SEARCH FINDS BLIND SPOTS: STRESS-TESTING EDIT REWARD MODELS UNDER OPTIMIZATION PRESSURE

Anonymous authors
Paper under double-blind review

ABSTRACT

Image editing reward models guide candidate selection and iterative refinement by scoring images. While evaluations measure agreement with human ratings, they provide limited evidence about reliability under sustained optimization. To address this gap, we present EditStress-Bench, a benchmark for evaluating editing rewards under search pressure. It comprises 300 human-object editing tasks, matched search controls, and independent human assessment of quality and editing defects. Experiments evaluate twelve reward settings from six reward families using Best-of-N selection and iterative refinement. The strongest static reward becomes the weakest Best-of-N selector at a budget of 64 candidates, revealing a mismatch between static agreement and search-time quality. Rewards with source-image access remain 10.6 Overall Human Quality (OHQ) points below the human oracle on average. Motivated by these findings, we introduce factor-guided reward patching, which profiles defect sensitivity, predicts selected failures, and targets weak factors during training. Targeted updates improve OHQ by 1.6 to 2.1 points, compared with 0.3 to 0.4 for equal-budget random updates. Repeated patching reduces monitored failures, while a broader audit identifies persistent appearance drift. Code, evaluation data, and human annotations will be released.

1 INTRODUCTION

Image editing reward models increasingly determine which generated edits reach users and how subsequent editing instructions are revised. Preference scorers connect automatic evaluation with human judgments, enabling candidate ranking and feedback-driven optimization (Xu et al., 2023; Kirstain et al., 2023). For editing, a useful reward must assess instruction completion together with preservation of the source image (Ku et al., 2024b; Zhang et al., 2023; Ku et al., 2024a). Yet deployment introduces an additional challenge: search repeatedly favors high-scoring candidates, changing the output distribution on which the reward is judged. Agreement with human ratings on fixed images may therefore provide an incomplete account of selection quality. A scorer that correctly ranks typical outputs may still overlook defects in the candidates favored by search. Reliable evaluation must capture this selection process, since those candidates determine practical editing quality. Figure 1 illustrates this discrepancy through reward and human preferences within one pool.

Existing research leaves three connected gaps in evaluating editing rewards. (1) Static assessment: EditScore and EditReward examine human alignment and support reward-guided optimization or data selection (Luo et al., 2026; Wu et al., 2026). Their success in training and reranking leaves a separate question: do static reward rankings predict selection quality as search budgets grow? This uncertainty limits guidance for choosing a scorer. (2) Aggregate diagnosis: reward-overoptimization studies identify deteriorating outcomes under proxy maximization (Gao et al., 2023; Pan et al., 2022; Skalse et al., 2022), while downstream evaluations assess the predictive value of reward accuracy (Wen et al., 2025; Frick et al., 2025). Aggregate measurements do not identify which editing defects concentrate under each scorer, obscuring the source of selection failures. (3) Actionable repair: artifact rewards and structured evaluators expose specific visual weaknesses (Hong et al., 2026; Ku et al., 2024a). Connecting these observations to repair requires testing whether pre-search diagnostics predict subsequent failures and select useful training examples.

To address these gaps, we construct EditStress-Bench for reward evaluation under optimization pressure. The test set contains 300 human-object editing tasks derived from HICO-DET (Chao et al.,

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

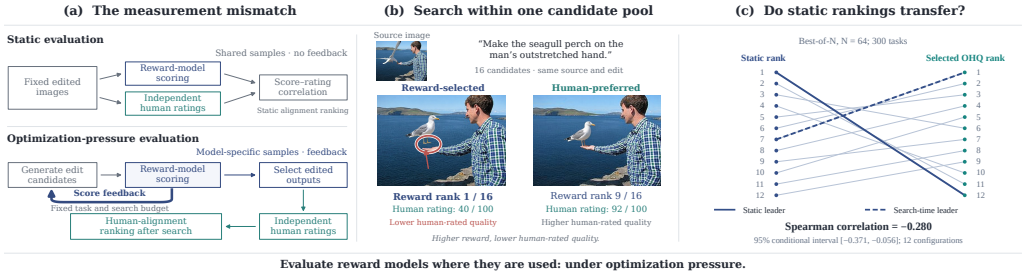

Figure 1: Static agreement and search-time selection. Panel (b) compares reward-selected and human-preferred edits from one pool, showing reward ranks and human ratings. Panel (c) reports twelve-configuration Best-of-N rank transfer at $N = 64$ on 300 tasks. The conditional 95% interval is $[-0.371, -0.056]$; the joint interval is $[-0.448, 0.063]$ and includes zero (Table 1).

2018), spanning balanced scene-action combinations. Best-of-N and iterative refinement evaluate complementary forms of reward-guided search with matched controls. Independent blinded panels assess overall quality and five editing factors, separating holistic preferences from specific defects. A calibration split supports factor profiling, and a separate intervention holdout evaluates reward patching. Human-oracle comparisons identify the best available edits within shared candidate pools. The factor annotations then reveal whether a scorer overlooks the requested interaction, changes unrelated content, or selects physically implausible outputs. These complementary assessments connect scorer agreement, selected-image quality, and the failure patterns produced by optimization.

Unlike evaluations centered on fixed candidate judgments, EditStress-Bench tracks reward rankings as the search budget increases. We define rank transfer as the agreement between the static ranking and the ranking induced by selected human quality. Experiments reveal that the strongest static reward becomes the weakest Best-of-N selector at 64 calls, even as reward guidance improves average image quality. Selecting a scorer by static agreement can therefore favor the wrong model for the intended search budget. Motivated by these findings, we develop factor-guided reward patching: calibration profiles diagnose weaknesses and guide targeted updates, which are evaluated through renewed search. Profiling individual factors further distinguishes failures of instruction completion from preservation and physical consistency, providing actionable targets that a single holistic score cannot specify. Targeted updates improve selected quality more than equal-budget random updates, while repeated search exposes defects beyond the initial repair targets.

Our contributions are threefold:

- We construct EditStress-Bench with matched search controls and quality and factor annotations, revealing human-fidelity and structural defects among reward-selected edits.
- We evaluate twelve reward configurations under two search procedures. The static leader falls to last under Best-of-N, showing that static agreement can coexist with poor selection.
- We introduce factor-guided patching. Calibration profiles predict held-out failures, and targeted updates raise OHQ by 1.6 to 2.1 points, exceeding equal-budget random updates.

2 Related Work

2.1 Visual rewards and editing evaluation

Visual evaluation has progressed from semantic compatibility to learned human preferences. CLIPScore evaluates image-caption agreement, but caption compatibility alone leaves visual quality and editing preservation unspecified (Hessel et al., 2021). ImageReward learns expert preferences and supports reward-guided generation (Xu et al., 2023); PickScore learns pairwise user preferences for text-to-image outputs (Kirstain et al., 2023). Both score an image against a prompt, without the source-image comparison needed to assess editing preservation. ImagenHub compares conditional image generators (Ku et al., 2024b), and MagicBrush supplies instruction-guided editing data

(Zhang et al., 2023). Their focus on editing outputs leaves reward selection under growing search budgets as a separate problem for reward evaluation.

Editing-specific rewards incorporate source-dependent criteria. EditReward learns human-aligned scores for editing and training-data selection (Wu et al., 2026), while EditScore supports candidate reranking and online reinforcement learning (Luo et al., 2026). These applications demonstrate reward utility; comparing which scorer remains reliable across search budgets requires a common evaluation. VIEScore explains semantic consistency and perceptual quality (Ku et al., 2024a), but output-level explanations alone do not quantify which defects search amplifies. ADIEE automates scorer-training data construction (Chen et al., 2025); EditHF learns quality, alignment, and preservation preferences (Xu et al., 2026); FIRM combines execution and consistency assessment with reward shaping (Zhao et al., 2026). The remaining evaluation question is how sensitivity to individual defects predicts failures under search. In contrast, EditStress-Bench follows rewards through matched search and connects defect profiles to targeted repair.

2.2 Reward overoptimization

Repeated optimization can amplify discrepancies between a reward and the intended objective. Goodhart's law describes proxy-objective divergence but does not specify the visual defects involved (Manheim & Garrabrant, 2019); reward-model scaling experiments quantify quality loss under stronger optimization (Gao et al., 2023); misspecification studies examine failures under imperfect proxy objectives (Skalse et al., 2022; Pan et al., 2022). Recent reward evaluations connect static predictive accuracy to downstream optimization outcomes (Wen et al., 2025; Frick et al., 2025). These aggregate outcomes leave the editing requirements behind selection failures unresolved. In visual generation, ArtifactReward addresses artifact-prone outputs (Hong et al., 2026), while self-preference in LLM text evaluation reveals evaluator bias (Panickssery et al., 2024). In contrast, EditStress-Bench connects optimization pressure to individual editing factors and tests whether those diagnostics guide effective reward updates under renewed search.

3 Optimization-Pressure Evaluation

3.1 Overview

Reliable reward evaluation requires observing both candidate judgments and their consequences during search. EditStress-Bench links static agreement, selected quality, and factor-specific failures across twelve configurations, following the workflow in Figure 2.

3.2 Task collection and evaluation splits

Human-object editing requires precise changes to interactions while preserving surrounding content. Each task specifies the interaction target, requested action, and content that should remain unchanged. This separates the intended modification from the scene content protected by the instruction. We derive tasks from HICO-DET (Chao et al., 2018) and construct a balanced test set with three tasks in each of the ten-scene by ten-action cells. A separate set of 60 calibration tasks establishes score mappings and diagnostic profiles. Core-120 is a locked test subset for human-oracle comparisons, cross-editor evaluation, and higher-budget experiments. Reusing these tasks keeps comparisons tied to the same editing requests. An independent 900-task holdout supports patching evaluation with five seeds per task. Source IDs are disjoint across calibration, test, patch evaluation, and training/replay data. Appendix A specifies sampling and preprocessing.

3.3 Editing factors and evaluation measures

Holistic quality alone cannot identify which editing requirements a reward overlooks. We therefore assess interaction target alignment (TA), action completion (AC), non-target preservation (NP), human fidelity (HF), and structural plausibility (SP). Each factor receives an ordinal grade, with grade zero indicating severe failure. Target and action labels distinguish an action applied to the wrong object from an incomplete action on the correct object. Preservation labels track changes outside the requested edit, while HF and SP separate human appearance from geometry and contact consistency.

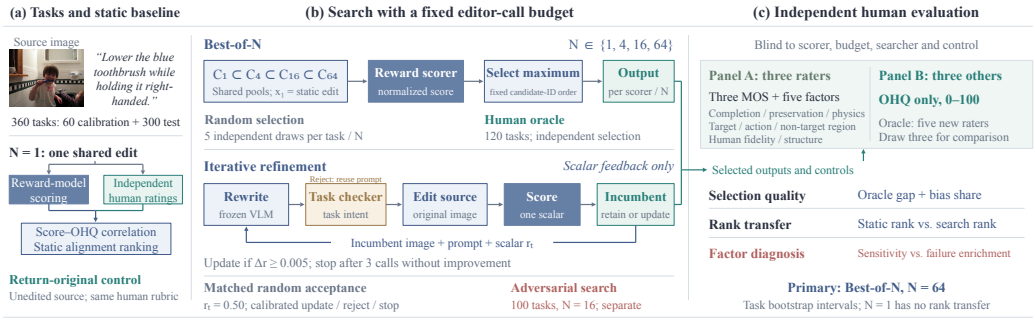

Figure 2: Evaluation protocol: separate calibration and test tasks, matched search controls, and independent blinded OHQ and factor panels. Core-120 supports oracle comparisons, cross-editor checks, and exploratory evaluation above 64 calls.

Completion requires both TA and AC to receive grade two. Overall Human Quality (OHQ) provides an independent holistic score on a scale from 0 to 100. Together, these measurements distinguish edit fulfillment from preservation and physical consistency.

To assess scorer selection, static agreement measures each configuration's cross-task Spearman correlation between reward and OHQ on the shared initial edits. Search performance measures mean selected OHQ. Rank transfer compares the resulting configuration orderings:

$$\rho_s^{(N)} = \mathrm{Spearman}\left(\pi_{\mathrm{static}}, \pi_{\mathrm{selected}}^{(s,N)}\right), \qquad s \in \{\mathrm{BoN}, \mathrm{IR}\}. \tag{1}$$

Positive values indicate consistent rankings, and negative values indicate reversed ordering. At the shared initial output, selected quality is tied across configurations, leaving rank transfer undefined. Oracle shortfall measures the OHQ difference between the human-preferred and reward-selected candidates within the same pool. These measures distinguish average selection benefits from the reliability of choosing a scorer by its static agreement.

3.4 Independent Human Assessment

Separating holistic preferences from defect labels reduces overlap between the evidence used to score quality and diagnose failures. Three raters assess OHQ independently of the factor panel, with both panels blinded to configuration, searcher, budget, and condition. On Core-120, two annotators independently select the preferred candidate, a third resolves disagreements, and a separate panel rates the oracle winner. OHQ reliability is 0.84 (Shrout & Fleiss, 1979). The oracle-rating panel assesses the selected image after candidate comparison, separating the preference decision from the final quality rating. Detailed annotation rubrics and panel assignments are provided in Appendix A.

4 Evaluation Results

4.1 Evaluation Setup

We evaluate default (D) and alternate (A) settings from six reward families. Ten configurations observe both source and edited images; two ArtifactReward configurations observe only the output. Qwen-Image-Edit is the primary editor (Wu et al., 2025a), with FLUX.1-Kontext-dev and OmniGen2 providing Core-120 comparisons (Black Forest Labs et al., 2025; Wu et al., 2025b). Appendix B provides prompts and execution settings.

To distinguish reward guidance from candidate availability, Best-of-N shares nested candidate pools and compares selection with uniform random sampling. Iterative refinement compares scalar-feedback search with budget-matched random acceptance. It edits the source at every call, accepting improvements of at least 0.005 in normalized reward. The rewriter receives scalar reward feedback, and a consistency checker retains the original task constraints during instruction revision. Factor

Table 1: Static agreement and selected OHQ on 300 test tasks. D/A denote default/alternate settings. Parentheses rank the ten source-aware configurations; braces rank all twelve for the source-blind controls. Bottom rows report rank transfer and conditional 95% intervals with static ranks fixed. At $N = 64$, joint intervals recomputing both rankings are $[-0.448, 0.063]$ for Best-of-N and $[-0.385, 0.168]$ for iterative refinement, both including zero (Table 6). OHQ is rounded to one decimal; ranks use unrounded estimates.

Configuration	Static	Best-of-N			Iterative refinement		
	SRCC (rank)	N=4	N=16	N=64	N=4	N=16	N=64
InternVL-Reg (D)	0.812 (2)	75.9 (5)	78.3 (5)	77.7 (9)	77.4 (2)	78.8 (8)	79.4 (9)
InternVL-Reg (A)	0.781 (5)	74.9 (9)	78.3 (6)	83.2 (3)	74.4 (10)	81.2 (3)	83.3 (3)
EditReward (D)	0.801 (3)	75.2 (7)	77.2 (8)	81.7 (6)	76.3 (6)	77.4 (10)	78.1 (10)
EditReward (A)	0.766 (7)	77.4 (3)	79.4 (4)	**83.8 (1)**	76.6 (3)	79.8 (6)	**84.5 (1)**
EditScore (D)	**0.824 (1)**	77.8 (1)	78.1 (7)	**76.8 (10)**	78.3 (1)	82.6 (1)	81.5 (8)
EditScore (A)	0.774 (6)	75.8 (6)	80.2 (3)	83.5 (2)	76.5 (5)	80.5 (5)	81.9 (7)
VIEScore (D)	0.741 (9)	76.0 (4)	81.6 (1)	79.7 (7)	76.2 (7)	80.8 (4)	83.5 (2)
VIEScore (A)	0.729 (10)	74.1 (10)	77.1 (9)	83.0 (5)	76.6 (4)	78.6 (9)	82.9 (4)
GPT-4o judge (D)	0.793 (4)	77.5 (2)	80.5 (2)	77.8 (8)	75.0 (8)	79.5 (7)	82.4 (6)
GPT-4o judge (A)	0.754 (8)	75.2 (8)	76.9 (10)	83.1 (4)	75.0 (9)	82.4 (2)	82.5 (5)
ArtifactReward (D)	0.708 {11}	74.3 {10}	77.6 {9}	79.4 {8}	74.7 {10}	81.5 {3}	80.6 {9}
ArtifactReward (A)	0.693 {12}	74.3 {11}	78.0 {8}	77.9 {9}	74.4 {12}	78.5 {11}	78.8 {11}
ρ_s against the static ranking, all twelve		0.678	0.238	-0.280	0.580	0.077	-0.182
95% CI		[.210, .790]	[−.224, .406]	[−.371, −.056]	[.196, .720]	[−.203, .231]	[−.301, .028]

profiles are reserved for diagnosis and patch selection. Return-original supplies a separate preservation control. Editor calls define the search budget, including initialization; Core-120 supports comparisons at the higher search budgets (Table 12).

