CV-Arena: An Open Benchmark for Instructional Computer Vision Problem Solving with Human-AI Collaborative Preferences

Fangzhou Lin[1,2,3], **Peiran Li**[1], **Lingyu Xu**[2], **Wenjing Chen**[1], **Qianwen Ge**[4], **Shuo Xing**[1], **Mingyang Wu**[1], **Xiangbo Gao**[1], **Siyuan Yang**[1], **Kazunori Yamada**[3], **Ziming Zhang**[2], **Haichong Zhang**[2], **Zhen Dong**[5], **Ming-Hsuan Yang**[6], **Zhengzhong Tu**[1⋆]

[1]Texas A&M University [2]Worcester Polytechnic Institute [3]Tohoku University
[4]Georgia Institute of Technology [5]NVIDIA [6]UC Merced

⋆Corresponding Author: tzz@tamu.edu.

Project Website: 4kagent.github.io

Abstract

Instruction-guided image editing is becoming a general interface for visual work, yet existing benchmarks still focus largely on narrow appearance edits and do not fully capture the diversity of real-image tasks 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 at professional scales. 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 while preserving human fidelity, we propose Active Elo, a human-AI collaborative preference protocol that leverages CV-Judge, a logic-gated, multi-dimensional VLM evaluator, to reject clear failures and resolve high-confidence comparisons; and to route 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 lightweight agentic model that combines planning, editing, and verification, and demonstrate 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, that is, given an image and a natural-language instruction, a system is expected to change only what is requested while preserving the rest [81, 77]. Recent multimodal generative models, including vision-language models (VLMs) and unified generative models [71, 80, 26, 70, 45], have made this interface increasingly practical. Both proprietary systems [51, 12] and open models [41, 77, 74] are now being used for everyday editing and increasingly complex visual workflows [48, 52, 42].

Preprint.

However, most prior work [29, 86, 47] still formulates instruction-guided editing as a relatively narrow set of appearance-centric or stylistic transformations, which only partially reflects the diversity of professional real-image workflows [77, 81, 57, 5, 36, 56]. We argue that this framing is too restrictive, and thereby define **instructional computer vision problem solving (iCVPS)** as a broader formulation of image editing. 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 physically consistent lighting, or perform structure-preserving outpainting [46, 30, 68, 90, 25, 82, 4, 61, 31, 85]. 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 [69]. Second, existing evaluation protocols are 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, current benchmarks rarely stress-test the multi-domain workload required by professional workflows, as we defined above, including restoration, computational photography, physically grounded composition, semantic manipulation, typography recovery, and geometry-driven structural edits. [50, 1, 63, 55].

To close this gap, we introduce **CV-Arena**, an open benchmark that targets a diverse set of iCVPS tasks that naturally fit the image–instruction interface, spanning restoration and enhancement, computational photography, physically grounded composition, semantic manipulation, geometry and structural control, and typography recovery. Crucially, CV-Arena focuses on high-resolution, in-the-wild images whose content and quality resemble those encountered in real visual workflows, making it more realistic than prior benchmarks [77, 81, 5, 36, 56], whose image sizes are mostly small (e.g., 512). 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 data across diverse high-resolution settings.

A second challenge is *scalable and reliable evaluation* for CV-Arena. Classical image quality metrics such as PSNR and SSIM [35, 72] ignore instruction adherence and semantic preservation, while embedding-based metrics capture only partial signals for high-fidelity professional edits [59]. Recent benchmarks increasingly adopt VLM-as-a-judge for scalable scoring [77, 75], 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 [11] is more faithful, but costly, difficult to scale, and vulnerable to low-quality voting in crowdsourced settings [87, 32]. 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 and aggregates both human and AI decisions through reliability-weighted Elo updates. This design concentrates human effort on the most informative cases while preserving scalable coverage across models, tasks, and high-resolution outputs, enabling stable comparison of both single-pass editors and agentic systems under fixed annotation budgets.

Beyond benchmarking existing editors, we also study whether agentic reasoning can improve iCVPS. To this end, we build **CV-Agent**, a lightweight 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. Although simple, this baseline helps validate an important finding of CV-Arena: many failures are not caused only by image generation quality, but by missing planning, constraint checking, and self-verification. 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.

Table 1: **Comparison of Existing iCVPS Benchmarks.** #Size and #Tasks represent the number of samples and editing types. 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 demonstrates the evaluation protocols used in the benchmark.

Dataset	#Size	#Tasks	Max Res. (px)↑	Real Image	Physics	Reasoning	Low Level	Complex	Metrics
MagicBrush [81]	10K	5	500	✓	✗	✗	✗	✓	L1, L2, CLIP, DINO
InstructPix2Pix [3]	313K	4	512	✗	✗	✗	✗	✗	CLIP
HQ-Edit [29]	197K	6	≥768	✗	✗	✗	✓	✗	GPT
SEED-Data-Edit [22]	3.7M	6	768	✗	✗	✗	✗	✓	N/A
UltraEdit [86]	4M	9	512	✗	✗	✗	✓	✗	L1, L2, CLIP, DINO
AnyEdit [78]	2.5M	25	512	✓	✓	✓	✗	✓	L1, CLIP, DINO
ImgEdit [77]	1.2M	13	≥ 1280	✓	✗	✓	✗	✗	GPT
CV-Arena	12K	16	≥ 2048	✓	✓	✓	✓	✓	GPT + Human

- **CV-Agent,** a lightweight agentic baseline that decouples high-level planning and verification from low-level image manipulation, demonstrating that closed-loop reasoning can improve instruction following and constraint satisfaction in professional-grade visual editing.

2 Related Work

Real-world visual understanding has driven AI progress since MNIST and ImageNet [38, 16, 66, 62], but while recognition-oriented tasks have largely saturated [19, 49, 91], higher-tier objectives involving image realism, visual naturalness, and the plausibility of edits [65, 40, 77, 9] remain far from solved. Existing instructional editing benchmarks reflect this gap: object-insertion datasets such as iHarmony4 [13] and ObjectDrop [34, 73] reduce the task to appearance harmonization or static placement, ignoring dynamic interactions with deformable media; semantic editing benchmarks such as MagicBrush [81] and RefCOCO-Edit [43] conflate insertion, replacement, and reconstruction, blurring evaluation signals; and geometry-related edits in MagicBrush, InstructPix2Pix, and AnyEdit [81, 3, 78] are entangled with appearance changes, while ImgEdit [77] frames complexity through interaction length rather than structural constraints. Typography and UI recovery [58, 20] are similarly under-served despite their importance in professional workflows. CV-Arena addresses these gaps with a geometry- and physics-aware task design that isolates dynamic interaction, semantic manipulation, structural transformation, and typography restoration as first-class categories on real, high-resolution images. 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 distinct tasks. By integrating physical interaction and geometric constraints alongside traditional restoration tasks, CV-Arena spans the full spectrum from low-level pixel recovery 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 legality, 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 [46, 30, 68, 90, 25, 82, 4, 61, 31, 85]. These constraints in turn drive the image selection criterion: 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.

