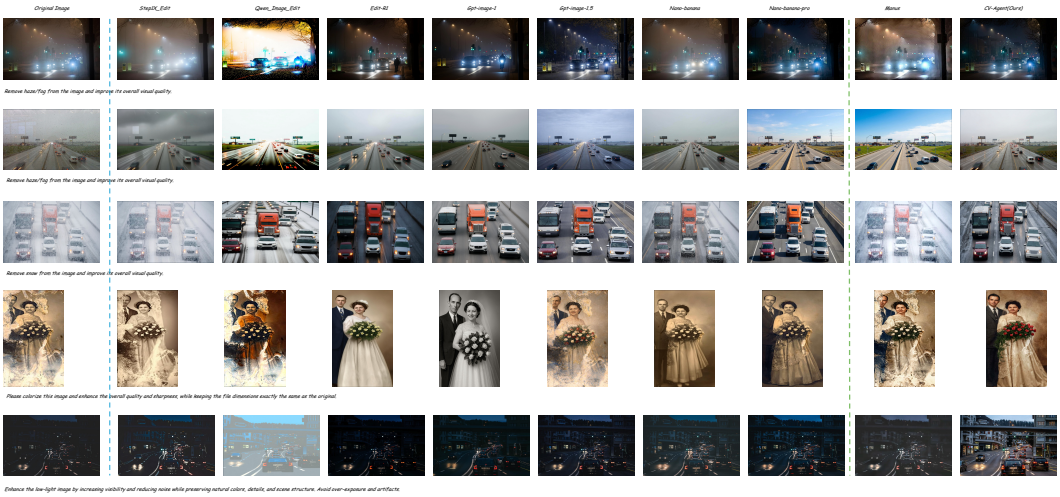

Figure 6: **Qualitative Comparisons on Low-Level Tasks.**



Figure 7: **Qualitative Comparisons of Failure Cases.**