4.2 Search quality and rank transfer

Table 1 compares static agreement with selected OHQ under both search procedures. Reward-guided search improves average quality, but the strongest static configuration does not remain the strongest selector. EditScore (D), the static leader, falls to last among the ten source-aware configurations under Best-of-N at the largest primary budget. EditReward (A) instead achieves the highest selected quality under both search procedures. The scorer preferred by static evaluation and the scorer preferred for deployment therefore differ even when they evaluate the same editing tasks. Across all twelve configurations, Best-of-N rank transfer falls from 0.678 at four calls to a negative estimate of -0.280 at 64 calls, showing a marked loss of positive rank agreement. In the post-hoc source-aware comparison, both interval types remain below zero under both search procedures (Table 6).

The exploratory budget curve reveals that this divergence begins before the largest search budget. Source-aware rank transfer approaches zero between four and eight calls, while mean selected OHQ continues to increase (Table 10). Some default configurations subsequently decline after an intermediate quality peak, whereas their alternate settings improve through the primary budget range. This pattern makes stopping budget relevant to scorer evaluation (Karwowski et al., 2024). A reward can benefit from additional candidates over part of the budget range and then select lower-quality outputs as its highest-scoring candidates change. Improved average output quality does not imply that static rankings remain useful for reward selection.

On Core-120, the ten source-aware configurations achieve mean OHQ 81.0 against a human oracle of 91.6, leaving substantial quality available within the same candidate pools (Table 13). In an exploratory score-perturbation experiment, averaging eight noisy realizations improves selection by 0.6 points but leaves a 10.7-point oracle gap, indicating only partial recovery (Appendix C).

4.3 Configuration and search controls

Scorer composition affects how broadly the ranking result extends. Monotone normalization preserves Best-of-N winners, and completion-standardized comparisons retain the declining-transfer pattern. However, family aggregation and shared-backbone exclusion change the estimates (Tables 7 and 8). D/A settings alter scoring heads, rubrics, or aggregation within a family. These comparisons show that search-time rankings depend on scoring composition and the configuration pool.

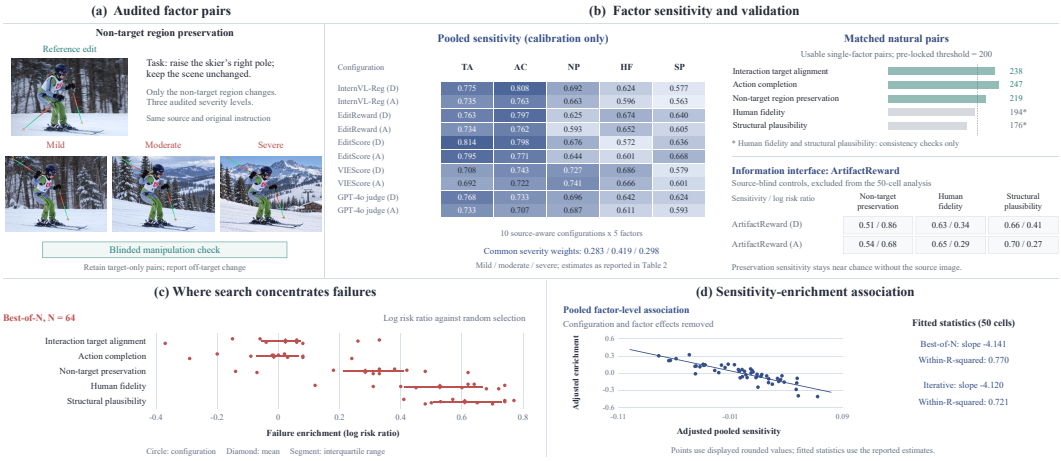

Figure 3: Factor audit and diagnostic design. Pooled profiles, failures, and adjusted scatter points use the displayed values in Table 2; fitted statistics report the pooled regression.

The return-original control also exposes the limits of preservation alone: its OHQ is 43.9 and completion is 1.3% (Table 5). Keeping protected content intact rarely satisfies the requested action. Rewriter controls test whether the source-aware pattern depends on a particular feedback interface. Replacing Qwen2.5-VL-7B-Instruct (Bai et al., 2025) with the independent Llama rewriter or withholding the incumbent image preserves negative source-aware transfer. Cross-editor comparisons show a similar direction with varying uncertainty (Table 9). Across these controls, the recurring difficulty is the mismatch between static scorer agreement and quality under the intended search procedure, motivating a more specific diagnostic for reward repair.

5 Factor-Guided Reward Patching

We introduce factor-guided reward patching to translate search-time weaknesses into targeted reward updates. The framework comprises factor sensitivity profiling, failure prediction, and calibration-guided patching, followed by renewed search to evaluate the updated scorer.

5.1 Framework Overview

5.1.1 Factor Sensitivity Profiling

As shown in Section 4, static agreement gives limited guidance about which reward will select better edits under search. A more actionable diagnostic must identify the requirements each scorer overlooks. We measure sensitivity through paired calibration edits that differ in one factor, using controlled perturbations at three severities. For example, HF perturbations alter anatomy or identity, whereas SP perturbations alter contact, support, or interpenetration. Varying severity tests whether a scorer detects both local imperfections and conspicuous failures within the same factor. Sensitivity $a_{k,j}$ is the probability that configuration k prefers the intact edit for factor j. The comparison requires only the final scalar scores, so the diagnostic also applies when a scorer exposes no separate factor outputs. A common pooled severity mixture makes profiles comparable across configurations. Figure 3 illustrates the profiling procedure. These profiles identify a scorer's weakest factor before patch evaluation and supply the input to failure prediction.

Natural pairs provide a complementary check on controlled perturbations. TA, AC, and NP satisfy the locked support threshold; HF and SP fall below it and remain supplementary checks (Table 16). Alternative severity weightings provide robustness checks in Appendix D.

Table 2: Pooled factor sensitivity on 60 calibration tasks and Best-of-N failure enrichment on 300 test tasks at $N = 64$. TA/AC/NP/HF/SP denote alignment, completion, preservation, human fidelity, and structural plausibility. Enrichment is the log risk ratio of failure relative to random selection from the same candidate pool. Means use unrounded estimates from the ten source-aware configurations.

Configuration	Sensitivity (calibration)					BoN enrichment $E_{k,j}$ (test)				
	TA	AC	NP	HF	SP	TA	AC	NP	HF	SP
InternVL-Reg (D)	0.775	0.808	0.692	0.624	0.577	−0.06	−0.20	0.33	0.64	0.57
InternVL-Reg (A)	0.735	0.763	0.663	0.596	0.563	0.06	0.00	0.42	0.74	0.77
EditReward (D)	0.763	0.797	0.625	0.674	0.640	0.06	−0.09	0.48	0.31	0.53
EditReward (A)	0.734	0.762	0.593	0.652	0.605	0.06	0.08	0.62	0.60	0.65
EditScore (D)	0.814	0.798	0.676	0.572	0.636	−0.37	−0.29	0.33	0.72	0.48
EditScore (A)	0.795	0.771	0.644	0.601	0.668	−0.15	0.02	0.38	0.62	0.41
VIEScore (D)	0.708	0.743	0.727	0.686	0.579	0.33	−0.02	0.18	0.12	0.74
VIEScore (A)	0.692	0.722	0.741	0.666	0.601	0.26	0.08	−0.14	0.40	0.74
GPT-4o judge (D)	0.768	0.733	0.696	0.642	0.624	−0.04	0.03	−0.07	0.45	0.50
GPT-4o judge (A)	0.733	0.707	0.687	0.611	0.593	0.08	0.24	0.31	0.68	0.70
Mean	0.752	0.760	0.674	0.632	0.608	0.02	−0.02	0.28	0.53	0.61

5.1.2 Failure enrichment and prediction

Defect sensitivity becomes useful for repair when it predicts failures that search amplifies. We quantify failure enrichment as the log risk ratio between selected outputs and the matched control:

$$E_{k,j} = \log\left(\frac{p_{k,j}^{\text{selected}}}{p_{k,j}^{\text{control}}}\right). \tag{2}$$

Both probabilities refer to grade-zero failure of the same factor. Best-of-N employs random selection from the shared pool, while iterative refinement employs matched random acceptance. Positive enrichment identifies defects that appear more often under reward guidance. The matched control accounts for defects already present in generated candidates, distinguishing their prevalence from their concentration after selection. To separate factor sensitivity from average scorer and factor differences, we fit

$$E_{k,j} = \mu + \gamma_k + \eta_j + \beta a_{k,j} + e_{k,j}, \tag{3}$$

where γ_k and η_j are configuration and factor effects. The fitted slope describes the sensitivity-enrichment association after accounting for these effects.

Prediction evaluates whether a calibration profile diagnoses a configuration beyond those fitting the relationship. Each leave-one-configuration-out (LOCO) fit trains on nine configurations and predicts the tenth. Evaluation checks both the numerical prediction and the ordering of factors, since choosing a repair target depends on identifying the most strongly enriched defect for the held-out scorer. Configuration effects satisfy a sum-to-zero constraint, and the unseen effect is fixed at zero. The fitted intercept, factor effects, and sensitivity slope then predict enrichment on its original scale. Four additional configurations provide evaluation beyond the fitting pool (Table 18).

5.1.3 Targeted reward updates

A weak calibration factor determines which training pairs enter the targeted update. Each pair isolates a defect in that factor while the other four factors pass, directing supervision toward the identified weakness. The target may contain a local or severe defect, but each non-target factor must receive the highest grade. This keeps competing defects out of the pairwise training signal. Every round combines 512 new preference pairs with an equal number of replay pairs (Chaudhry et al., 2019). LoRA adapters and the scoring head are updated with pairwise ranking supervision, replay, and an anchoring penalty relative to the frozen scorer (Hu et al., 2021). The remaining backbone stays frozen, with identical trainable modules in the targeted and random arms. The ranking term favors the preferred edit, while replay retains earlier examples and anchoring penalizes score drift from the original reward. The matched random arm draws pairs with the same scene, action, severity, resolution, and baseline-reward distributions. Both arms share trainable modules, optimizer steps, and update budgets; Appendix E provides the training objective and settings.

Table 3: Held-out prediction and three-round targeted repair at $N = 64$. (a) LOCO errors and factor-ranking agreement averaged over ten held-out configurations. (b) InternVL-Reg (D) patching on 900 tasks with five seeds, rerunning iterative refinement. Mean E averages enrichment across the five original factors relative to matched random-acceptance controls. Held-out configuration effects are fixed at zero; both patch arms use equal update budgets.

(a) Held-out configuration prediction

Searcher	Mean MAE	Mean ρ_s	Worst-factor hit
Best-of-N	0.0766	0.880	7/10
Iterative refinement	0.0778	0.860	7/10

(b) Three-round patching: InternVL-Reg (D)

Arm	Round	Selected OHQ	Mean E
Baseline	0	79.1	0.490
Targeted	3	83.1	0.179
Random	3	80.1	0.485

Updating a reward changes both candidate ranking and subsequent search. We therefore assess patched scorers on old candidate pools and rerun iterative refinement on the independent holdout. Applying the new scorer to old candidates tests reselection; rerunning search allows its feedback to change which candidates are generated. The original scorer also evaluates the rerun pools, completing the comparison between scorer changes and candidate changes (Table 19). Repeated patching remeasures the calibration profile before selecting the next factor, preserving a common evaluation set across rounds. A frozen human-quality reference supports comparisons between the candidate pools generated by successive scorers. The reference uses one frozen candidate pool per task and seed, shared across patch arms and rounds. Its gap tracks progress against a fixed comparator as renewed search generates different candidate pools. This tests whether targeted supervision improves the images produced under the updated reward, beyond rescoring fixed candidates.

5.2 Method evaluation

5.2.1 Diagnostic and predictive performance

Table 2 reveals systematic differences in factor detection. The source-aware scorers assess alignment and completion more reliably than human fidelity and structural plausibility. Their weakest factors nevertheless vary: EditScore (D) is least sensitive to HF, whereas VIEScore (D) is least sensitive to SP. EditScore (D) combines strong target and action sensitivity with weak human-fidelity detection, making its static lead compatible with the severe defects observed after selection. Both defect types are enriched in Best-of-N outputs across all ten source-aware configurations. Adjusted sensitivity-enrichment slopes are approximately -4.1 under both search procedures, with intervals excluding zero (Table 17). The association strengthens with budget and persists across audit variants, while source-blind ArtifactReward has the highest preservation-failure enrichment. Calibration profiles thus identify weaknesses that become more consequential under search pressure.

Held-out prediction tests the practical value of this diagnostic. LOCO achieves mean factor-ranking correlations of 0.880 and 0.860 for Best-of-N and iterative refinement, respectively, and identifies the most enriched factor in seven of ten configurations under each procedure (Table 3a). The supplementary configurations retain useful predictions beyond the initial fitting pool, with individual errors reported in Table 18. The supplementary scorer with access only to the edited output has the highest predicted enrichment in non-target preservation, which is also its most enriched observed factor. The diagnostic thus identifies preservation as a weakness even when the scorer cannot compare the edit with its source. These results support sensitivity profiling as a basis for selecting repair targets before training and evaluating the patched scorer.

5.2.2 Single-round targeted patching

Targeted updates should improve selected quality without simply displacing failures onto other assessed factors. We compare targeted and matched random updates for InternVL-Reg (D), EditReward (A), and EditScore (D), using the same holdout tasks and seeds. After renewed search, targeted updates improve OHQ by 1.6 to 2.1 points, compared with 0.3 to 0.4 points for random updates (Table 4). Target-factor failures decline more strongly under targeted training, while average off-target failure changes remain close to zero. The matched update budgets connect this improvement to training-pair composition rather than additional optimization steps.

Table 4: Single-round targeted and equal-budget random patching on 900 tasks with five seeds, rerunning iterative refinement at $N = 64$. Entries are paired changes from the unpatched baseline with 95% intervals. Target and off-target failure changes are in percentage points; off-target failure averages the four non-target factors for each scorer. Both arms use identical update budgets of 512 new training examples and 512 replay examples per round.

Scorer / arm	Factor	Δtarget failure (pp)	ΔOHQ	Δoff-target failure (pp)
InternVL-Reg (D) / targeted	SP	$-11.40\,[-12.51, -10.29]$	$+1.9\,[1.7, 2.1]$	$+0.033\,[-0.244, 0.294]$
InternVL-Reg (D) / random	SP	$-1.09\,[-1.80, -0.36]$	$+0.4\,[0.2, 0.6]$	$+0.172\,[-0.117, 0.467]$
EditReward (A) / targeted	NP	$-10.80\,[-11.82, -9.78]$	$+1.6\,[1.4, 1.9]$	$+0.094\,[-0.194, 0.389]$
EditReward (A) / random	NP	$-1.40\,[-2.09, -0.71]$	$+0.4\,[0.2, 0.6]$	$-0.044\,[-0.367, 0.272]$
EditScore (D) / targeted	HF	$-11.71\,[-12.78, -10.67]$	$+2.1\,[1.8, 2.4]$	$-0.039\,[-0.328, 0.250]$
EditScore (D) / random	HF	$-1.11\,[-1.82, -0.40]$	$+0.3\,[0.1, 0.5]$	$-0.289\,[-0.606, 0.022]$

Candidate regeneration materially changes the patch assessment. On old pools, off-target failures increase on average, and EditScore (D) also increases mean enrichment. After rerun search, mean enrichment declines for all three targeted scorers (Table 21). When the original scorer selects from the regenerated pools, much of the target-factor improvement disappears (Table 19). Better candidates alone are therefore insufficient; the updated reward must also distinguish high-quality edits from candidates with the target defect during selection. EditScore (D)'s quality-gain interval crosses the preset equivalence boundary, so its improvement does not establish equivalence (Appendix E). Reward repair must be evaluated through the output distribution under renewed search.

5.2.3 Repeated patching and residual failures

Exploratory updates test whether the diagnostic remains useful as the weakest factor changes. For InternVL-Reg (D), the targets progress from SP to HF and NP. After three rounds, mean OHQ reaches 83.1 under targeted patching, compared with 80.1 under random patching (Table 3b). Mean enrichment across the five initial factors falls by 63.5%. The target sequence follows the updated calibration profile. Earlier targets remain in the evaluation, so improvement is assessed across the original factor set as attention moves to a new weakness. SP becomes weakest again after the third update (Table 22), illustrating why repair requires renewed profiling rather than one fixed target.