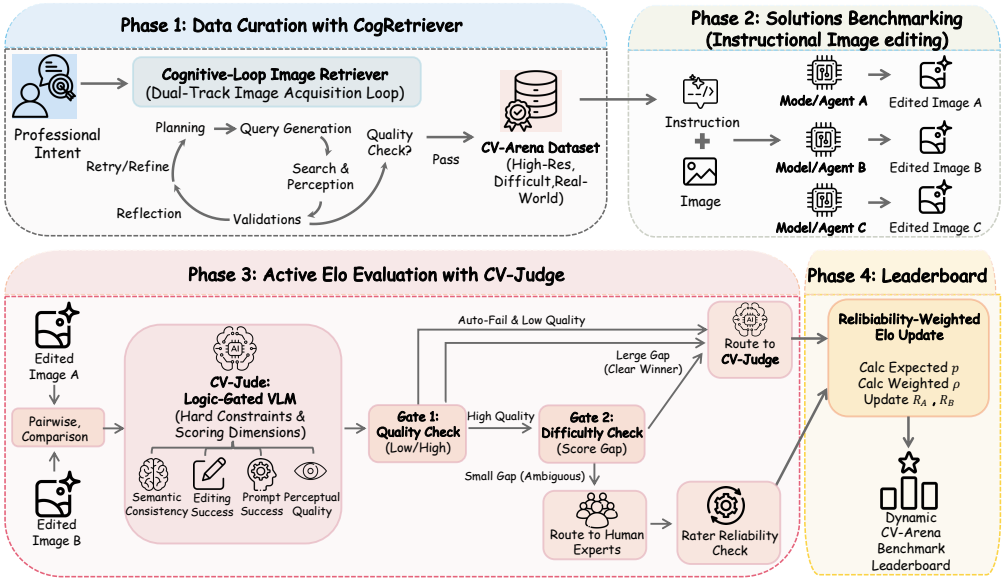

Figure 1: **The Overall Pipeline.** The framework starts with data curation, where CogRetriever constructs a professional-grade dataset. Then, it is followed by model/agent benchmarking and an Active Elo Evaluation, where CV-Judge generates scores and filters outputs using two-gate constraints, while routing ambiguous and high-quality comparisons to human experts. Final rankings are produced through Active Elo with a reliability-weighted update mechanism.

3.2 Task Design and Taxonomy

The taxonomy is designed that collected images are guided by professional editing intents rather than organized post hoc. 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 distribution and instruction keywords. 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 task definitions are provided in Appendix B.

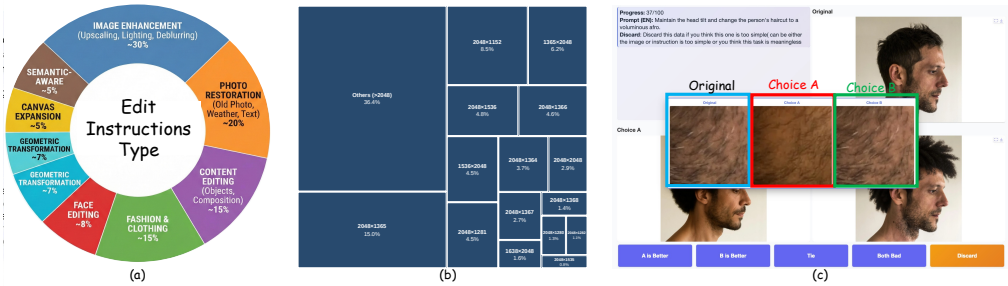

Figure 2: **Dataset statistics and User Interface.** From left to right: (a) the data composition across different sources; (b) shows the image-resolution distributions; and (c) a *zoom-in* function to check details during human rating.

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 T 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 writes a reflection m_{t+1} if the pool fails to reach $K_p = 3$ qualified samples. The final set is $\mathcal{X}^* = \mathrm{TopK}_{\mathbf{x} \in \mathcal{P}_t} s(\mathbf{x}; \mathbf{I_i})$. Full algorithmic details and hyperparameters are in Appendix C.

All retrieved images undergo automatic filtering for legality (Creative Commons rights filter for the Base Track; `cc_publicdomain` and `cc_attribute` restrictions via Google Custom Search API for the Agentic Track), near-duplicate removal, and low-quality rejection. Surviving pairs are then verified by human experts who check task-category match, target-region visibility, instruction feasibility, and the existence of a consistent success criterion, using the interactive *zoom-in* tool (Figure 2c) for detail-sensitive cases. Complete filtering criteria, traceability logs, and the human-verification protocol are detailed in Appendix D.

4 Evaluation: Active Elo with CV-Judge

We evaluate instructional computer vision problem-solving models through pairwise comparisons under identical conditions, aiming to obtain a reliable ranking rather than only absolute quality scores. The bottom of Figure 1 summarizes the overall evaluation pipeline. Our evaluation stack consists of two components: **CV-Judge**, a multi-modal evaluation protocol for instructional image editing, and **Active Elo**, a human-AI collaborative ranking framework that allocates expert annotation to ambiguous comparisons and aggregates mixed supervision through reliability-aware updates.

4.1 Preliminaries: Arena and Elo Ranking

Arena-style evaluation has become a standard protocol for comparing open-ended generative systems [11]: rather than assigning absolute scores, annotators provide blinded pairwise preferences over two outputs produced under the same input, which is typically more stable when outputs are diverse and hard to calibrate on a universal scale. An Elo-style system then converts these wins and losses into a global leaderboard: each model m maintains a rating R_m, and the win probability of A over B is a monotonic function following the Bradley-Terry-Luce model [2]. CV-Arena adopts this pairwise ranking view but modifies the standard protocol to account for expert annotation cost and the varying reliability of automatic judgments.

4.2 Active Elo with Human-AI Collaboration

Pairwise sampling. 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

$$\hat{x}_{i,m} = \mathrm{Edit}(\mathbf{x_i}, \mathbf{I_i}; m). \tag{1}$$

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. The evaluator produces a scalar score

$$s_{i,m} := \text{CV-Judge}\left(\mathbf{x_i}\mathbf{I_i}, \mathrm{Edit}(\mathbf{x_i}, \mathbf{I_i}; m)\right), \tag{2}$$

and induces a binary preference outcome $z_{i,A,B} \in \{0, 1\}$, where $z_{i,A,B} = 1$ indicates $\hat{x}_{i,A} \succ \hat{x}_{i,B}$, i.e., $s_{i,A} \geq s_{i,B}$. Human-routed pairs follow the same blinded pairwise format, but the final outcome is determined by expert preference rather than the automatic score difference.

CV-Judge evaluation. Evaluating instructional image editing requires checking whether an edited image faithfully follows a given instruction while preserving the original image content that should remain unchanged. Given an original image a, an instruction I, and an edited image A, CV-Judge operates original image, instruction and an edited image to produces a structured evaluation consisting of a scalar score, a binary success flag, and four auxiliary dimension scores retained for analysis and debugging. The four dimensions are 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 actually 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, or structural distortions. Together, these dimensions disentangle the correctness of editing from perceptual appearance.

We denote the four dimension scores as S_{sem}, S_{edit}, S_{prompt}, and S_{perc}, respectively. Each dimension is internally scored on $[0, 1000]$, and the initial overall score is computed as a weighted sum:

$$S_{\text{init}} = \omega_s \, S_{\text{sem}} + \omega_e \, S_{\text{edit}} + \omega_i \, S_{\text{prompt}} + \omega_p \, S_{\text{perc}}. \tag{3}$$

The weighting prioritizes correct realization of the instruction over purely perceptual improvements, preventing visually pleasing but incorrect edits from receiving high scores. To enforce logical consistency, CV-Judge applies hard constraints: if the core edit is largely unsuccessful ($S_{\text{edit}} < \omega_e \cdot 1000$), the final score is capped at $(\omega_e + \omega_i) \cdot 1000$ and marked unsuccessful; if semantic consistency or perceptual quality is severely degraded ($S_{\text{sem}} < \omega_s \cdot 1000$ or $S_{\text{perc}} < \omega_p \cdot 1000$), the score is capped at $(\omega_p + \omega_s) \cdot 1000$ and marked unsuccessful. An edit is considered successful only when all dimensions exceed moderate thresholds, ensuring both correctness and usability. We denote the final score after the capping operation as S. In our implementation, CV-Judge is instantiated with GPT-4o as the backbone VLM; cross-VLM sensitivity and dimension-weight sensitivity are reported in Appendix I and Appendix M, respectively.

Human-AI routing. Pure VLM judging scales well but can be unreliable on subtle comparisons, while human judgments are high-fidelity but expensive. We therefore use a two-gate routing policy to decide which pairs should be sent to human experts. For a pair (A, B) on instance i, define the score gap$g_i(A, B) = |s_{i,A} - s_{i,B}|$. We route the pair to human annotation iff

$$\min(s_{i,A}, s_{i,B}) \geq \tau \quad \text{and} \quad g_i(A, B) < \Delta. \tag{4}$$

The *quality gate* $\min(\cdot) \geq \tau$ avoids spending human budget on obvious failure regimes where both outputs are unusable. The *ambiguity gate* $g < \Delta$ targets cases where the automatic judge is least reliable and where additional supervision most improves the ranking. Pairs that do not pass the routing condition are resolved automatically by CV-Judge. Appendix F provides an information-per-cost interpretation of this routing rule. Empirically, AI-human agreement rises monotonically with g, from 56.3% at $g < 50$ to 94.8% at $g \geq 200$ (Appendix J), confirming that the routing policy concentrates human effort where the VLM is least reliable. The gate is also task-adaptive: deferral rates range from 46.8% for geometry-driven warping down to 26.2% for restoration (Appendix K), showing that the policy automatically reallocates supervision to harder task families.

Reliability-weighted Elo update. Each model m maintains an Elo rating R_m, which is updated online after each pairwise comparison. For a match between A and B, we model the probability that A beats B following the Bradley-Terry-Luce model [2]:

$$p_{AB} = \sigma\left(\frac{R_A - R_B}{S_{AB}}\right), \tag{5}$$

where σ is the sigmoid function and $S_{AB} = \frac{s_{i,A} + s_{i,B}}{2}$ is an instance-dependent scale derived from the two CV-Judge scores. To combine heterogeneous supervision, we downweight noisy outcomes with a credibility weight $\rho \in [0, 1]$. We model a rater with reliability $q \in [0, 1]$ as producing the correct preference with probability q and a random guess otherwise. Given $(p_{AB}, z_{i,A,B})$, define

$$w = \begin{cases} p_{AB}, & z_{i,A,B} = 1, \\ 1 - p_{AB}, & z_{i,A,B} = 0, \end{cases} \qquad \rho = \frac{q\, w}{q\, w + (1-q)\frac{1}{2}}. \tag{6}$$

Human labels use $q \approx 1$, while AI-resolved matches use an instance-dependent reliability $q = q_{\text{AI}}(g_i(A, B))$ calibrated on a small held-out set (Appendix E).

We then update Elo ratings by

$$R_A \leftarrow R_A + K_r \, \rho \, \big(z_{i,A,B} - p_{AB}\big), \; R_B \leftarrow R_B - K_r \, \rho \, \big(z_{i,A,B} - p_{AB}\big), \tag{7}$$

where K_r is rater-dependent. We use a larger step size for human matches (K_H) and a smaller one for AI matches ($K_{AI} = \alpha K_H$).

This design leverages AI for scalability while preventing abundant but noisier AI supervision from overwhelming high-fidelity evidence. Appendix G connects this update to a rater-aware BT mixture objective.

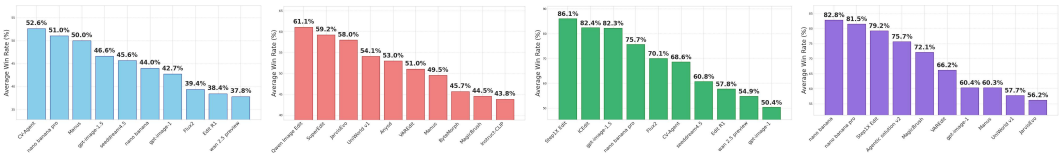

Figure 3: Average Win Rate Against Top-10 Models with three settings from left to right: Active Elo (Ours), Human Only, CV-Judge only, and EdiReward only(Assuming Uniform Sampling and No Ties).

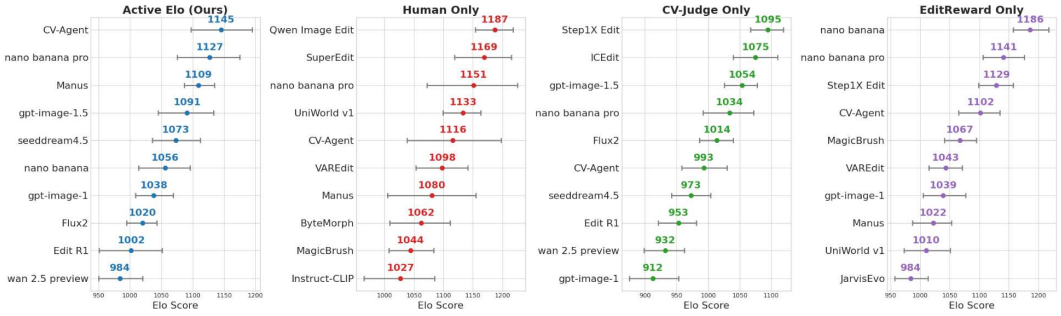

Figure 4: Bootstrap of Elo Estimates (1000 Rounds of Random Sampling) on Top-10 Models with four settings from left to right: Active Elo (Ours), Human Only, CV-Judge only, and EdiReward only.

Validation of the routing policy. We validate the two-gate routing policy through a budget-controlled ablation with a fixed number of expert comparisons (B_H). Following an LMArena-style blinded pairwise protocol [11], we construct a small, high-confidence human ground-truth test set $\mathcal{H}_{\text{test}}$ with 4 stable models, 8 curated task categories, and 10 expert annotators. We evaluate each routing strategy by agreement with humans (Acc_H) [54, 89] and leaderboard stability, measured by bootstrap Spearman rank correlation ρ_S [33, 17] and RankStd [18, 32]. As shown in Table 2, the proposed two-gate routing policy substantially improves human consistency while producing a stable ranking. Ablation details are provided in Appendix H.

5 CV-Agent: Simple Agentic Baseline

Table 2: **Validation of The Two-Gate Routing Policy.** Fixed human budget B_H.

Method	$\text{Acc}_H \uparrow$	$\rho_S \uparrow$	RankStd $\downarrow$
Human-only	60.7%	0.68	38.5
CV-Judge only	51.4%	0.63	23.2
Quality-only gate	68.8%	0.81	27.9
Ambiguity-only gate	73.2%	0.75	26.7
Two-gate (Ours)	**82.6%**	**0.94**	**22.3**

In addition to evaluating standalone editing models, we introduce a simple agentic editing baseline that decouples high-level reasoning from low-level image manipulation. The baseline is powered by strong LVLMs and off-the-shelf expert editors [23, 15] and follows a lightweight ReAct-style loop [76]; it is modular and requires no additional supervision or task-specific tuning. Although deliberately minimal, CV-Agent serves as a paradigm-validating baseline; a per-stage module ablation isolating Understanding, Planning, and Closed-Loop Refinement is reported in Appendix N. The pipeline proceeds in three stages:

Stage❶: Understanding. Conditioned on $(\mathbf{x}, \mathbf{I})$, the VLM (gemini-2.5-pro [23]) rewrites I into a precise, executable instruction and extracts the required visual changes and constraints. The output is a compact task specification that reduces ambiguity while preserving intent.

Stage❷: Planning. The LVLM generates a structured plan. It also predicts whether the edit should be executed in one step or many, and sets a step budget capped by T to prevent unbounded iteration.

Stage❸: Closed-loop editing. For step t, the editor (nano banana pro [15]) applies an edit to the current image using a step-specific prompt, producing A_t. The LVLM then evaluates A_t against $(\mathbf{x}, \mathbf{I})$ and outputs (i) a scalar quality score, (ii) a success indicator, and (iii) brief corrective feedback if needed. The loop stops early if the judge declares success; otherwise, it continues until $t = T$. The agent tracks the highest-scoring intermediate result and returns it as final output.