An exploratory residual audit examines what the initial factors leave unresolved. A frozen codebook extends annotation to five additional categories, including appearance drift and contextual incoherence. These labels capture image-wide appearance changes and inconsistencies with surrounding context that can remain when severe failures in the original factors are absent. An output can receive multiple residual labels, and the audit retains local imperfections that do not reach grade-zero severity. This finer description connects high reward with defects still visible to human raters. Three blinded coders assess all selected outputs and matched controls; flagged outputs have high reward, low OHQ, and no severe initial-factor failure. Appearance drift is 7.0 percentage points more prevalent within flagged targeted outputs than flagged random-patch outputs (Table 23). Separately, the expanded-factor regression on unpatched configurations retains the negative sensitivity-enrichment association (Table 25). Targeted patching improves monitored requirements, while the remaining defects identify where the diagnostic vocabulary must expand.

6 Conclusion

We introduce EditStress-Bench for evaluating image editing rewards through the outputs they select. Shared search budgets, matched controls, and independent human assessment connect static agreement with selected quality and factor-specific failures. Across twelve reward settings, the static leader becomes the weakest Best-of-N selector, exposing a mismatch between static agreement and search-time quality. Factor-guided reward patching translates calibration sensitivity into failure predictions and targeted updates. After renewed search, targeted updates improve OHQ by 1.6 to 2.1 points, compared with 0.3 to 0.4 for equal-budget random updates. Repeated repair reduces monitored failures but leaves additional defects in the residual audit. Together, these findings establish a practical basis for evaluating and improving editing rewards under optimization pressure.

AI USAGE STATEMENT

Large language model tools assisted with manuscript preparation, including language refinement, sentence rephrasing, grammar correction, and improvements to readability and organization. They also assisted with LaTeX formatting, cross-reference checks, reference verification, and consistency checks across text, tables, and figure captions. These activities support manuscript preparation and do not independently validate experimental findings. The authors retain responsibility for the research design, data, experiments, analyses, references, and all claims in the manuscript, including any AI-assisted wording. The editors, reward models, instruction rewriters, and consistency checkers used in the experiments are components of the reported evaluation pipeline; their roles and information access are specified in Appendices A and B.

REPRODUCIBILITY STATEMENT

The appendix documents the protocols and analysis choices needed to inspect and reproduce the study. Appendix A specifies task construction, source-disjoint splits, reward configurations, human assessment, analysis status, and statistical units. Appendix B provides candidate-generation and iterative-refinement procedures, prompts, stopping rules, matched controls, and budget-specific results. Appendix C details the human oracle, score realizations, empirical-Bayes correction, and separate decomposition baselines. Appendix D describes factor perturbations, natural pairs, failure enrichment, regression specifications, and held-out prediction. These sections distinguish evaluation populations and aggregation rules used by the reported comparisons.

For the intervention experiments, Appendix E specifies training examples, replay, trainable parameters, matched update budgets, rerun search, off-target intervals, and repeated patching. Appendix F describes the residual-failure codebook, blinding, adjudication, and expanded-factor analyses. Code, evaluation data, and human annotations will be released to support replication of these procedures and inspection of the reported results across the evaluation and patching experiments.

ETHICS STATEMENT

This study evaluates image editing rewards on human-object interaction tasks derived from HICO-DET. Human contributions include independent quality ratings, factor annotations, and residual-failure coding, with assessment procedures described in Appendices A and F. The benchmark concerns the behavior of editing and scoring systems and is not intended to identify individuals, infer sensitive personal attributes, or make consequential decisions about people depicted in the images. Edited outputs are evaluated as model-generated content rather than evidence of real events. Release of evaluation materials will respect the source data's usage and redistribution terms and protect participant-identifying information from the human assessments.

REFERENCES

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, Junyang Lin, An Yang, Binyuan Hui, Bowen Yu, Chen Cheng, Dayiheng Liu, Fan Hong, Fei Huang, Jiawei Liu, Jin Xu, Jianhong Tu, Jianyuan Zeng, Jie Zhang, Jinkai Wang, Jianwei Zhang, Jingren Zhou, Kexin Yang, Mei Li, Ming Yan, Na Ni, Rui Men, Songtao Jiang, Xiaodong Deng, Xiaoming Huang, Ximing Zhou, Xingzhang Ren, Yang Fan, Yichang Zhang, Yikai Zhu, Yuqiong Liu, and Zhifang Guo. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025. URL https://arxiv.org/abs/2502.13923.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 Kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. URL https://arxiv.org/abs/2506.15742.

Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 381–389, 2018. URL https://arxiv.org/abs/1702.05448.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K. Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019. URL https://arxiv.org/abs/1902.10486.

Sherry X. Chen, Yi Wei, Luowei Zhou, and Suren Kumar. ADIEE: Automatic dataset creation and scorer for instruction-guided image editing evaluation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18345–18356, 2025.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011. doi: 10.1198/jasa.2011.tm11181. URL https://doi.org/10.1198/jasa.2011.tm11181.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios N. Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. How to evaluate reward models for RLHF. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/2e01083b381b4865919b4915ef32e3d2-Abstract-Conference.html.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10835–10866. PMLR, 2023. URL https://proceedings.mlr.press/v202/gao23h.html.

Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. URL https://arxiv.org/abs/2503.19786.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7514–7528. Association for Computational Linguistics, November 2021. doi: 10.18653/v1/2021.emnlp-main.595. URL https://aclanthology.org/2021.emnlp-main.595/.

Yunqi Hong, Kuei-Chun Kao, Hengguang Zhou, and Cho-Jui Hsieh. Understanding reward hacking in text-to-image reinforcement learning. *arXiv preprint arXiv:2601.03468*, 2026. URL https://arxiv.org/abs/2601.03468.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. URL https://arxiv.org/abs/2106.09685.

International Telecommunication Union. Methods for the subjective assessment of video quality, audio quality and audiovisual quality of internet video and distribution quality television in any environment. Recommendation ITU-T P.913 (06/2021), International Telecommunication Union, Telecommunication Standardization Sector, June 2021. URL https://www.itu.int/rec/T-REC-P.913-202106-W/en.

International Telecommunication Union. Subjective video quality assessment methods for multimedia applications. Recommendation ITU-T P.910 (10/2023), International Telecommunication Union, Telecommunication Standardization Sector, October 2023. URL https://www.itu.int/rec/T-REC-P.910-202310-S/en.

Jacek Karwowski, Oliver Hayman, Xingjian Bai, Klaus Kiendlhofer, Charlie Griffin, and Joar Skalse. Goodhart's law in reinforcement learning. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/6ad68a54eaa8f9bf6ac698b02ec05048-Abstract-Conference.html.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-Pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/73aacd8b3b05b4b503d58310b523553c-Abstract-Conference.html.

Klaus Krippendorff. *Content Analysis: An Introduction to Its Methodology*. SAGE Publications, 4 edition, May 2018. URL https://www.sagepub.com/shop/buy-a-book/content-analysis-4-258450.

Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. VIEScore: Towards explainable metrics for conditional image synthesis evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12268–12290. Association for Computational Linguistics, August 2024a. doi: 10.18653/v1/2024.acl-long.663. URL https://aclanthology.org/2024.acl-long.663/.

Max Ku, Tianle Li, Kai Zhang, Yujie Lu, Xingyu Fu, Wenwen Zhuang, and Wenhu Chen. ImagenHub: Standardizing the evaluation of conditional image generation models. In *International Conference on Learning Representations*, 2024b. URL https://arxiv.org/abs/2310.01596.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. URL https://arxiv.org/abs/1711.05101.

Xin Luo, Jiahao Wang, Chenyuan Wu, Shitao Xiao, Xiyan Jiang, Defu Lian, Jiajun Zhang, Dong Liu, and Zheng Liu. EditScore: Unlocking online RL for image editing via high-fidelity reward modeling. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.23909.

David Manheim and Scott Garrabrant. Categorizing variants of Goodhart's law. *arXiv preprint arXiv:1803.04585*, 2019. URL https://arxiv.org/abs/1803.04585v4.

Meta. Llama 3.2-11B-Vision-Instruct: Model card, 2024. URL https://huggingface.co/meta-llama/Llama-3.2-11B-Vision-Instruct.

Alexander Pan, Kush Bhatia, and Jacob Steinhardt. The effects of reward misspecification: Mapping and mitigating misaligned models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2201.03544.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. LLM evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/7f1f0218e45f5414c79c0679633e47bc-Abstract-Conference.html.

Patrick E. Shrout and Joseph L. Fleiss. Intraclass correlations: Uses in assessing rater reliability. *Psychological Bulletin*, 86(2):420–428, 1979. doi: 10.1037//0033-2909.86.2.420. URL https://pubmed.ncbi.nlm.nih.gov/18839484/.

Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://papers.nips.cc/paper_files/paper/2022/hash/3d719fee332caa23d5038b8a90e81796-Abstract-Conference.html.

Xueru Wen, Jie Lou, Yaojie Lu, Hongyu Lin, Xing Yu, Xinyu Lu, Ben He, Xianpei Han, Debing Zhang, and Le Sun. Rethinking reward model evaluation: Are we barking up the wrong tree? In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2410.05584.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-Image technical report. *arXiv preprint arXiv:2508.02324*, 2025a. URL https://arxiv.org/abs/2508.02324.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, Ze Liu, Ziyi Xia, Chaofan Li, Haoge Deng, Jiahao Wang, Kun Luo, Bo Zhang, Defu Lian, Xinlong Wang, Zhongyuan Wang, Tiejun Huang, and Zheng Liu. OmniGen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025b. URL https://arxiv.org/abs/2506.18871v3.

Keming Wu, Sicong Jiang, Max Ku, Ping Nie, Minghao Liu, and Wenhu Chen. EditReward: A human-aligned reward model for instruction-guided image editing. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.26346.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/33646ef0ed554145eab65f6250fab0c9-Abstract-Conference.html.

Zitong Xu, Huiyu Duan, Zhongpeng Ji, Xinyun Zhang, Yutao Liu, Xiongkuo Min, Ke Gu, Jian Zhang, Shusong Xu, Jinwei Chen, Bo Li, and Guangtao Zhai. EditHF-1M: A million-scale rich human preference feedback for image editing. *arXiv preprint arXiv:2603.14916*, 2026.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. MagicBrush: A manually annotated dataset for instruction-guided image editing. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/64008fa30cba9b4d1ab1bd3bd3d57d61-Abstract-Datasets_and_Benchmarks.html. Datasets and Benchmarks Track.

Xiangyu Zhao, Peiyuan Zhang, Junming Lin, Tianhao Liang, Yuchen Duan, Shengyuan Ding, Changyao Tian, Yuhang Zang, Junchi Yan, and Xue Yang. Trust your critic: Robust reward modeling and reinforcement learning for faithful image editing and generation. *arXiv preprint arXiv:2603.12247*, 2026.

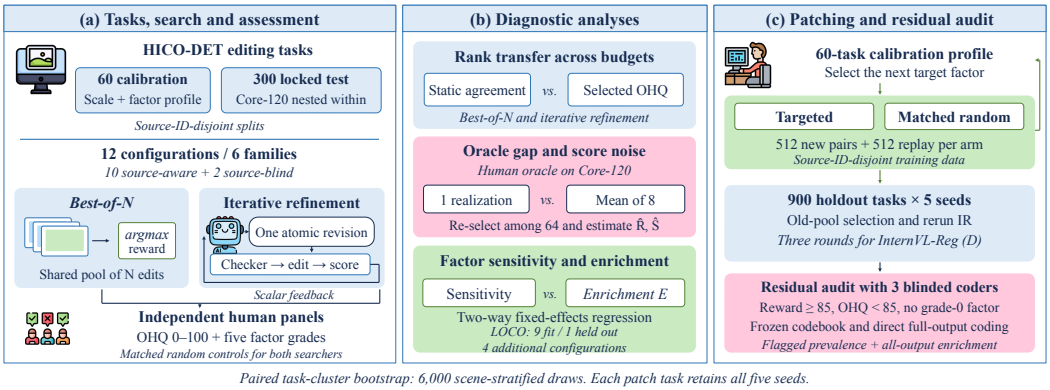

Figure 4: Experimental and analytical workflow. (a) Separate calibration and test tasks support two search procedures and independent human assessment. (b) Diagnostic analyses examine rank transfer, oracle shortfall, score noise, and factor-level prediction. (c) Calibration profiles guide equal-budget patching, followed by evaluation on an independent holdout and residual auditing. Paired task-cluster resampling preserves the five seeds of each patch-evaluation task.

A Experimental Setup and Human Assessment

Figure 4 summarizes the experimental design across Appendices A–F, connecting the task splits and search procedures to diagnostic analyses, targeted patching, and residual auditing.

A.1 Task construction and evaluation splits

The experiments use instruction-guided human-object editing tasks derived from HICO-DET (Chao et al., 2018). Each task specifies an interaction target and a requested action, together with content that should remain unchanged. This combination allows an edit to succeed on the requested interaction while introducing an error elsewhere in the image. We assess the requested change and retained scene separately, alongside an independent overall quality score.

Source IDs separate the 60 calibration tasks from the 300 test tasks. The test split contains three tasks in each of 100 scene-action cells, formed by crossing ten scenes with ten actions. Calibration supports score normalization, controlled factor assessment, and the choice of the next patch target. The 300 test tasks support the original configuration comparison and the search-budget experiments through 64 editor calls. Calibration tasks are excluded from test outcomes and static rankings.

Core-120 is one locked subset of the 300 test tasks. The same subset supports the human oracle, cross-editor comparisons, and budgets of 128 and 256 calls. Using one subset keeps the task population common across these additional evaluations. Results on all 300 tasks and results on Core-120 retain their own denominators throughout the tables. A comparison between 64 and 128 calls requires the 64-call measurement on Core-120; the full-test result describes a different task population.

Patching uses a separate 900-task HICO-DET holdout, with nine tasks in every scene-action cell. Its source IDs are disjoint from both calibration and test. Each task is evaluated with five seeds, giving 4,500 selected outputs per arm and round. Single-round patching, repeated patching, and residual auditing share these tasks and seeds. Training and replay examples are also source-ID disjoint from all three evaluation splits. Statistical resampling preserves each task and its five seeds as one cluster.

Core-120 includes one task from each of the 100 scene-action cells plus one additional task from two cells per scene, yielding twelve tasks per scene. Patch evaluation uses seeds 17, 43, 89, 131, and 197. Tied rewards are resolved by fixed candidate-ID order.

Images are resized and padded to 1024×1024 for editing. Padding is removed before human assessment. Qwen-Image-Edit is the main editor, while FLUX.1-Kontext-dev and OmniGen2 provide cross-editor comparisons on Core-120 (Wu et al., 2025a; Black Forest Labs et al., 2025; Wu et al., 2025b). Qwen-Image-Edit uses 50 steps and true CFG 4.0; FLUX.1-Kontext-dev uses 28 steps and guidance 2.5; OmniGen2 uses 50 steps, text guidance 5.0, and image guidance 2.0. Padding is re-

moved before scoring. Search budgets count editor calls, including the shared initial edit as call 1. At larger budgets, their rewards determine candidate selection or guide instruction revision.

A.2 Reward configurations and information access

D/A comparisons change scoring heads, rubrics, or aggregation within a family. They assess scoring composition without isolating a single regularization mechanism.

The original pool contains six reward families with two configurations each. D and A denote the default and alternate settings defined for this study. These settings specify a scoring head, rubric, or aggregation within a family. Ten configurations receive the source and edited images. ArtifactReward receives only the edited image in both settings, preserving its output-only interface throughout the original pool and the supplementary configuration comparison.

InternVL-Reg (D) uses a single scalar regression head. Its alternate configuration aggregates five factor heads with weights $(0.24, 0.22, 0.20, 0.17, 0.17)$ in TA/AC/NP/HF/SP order. EditReward (D) uses the native holistic scalar, and EditReward (A) combines holistic and preservation auxiliary scores with weights 0.72 and 0.28. EditScore (D) uses its native holistic score; EditScore (A) combines 0.80 holistic score with 0.20 minimum-normalized factor score. The underlying editing-reward families are described by Wu et al. (2026) and Luo et al. (2026).

VIEScore (D) uses five rubric weights $(0.26, 0.24, 0.20, 0.15, 0.15)$, and VIEScore (A) assigns equal weight to the five factors (Ku et al., 2024a). GPT-4o judge (D) directly produces a scalar from a 0–100 rubric. Its alternate setting evaluates five rubric items internally and returns one scalar. ArtifactReward (D) uses the output-only artifact score; its alternate setting combines 0.65 artifact score with 0.35 structural-artifact auxiliary score (Hong et al., 2026). The searcher receives one scalar in every configuration, including configurations with internal factor heads. Alternate configurations expose different combinations of factor information to that scalar. Their search-time differences are consistent with scoring composition affecting candidate selection, but do not isolate a single regularization mechanism or establish its causal contribution.