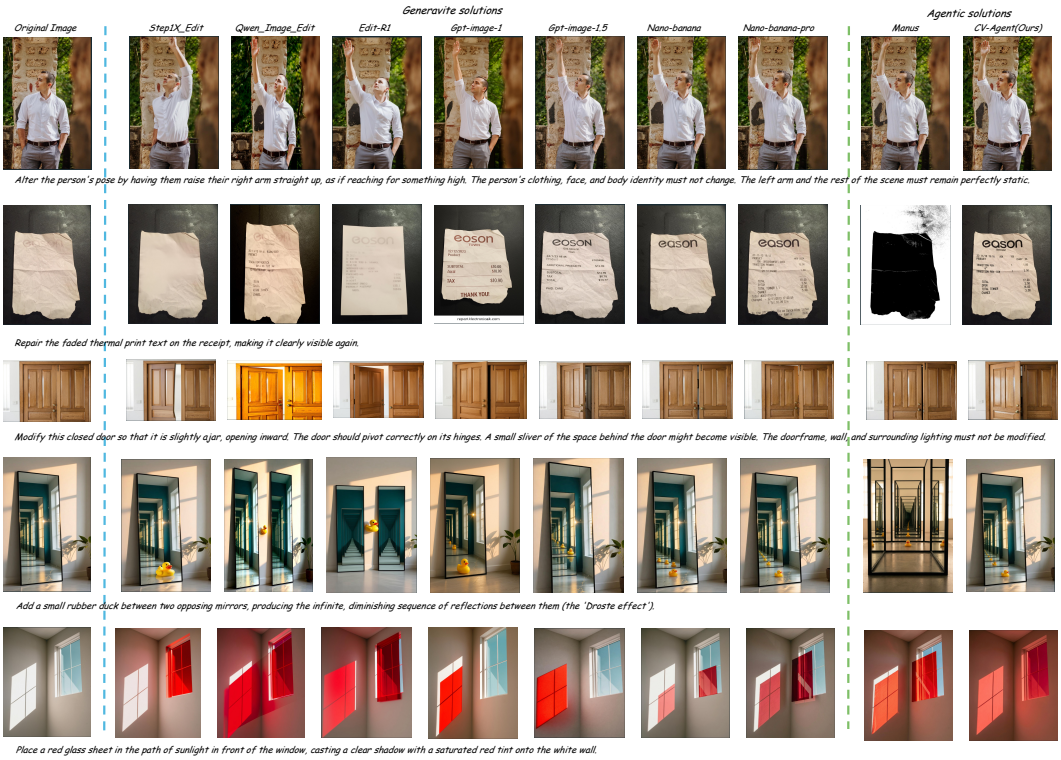

Figure 5: **Qualitative comparison among different editing solutions with reasoning and complex tasks**. Our proposed simple baseline CV-Agent consistently produces more faithful, constraint-satisfying edits, preserving non-target content and structure while better following the instruction than strong single-pass editors.

6 Experiments

We benchmark a broad set of instructional image editing systems on CV-Arena, including both single-pass editors and agentic solutions. Our evaluation follows the *Active Elo* introduced in Section 4.2, with three reference settings (CV-Judge only, Human-only [11, 32], and EditReward Only [75]) to isolate the effect of evaluation strategy. We first describe the evaluated models and experimental setup (Section 6.1), and then report quantitative rankings (Table 3) and qualitative analysis (Section 6.2).

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 [51], gpt-image-1.5 [53], nano banana [14], nano banana pro [15], Flux2 [37], and Seedream 4.5 [60], wan 2.5 preview [67]. Open-source baselines include Edit-R1 [41, 77], VAREdit [47], ICEdit [83], AnyEdit [78], Instruct-CLIP [7], Step1X-Edit [44], MagicBrush [81], UniWorld [41], Qwen Image Edit [74], ByteMorph [6], and SuperEdit [39, 24]. We also evaluate three agentic systems: Manus 1.6 [48], JarvisEvo [42], and our CV-Agent.

Inference protocol Editing is performed at each model's *native output resolution* under their default/highest-quality inference setting. Unless otherwise stated, we do not apply post-processing that could alter the outputs, ensuring that measured differences reflect model behavior.

Human evaluation interface Human comparisons are conducted on uniformly resized renderings for *display only* to eliminate perceptual advantages from differing native resolutions; this does not affect any model output or any evaluation input to the judge. Importantly, because our protocol routes human primarily to *high-quality and close* pairs, annotations often hinge on subtle local artifacts (e.g., detail legibility, boundary consistency, texture recovery). We therefore implement an interactive *zoom-in* tool that allows humans to inspect fine details via mouse-controlled magnification (as shown in Figure 2 c). This capability is critical for reliably differentiating near-tied outputs, and improves annotation fidelity in the regime our protocol explicitly targets.

Table 3: **Top-5 leaderboard comparison across evaluation settings.** We compare Active Elo (ours), human-only evaluation, CV-Judge-only evaluation, and EditReward-only evaluation. Active Elo combines scalable automatic judgments with selective expert supervision and aggregates mixed outcomes through reliability-weighted Elo updates.

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
seeddream4.5	1073	+39/-37	CV-Agent	1116	+82/-78	Flux2	1014	+26/-28	MagicBrush	1067	+28/-26

Evaluation settings We report three complementary evaluation settings: (i) **Active Elo** (ours), as described in Section 4.2; (ii) **Human-only**, where comparisons are resolved by humans [11, 32] (within budget); (iii) **CV-Judge only**, where all comparisons are resolved by our proposed automated judge, and (iv) **EditReward only**, where all comparisons are resolved by a concurrent automated judge [75]. These controlled references enable an apples-to-apples analysis of how hybrid routing and reliability-aware aggregation affect leaderboard quality and stability.

6.2 Benchmark Results

Leaderboard. We rank solutions using the Human-AI Collaborative Preferences Active Elo protocol via credibility-weighted updates (Section 4.2). Our main leaderboard is shown in Table 3 (Please refer to the full 21 solutions in Table 4 in the supplementary). In addition to our setting, we report the CV-Judge only, human-only, and EditReward Only leaderboard as a reference baseline. As shown in Table 3, Human Only, CV-Judge, and EditReward alone cannot reflect the true competence of different solutions, whereas our Active Elo provides the most reliable ranking. We also include more results in supplementary, please refer Figure 6 and Figure 7.

Discussion and Analysis. As shown in Figure 3, we observe that the average win rates of the Top-10 ranked solutions in Active Elo and Human-only are similar, none exceeding 80%. This indicates that there is no dominant, highly powerful solution in our CV-Arena Dataset at the current time. While CV-Judge and EditReward only clearly show untrustworthy results with exceeding 85% in an open source model (Step1X Edit [44]) or messed up ranking (nano banana [14] is better than nano banana pro [15]), showing pure AI cannot handle our dataset. Moreover, we also included more analysis in Figure 4, the error bar of the Elo Score shows the reliability of Active Elo compared with Human Only, and is almost the same as CV-Judge only and EditReward only. As a result, our dynamic CV-Arena Benchmark more faithfully reflects human preference in the subtle, high-stakes regime that is most relevant for professional-grade instructional image editing. We also decompose CV-Judge scores along the four dimensions (S_{sem}, S_{edit}, S_{prompt}, S_{perc}) for the top solutions. Agentic methods (CV-Agent, Manus) lead on S_{edit} and S_{prompt}, while strong single-pass generative models (nano banana pro, gpt-image-1.5, seeddream4.5) achieve higher S_{perc}. This separation suggests purely generative pipelines retain a perceptual edge that becomes decisive only when instruction adherence is otherwise comparable. Full per-dimension scores are reported in Appendix K.