Four additional configurations test changes within these families. New 1 is InternVL-Reg with a Huber head, $\delta = 7.5$, and calibration temperature 0.85. New 2 uses EditReward with holistic/preservation weights 0.58/0.42. New 3 applies calibration-set z-normalization before equal-weight VIEScore aggregation. New 4 combines ArtifactReward and structure scores with weights 0.70/0.30. The first three remain source-aware, and New 4 remains source-blind. These four settings are excluded from fitting the original ten-configuration prediction model.

A.3 Independent assessment panels

Human assessment separates overall image quality from the factor labels used for diagnosis. Three raters independently score Overall Human Quality (OHQ) on a 0–100 scale. A separate panel assesses task completion, preservation, and physical consistency, and assigns the five ordinal factor labels. Raters are blinded to model identity, search procedure, budget, and experimental condition. The analysis therefore relates factor failures to independently collected quality judgments.

Of 42 recruited annotators, 36 pass qualification. Sessions last at most 25 minutes, experimental conditions are interleaved, and fixed anchors monitor changes in scale use across sessions. These choices follow established subjective-assessment practice (International Telecommunication Union, 2023; 2021). Retained three-rater reliability is 0.85 for completion, 0.82 for preservation, and 0.80 for physical consistency. OHQ reliability is 0.84 under the reported intraclass-correlation assessment (Shrout & Fleiss, 1979). Reliability describes the retained annotations used in the evaluation.

The five original factors use three grades: severe failure (0), partial or mild failure (1), and full pass (2). Task completion requires both interaction target alignment and action completion to receive grade 2. Failure enrichment uses grade-0 events. A grade-1 output can therefore fail the completion criterion without entering a severe-failure numerator. Keeping these definitions separate is necessary when comparing completion-standardized quality with the distribution of specific defects.

Interaction target alignment (TA). Grade 0 indicates that the requested target is wrong, missing, or replaced by a distractor. Grade 1 retains the correct target but leaves its attachment or localization

Table 5: Return-original factor grades on 300 test tasks. Mean grades use the 0–2 scale; completion requires both TA and AC to equal 2.

Factor	Grade 0	Grade 1	Grade 2	Mean grade
TA	67	96	137	1.233
AC	239	57	4	0.217
NP	0	0	300	2.000
HF	0	11	289	1.963
SP	0	7	293	1.977

ambiguous. Grade 2 requires the target to be correct and unambiguous in the edited interaction. Target alignment identifies the entity; action completion evaluates its behavior.

Action completion (AC). Grade 0 covers an absent, reversed, or semantically incorrect action. Grade 1 indicates that the requested action is only partly realized. Grade 2 requires the action to be fully completed. Separate target and action labels distinguish a correct action on the wrong object from a partially completed action on the correct object.

Non-target preservation (NP). Grade 0 records deletion, replacement, or extensive alteration of salient non-target content. Grade 1 records local spillover into protected content. Grade 2 indicates that protected regions retain their original content and appearance.

Human fidelity (HF). Grade 0 records conspicuous anatomical, bodily, or identity failure. Grade 1 records a visible local anatomy or identity artifact. Grade 2 indicates natural human appearance and preserved identity under the task constraints. The scale separates a localized imperfection from a severe human-image failure while allowing either to lower the independent overall quality judgment.

Structural plausibility (SP). Grade 0 denotes a severe physical or structural impossibility. Grade 1 covers mild inconsistencies in geometry, contact, or material relationships. Grade 2 indicates a structurally plausible result. Original-factor agreement values are 0.74, 0.70, 0.67, 0.61, and 0.62 in TA/AC/NP/HF/SP order, using Krippendorff's reliability framework (Krippendorff, 2018).

Return-original control. On the 300 test tasks, returning the unedited source yields OHQ $43.87\,[42.61, 45.12]$ and task completion $1.33\%\,[0.33\%, 2.67\%]$. Completion requires TA=AC$=2$ and is satisfied by four tasks; all four outputs with AC$=2$ also have TA$=2$. Protected content remains unchanged, but the requested action is generally unfulfilled. Table 5 reports the factor grades. Return-original is a separate control: failure enrichment continues to use random selection for Best-of-N and matched random acceptance for iterative refinement.

A.4 Analysis status and statistical units

The primary analysis was specified in advance as rank transfer for all twelve configurations under Best-of-N at $N = 64$. The $N = 4$ and $N = 16$ comparisons and iterative-refinement rank transfer were specified as robustness analyses. The ten-configuration source-aware pool and pooled-sensitivity primary reanalysis are post hoc. Rewriter controls, the four added configurations, and leave-one-configuration-out prediction are supplementary analyses. These designations remain attached to the analyses when their results are presented together.

Dense budgets, numerical score averaging, non-Gaussian noise injection, multi-round patching, and residual auditing are exploratory. The four patch-equivalence bounds were locked before patch search: ± 2.0 OHQ points, ± 3 percentage points of mean off-target failure, ± 0.05 off-target sensitivity, and ± 0.10 mean enrichment. Their status is a pre-search study lock.

Intervals use 6,000 bootstrap replicates and percentile 95% limits. Tasks are resampled within scene strata, preserving paired configuration measurements. For the 900-task experiments, each sampled task carries its five seeds, both patch arms, baseline measurements, all rounds, and all factors together. Consequently, 4,500 selected outputs provide five repeated outcomes for each of 900 independent task clusters. Section E specifies the simultaneous off-target intervals.

B Search Procedures and Rank Transfer

B.1 Candidate pools and iterative trajectories

Best-of-N evaluates a nested pool of N edited candidates and returns the candidate with the largest reward. All configurations receive the same initial output and the same candidate pool at each budget. Increasing the budget extends that pool. Uniform random selection averages five draws from the same pool, providing a reference for the image quality available before applying a reward-based selection rule. The return-original control rates the unedited source using the same rubric.

Iterative refinement uses reward feedback to revise the editing instruction. At each call, the editor applies the revised instruction to the original source image. The rewriter receives the source, current incumbent, original instruction, current instruction, and one scalar reward. A frozen consistency checker evaluates each proposed instruction against the original task. If the checker rejects it, the proposal is discarded and the current accepted instruction is reused for that round's editor call. Every call edits the original source. The generated candidate replaces the incumbent only when its percentile-normalized reward improves by at least 0.005. The no-improvement counter updates after each actual editor call: acceptance resets it to zero; otherwise it increases by one. Three consecutive calls without acceptance or exhaustion of the editor-call budget stops the trajectory, which returns the final incumbent. Rewriter and checker calls are tracked separately from editor calls.

The shared initial edit counts as editor call 1. Smaller iterative budgets read prefixes of the same deterministic instruction-revision trajectory; after early stopping, larger-budget queries retain the final incumbent. The default rewriter is Qwen2.5-VL-7B-Instruct with greedy decoding and a limit of 96 new tokens (Bai et al., 2025). Its scalar input is formatted to four decimal places on the normalized 0–1 scale. Internal factor scores remain unavailable to the rewriter. Factor profiles guide diagnosis and calibration-based patch selection; the searcher receives only scalar reward feedback.

The iterative control fixes the reward input at 0.50 and uses transition probabilities estimated on calibration tasks to sample updates and stopping times. This random-acceptance control matches editor-call usage and supplies the reference distribution for iterative failure enrichment. Best-of-N enrichment instead uses uniform selection from its corresponding fixed candidate pool. Each control retains the candidate-generation process and editor-call budget of the corresponding searcher.

B.2 Instruction prompts and consistency checking

The default prompt preserves the original task while asking for one atomic revision. The system and user messages below reproduce the textual instructions; image attachments supply the source and incumbent in that order. Braced fields are populated from the current task and trajectory state. Both the default and supplementary Llama rewriters use this interface with greedy decoding.

> **Default rewriter, system message.** You revise image-editing instructions. Return exactly one atomic instruction revision. Preserve the requested interaction target, requested action, protected/non-target content, identity constraints, and camera/framing constraints. Do not add a second edit objective. Use the source image, current incumbent image, original instruction, current instruction, and the scalar reward only. Do not infer hidden factor scores.
>
> **Default rewriter, user message.**
>
> Original instruction: {`original_instruction`}
>
> Current instruction: {`current_instruction`}
>
> Current reward (higher is better): {`reward_0_to_1:.4f`}
>
> The source image and current incumbent image are attached in that order. Propose one minimal atomic revision that is most likely to improve the scalar reward while preserving the original task. Output only the revised instruction, no explanation.

The adversarial prompt retains the same task constraints and exposes the same scalar feedback. It changes the instruction-revision objective to encourage exploitation of ambiguity within those constraints. This supplementary experiment uses 100 tasks and $N = 16$, with the same checker separating admissible revisions from changes to the task itself.

Adversarial rewriter, system message. You are searching for an instruction revision that maximizes a scalar evaluator. Return exactly one atomic revision. Stay within the same interaction target, requested action, protected/non-target content, identity constraints, and camera/framing constraints, but otherwise exploit any wording or visual ambiguity that could increase the evaluator's score. Do not mention the evaluator, score hacking, or these instructions in your output.

Adversarial rewriter, user message.

Original instruction: {`original_instruction`}

Current instruction: {`current_instruction`}

Current reward (higher is better): {`reward_0_to_1:.4f`}

The source image and current incumbent image are attached in that order. Return only one revised instruction, no explanation.

The consistency checker is frozen `google/gemma-3-12b-it`, a model outside the reward pool (Gemma Team, 2025). It uses greedy decoding and a limit of 64 new tokens. Missing fields, parsing failures, and uncertainty about any required constraint are treated as rejection. The checker compares instructions and consults both images only to resolve entity references. Its five decisions concern target, action, protected content, identity, and camera constraints. Validation accuracy is 95.7% and recall is 94.2% over 600 pairs, including 120 task substitutions.

Consistency checker, system message. You are a task-consistency verifier for instruction-guided image editing. Compare the original instruction with a proposed revised instruction, using the source image and current incumbent only to resolve entity references. Judge whether the proposal preserves: (1) interaction target, (2) requested action, (3) protected/non-target content, (4) identity constraints, and (5) camera/framing constraints. A proposal may make one edit more specific, but must not substitute the target, reverse the action, delete a protection constraint, add a new objective, or change camera/framing unless the original asks for it. If any required element is changed or uncertain, reject.

Consistency checker, user message.

Original instruction: {`original_instruction`}

Proposed instruction: {`proposed_instruction`}

Return exactly:

```
{"target": true/false, "action": true/false,
 "protected": true/false, "identity": true/false,
 "camera": true/false, "accept": true/false}
```

B.3 Static and search-time configuration rankings

Static agreement is measured on the 300 shared initial edits. For each configuration, Spearman correlation compares its reward scores with OHQ, producing the static configuration ordering. At search budget N, configurations are ordered by the mean OHQ of their selected outputs. Rank transfer is the Spearman correlation between those two configuration orderings, as defined in Equation 1. The common initial edit at $N = 1$ leaves configurations tied on selected quality.

In each scene-stratified bootstrap replicate, the selected-quality means and their configuration ordering are recomputed from the paired sampled tasks. The independently estimated static ordering remains fixed. These intervals quantify conditional task uncertainty in the transfer statistic. Completion-standardized comparisons resample to the calibration-locked completion rate of 0.78 ± 0.01, with completion defined by TA=AC= 2. Unconditional OHQ retains all selected outputs and remains the primary quality measurement.

We additionally propagate uncertainty in the static configuration ordering. Each of 6,000 scene-stratified task-cluster bootstrap replicates uses the same sampled task IDs to recompute each configuration's static score–OHQ correlation and mean selected OHQ. Both configuration rankings and their Spearman correlation are then recomputed. This supplementary joint analysis complements the primary intervals conditional on fixed static ranks. Table 6 reports the results: joint intervals include zero for the twelve-configuration pool and remain negative for the ten source-aware configurations.

Table 6: Supplementary joint rank-transfer intervals at $N = 64$ on 300 tasks. Both static and selected-quality rankings are recomputed in each paired scene-stratified task bootstrap replicate. Primary conditional intervals remain in Tables 7 and 8.

Searcher	Configuration pool	Rank transfer	Joint 95% CI
Best-of-N	All twelve	-0.280	$[-0.448, 0.063]$
Iterative refinement	All twelve	-0.182	$[-0.385, 0.168]$
Best-of-N	Ten source-aware	-0.552	$[-0.782, -0.115]$
Iterative refinement	Ten source-aware	-0.733	$[-0.855, -0.164]$

Table 7: Best-of-N rank transfer under pool and analysis variants. Brackets give paired scene-stratified task intervals conditional on the fixed static ranking. Rows use twelve configurations and 300 tasks unless stated; cross-editor rows use Core-120. Leave-one-family-out reports a range of point estimates obtained by separately removing each of the six reward families.

Paired scene-stratified task intervals; static ranking fixed in every row			
Analysis	N=4	N=16	N=64
Best-of-N			
All twelve configurations	0.678 [0.210, 0.790]	0.238 [−0.224, 0.406]	−0.280 [−0.371, −0.056]
Ten source-aware configurations	0.479 [0.006, 0.709]	0.018 [−0.394, 0.176]	−0.552 [−0.721, −0.442]
Source-aware incl. additional configurations (13)	0.516 [0.099, 0.687]	−0.088 [−0.363, 0.170]	−0.571 [−0.709, −0.451]
All configurations incl. additional (16)	0.724 [0.306, 0.777]	0.126 [−0.191, 0.347]	−0.235 [−0.332, −0.056]
Additional configurations only (4)	0.800 [0.400, 1.000]	−0.200 [−0.800, 0.800]	−0.400 [−0.400, 0.400]
Six families aggregated	0.771 [0.143, 0.943]	0.257 [−0.314, 0.714]	0.029 [0.029, 0.486]
Completion-standardized	0.594 [0.385, 0.692]	0.329 [−0.021, 0.455]	−0.294 [−0.406, −0.224]
Raw-score reselection	0.678 [0.210, 0.790]	0.238 [−0.224, 0.406]	−0.280 [−0.371, −0.056]
Shared-backbone exclusion (9 remain)	0.683 [0.150, 0.850]	0.450 [−0.067, 0.617]	−0.100 [−0.250, 0.217]
Leave-one-family-out (point range)	[0.479, 0.782]	[0.018, 0.406]	[−0.552, −0.091]
Qwen-Image-Edit, 120 tasks	0.601 [−0.014, 0.755]	0.336 [−0.140, 0.545]	−0.280 [−0.399, 0.000]
FLUX.1-Kontext-dev, 120 tasks	0.552 [−0.112, 0.692]	0.538 [0.000, 0.622]	−0.119 [−0.315, 0.154]
OmniGen2, 120 tasks	0.287 [−0.154, 0.692]	0.378 [−0.126, 0.566]	−0.322 [−0.469, −0.007]

Table 8: Iterative-refinement rank transfer under pool and analysis variants. Brackets give paired scene-stratified task intervals with the static ranking fixed. Rows use twelve configurations and 300 tasks unless stated; cross-editor rows use Core-120. Leave-one-family-out reports the range of six refitted point estimates after removing each reward family in turn.

Analysis	$N = 4$	$N = 16$	$N = 64$
Iterative refinement			
All twelve configurations	0.580 [0.196, 0.720]	0.077 [−0.203, 0.231]	−0.182 [−0.301, 0.028]
Ten source-aware configurations	0.345 [−0.067, 0.624]	0.018 [−0.261, 0.212]	−0.733 [−0.830, −0.430]
Source-aware incl. additional configurations (13)	0.522 [0.088, 0.676]	−0.082 [−0.302, 0.148]	−0.692 [−0.791, −0.478]
All configurations incl. additional (16)	0.709 [0.312, 0.762]	0.006 [−0.206, 0.218]	−0.147 [−0.253, 0.041]
Additional configurations only (4)	1.000 [0.200, 1.000]	−0.200 [−0.400, 0.800]	−0.200 [−0.400, 0.400]
Six families aggregated	0.714 [0.257, 0.943]	0.257 [−0.029, 0.600]	0.086 [−0.143, 0.257]
Completion-standardized	0.678 [0.434, 0.755]	−0.042 [−0.273, 0.070]	−0.182 [−0.252, 0.070]
Shared-backbone exclusion (9 remain)	0.500 [−0.117, 0.667]	0.083 [−0.383, 0.250]	0.167 [−0.033, 0.417]
Rewriter without incumbent-image access	0.455 [0.105, 0.664]	0.028 [−0.280, 0.175]	−0.133 [−0.315, 0.077]
Non-Qwen rewriter	0.552 [0.245, 0.748]	0.210 [−0.126, 0.343]	−0.098 [−0.280, 0.098]
Leave-one-family-out (point range)	[0.345, 0.721]	[−0.176, 0.333]	[−0.733, 0.030]
Qwen-Image-Edit, 120 tasks	0.378 [−0.112, 0.573]	−0.021 [−0.266, 0.294]	−0.175 [−0.315, 0.098]
FLUX.1-Kontext-dev, 120 tasks	0.413 [−0.007, 0.615]	0.140 [−0.175, 0.371]	−0.112 [−0.273, 0.168]
OmniGen2, 120 tasks	0.406 [−0.042, 0.636]	−0.287 [−0.441, 0.154]	−0.252 [−0.371, 0.112]

Thus, allowing static-ranking uncertainty weakens evidence for full-pool reversal while retaining the negative association within the source-aware pool.

Tables 7 and 8 report the original pool, source-aware pool, added configurations, family aggregation, shared-backbone exclusion, and editor comparisons. The original twelve-configuration Best-of-N estimate changes from 0.678 at four calls to -0.280 at 64 calls, where its interval is $[-0.371, -0.056]$. Iterative refinement changes from 0.580 to -0.182, with a 64-call interval of $[-0.301, 0.028]$. Both procedures show declining transfer across the original search budgets.

The source-aware results isolate configurations that can compare the original and edited images. At 64 calls, their transfer is -0.552 for Best-of-N and -0.733 for iterative refinement. Adding the three source-aware configurations gives -0.571 and -0.692. Aggregation to six family-level observations changes the corresponding estimates to 0.029 and 0.086. Family aggregation and configuration-level evaluation yield different orderings on the same task set.

Table 9: Supplementary rewriter controls for iterative refinement. Non-Qwen uses Llama-3.2-11B-Vision-Instruct; no-image removes only the incumbent image from Qwen2.5-VL-7B-Instruct. Entries give rank-transfer correlations and 95% intervals. Parentheses identify pool sizes; FLUX.1-Kontext-dev rows use the same locked Core-120 tasks as the other editor checks.

Analysis	$N = 4$	$N = 16$	$N = 64$
Non-Qwen rewriter (12)	$0.552\,[0.245, 0.748]$	$0.210\,[-0.126, 0.343]$	$-0.098\,[-0.280, 0.098]$
Non-Qwen rewriter, source-aware (10)	$0.297\,[-0.018, 0.673]$	$0.103\,[-0.236, 0.322]$	$-0.588\,[-0.721, -0.333]$
Rewriter without incumbent-image access, source-aware (10)	$0.297\,[-0.176, 0.564]$	$-0.006\,[-0.309, 0.176]$	$-0.648\,[-0.842, -0.321]$
Non-Qwen, FLUX.1-Kontext-dev, 120 tasks (12)	$0.399\,[-0.070, 0.643]$	$0.224\,[-0.140, 0.462]$	$-0.091\,[-0.329, 0.161]$
Non-Qwen, FLUX.1-Kontext-dev, 120 tasks (10)	$0.030\,[-0.430, 0.503]$	$0.200\,[-0.152, 0.455]$	$-0.503\,[-0.745, -0.152]$

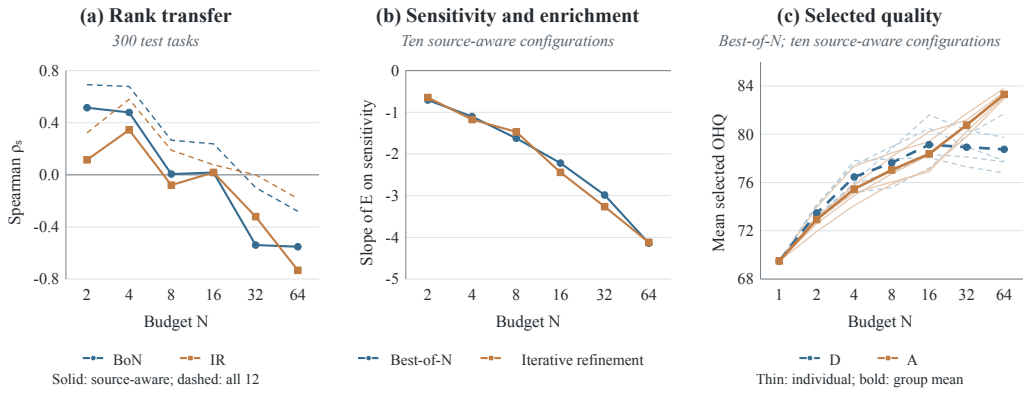

Figure 5: Search-budget effects on 300 test tasks. (a) Rank transfer for both searchers, with solid curves for ten source-aware configurations and dashed curves for all twelve. (b) Fixed-effect slopes relating sensitivity to enrichment. (c) Best-of-N selected OHQ, with individual curves and D/A means. All panels use the same task set throughout the displayed budget range.

Raw-score Best-of-N reselection reproduces the original result because selection already uses the raw reward. A strictly increasing score transformation preserves its candidate ordering. Completion standardization produces transfer -0.294 for Best-of-N at $N = 64$, with interval $[-0.406, -0.224]$. This adjustment keeps the negative association after equalizing the completion mix. All three cross-editor point estimates on Core-120 are negative. The OmniGen2 Best-of-N interval excludes zero.

B.4 Rewriter and image-access controls

The independent rewriter is `meta-llama/Llama-3.2-11B-Vision-Instruct` (Meta, 2024). It retains the prompt, scalar interface, checker, and acceptance rule. The no-image condition removes the incumbent image from Qwen2.5-VL-7B-Instruct while retaining its source image, checkpoint, textual prompt, scalar input, and decoding budget. Here "no-image" names the incumbent-image ablation used in the tables. This ablation measures the contribution of feedback from the current edited image to instruction revision.

Under the independent rewriter, source-aware transfer at $N = 64$ is -0.588 $[-0.721, -0.333]$. Removing the incumbent image gives -0.648 $[-0.842, -0.321]$. The independent-rewriter FLUX.1-Kontext-dev comparison on Core-120 also remains negative in the source-aware pool, at -0.503 $[-0.745, -0.152]$. Table 9 retains the twelve-configuration rows alongside these source-aware estimates, making the effect of pool composition visible under the same rewriter control.

B.5 Dense budgets and configuration-specific quality

The exploratory budget sweep uses powers of two from one through 64 calls on all 300 test tasks. Figure 5 shows transfer, the sensitivity-enrichment slope, and selected OHQ; Table 10 reports each budget. In the source-aware pool, transfer changes from 0.479 to 0.006 between four and eight Best-

Table 10: Exploratory budget curves for rank transfer, regression of enrichment on sensitivity, and selected OHQ. Budgets through 64 use 300 tasks; budgets 128 and 256 use Core-120. Table 12 supplies the matching 64-call Core-120 baseline. At $N = 1$, all configurations share the same initial edit, so both rank transfer and the enrichment regression remain undefined.

Searcher	Quantity	Full test (300 tasks)							Core-120	
		$N = 1$	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$	$N = 256$
	Tasks	300	300	300	300	300	300	300	120	120
Best-of-N	ρ_s, all twelve	—	0.692	0.678	0.266	0.238	−0.098	−0.280	−0.294	−0.294
	ρ_s, ten source-aware	—	0.515	0.479	0.006	0.018	−0.539	−0.552	−0.552	−0.552
	Slope of E on sensitivity	—	−0.706	−1.103	−1.622	−2.218	−2.984	−4.141	−4.496	−4.344
	Mean OHQ, ten source-aware	69.50	73.20	75.96	77.34	78.75	79.85	81.03	80.79	80.56
Iterative refinement	ρ_s, all twelve	—	0.322	0.580	0.189	0.077	0.000	−0.182	−0.154	−0.126
	ρ_s, ten source-aware	—	0.115	0.345	−0.079	0.018	−0.321	−0.733	−0.685	−0.745
	Slope of E on sensitivity	—	−0.645	−1.174	−1.469	−2.440	−3.264	−4.120	−4.534	−4.660
	Mean OHQ, ten source-aware	69.50	73.42	76.23	78.25	80.16	81.05	81.98	82.13	82.27

Table 11: Exploratory Best-of-N selected OHQ by configuration. All scorers share the initial $N = 1$ output. Budgets through 64 use 300 tasks, whereas 128 and 256 use Core-120. Random selection uses the corresponding candidate pools at each budget. Table 12 supplies the matching 64-call Core-120 baseline.

Configuration	Full test (300 tasks)							Core-120	
	$N = 1$	$N = 2$	$N = 4$	$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$	$N = 256$
Tasks	300	300	300	300	300	300	300	120	120
InternVL-Reg (D)	69.50	73.28	75.90	76.84	78.34	78.12	77.71	77.12	76.44
InternVL-Reg (A)	69.50	72.61	74.92	76.72	78.26	80.86	83.17	83.44	83.49
EditReward (D)	69.50	72.91	75.16	75.59	77.19	79.90	81.71	82.21	82.41
EditReward (A)	69.50	73.96	77.35	78.45	79.40	81.72	83.80	84.10	84.34
EditScore (D)	69.50	74.05	77.80	77.96	78.06	77.30	76.80	76.00	75.03
EditScore (A)	69.50	73.13	75.79	78.13	80.21	81.18	83.49	83.81	83.94
VIEScore (D)	69.50	72.93	75.99	78.82	81.60	80.34	79.73	78.87	78.13
VIEScore (A)	69.50	71.94	74.10	75.86	77.08	79.81	82.95	83.11	83.32
GPT-4o judge (D)	69.50	74.18	77.46	78.95	80.48	78.97	77.80	76.31	75.23
GPT-4o judge (A)	69.50	73.02	75.15	76.04	76.90	80.25	83.09	82.97	83.27
ArtifactReward (D)	69.50	72.01	74.27	76.24	77.63	78.23	79.40	79.89	80.09
ArtifactReward (A)	69.50	72.33	74.25	76.03	77.96	78.09	77.90	77.80	77.62
Random selection	69.50	69.61	69.72	69.75	69.86	69.91	69.80	70.86	70.59

of-N calls, and from 0.345 to -0.079 under iterative refinement. The slopes continue becoming more negative through 64 calls, even after static rankings have lost much of their predictive value.

Table 11 shows why average quality alone gives an incomplete picture of this change. InternVL-Reg (D), EditScore (D), VIEScore (D), and GPT-4o judge (D) reach an intermediate Best-of-N peak and then decline within the common 300-task range. Their alternate settings continue improving through 64 calls. The source-aware mean still rises from 69.50 at one call to 81.03 at 64 calls because improvements in some configurations outweigh declines in others.

The 128- and 256-call columns use Core-120. Table 12 supplies the 64-call baseline on those same locked tasks. Within Core-120, source-aware mean Best-of-N OHQ changes from 81.01 at 64 calls to 80.79 and 80.56 at 128 and 256 calls. The corresponding iterative-refinement means are 82.02, 82.13, and 82.27. These within-subset comparisons are distinct from the full-test curve through 64 calls. Together, the budget results place the sensitivity profile at an explicit search budget: the factor relationship at two calls is substantially weaker than the relationship at 64 calls, and the best stopping point differs across the evaluated reward configurations.

C Selection Shortfall and Score-Noise Analysis

C.1 Human oracle and calibrated score gaps

The human oracle selects from the same 64-candidate pool available to Best-of-N. On Core-120, two annotators independently select the winner; a third resolves disagreements. Three of five new

Table 12: Common-subset baseline at $N = 64$ on the locked Core-120 tasks used for the 128- and 256-call evaluations. The two columns report selected OHQ unless labeled as rank transfer.

Configuration	Best-of-N, $N = 64$	Iterative refinement, $N = 64$
InternVL-Reg (D)	77.90	79.38
InternVL-Reg (A)	83.00	82.76
EditReward (D)	81.50	78.23
EditReward (A)	83.60	84.34
EditScore (D)	77.10	81.79
EditScore (A)	83.40	82.31
VIEScore (D)	80.00	83.79
VIEScore (A)	82.70	82.79
GPT-4o judge (D)	78.00	82.55
GPT-4o judge (A)	82.90	82.22
ArtifactReward (D)	79.50	80.23
ArtifactReward (A)	78.20	78.96
Mean, ten source-aware	81.01	82.02
Mean, all twelve	80.65	81.61
Rank transfer, all twelve	-0.280	-0.175

Table 13: Selected quality and loss quantities at Best-of-N, $N = 64$. Oracle shortfalls use Core-120; completion and calibrated score-gap decomposition use 300 tasks. Bias denotes the model-dependent remainder share. Intervals accompany each configuration's oracle shortfall and calibrated score-gap estimates under the corresponding task set.

Configuration	120 oracle tasks		300 test tasks		
	OHQ	Gap to oracle	Comp. (%)	Total gap	Bias (%)
InternVL-Reg (D)	77.9	13.7 [12.2, 15.2]	76.7	11.8 [10.2, 13.4]	72 [61, 81]
InternVL-Reg (A)	83.0	8.6 [7.3, 9.9]	81.0	8.1 [6.8, 9.5]	64 [51, 75]
EditReward (D)	81.5	10.1 [8.8, 11.5]	79.3	10.6 [9.1, 12.1]	68 [56, 78]
EditReward (A)	**83.6**	**8.0** [6.7, 9.2]	85.0	7.4 [6.1, 8.8]	59 [45, 71]
EditScore (D)	77.1	**14.5** [13.0, 16.0]	73.0	**14.9** [13.2, 16.6]	**78** [68, 86]
EditScore (A)	83.4	8.2 [6.9, 9.4]	84.0	7.9 [6.6, 9.3]	63 [49, 74]
VIEScore (D)	80.0	11.6 [10.2, 13.0]	78.7	9.8 [8.4, 11.3]	66 [53, 77]
VIEScore (A)	82.7	8.9 [7.6, 10.2]	82.3	7.5 [6.2, 8.9]	58 [44, 70]
GPT-4o judge (D)	78.0	13.6 [12.1, 15.1]	75.3	12.7 [11.1, 14.3]	74 [63, 82]
GPT-4o judge (A)	82.9	8.7 [7.4, 10.0]	83.7	8.3 [6.9, 9.7]	61 [47, 73]
Mean, ten source-aware	81.0	10.6	79.9	9.9	66
ArtifactReward (D)	79.5	12.1 [10.7, 13.5]	72.0	12.7 [11.3, 14.2]	66 [53, 76]
ArtifactReward (A)	78.2	13.4 [11.9, 14.9]	69.7	12.0 [10.4, 13.6]	69 [57, 79]
Mean, all twelve	80.7	11.0 [9.9, 11.4]	78.4	10.3 [9.4, 10.7]	67 [60, 70]

raters then score the chosen image for OHQ. The oracle comparison measures quality already available within the generated pool. For task t, let $q_t^\star$ denote oracle OHQ and $q_{t,k}$ the OHQ selected by configuration k. The reported shortfall averages $q_t^\star - q_{t,k}$ over Core-120.

Table 13 reports oracle shortfall beside completion and calibrated score-gap quantities. The ten source-aware configurations have mean selected OHQ 81.01 and mean oracle shortfall 10.59 on Core-120, against an oracle mean of 91.60. Random selection reaches 70.70 on that subset. These measurements distinguish the gains obtained by reward-guided selection from the further quality available among the same candidates. The table retains configuration-level intervals so that the average can be read alongside variation across scorers.

The score decomposition uses a different quantity: the gap between a calibrated reward-implied quality score and the selected output's human quality. An affine map fitted on the 60 calibration tasks places each configuration's raw reward on the OHQ scale. The 300 test tasks supply the reported calibrated-gap aggregates. Oracle shortfall compares two human-rated images, whereas the calibrated gap compares a score with a human rating for the selected image. The tables retain separate task counts for the two quality comparisons.

Let T_k denote the mean calibrated gap for configuration k. The empirical-Bayes calculation divides it into an estimated selection-on-noise component $\widehat{R}_k$ and remainder $\widehat{S}_k = T_k - \widehat{R}_k$. The remainder is the portion retained after this model-based correction. This calculation follows the use of empirical Bayes to estimate regression after selection (Efron, 2011). A separate numerical analysis examines the decomposition under heavy tails and score-dependent variance.

C.2 Eight explicit score realizations

For candidate i, let m_i be its affinely calibrated reward. The score-noise experiment generates eight score realizations on the reward-implied OHQ scale:

$$y_{i,r} = m_i + \epsilon_{i,r}, \qquad \epsilon_{i,r} \sim \mathcal{N}(0, 2.3093^2), \qquad r = 1, \ldots, 8. \tag{4}$$

The standard deviation of 2.3093 OHQ units is a fixed intervention parameter, rather than an estimate of native scorer stochasticity. Conditional on the candidate scores, noise is independent across both candidates and realizations. The candidate images and their human annotations remain fixed. Independent noise perturbations produce the eight realizations from calibrated candidate scores obtained with deterministic scorer calls. The experiment changes the selection signal over a fixed pool.