Comparison with Traditional Metrics. We also evaluate embedding-based similarity (CLIP-I [27], DINO [79]) and text-image alignment (CLIPScore [28]) on a $\sim$1K subset. Top models cluster within a 3.4% range under CLIP-I/DINO, and although paired bootstrap tests confirm many pairwise differences are statistically significant, the resulting ranking correlates only weakly with Active Elo (Spearman $\rho = 0.50$) and produces rank reversals among competitive models, because input-output similarity rewards timid edits regardless of whether the instruction was actually realized. CLIPScore is substantially better aligned with human judgment ($\rho = 0.90$) but still cannot resolve fine-grained perceptual artifacts and hard constraint violations that are decisive in our high-resolution professional setting. These observations support the use of traditional metrics cannot serve as a stand-alone leaderboard for this constraint-heavy task. Fullanalyses are in Appendix L.

7 Conclusion

We introduce **CV-Arena**, an open benchmark designed to evaluate this capability at professional scales, together with **Active Elo**, a human-AI collaborative ranking protocol. Active Elo achieves substantially higher agreement with expert judgment and more stable leaderboards than VLM-only, reward-model-only, or budget-matched human-only baselines, and our simple **CV-Agent**, a neutral

closed-loop agentic baseline to complete the benchmark and enable fair, end-to-end evaluation under a unified protocol.

Limitations. The current 12K release, while sufficient for stable ranking, remains modest relative to the long tail of professional editing scenarios; scaling the dataset and broadening rare task coverage are left to future work.

References

[1] 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.

[2] 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.

[3] 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)*, pages 18392–18402, 2022.

[4] 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*, pages 2704–2713, 2025.

[5] 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*, 2025.

[6] 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*, 2025.

[7] Sherry X. Chen, Misha Sra, and Pradeep Sen. Instruct-clip: Improving instruction-guided image editing with automated data refinement using contrastive learning, 2025.

[8] 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, 2023.

[9] 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*, 2025.

[10] 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*, 2023.

[11] 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.

[12] 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.

[13] 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.

[14] Google DeepMind. Introducing gemini 2.5 flash image, our state-of-the-art image model, 2025.

[15] Google DeepMind. Introducing nano banana pro, 2025.

[16] 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*, pages 248–255. Ieee, 2009.

[17] 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.

[18] Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36:30039–30069, 2023.

[19] 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.

[20] 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*, pages 13104–13113, 2025.

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

[22] 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.

[23] Google. Gemini 2.5 pro model card, 2025. Accessed: 2026-01-11.

[24] 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*, 2025.

[25] 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*, pages 16071–16080, 2025.

[26] 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.

[27] Markus Hafner, Maria Katsantoni, Tino Köster, James Marks, Joyita Mukherjee, Dorothee Staiger, Jernej Ule, and Mihaela Zavolan. Clip and complementary methods. *Nature Reviews Methods Primers*, 1(1):20, 2021.

[28] 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*, pages 7514–7528, 2021.

[29] 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.

[30] 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*, pages 5734–5743, 2026.

[31] 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.

[32] 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.

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

[34] 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*, pages 3051–3060, 2025.

[35] Jari Korhonen and Junyong You. Peak signal-to-noise ratio revisited: Is simple beautiful? In *2012 Fourth International Workshop on Quality of Multimedia Experience*, pages 37–38, 2012.

[36] 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.

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

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

[39] 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.

[40] 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*, pages 12324–12333, 2023.

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

[42] 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*, 2025.

[43] 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*, pages 13128–13138, 2024.

[44] 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.

[45] 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.

[46] 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*, pages 12764–12777, 2025.

[47] 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.

[48] Manus Meta. Introducing manus 1.6: Max performance, mobile dev, and design view, 2025.

[49] 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.

[50] 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.

[51] OpenAI. Gpt image 1, 2025.

[52] OpenAI. Introducing chatgpt agent: bridging research and action, 2025. Accessed: 2026-01-26.

[53] OpenAI. The new chatgpt images is here, 2025.

[54] 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.

[55] Anthropic PBC. The claude 3 model family: Opus, sonnet, haiku.

[56] 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.

[57] 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.

[58] 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, pages 2119–2127, 2023.

[59] 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*, pages 8748–8763. PmLR, 2021.

[60] 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.

[61] 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, pages 9097–9105, 2026.

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

[63] 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.

[64] 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.

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

[66] 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.

[67] 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.

[68] 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*, pages 23539–23550, 2025.

[69] 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*, pages 13391–13397, 2025.

[70] 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*, 2025.

[71] 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.

[72] 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.

[73] 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*, pages 112–129. Springer, 2024.

[74] 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, 2025.

[75] 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*, 2025.

[76] 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.

[77] 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.

[78] 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*, pages 26125–26135, 2025.

[79] Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection. *arXiv preprint arXiv:2203.03605*, 2022.

[80] 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, 2024.

[81] 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.

[82] 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, pages 12888–12896, 2026.

[83] 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)*, 2025. arXiv:2504.20690.

[84] 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*, 2024.

[85] 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*, 2025.

[86] Haozhe Zhao, Xiaojian Shawn Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale. *Advances in Neural Information Processing Systems*, 37:3058–3093, 2024.

[87] 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*, pages 3359–3365, 2025.

[88] 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, 2024.

[89] 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.

[90] 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, pages 10752–10760, 2025.

[91] 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 dataset [38, 16], 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 [66, 21, 62, 84, 88]. With the rapid progress of modern architectures and large-scale training, many closed-form vision tasks have become largely saturated: such as image classification, semantic segmentation, and object detection [19, 49, 91]. However, tasks at a higher semantic and perceptual tier, involving notions such as image realism [65, 8], visual

naturalness [40, 10], and the plausibility of edits [77, 9], 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 [13], 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 [34, 73] 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 [64] 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 [81] and RefCOCO-Edit [43] 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 [81, 3, 78]; 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 ImageEdit [77], explicitly introduce *single-turn* and *multi-turn* editing formulations to better support complex editing behaviors, partially addressing the limitations of one-shot editing protocols. While this design improves task coverage, it still frames complexity primarily from the perspective of interaction length rather than the underlying geometric constraints.

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. It 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.

Typography and UI restoration targets the correction, reconstruction, or removal of textual and graphical elements embedded in real-world images [58, 20]. 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 the full definitions of the three signature task families that distinguish CV-Arena from prior benchmarks.

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 $|\mathcal{P}_t| < K_p$ (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 next iteration. Once a sufficient pool is reached, the final candidate set is taken as the top-K_p scoring elements:

$$\mathcal{X}^* = \mathrm{TopK}_{\mathbf{x} \in \mathcal{P}_t} \, s(\mathbf{x}; \mathbf{I_i}). \tag{8}$$

Iteration cap and termination. The closed loop terminates either when $|\mathcal{P}_t| \geq K_p$ or when a maximum iteration budget T is exhausted, in which case the instruction is flagged for manual review rather than silently producing low-quality samples.

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.

Table 4: Full Leaderboard comparison across different settings (21 Models).

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
seeddream4.5	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	seeddream4.5	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	seeddream4.5	920	+70/-70	VAREdit	804	+18/-22	seeddream4.5	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

Legality and copyright compliance. All imagery is retrieved through strictly filtered channels to ensure permissible usage. For manual acquisition in the Base Track, we use the `Creative Commons` rights filter in Google Images. The Agentic Track uses the Google Custom Search API with parameters restricted to `cc_publicdomain` and `cc_attribute` content.

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 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 experts further verify the remaining image-instruction pairs. The verification process checks four criteria: (i) whether the selected image matches the intended task category, (ii) whether the target region is visible, (iii) whether the instruction is feasible, and (iv) whether the expected edit can be judged consistently. For detail-sensitive cases, annotators use the *zoom-in* function shown in Figure 2c to inspect local regions such as text, boundaries, object parts, and fine structural details. Pairs that fail any criterion are either re-routed to manual repair or discarded; the remaining examples constitute the final 12K dataset.