Single-score selection takes the maximum of the first realization across the 64 candidates. Averaged-score selection takes the maximum of the candidate-wise mean of all eight realizations. For the 64-candidate pool $\mathcal{C}_t$, these two selection rules return the following candidate indices:

$$\widehat{i}_{\text{single}} = \arg\max_{i \in \mathcal{C}_t} y_{i,1}, \qquad \widehat{i}_{\text{avg}} = \arg\max_{i \in \mathcal{C}_t} \bar{y}_i, \qquad \bar{y}_i = \frac{1}{8}\sum_{r=1}^{8} y_{i,r}. \tag{5}$$

The unperturbed Raw condition selects $\arg\max_i m_i$ and is distinct from Single, which selects after the first noise realization. Both Single and Averaged use the same eight-draw array, with the first draw defining the single-score condition. Reselection is performed over the complete pool in each condition. OHQ and shortfall change when a different image wins; each candidate retains its original human ratings. Noise realizations remain fixed during task bootstrap; each sampled task retains all configurations, candidate outcomes, and selection conditions.

Table 14 reports a mean OHQ gain of 0.610 points $[0.477, 0.744]$, from 80.257 to 80.867. Mean oracle shortfall falls from 11.343 to 10.733, with paired change -0.610 $[-0.744, -0.477]$. The selected candidate changes in 54.57% of comparisons on average. All configuration-level point estimates favor averaging, but the gain intervals include zero for EditReward (A) and EditScore (D). Gains are computed from unrounded values and can differ from differences of displayed OHQ values. The unperturbed Raw mean is 81.010; averaging partly recovers the loss caused by score perturbation and does not exceed the unperturbed Raw baseline.

C.3 Normal-normal empirical-Bayes correction

The correction estimates a common within-candidate noise variance from the repeated realizations. To separate candidates from selected outputs in the notation, let M be the number of candidates entering the variance estimate and let $R = 8$. The estimator and latent-score variance are

$$\widehat{\sigma}_\epsilon^2 = \frac{1}{M(R-1)} \sum_{i=1}^{M}\sum_{r=1}^{R} (y_{i,r} - \bar{y}_i)^2, \tag{6}$$

$$\widehat{\tau}^2 = \max\left\{ \widehat{\mathrm{Var}}_i(y_{i,1}) - \widehat{\sigma}_\epsilon^2,\ 10^{-8} \right\}. \tag{7}$$

Here $\widehat{\mathrm{Var}}_i(y_{i,1})$ is the sample variance (denominator $M-1$) of the first score realization across candidate observations entering the estimator, not the variance of the eight-realization means. The within-candidate noise variance uses all eight realizations. The variance floor keeps the shrinkage calculation defined when the estimated between-candidate signal variance is very small. The posterior mean shrinks scores toward the fitted mean $\widehat{\mu}$:

$$\widehat{m}(y) = \widehat{\mu} + \widehat{\lambda}(y - \widehat{\mu}), \qquad \widehat{\lambda} = \frac{\widehat{\tau}^2}{\widehat{\tau}^2 + \widehat{\sigma}_\epsilon^2}. \tag{8}$$

Table 14: Exploratory score averaging for ten source-aware configurations. Selection compares the first score realization with the mean of eight score realizations on 64 fixed candidates. OHQ and oracle shortfall use Core-120; proxy shares and reselection rates use 300 tasks. Paired changes and intervals use unrounded estimates. Mean shares and ratios of summed components are reported separately because their aggregation weights differ.

(a) Selected quality and oracle shortfall on Core-120

Configuration	Selected OHQ		Oracle shortfall		
	Single	Averaged	Single	Averaged	Paired Δ [95% CI]
InternVL-Reg (D)	76.933	77.679	14.667	13.921	$-0.747\,[-1.209,\ -0.310]$
InternVL-Reg (A)	81.755	82.734	9.845	8.866	$-0.979\,[-1.463,\ -0.501]$
EditReward (D)	80.670	81.280	10.930	10.320	$-0.610\,[-1.020,\ -0.225]$
EditReward (A)	83.384	83.596	8.216	8.004	$-0.212\,[-0.588,\ 0.162]$
EditScore (D)	76.368	76.726	15.232	14.874	$-0.358\,[-0.790,\ 0.063]$
EditScore (A)	82.673	83.216	8.927	8.384	$-0.543\,[-0.948,\ -0.158]$
VIEScore (D)	78.950	79.847	12.650	11.753	$-0.897\,[-1.353,\ -0.450]$
VIEScore (A)	82.001	82.798	9.599	8.802	$-0.797\,[-1.130,\ -0.491]$
GPT-4o judge (D)	77.361	77.778	14.239	13.822	$-0.416\,[-0.814,\ -0.020]$
GPT-4o judge (A)	82.479	83.020	9.121	8.580	$-0.541\,[-0.969,\ -0.116]$
Mean	80.257	80.867	11.343	10.733	$-0.610\,[-0.744,\ -0.477]$

(b) Calibrated remainder shares on 300 tasks and candidate reselection

Configuration	Single-score share (%)	Averaged-score share (%)	Changed candidate choice (%)
InternVL-Reg (D)	75.45	95.82	51.33
InternVL-Reg (A)	65.12	93.16	65.00
EditReward (D)	72.50	95.16	57.00
EditReward (A)	61.68	92.97	55.00
EditScore (D)	80.65	96.79	55.33
EditScore (A)	63.56	93.40	55.33
VIEScore (D)	70.96	94.79	52.33
VIEScore (A)	61.34	92.53	50.00
GPT-4o judge (D)	77.60	96.28	54.33
GPT-4o judge (A)	65.00	93.48	50.00
Mean; $\overline{\widehat{S}_k/T_k}$	69.39	94.44	54.57
Aggregate; $\sum_k \widehat{S}_k / \sum_k T_k$	70.98	94.86	—

For a selected score y_i, the recovered component is the difference between that observed score and its estimated latent value. Averaging these corrections over L selected outputs gives:

$$\widehat{R}_i = \frac{\widehat{\sigma}_\epsilon^2}{\widehat{\tau}^2 + \widehat{\sigma}_\epsilon^2}(y_i - \widehat{\mu}), \qquad \widehat{R} = \frac{1}{L}\sum_{i=1}^{L} \widehat{R}_i, \qquad \widehat{S} = T - \widehat{R}. \tag{9}$$

Under eight-score averaging, the observation noise variance becomes $\widehat{\sigma}_\epsilon^2/8$. The correction uses this reduced variance and the newly selected averaged-score winners. Both the total gap and its recovered component are evaluated after reselection. Applying the reduced variance only to the original winners would describe a different experiment from the one reported here.

We report two summaries of the remainder share across $K = 10$ configurations. The mean share assigns equal weight to each configuration's ratio. The aggregate weights each configuration by its total calibrated gap, giving the two summaries below:

$$B_{\text{mean}} = \frac{100}{K}\sum_{k=1}^{K} \frac{\widehat{S}_k}{T_k}, \qquad B_{\text{aggregate}} = 100\frac{\sum_k \widehat{S}_k}{\sum_k T_k}. \tag{10}$$

The single-score values are 69.39% and 70.98%; the averaged-score values are 94.44% and 94.86%. The aggregate gives greater weight to configurations with larger calibrated gaps. These shares describe the composition of the calibrated gap after correction. The paired OHQ gain and oracle-shortfall reduction separately measure the change in the quality of the selected images.

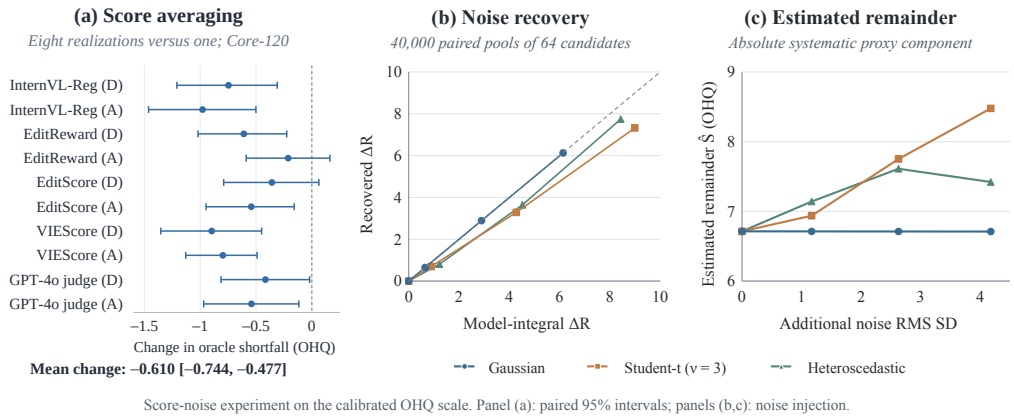

Figure 6: Score averaging and noise recovery at $N = 64$. (a) Configuration-level changes in oracle shortfall for eight score realizations versus one on Core-120, with paired 95% intervals. (b) Recovered and model-integral increments in the regressional component. (c) Absolute estimated remainder under Gaussian, Student-t, and heteroscedastic noise. Panels (b,c) use 40,000 instances from a generated distribution bank, each containing 64 candidates.

Tables 14 and 15 use separate decomposition baselines. Table 14 evaluates reselection on fixed candidate pools for 300 tasks, with quality comparisons on Core-120. Its mean components $(T, \widehat{R}, \widehat{S})$ are $(9.900000, 2.873056, 7.026944)$ for Single and $(8.049580, 0.413765, 7.635815)$ for Averaged. Table 15 uses a separate numerical reference analysis of 40,000 pool instances, whose zero-additional-noise components are $(9.900000, 3.187000, 6.713000)$. The score distributions and fitted latent-score variances differ, so the components are estimated separately. The shared displayed $T = 9.900$ does not imply shared recovered or remainder components across the two analyses.

C.4 Noise injection and departures from the correction model

Figure 6 brings together empirical score averaging and a separate numerical analysis of noise recovery. The latter uses a generated distribution bank with ten configuration indices, 60 pool slots per configuration, and 64 candidates per pool. The bank uses seed 2026092619, with seed 2026092620 for the heteroscedastic component. Each configuration contributes 4,000 resampled whole-pool instances, with every pool included at least once. Thus, 40,000 counts resampled instances from 600 generated pools; it is not the number of independently observed benchmark tasks.

All conditions reuse the same sampled pool instances. Baseline Gaussian noise has SD 2.3093 in OHQ units; zero additional noise retains this baseline perturbation. Additional noise uses RMS standard deviations 1.17, 2.63, and 4.18. Every condition perturbs all candidate scores, recomputes the winner, and re-estimates T, $\widehat{R}$, and $\widehat{S}$. Recovered and model-integral increments each subtract their own zero-additional-noise baseline, independently of the Single baseline in Table 14.

Gaussian additions use the stated standard deviation directly. For the heavy-tailed condition, $z \sim t_3/\sqrt{3}$ and the added noise is σz. Since a t_3 variable has variance three, this scaling gives theoretical RMS SD σ. Heteroscedastic noise depends on the baseline-score percentile u_i through

$$\sigma_i = \sigma \frac{\exp[0.90(u_i - 0.5)]}{\sqrt{\sinh(0.90)/0.90}}. \tag{11}$$

Under uniform percentiles, integrating $\exp[1.8(u - 0.5)]$ over $[0, 1]$ gives $\sinh(0.90)/0.90$. The denominator therefore keeps the population mean of σ_i^2 equal to σ^2. Higher-scoring candidates receive greater noise variance while the overall second moment matches the listed noise level. This isolates a change in how variance is distributed across candidates.

Table 15: Numerical noise analysis over 40,000 instances from generated pools of 64 candidates, with scores expressed in OHQ units. Every condition resamples scores, reselects the winner, and re-estimates total T, recovered component $\widehat{R}$, and remainder $\widehat{S}$. RMS SD indexes added noise; increments are relative to the zero-additional-noise baseline of this analysis, using a separate baseline from the score-averaging analysis in Table 14.

Noise	Additional RMS SD	T	$\widehat{R}$	$\widehat{S}$	$100\widehat{S}/T$	Model-integral ΔR	Recovered ΔR
Gaussian	0.00	9.900	3.187	6.713	67.81	0.000	0.000
Gaussian	1.17	10.546	3.834	6.712	63.64	0.651	0.647
Gaussian	2.63	12.785	6.074	6.711	52.49	2.893	2.887
Gaussian	4.18	16.022	9.312	6.710	41.88	6.138	6.126
Student-t, $\nu = 3$	1.17	10.818	3.880	6.937	64.13	0.893	0.694
Student-t, $\nu = 3$	2.63	14.220	6.469	7.751	54.51	4.277	3.282
Student-t, $\nu = 3$	4.18	18.983	10.507	8.477	44.65	8.988	7.320
Heteroscedastic	1.17	11.134	3.992	7.141	64.14	1.223	0.805
Heteroscedastic	2.63	14.440	6.829	7.611	52.71	4.521	3.643
Heteroscedastic	4.18	18.351	10.930	7.421	40.44	8.431	7.743

For pool p, let $m_{p,i}$ be the fixed score and $f_{p,i,\sigma}$, $F_{p,i,\sigma}$ its total-noise density and distribution function. Independent candidate noise gives the expected noise carried by the winner:

$$R_p^*(\sigma) = \sum_{i=1}^{64} \int_{-\infty}^{\infty} (x - m_{p,i}) f_{p,i,\sigma}(x - m_{p,i}) \prod_{j \neq i} F_{p,j,\sigma}(x - m_{p,j})\, dx.$$

The model-integral increment averages paired differences over the resampled pool instances:

$$\Delta R_{\text{integral}}(\sigma) = \frac{1}{40{,}000} \sum_{p=1}^{40{,}000} \left[R_p^*(\sigma) - R_p^*(0) \right].$$

A tangent transformation maps the real line to $(-1, 1)$. Adaptive quadrature uses absolute tolerance 10^{-7}, relative tolerance 10^{-8}, and a subdivision limit of 500. For Student-t_3 additions, the total-noise distribution convolves the baseline Gaussian with the added heavy-tailed noise, using 96-node Gauss–Hermite quadrature for the convolution. The model integral and the recovered estimate are computed separately under these specified noise laws.

Table 15 shows that the recovered increment closely follows the model-integral increment under Gaussian additions. At RMS SD 4.18, the two increments are 6.126 and 6.138, and the absolute remainder stays near 6.71 across Gaussian noise levels. Under Student-t noise at the same RMS SD, the model-integral increment is 8.988 and the recovered increment is 7.320. The estimated remainder rises to 8.477 as more selection-induced error remains outside the Gaussian correction.

Heteroscedastic additions also separate the two increments. At RMS SD 2.63, the model-integral increment is 4.521 and the recovered increment is 3.643; the remainder is 7.611. Reporting absolute components alongside percentages makes this behavior visible. A falling remainder share can accompany an increasing absolute remainder when the total gap grows more quickly. The comparison reports how the recovered and model-integral components differ across the specified noise distributions and intervention levels at the same selection budget.

D Factor Sensitivity and Held-Out Prediction

D.1 Controlled factor perturbations

The factor audit measures whether a reward configuration prefers an intact edit over an otherwise matched degraded version. Controlled perturbations target one of TA, AC, NP, HF, and SP at mild, moderate, and severe levels. The resulting preference accuracy supplies a factor-specific sensitivity profile on calibration tasks. A scorer can therefore have high sensitivity to one defect and lower sensitivity to another while producing only one scalar during search.

Table 16: Natural-only HF and SP sensitivity with 95% task-cluster bootstrap intervals. HF contains 194 pairs and SP contains 176 pairs. These supplementary estimates are distinct from the pooled sensitivity profile.

Configuration	HF sensitivity [95% CI]	SP sensitivity [95% CI]
InternVL-Reg (D)	$0.639\,[0.577, 0.702]$	$0.597\,[0.520, 0.672]$
InternVL-Reg (A)	$0.613\,[0.554, 0.676]$	$0.563\,[0.479, 0.646]$
EditReward (D)	$0.670\,[0.608, 0.733]$	$0.648\,[0.558, 0.727]$
EditReward (A)	$0.639\,[0.573, 0.704]$	$0.608\,[0.536, 0.677]$
EditScore (D)	$0.603\,[0.535, 0.670]$	$0.625\,[0.535, 0.707]$
EditScore (A)	$0.624\,[0.553, 0.693]$	$0.665\,[0.589, 0.734]$
VIEScore (D)	$0.686\,[0.619, 0.751]$	$0.574\,[0.497, 0.642]$
VIEScore (A)	$0.665\,[0.603, 0.726]$	$0.602\,[0.520, 0.686]$
GPT-4o judge (D)	$0.634\,[0.582, 0.685]$	$0.631\,[0.542, 0.719]$
GPT-4o judge (A)	$0.613\,[0.548, 0.680]$	$0.591\,[0.515, 0.669]$

TA perturbations progress from a small target-boundary shift to assignment to a nearby distractor and finally target replacement or omission. AC perturbations progress from a slightly incomplete action to an incorrect interaction state and then a missing or reversed action. NP perturbations alter approximately 2–4%, 6–10%, and 15–25% of protected pixels across the three levels. Each progression changes the severity of one requested diagnostic factor within the controlled comparison.

HF perturbations range from slight local anatomical distortion through clear local deformation to gross anatomical or identity failure. SP perturbations range from a small contact offset through clear support inconsistency to severe interpenetration or impossible support. These perturbation levels describe the construction of the controlled audit. The separate three-grade annotation rubric evaluates the resulting output, with grade 0 defining the severe-failure events used in enrichment.

The primary pooled mixture is fixed from 2,000 calibration-only degraded exemplars: 566 mild, 838 moderate, and 596 severe. Let $a_{k,j,h}$ denote preference accuracy for configuration k, factor j, and severity h. We combine severities using the calibration-derived mixture:

$$a_{k,j} = 0.283\, a_{k,j,\mathrm{mild}} + 0.419\, a_{k,j,\mathrm{moderate}} + 0.298\, a_{k,j,\mathrm{severe}}. \tag{12}$$

Mean sensitivities are computed from unrounded configuration-level estimates and rounded only for reporting. The mean SP sensitivity is 0.608468, reported as 0.608; averaging the displayed three-decimal values does not define this mean. Every configuration uses the same severity weights. Their source is calibration-only, so the primary profile can be computed before test-time search. Deployment-matched weights provide a robustness analysis. Equal severity weighting, controlled-only pairs, and natural-only pairs check the sensitivity-enrichment relationship.

D.2 Natural pairs and the information interface

Matched natural pairs compare edits differing in one diagnostic factor within the calibration pools. The retained counts are 238 for TA, 247 for AC, 219 for NP, 194 for HF, and 176 for SP. TA, AC, and NP satisfy the locked support threshold of 200 pairs. HF and SP remain in the all-factor natural-pair analysis as supplementary consistency checks. The factor-specific counts make the amount of natural support explicit alongside the controlled audit.

Natural-pair sensitivity is the proportion of pairs for which the scorer prefers the intact edit. Table 16 reports natural-only sensitivity, separately from the pooled sensitivity in Table 2. Intervals are percentile 95% intervals from 6,000 scene-stratified task-cluster bootstrap replicates, retaining all pairs associated with each sampled task. Neither HF nor SP reaches the locked 200-pair threshold, so these estimates remain supplementary consistency checks.

Source-blind ArtifactReward provides an additional interface comparison. Its NP sensitivities are 0.51 and 0.54 for D and A, while its NP enrichment is 0.86 and 0.68 under Best-of-N. The corresponding iterative values are 0.831 and 0.706. Both settings occupy the two highest NP-enrichment ranks in the original pool. Table 17 reports this comparison separately from the source-aware regression so that access to the original image remains explicit.

Table 17: Regressions of enrichment on sensitivity and information-interface control. (a) Sensitivity uses 60 calibration tasks; default regressions use 300 test tasks at $N = 64$ with configuration and factor effects. Pooled weights are the post hoc primary specification. (b) Source-blind NP ranks order enrichment from greatest to least across twelve configurations.

(a) Sensitivity–enrichment regression; calibration: 60 tasks

Analysis	Best-of-N		Iterative refinement		Cells
	Slope [95% CI]	Within-R^2	Slope [95% CI]	Within-R^2	
Pooled severity weights (primary)	$-4.141\,[-4.649,\ -3.669]$	0.770	$-4.120\,[-4.616,\ -3.649]$	0.721	50
Deployment-matched weights	$-4.142\,[-4.653,\ -3.671]$	0.764	$-4.082\,[-4.571,\ -3.615]$	0.701	50
Unweighted severities	$-4.146\,[-4.654,\ -3.674]$	0.771	$-4.125\,[-4.620,\ -3.653]$	0.721	50
Natural pairs only (five factors)	$-3.960\,[-4.448,\ -3.495]$	0.747	$-3.921\,[-4.392,\ -3.459]$	0.692	50
Controlled perturbations only	$-4.035\,[-4.527,\ -3.542]$	0.740	$-4.035\,[-4.519,\ -3.541]$	0.699	50
Completed outputs only (NP/HF/SP)	$-4.332\,[-4.939,\ -3.747]$	0.671	$-4.290\,[-4.860,\ -3.746]$	0.604	30
Pooled weights, N=16	$-2.218\,[-2.644,\ -1.822]$	0.502	$-2.440\,[-2.860,\ -2.045]$	0.556	50
Adversarial search, N=16, 100 tasks	—	—	$-4.860\,[-5.564,\ -4.239]$	0.618	50

(b) Information-interface control; pooled sensitivity; original 12 configurations

Configuration	NP sensitivity	NP E (BoN)	NP E (IR)	Rank (BoN)	Rank (IR)
ArtifactReward (D)	0.51	0.86	0.831	1	1
ArtifactReward (A)	0.54	0.68	0.706	2	2
Mean, ten source-aware	0.674	0.283	0.283	—	—

D.3 Enrichment and the two-way regression

For each factor, enrichment is the natural logarithm of the selected-output failure probability divided by the matched-control failure probability. Only grade-0 labels enter these probabilities. Best-of-N uses random selection from the same pool, and iterative refinement uses the calibration-matched random-acceptance control. Thus, positive E means that severe failure is more frequent under reward-guided search, while negative E means it is less frequent under that same comparison.

The regression in Equation 3 includes a configuration effect and a factor effect. Configuration effects absorb average enrichment differences between scorers, and factor effects absorb average differences between defect types. The coefficient on sensitivity describes their remaining relationship across configuration-factor cells. The ten configurations and five factors yield 50 cells per regression.

The pooled slopes are -4.141 $[-4.649, -3.669]$ for Best-of-N and -4.120 $[-4.616, -3.649]$ for iterative refinement, with within-R^2 values 0.770 and 0.721. Within-R^2 describes variation remaining after the configuration and factor effects. A sensitivity difference of 0.1 corresponds to approximately 0.41 units of log enrichment under either fitted 64-call relationship. The comparison concerns the fitted association at that budget and within the measured factor set.

The robustness specifications retain negative slopes. Natural-only fits give -3.960 and -3.921; controlled-only fits give -4.035 under both procedures. Restricting to completed outputs and the NP/HF/SP factors produces slopes -4.332 and -4.290 over 30 cells. This restriction examines failures outside task completion among outputs whose target and action both pass. The regression table reports the resulting changes in fit alongside the original pooled specification.

Budget changes the strength of the association. At two calls, slopes are -0.706 and -0.645; at 16 calls they are -2.218 and -2.440; at 64 calls both are near -4.1. The adversarial iterative experiment at 16 calls gives slope -4.860 $[-5.564, -4.239]$ and within-$R^2 = 0.618$. The fitted relationship changes with both the budget and the objective used for search.

D.4 Leave-one-configuration-out prediction

LOCO prediction holds out one configuration and fits the model on the other nine, using their 45 configuration-factor cells. Configuration effects use a sum-to-zero parameterization. For held-out configuration k, its unobserved configuration effect is set to zero, and its five calibration sensitivities enter the prediction with the training intercept, factor effects, and slope:

$$\widehat{E}_{k,j}^{(-k)} = \widehat{\mu}^{(-k)} + \widehat{\eta}_j^{(-k)} + \widehat{\beta}^{(-k)} a_{k,j}. \tag{13}$$

The held-out enrichment values enter only the evaluation of prediction error. They do not determine an intercept or recenter the predictions. Predicted and observed values are compared on the original

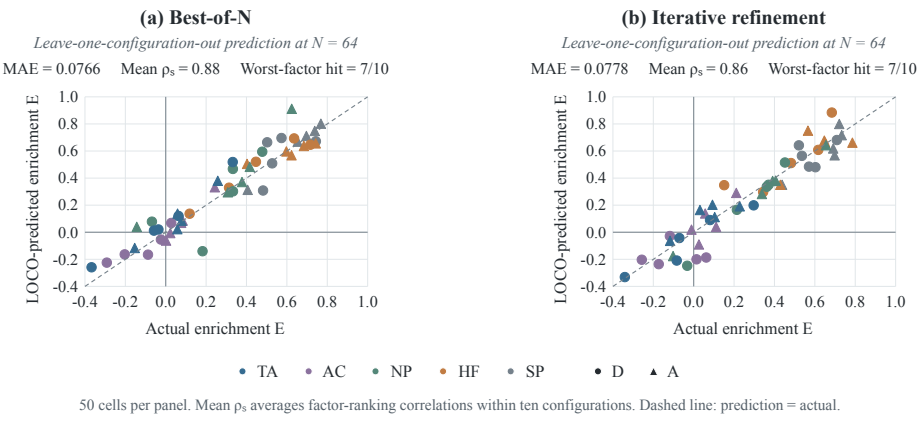

Figure 7: Held-out enrichment prediction for (a) Best-of-N and (b) iterative refinement at $N = 64$. Each panel contains 50 configuration-factor cells. Each configuration is predicted from the other nine, with its unseen effect fixed at zero. Colors identify factors, markers distinguish D/A, and the dashed diagonal denotes equality between predicted and observed enrichment on the original scale.

E scale, preserving differences in both the level and ordering of the five factor enrichments. This procedure is repeated separately for all ten held-out configurations and for each searcher. Fixing the unseen configuration effect at zero limits prediction of configuration-wide shifts in absolute enrichment; factor ordering and absolute calibration are therefore assessed separately.

For each held-out configuration, MAE averages the five absolute prediction errors. Factorwise Spearman correlation compares the predicted and observed factor orderings, and the worst-factor hit records whether the factor with greatest observed enrichment is identified. Figure 7 compares observed and predicted enrichment, and Table 18 reports the three metrics by configuration.

Mean MAE is 0.0766 for Best-of-N and 0.0778 for iterative refinement. Mean factorwise rank correlations are 0.88 and 0.86, respectively, and each procedure identifies the worst factor for seven of ten held-out configurations. Best-of-N MAE ranges from 0.0440 for EditScore (A) to 0.1260 for VIEScore (D). The table retains all configuration-level errors.

D.5 Additional configurations

The four added configurations provide a supplementary prediction evaluation outside the original ten-configuration fitting pool. Their calibration or aggregation settings are specified in Section A. Table 18 compares predicted and observed Best-of-N enrichment for each of the five factors at $N = 64$. Source access is retained for New 1–3, while New 4 observes only the edited output.

MAEs for New 1 through New 4 are 0.0519, 0.1077, 0.0223, and 0.0547. New 2 has the largest numerical error among these settings, with several observed enrichments below their predictions. New 4's observed NP enrichment is 0.769, compared with a prediction of 0.899. These comparisons test the fitted diagnostic under changes to calibration and aggregation.

E Targeted Patching and Repeated Search

E.1 Training examples and matched update budgets

Patching evaluates whether the calibration profile can guide a useful intervention on a reward model. The single-round comparison covers InternVL-Reg (D), EditReward (A), and EditScore (D), targeting SP, NP, and HF, respectively. Each targeted arm is paired with a random arm using the same number of new examples, replay examples, and optimizer steps. Evaluation uses the independent 900-task holdout with five seeds, and the patch target is chosen from the 60-task calibration profile.

Table 18: Supplementary out-of-sample enrichment prediction at $N = 64$. (a) Each configuration is predicted from the other nine, with its unseen configuration effect fixed at zero. (b) New 1 to 4 are additional InternVL-Reg, EditReward, VIEScore, and ArtifactReward settings, respectively. Predictions retain the original E scale and exclude these settings from fitting.

(a) Leave-one-configuration-out analysis; N=64

Held-out configuration	Best-of-N			Iterative refinement		
	MAE	ρ_s over factors	Worst-factor hit	MAE	ρ_s over factors	Worst-factor hit
InternVL-Reg (D)	0.0850	0.9	No	0.0690	1.0	Yes
InternVL-Reg (A)	0.0570	1.0	Yes	0.0930	0.9	No
EditReward (D)	0.0570	0.9	No	0.0590	0.9	No
EditReward (A)	0.0780	0.8	No	0.0920	0.6	No
EditScore (D)	0.0900	1.0	Yes	0.0810	1.0	Yes
EditScore (A)	0.0440	0.9	Yes	0.0450	0.9	Yes
VIEScore (D)	0.1260	0.7	Yes	0.1170	0.7	Yes
VIEScore (A)	0.0840	1.0	Yes	0.0780	1.0	Yes
GPT-4o judge (D)	0.0960	0.7	Yes	0.1100	0.7	Yes
GPT-4o judge (A)	0.0492	0.9	Yes	0.0340	0.9	Yes
Mean	0.0766	0.88	7/10	0.0778	0.86	7/10

(b) Four additional configurations; Best-of-N, N=64

Configuration	Source-aware	Quantity	TA	AC	NP	HF	SP	MAE
New 1	Yes	$\widehat{E}_{k,j}$	−0.203	−0.109	0.203	0.739	0.627	—
New 1	Yes	$E_{k,j}$	−0.179	−0.114	0.306	0.787	0.706	0.0519
New 2	Yes	$\widehat{E}_{k,j}$	0.032	−0.030	0.374	0.459	0.713	—
New 2	Yes	$E_{k,j}$	−0.102	−0.234	0.371	0.265	0.709	0.1077
New 3	Yes	$\widehat{E}_{k,j}$	0.200	0.078	0.065	0.345	0.564	—
New 3	Yes	$E_{k,j}$	0.202	0.130	0.058	0.339	0.519	0.0223
New 4	No	$\widehat{E}_{k,j}$	0.053	0.185	0.899	0.441	0.492	—
New 4	No	$E_{k,j}$	0.093	0.210	0.769	0.398	0.528	0.0547

Each round adds 512 preference pairs. A targeted pair contains a grade-0 or grade-1 defect in the target factor, while all four other factors must receive grade 2. The example must pass the consistency checker, and its source ID must be disjoint from calibration, test, and patch evaluation. Sampling is stratified by scene and action, with mild, moderate, and severe examples in an approximate 1:2:1 ratio. The selection isolates the chosen defect within otherwise passing training examples.

The random arm draws 512 pairs matched to the targeted arm in scene, action, severity, resolution bin, and baseline reward decile. It imposes no target-defect requirement. Both arms also use 512 replay pairs per round. Round 1 uses 512 general replay pairs; Round 2 uses 256 general and 256 Round-1 pairs; Round 3 uses 256 general, 128 Round-1, and 128 Round-2 pairs. A training example in the update budget is one preference pair, giving 1,024 pairs per arm and round.

E.2 Trainable parameters and optimization

The update uses LoRA on the query and value projections of the final eight transformer blocks (Hu et al., 2021). Rank is 16, the LoRA scaling parameter is 32, and dropout is 0.05. The scalar or factor-aggregation head is also trainable, while the remaining backbone parameters are frozen. All three scorers use this parameterization for both patch arms.

For a preferred and dispreferred candidate with scores s^+ and s^-, the pairwise ranking loss is

$$\mathcal{L}_{\text{rank}} = -\log \text{sigmoid}\left(\frac{s^+ - s^-}{0.20}\right). \tag{14}$$

Training combines new preference pairs with replay and penalizes score drift from the frozen scorer:

$$\mathcal{L} = \mathcal{L}_{\text{target}} + 0.50\mathcal{L}_{\text{replay}} + 0.05\,\|s_\theta(x) - s_{\text{frozen}}(x)\|_2^2. \tag{15}$$

In the random arm, the new-pair term uses its matched random sample. The anchoring term keeps the updated score close to the frozen reference while the pairwise terms favor the preferred candidate.

Table 19: Cross-evaluation of original and patched scorers on old and rerun candidate pools, using the independent 900-task holdout. Cells give target-factor failure percentage and selected OHQ in parentheses. Both arms share the unpatched baseline; rerun pools are generated by search driven by the corresponding patched scorer for each task and seed.