This final verification step ensures that the retained examples are legally traceable, visually interpretable, and aligned with the professional task taxonomy of CV-Arena.

E Appendix: Calibrating AI Reliability from Score Gap

Our hybrid protocol relies on an instance-dependent reliability for AI-resolved comparisons, denoted by $q_{\mathrm{AI}}(g) \in [0, 1]$, where the score gap

$$g_i(A, B) = |s_{i,A} - s_{i,B}| \tag{9}$$

serves as a practical proxy for comparison ambiguity.

E.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
$$\hat{z}^{\mathrm{AI}}_{i,A,B} = \mathbb{I}[s_{i,A} \geq s_{i,B}], \tag{10}$$
- 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

$$a_{i,A,B} = \mathbb{I}\big[\hat{z}^{\mathrm{AI}}_{i,A,B} = z^{\mathrm{H}}_{i,A,B}\big] \in \{0,1\}. \tag{11}$$

E.2 Binned estimation and monotone fitting

We partition the observed gaps $\{g_i(A,B)\}$ into J bins $\{\mathcal{B}_j\}_{j=1}^{J}$ (e.g., equal-count bins for robustness). We compute the empirical AI reliability as the empirical probability of whether the AI preference agrees with human preference. In particular, the empirical AI reliability in bin j is

$$\hat{q}_j = \frac{1}{|\mathcal{B}_j|} \sum_{(i,A,B)\in\mathcal{B}_j} a_{i,A,B}. \tag{12}$$

Since reliability should be non-decreasing with g, we enforce monotonicity via either:

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

$$q_{\mathrm{AI}}(g) = \sigma\big(\beta(g - g_0)\big) \tag{13}$$

  where $\beta > 0$ controls the sharpness of the transition, and g_0 denotes the ambiguity threshold, aligned with the routing criterion used in Section 4.2.

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

E.3 Final reliability map used in ranking

The calibrated function $q_{\mathrm{AI}}(g)$ is used in the credibility weight ρ in the Elo updates (Section **??**). For AI-resolved comparisons we set

$$q = q_{\mathrm{AI}}\big(g_i(A,B)\big), \tag{14}$$

while for human-labeled comparisons we use $q \approx 1$ (practically, $q = 1 - \varepsilon$ with a small ε for numerical stability).

F Appendix: Two-Gate Selection as Cost-Effective Experimental Design

We provide an interpretation of the two-gate routing rule as an approximate cost-effective design choice: allocate scarce human budget to comparisons that are both (i) relevant to the benchmark objective (high-quality regime) and (ii) most informative for refining the ranking (ambiguous regime).

F.1 Information is concentrated in ambiguous comparisons

Consider an online ranking step comparing models A and B. Under Elo, the predicted win probability of A is

$$p_{AB} = \sigma\left(\frac{R_A - R_B}{S}\right), \tag{15}$$

where σ is the sigmoid function, $S = \frac{s_{i,A}+s_{i,B}}{2}$, which follows the BTL model. The informativeness of a single comparison can be measured by the variance of the Bernoulli outcome:

$$\mathrm{Var}(A \text{ is ranked over B} \mid p_{AB}) = p_{AB}(1 - p_{AB}). \tag{16}$$

This variance is maximized when $p_{AB} = 0.5$ (a toss-up) and decreases toward zero as p_{AB} approaches 0 or 1. Intuitively, observing the outcome of a close match provides more information for refining the ranking than observing a heavily favored model win as expected.

Our routing rule leverages the CV-Judge score gap

$$g_i(A,B) = |s_{i,A} - s_{i,B}| \tag{17}$$

as a proxy for comparison ambiguity. Smaller gaps indicate harder judgments for the automatic judge and greater uncertainty in the ordering. In such cases, human supervision provides the highest marginal benefit for improving ranking accuracy.

F.2 Mixture model for noisy pairwise labels

We adopt the observation model in Section 6. In particular, We introduce a latent variable $c \in \{0, 1\}$ indicating whether the observed label is *credible* ($c = 1$) or a random guess ($c = 0$). Given rater reliability $q \in [0, 1]$, we assume

$$P(c = 1) = q, \qquad P(z = 1 \mid c = 1) = p_{AB}, \qquad P(z = 1 \mid c = 0) = \tfrac{1}{2}. \tag{18}$$

This yields the marginal observation model, i.e.,

$$P(\mathbb{I}(A \succ B) \mid p_{AB}, q) = q\, p_{AB} + (1 - q)\tfrac{1}{2}. \tag{19}$$

matching Section 6. Human labels correspond to $q \approx 1$, while AI labels use $q = q_{\mathrm{AI}}(g)$.

F.3 Rater reliability and effective information per cost

Let us define $p_{\text{eff}} = q\, p_{AB} + (1 - q)/2$. The uncertainty of the observed label is governed by its variance $p_{\text{eff}}(1 - p_{\text{eff}})$, while the *informativeness* about p_{AB} is attenuated when q is small, since p_{eff} moves toward $1/2$ regardless of p_{AB}. Intuitively, if the judge is near-random on a subset of cases (low q), labels from that judge contribute little useful signal.

Let c_{AI} and c_{H} denote the per-comparison costs for AI and human supervision, respectively. A cost-effective design allocates human labels to cases where the expected gain in ranking quality per unit cost is larger. A simple proxy criterion is:

$$\text{prefer human if} \quad \frac{\text{info}(q_{\mathrm{H}}, p_{AB})}{c_{\mathrm{H}}} > \frac{\text{info}(q_{\mathrm{AI}}(g(A, B)), p_{AB})}{c_{\mathrm{AI}}}, \tag{20}$$

where $\text{info}(q, p_{AB}) = (q\, p_{AB} + (1 - q)/2)((1 + q)/2 - q\, p_{AB})$ increases with both ambiguity (near $p_{AB} = 0.5$) and rater reliability.

Because $q_{\mathrm{H}} \approx 1$ and $q_{\mathrm{AI}}(g(A, B))$ decreases as g becomes small (Appendix E), human labeling becomes comparatively more valuable precisely in the ambiguous regime. This motivates the ambiguity gate $g(A, B) < \Delta$.

F.4 Why the quality gate is necessary

The benchmark objective emphasizes professional-grade competence, and low-quality outputs often lead to low-information human outcomes (e.g., "both unusable") that do not refine fine-grained ordering among competitive systems. We therefore restrict human effort to the regime where both candidates are at least moderately viable:

$$\min(s_{i,A}, s_{i,B}) \geq \tau. \tag{21}$$

This can be viewed as multiplying the information-per-cost objective by a relevance indicator $u_i = \mathbb{I}[\min(\cdot) \geq \tau]$, effectively focusing annotation budget on comparisons aligned with the benchmark's evaluation regime.

G Appendix: Online-EM Interpretation of Reliability-Weighted Elo

We provide an interpretation of the credibility-weighted Elo update as stochastic optimization of a rater-aware mixture objective. This is an *interpretation* that explains why the weight ρ is a principled way to combine heterogeneous supervision; the benchmark itself only requires the update rule in Section 4.

G.1 Posterior credibility

Let us denote the observed outcome $z := \mathbb{I}(A \succ B)$, and that $z \in \{0, 1\}$, define

$$w = \begin{cases} p_{AB}, & z = 1, \\ 1 - p_{AB}, & z = 0. \end{cases} \tag{22}$$

By Bayes' rule, the posterior probability that the label was generated from the credible component is

$$\rho := P(c = 1 \mid z, p_{AB}, q) := \frac{P(c = 1, z \mid p_{AB}, q)}{P(c = 1, z \mid p_{AB}, q) + P(c = 0, z \mid p_{AB}, q)} = \frac{q\, w}{q\, w + (1 - q)\frac{1}{2}}. \tag{23}$$

This is exactly the credibility weight used in our Elo updates.

G.2 Weighted log-likelihood and stochastic updates

Consider the conditional log-likelihood of the credible component for a single comparison:

$$\ell(R_A, R_B) = z \log p_{AB} + (1 - z) \log(1 - p_{AB}). \tag{24}$$

A credibility-weighted objective corresponds to maximizing $\rho\, \ell(R_A, R_B)$ online. By combining the definition of p_{AB} in (15), we can compute the derivative of l as follows

$$\frac{\partial \ell}{\partial (R_A - R_B)} = \frac{1}{S}\,(z - p_{AB}), \tag{25}$$

then update Elo ratings by

$$R_A \leftarrow R_A + \eta\, \rho\, (z - p_{AB}), \qquad R_B \leftarrow R_B - \eta\, \rho\, (z - p_{AB}), \tag{26}$$

which matches the reliability-weighted Elo update in Section 4 (with η corresponding to K_r). Rater-dependent step sizes (K_H vs. K_{AI}) can be viewed as an additional control that caps the influence of noisier supervision sources.

This perspective clarifies why ρ is preferable to using the score gap alone as a weight: ρ jointly captures (i) rater reliability q and (ii) match difficulty through p_{AB}, and therefore directly controls how much each observed outcome should move the online ranking.

H Appendix: Two-Gate Routing Policy Ablations

We test whether our Active Elo design choices are *necessary* for producing a faithful and stable leaderboard. We ablate (i) *routing* (which pairs receive human labels), (ii) *reliability modeling* (whether AI trust must be instance-dependent), and (iii) *aggregation* (whether noisy supervision should be downweighted). Notation follows the main text: CV-Judge scores $s_{i,m}$, gap $g_i(A, B) = |s_{i,A} - s_{i,B}|$, binary preference $z \in \{0, 1\}$, Elo ratings R_m, and

$$p_{AB} = \sigma\left(\frac{R_A - R_B}{S}\right), \qquad \rho = \frac{q\, w}{q\, w + (1 - q)\frac{1}{2}}, \;\; w = \begin{cases} p_{AB}, & z = 1 \\ 1 - p_{AB}, & z = 0. \end{cases} \tag{27}$$

We follow standard pairwise practice and use binary preferences (no ties).

H.1 Ablation Settings

Human GT for Evaluation. We construct a small but high-confidence human ground truth (GT) as a set $\mathcal{H}_{\text{test}}$ to evaluate ranking faithfulness. Following the standard [32] protocol (blinded pairwise comparisons under identical display conditions), we (i) select 4 empirically stable models, (ii) curate 8 task categories that yield consistent discrimination, and (iii) recruit 10 expert annotators. After repeated checks for consistency, we aggregate human preferences with a human-only pairwise ranker (Elo/BT) to obtain a GT *ranking* and associated *scores*. GT is used only for evaluation.

Routing (human budget allocation). Under a fixed human budget B_H, 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 held-out comparison (A_k, B_k) as $\mathbb{I}[R_{A_k} > R_{B_k}]$ and compute agreement with human labels [54, 89]:

$$\mathrm{Acc}_H = \frac{1}{|\mathcal{H}_{\text{test}}|} \sum_{k \in \mathcal{H}_{\text{test}}} \mathbb{I}\big[\big(R_{A_k} > R_{B_k}\big) \iff \big(z_k^{\mathrm{H}} = 1\big)\big]. \tag{28}$$

Higher Acc_H indicates that the learned ranking better matches human pairwise preferences on the comparisons.

(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 $\pi^{(\text{full})}$ denote the ranking from the full comparison set under the same protocol. We report the average Spearman correlation between bootstrap and full-data ranks [33, 17]:

$$\rho_S = \frac{1}{B} \sum_{b=1}^{B} \mathrm{Spearman}\Big(\pi^{(b)}, \pi^{(\text{full})}\Big), \tag{29}$$

where B is the number of bootstrap replicates. Larger ρ_S indicates that the relative ordering of models is stable under finite supervision.

(3) Rank Std (Standard deviation of ranks / Bootstrap stability). Let $r_m^{(b)}$ be the rank of model m in bootstrap replicate b. The rank standard deviation for model m is

$$\mathrm{StdRank}(m) = \mathrm{Std}\big(\{r_m^{(b)}\}_{b=1}^{B}\big), \tag{30}$$

and we summarize stability by the average rank standard deviation across models:

$$\mathrm{RankStd} = \frac{1}{|\mathcal{M}|} \sum_{m \in \mathcal{M}} \mathrm{StdRank}(m), \tag{31}$$

where $\mathcal{M}$ is the model set. Lower $\mathrm{RankStd}$ indicates a more stable leaderboard (less sensitivity to the specific sampled comparisons) [18, 32].

As we can see in Table 2, two-gate routing improves sample efficiency under fixed B_H; it also improves both human agreement and stability relative to a constant trust level.

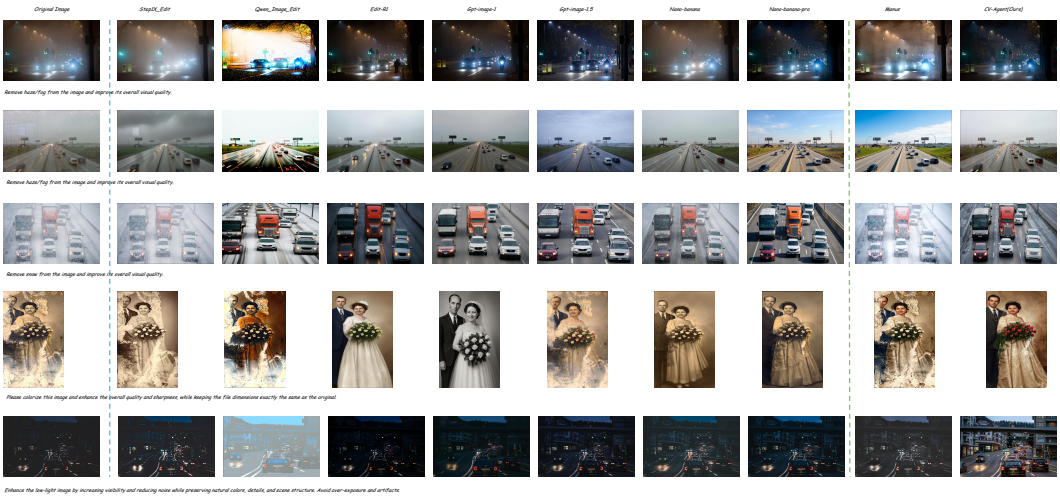

Figure 6: **Qualitative Comparison Among Different Editing Solutions with low level tasks**.

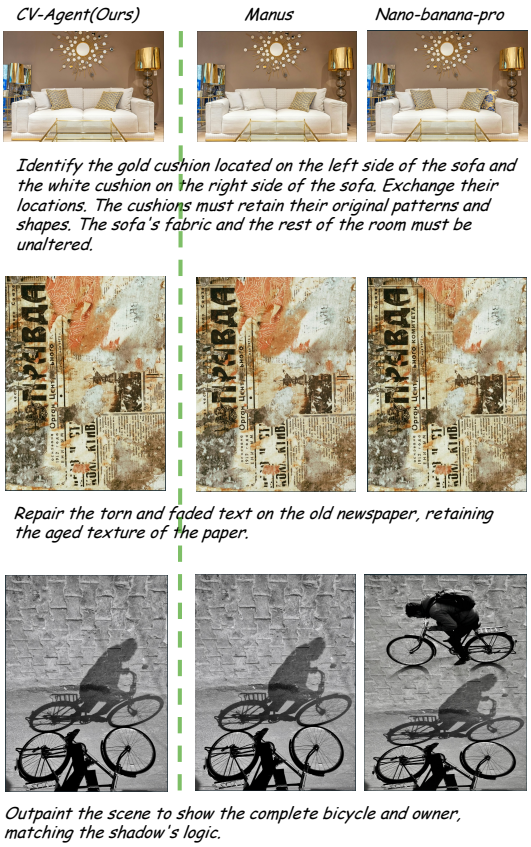

Figure 7: **Qualitative Comparison Among Different Editing Solutions with failure cases**.

I Appendix: CV-Judge VLM Backbone Sensitivity

CV-Judge is instantiated with GPT-4o as its backbone VLM, primarily due to the favorable balance between evaluation quality and API cost at the scale of CV-Arena. To verify that our findings are not artifacts of this specific choice, we conducted preliminary cross-VLM comparisons using alternative backbones from the GPT and Gemini families on a representative subset.

While per-category scoring distributions exhibit minor differences (notably for tasks requiring fine geometric reasoning), the overall Active Elo ranking remains largely stable across backbone choices. This robustness arises from three design choices working in concert: (i) the conservative quality gate operates only in the high-agreement regime ($g \geq 200$, 94.8% AI-human agreement; Appendix J); (ii) the ambiguity gate routes residual close-call cases to humans regardless of which backbone is used; and (iii) the reliability-weighted Elo update down-weights AI-resolved outcomes whose calibrated reliability is low. Together, these mechanisms localize backbone-specific biases to a small, gated portion of the budget rather than allowing them to dominate the leaderboard.

J Appendix: AI-Human Agreement Stratified by Score Gap

To diagnose where automatic CV-Judge decisions are reliable and where human supervision adds the most value, we stratify AI-human agreement by the score gap $g_i(A, B) = |s_{i,A} - s_{i,B}|$ on the held-out human GT set $\mathcal{H}_{\text{test}}$.

Agreement increases monotonically with the gap. This validates two key design choices simultaneously: (i) the ambiguity gate $g < \Delta$ correctly identifies the regime where CV-Judge alone is unreliable and routes those cases to humans, and (ii) the AI reliability function $q_{\text{AI}}(g)$ used in the credibility weight (Appendix E) is well-calibrated, since AI-resolved outcomes outside the routing window already exceed 90% agreement and can safely contribute to the Elo updates.

Table 5: AI-Human agreement and pair fraction by score gap.

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

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

K.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.

Table 6: 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
seeddream4.5	734	711	703	761

Two complementary patterns emerge: agentic solutions (CV-Agent, Manus) lead on S_{edit} and S_{prompt}, while single-pass generative models (nano banana pro, gpt-image-1.5, seeddream4.5) achieve higher S_{perc} but exhibit weaker instruction fidelity. This separation indicates that planning, verification, and closed-loop refinement most directly benefit instruction adherence, whereas purely generative pipelines retain a perceptual edge that is decisive only when instruction adherence is otherwise comparable.

K.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 7: 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%

Tasks that hinge on subtle constraint satisfaction (geometry, physics) generate more close-call comparisons and therefore consume proportionally more human budget; conversely, restoration tasks produce larger and more unambiguous quality gaps that CV-Judge resolves reliably. This adaptive allocation is an emergent property of the two-gate policy: human effort flows automatically toward task families where it is most informative, without any task-specific tuning.

L Appendix: Comparison with Traditional Metrics

For completeness, we evaluate three widely used reference-light metrics on a randomly sampled subset of $\sim$1K pairs: CLIP-I (input-output image similarity), DINO (visual feature similarity), and

CLIPScore (text-image similarity). We caution at the outset that CV-Arena samples lack ground-truth edited references, so CLIP-I and DINO can only measure preservation of input content, which is not uniformly desirable across our 16 task families (e.g., object insertion legitimately deviates from the input, while exposure correction should preserve it).

L.1 Aggregate Scores

Table 8: 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

CLIP-I spans only 0.857–0.891 (a 3.4% range) and DINO spans 0.809–0.842 (3.3%). The narrow margins limit their power to discriminate competitive solutions.

L.2 Paired Significance Testing

We ran paired bootstrap tests ($B = 10000$). With $N \approx 1000$, statistical power is high: CLIP-I yields $7/10$ significant pairs and DINO yields $6/10$. However, the resulting rankings are nearly identical across CLIP-I and DINO (CV-Agent $>$ Manus $>$ nano banana $>$ nano banana pro $>$ gpt-image-1.5) and Spearman-correlate only weakly with Active Elo ($\rho = 0.50$, $p = 0.39$). Two of the significant pairs are direct rank reversals relative to Active Elo, indicating that the disagreement is not statistical noise but systematic: input-output similarity rewards timid edits regardless of whether the instruction was actually realized.

L.3 CLIPScore: Better but Still Insufficient

CLIPScore correlates substantially better with Active Elo ($\rho = 0.90$): top-1 and bottom-1 match exactly, with only a single adjacent swap among the middle ranks. This confirms that text-image alignment is a more faithful proxy for instruction adherence than input-output similarity. Nevertheless, CLIPScore's text encoder lacks the resolution to detect (i) fine-grained perceptual artifacts (boundary inconsistencies, texture corruption) and (ii) hard constraint violations (e.g., introducing an object the instruction explicitly forbids), both of which are decisive in professional-grade settings. Active Elo's multi-dimensional CV-Judge combined with selective human routing captures both axes that no single embedding-based metric covers, while still using these metrics as useful supplementary diagnostics.

M Appendix: Hyperparameter Sensitivity

M.1 Routing Thresholds τ and Δ

The structural ablation in Table 2 already isolates the role of each gate. We supplement it with a finer-grained sweep around the default operating point $(\tau^\star, \Delta^\star)$.

Table 9: Sensitivity of Active Elo to the routing thresholds.

Configuration	Acc_H ↑	Spearman ρ_S ↑	RankStd ↓
Default $(\tau^\star, \Delta^\star)$	**82.6%**	**0.94**	**22.3**
Loose $(\tau^\star - 50, \Delta^\star + 50)$	80.1%	0.91	23.6
Tight $(\tau^\star + 50, \Delta^\star - 50)$	81.4%	0.92	22.8

Performance degrades gracefully under modest perturbations, confirming the protocol does not rely on knife-edge tuning.

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

The dimension weights structurally determine both the weighted score and the hard constraint caps in CV-Judge. Our default $(0.20, 0.30, 0.30, 0.20)$ prioritizes instruction correctness over purely perceptual appearance.

Table 10: 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

Uniform weighting over-rewards visually pleasing but instruction-violating outputs. Edit-heavy weights raise the edit-failure cap aggressively (to $\omega_e \times 1000 = 400$), which is too lenient: outputs that miss the edit by a small margin escape being flagged as failures. The default weighting occupies the operating point that best balances correctness against perceptual quality, and the top-5 ranking is stable across all three configurations.

N Appendix: CV-Agent Module Ablation

CV-Agent is an intentionally minimal agentic baseline (Section 5) whose purpose is to validate the agentic paradigm rather than to demonstrate architectural novelty. We isolate the contribution of each stage in the three-stage pipeline.

Table 11: CV-Agent module ablation under the same Active Elo protocol.

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

Each stage contributes a monotonic, non-trivial gain: instruction rewriting (Stage 1) improves the precision of low-level edits; planning (Stage 2) enables multi-step decomposition for complex requests; and closed-loop refinement (Stage 3) recovers the largest portion of remaining failures by detecting and correcting under-edits. Even with these very simple modules, the combined pipeline already achieves the top Active Elo position, supporting our claim that the agentic paradigm itself, rather than any specific architectural choice, is the primary driver of the observed improvement, and pointing to planning, verification, and closed-loop refinement as a promising direction for future work.