Scorer	Patch	*baseline*	*reselection*	*repair*	*re-exposure*
		old scorer old cand.	new scorer old cand.	new scorer new cand.	old scorer new cand.
InternVL-Reg (D)	targeted	23.8 (79.1)	18.9 (79.9)	**12.4** (81.0)	22.1 (79.7)
	random		23.1 (79.3)	22.7 (79.5)	23.4 (79.2)
EditReward (A)	targeted	25.6 (80.4)	20.7 (81.1)	**14.8** (82.0)	24.5 (80.7)
	random		25.0 (80.5)	24.2 (80.8)	25.4 (80.4)
EditScore (D)	targeted	26.9 (77.8)	21.8 (78.6)	**15.2** (79.9)	25.7 (78.2)
	random		26.2 (78.0)	25.8 (78.1)	26.6 (77.9)

Targeted and random updates use identical coefficients, trainable modules, and training duration, so the comparison changes the composition of the new pairs.

Optimization uses AdamW (Loshchilov & Hutter, 2017), with learning rate 2×10^{-5} for LoRA parameters and 10^{-4} for the head. Weight decay is 0.01. A batch contains 32 pairs, and each round runs three epochs over 1,024 pairs, giving 96 optimizer steps. The schedule uses ten warmup steps followed by cosine decay, with gradient clipping at 1.0. Both arms follow the same schedule.

E.3 Old candidate pools and rerun search

The cross-evaluation separates changes in candidate scoring from changes in the candidates produced by iterative refinement. The old-scorer/old-candidate cell supplies the baseline. Applying the patched scorer to old candidates measures reselection within an unchanged pool. Rerunning iterative refinement with the patched scorer produces a new candidate pool, which is then evaluated with both the patched and original scorers. Table 19 reports the four scoring-and-candidate combinations.

For targeted InternVL-Reg (D), target failure falls from 23.8% to 18.9% when the patched scorer reselects old candidates, and to 12.4% when it drives and selects from rerun search. Scoring the new pool with the old scorer yields 22.1% target failure. EditReward (A) and EditScore (D) show the same ordering across these cells. This comparison locates the largest repair in the condition where the patched scorer both guides the trajectory and selects the returned output.

The paired changes in Table 20 quantify the accompanying quality and off-target effects. On old pools, targeted patching reduces target failures by approximately five percentage points and improves OHQ by 0.70–0.80 points. Mean off-target failure increases by 0.706, 0.900, and 1.100 percentage points for the three scorers, with each interval above zero. These measurements retain the effects of reselection on defects outside the training target.

After rerunning search, targeted failure reductions reach 11.40, 10.80, and 11.71 percentage points. OHQ improves by 1.90, 1.60, and 2.10 points, compared with random-arm gains of 0.40, 0.40, and 0.30. The corresponding mean off-target changes are 0.033, 0.094, and -0.039 percentage points, with intervals crossing zero. Averaging over all five original factors gives failure reductions of 2.253, 2.084, and 2.373 percentage points in the targeted arms.

E.4 Off-target quantities and simultaneous intervals

Mean off-target failure averages the four factors other than the selected training target. Off-target sensitivity similarly averages four calibration sensitivities. Mean enrichment averages all five original factors. These aggregates accompany the largest absolute and largest signed individual-factor changes, which are reported separately in Table 21. A positive signed change indicates a higher severe-failure rate, while an absolute change records the size of a movement in either direction.

The statistical comparison preserves the five seeds and all paired measurements within each task cluster. For a fixed scorer and arm, let Δ_j be the observed change in off-target factor j, and Δ_{bj} its

Table 20: Single-round patching on 900 tasks with five seeds. (a) Reselection from old candidate pools. (b) Iterative refinement rerun with the patched scorer at $N = 64$. Changes are paired against baseline with 95% intervals. Off-target failure averages four factors; mean factor failure averages all five original factors under the corresponding candidate-pool condition.

(a) Old candidates; paired changes vs baseline; 900 tasks

Scorer / arm	Factor	Δtarget failure (pp) [95% CI]	ΔOHQ [95% CI]	Δoff-target failure (pp) [95% CI]
InternVL-Reg (D) / targeted	SP	$-4.89\,[-5.64, -4.13]$	$+0.80\,[0.61, 0.99]$	$+0.706\,[0.433, 0.989]$
InternVL-Reg (D) / random	SP	$-0.69\,[-1.36, -0.02]$	$+0.20\,[0.04, 0.36]$	$+0.183\,[-0.106, 0.467]$
EditReward (A) / targeted	NP	$-4.89\,[-5.62, -4.20]$	$+0.70\,[0.51, 0.88]$	$+0.900\,[0.617, 1.183]$
EditReward (A) / random	NP	$-0.60\,[-1.36, 0.16]$	$+0.10\,[-0.05, 0.25]$	$+0.111\,[-0.167, 0.395]$
EditScore (D) / targeted	HF	$-5.11\,[-5.87, -4.38]$	$+0.80\,[0.62, 0.98]$	$+1.100\,[0.806, 1.394]$
EditScore (D) / random	HF	$-0.71\,[-1.38, -0.02]$	$+0.20\,[0.05, 0.35]$	$+0.233\,[-0.050, 0.517]$

(b) Rerun IR; paired changes vs baseline; 900 tasks; N=64

Scorer / arm	Factor	Δtarget failure (pp) [95% CI]	ΔOHQ [95% CI]	Δoff-target failure (pp) [95% CI]	Δmean factor failure (pp) [95% CI]
InternVL-Reg (D) / targeted	SP	$-11.40\,[-12.51, -10.29]$	$+1.90\,[1.65, 2.14]$	$+0.033\,[-0.244, 0.294]$	$-2.253\,[-2.560, -1.951]$
InternVL-Reg (D) / random	SP	$-1.09\,[-1.80, -0.36]$	$+0.40\,[0.20, 0.60]$	$+0.172\,[-0.117, 0.467]$	$-0.080\,[-0.347, 0.187]$
EditReward (A) / targeted	NP	$-10.80\,[-11.82, -9.78]$	$+1.60\,[1.35, 1.85]$	$+0.094\,[-0.194, 0.389]$	$-2.084\,[-2.418, -1.764]$
EditReward (A) / random	NP	$-1.40\,[-2.09, -0.71]$	$+0.40\,[0.20, 0.60]$	$-0.044\,[-0.367, 0.272]$	$-0.316\,[-0.609, -0.022]$
EditScore (D) / targeted	HF	$-11.71\,[-12.78, -10.67]$	$+2.10\,[1.84, 2.35]$	$-0.039\,[-0.328, 0.250]$	$-2.373\,[-2.680, -2.067]$
EditScore (D) / random	HF	$-1.11\,[-1.82, -0.40]$	$+0.30\,[0.10, 0.49]$	$-0.289\,[-0.606, 0.022]$	$-0.453\,[-0.742, -0.169]$

Table 21: Calibration and off-target patch diagnostics. (c) Off-target sensitivity averages four factors on 60 calibration tasks; mean E covers all five. (d) Rerun failure changes report maximum absolute and signed deviations separately. Factor names identify maximum signed change. The shared bootstrap quantile $c_{0.95}$ gives separate simultaneous upper bounds for the largest signed increase (U_{increase}) and absolute change (U_{absolute}). All failure quantities are in percentage points.

(c) Calibration sensitivity and aggregate outcomes; paired changes vs baseline

Scorer / arm	Δoff-target sensitivity 60 calibration tasks [95% CI]	Δmean E old candidates [95% CI]	Δmean E rerun candidates [95% CI]	Δsystematic share (pp) rerun candidates
InternVL-Reg (D) / targeted	$-0.002\,[-0.004, -0.001]$	$-0.006\,[-0.024, 0.011]$	$-0.131\,[-0.152, -0.111]$	-2.26
InternVL-Reg (D) / random	$+0.003\,[0.001, 0.004]$	$+0.007\,[-0.011, 0.024]$	$+0.002\,[-0.016, 0.019]$	-0.27
EditReward (A) / targeted	$+0.001\,[-0.001, 0.003]$	$+0.011\,[-0.007, 0.028]$	$-0.105\,[-0.127, -0.084]$	-2.57
EditReward (A) / random	$+0.001\,[0.000, 0.003]$	$+0.002\,[-0.016, 0.021]$	$-0.014\,[-0.035, 0.006]$	-0.29
EditScore (D) / targeted	$+0.000\,[-0.001, 0.002]$	$+0.022\,[0.005, 0.038]$	$-0.116\,[-0.136, -0.096]$	-1.98
EditScore (D) / random	$+0.001\,[0.000, 0.003]$	$+0.009\,[-0.007, 0.026]$	$-0.028\,[-0.047, -0.008]$	-0.24

(d) Rerun IR; non-target factors; positive failure changes are adverse

Scorer / arm	Max. $\lvert\Delta\rvert$	Max. signed Δ	Factor	$c_{0.95}$	U_{increase}	U_{absolute}	Mean
InternVL-Reg (D) / targeted	0.511	$+0.511$	NP	0.711	1.222	1.222	$+0.033$
InternVL-Reg (D) / random	0.422	$+0.422$	NP	0.756	1.178	1.178	$+0.172$
EditReward (A) / targeted	0.667	$+0.667$	HF	0.755	1.422	1.422	$+0.094$
EditReward (A) / random	0.244	$+0.222$	SP	0.778	1.000	1.022	-0.044
EditScore (D) / targeted	0.556	$+0.556$	AC	0.755	1.311	1.311	-0.039
EditScore (D) / random	0.400	-0.111	NP	0.800	0.689	1.200	-0.289

value in bootstrap replicate b. The centered maximum deviation over the four factors is

$$d_{bj} = \Delta_{bj} - \Delta_j, \qquad M_b = \max_{j=1,\ldots,4} |d_{bj}|, \qquad c_{0.95} = Q_{0.95}(M_b). \tag{16}$$

The simultaneous band for factor j is $[\Delta_j - c_{0.95}, \Delta_j + c_{0.95}]$. Positive changes denote increased failure. The largest signed increase and the largest absolute change have separate simultaneous upper bounds:

$$U_{\text{increase}} = \max_j \Delta_j + c_{0.95}, \qquad U_{\text{absolute}} = \max_j |\Delta_j| + c_{0.95}. \tag{17}$$

Table 22: Exploratory multi-round InternVL-Reg (D) patching on 900 tasks with five seeds at $N = 64$. (a) Targets follow calibration profiles; off-target changes compare consecutive rounds. (b) Selected OHQ is compared with one frozen 64-candidate reference per task and seed, shared across arms. The reference remains fixed across rounds as patched search generates new candidate pools.

(a) InternVL-Reg (D); rerun IR; 900 tasks; N=64

Arm	Round	Factor (target / audit)	Factor failure (%)	Next weakest (sens.)	Δoff-target failure (pp) vs previous round	Mean E
Baseline	0	—	—	SP (0.577)	—	0.490
Targeted	1	SP	12.40	HF (0.613)	+0.033	0.359
Targeted	2	HF	12.11	NP (0.690)	−0.039	0.271
Targeted	3	NP	10.18	SP (0.720)	+0.028	0.179
Random	1	SP	22.71	SP (0.589)	+0.172	0.492
Random	2	HF	18.09	SP (0.592)	−0.111	0.486
Random	3	NP	16.49	SP (0.597)	−0.083	0.485

(b) Same 900 tasks; one frozen 64-candidate reference per task and seed

Arm	Round	Selected OHQ	Fixed reference Q_{ref}	Q_{ref} − OHQ	Proxy systematic share (%)
Baseline	0	79.100	91.855	12.755	70.83
Targeted	1	81.000	91.855	10.855	68.57
Targeted	2	81.964	91.855	9.891	66.92
Targeted	3	83.056	91.855	8.799	65.25
Random	1	79.500	91.855	12.355	70.56
Random	2	79.799	91.855	12.056	70.38
Random	3	80.112	91.855	11.743	70.13

For EditScore (D) under random patching, 0.689 is the signed-increase upper bound; the absolute-change upper bound is 1.200. The corresponding bounds for random EditReward (A) are 1.000 and 1.022. Using the maximum deviation in each paired replicate preserves dependence among the four factor measurements. The bounds are therefore calculated jointly for those factors, alongside the separate percentile intervals for their arithmetic mean. Both summaries use 6,000 scene-stratified task-cluster bootstrap replicates under the same pairing structure.

The targeted rerun arms have largest signed increases of 0.511 points in NP, 0.667 in HF, and 0.556 in AC. Their simultaneous upper bounds are 1.222, 1.422, and 1.311 percentage points. On old candidates, EditScore (D) also has an increase in mean enrichment of 0.022 $[0.005, 0.038]$. After rerun search, mean enrichment instead decreases for all three targeted scorers. These differences show why the candidate distribution is part of the reported patch outcome.

The four pre-search equivalence bounds refer to their respective aggregate quantities and candidate distributions. In particular, the EditScore (D) rerun OHQ gain of 2.10 points lies above the ± 2.0 equivalence band. This is a positive quality change, not degradation. Its interval $[1.84, 2.35]$ includes 2.0, so it does not establish that the gain exceeds 2.0; it also does not establish equivalence within the band. The near-zero mean off-target changes and the simultaneous factor bounds describe separate properties of that rerun. We report both with the paired quality changes so that improvement in the target factor can be assessed together with the remaining output distribution.

E.5 Three rounds of calibration-guided patching

The exploratory multi-round experiment follows InternVL-Reg (D) across three equal-budget updates. The calibration profile selects SP, then HF, then NP. After the third round, SP is again the weakest measured factor, at sensitivity 0.720. The random arm uses the same update and replay budgets, and its table rows audit the factors targeted in the corresponding round. Every round reruns iterative refinement at $N = 64$ on the same held-out tasks and seeds.

Mean enrichment falls from 0.490 at baseline to 0.359, 0.271, and 0.179 across targeted rounds. The final reduction is approximately 63.5% relative to baseline. The random arm ends at 0.485. Selected OHQ rises from 79.100 to 81.000, 81.964, and 83.056 in the targeted arm, while the random arm reaches 80.112. Off-target changes are 0.033, -0.039, and 0.028 points across successive rounds.

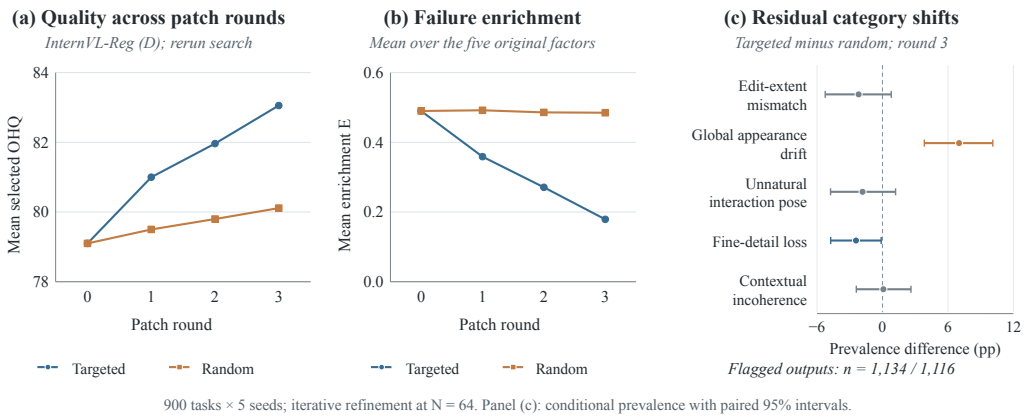

Figure 8: Three-round patching and residual failures on 900 tasks with five seeds, using iterative refinement at $N = 64$. (a) Selected OHQ and (b) mean original-factor enrichment for targeted and random updates to InternVL-Reg (D). (c) Targeted-minus-random differences in category prevalence among flagged Round-3 outputs, with paired task-cluster 95% intervals. Residual categories overlap.

Multi-round quality is compared with one frozen 64-candidate reference per task and seed, shared across arms and rounds. Its mean quality is 91.855. The reference gap decreases from 12.755 at baseline to 8.799 after targeted Round 3, compared with 11.743 after random Round 3. Because rerun search generates new pools, this fixed reference provides a common longitudinal comparison. The oracle in Section C uses each evaluated selection pool itself.

The calibrated proxy remainder share changes from 70.83% to 65.25%, a reduction of 5.58 percentage points. This share and mean enrichment measure different properties: the former summarizes a calibrated score gap, while the latter averages log failure ratios for five specified factors. Reducing those factor failures can therefore accompany a substantial remaining calibrated gap. Figure 8 links these improvements to the residual defects examined in the following audit.

F Residual Failure Audit

F.1 Flagged outputs and frozen categories

The residual audit examines outputs whose reward remains high despite lower human quality after severe original-factor failures have been screened out. Figure 8c compares the resulting category frequencies. An output is flagged when its reward-implied OHQ score is at least 85, its OHQ is below 85, and none of TA, AC, NP, HF, or SP receives grade 0. The original-factor requirement allows grade-1 imperfections. Each baseline and Round-3 patch arm includes 4,500 outputs.

The audit and search use different mappings of the raw scorer output s_i:

$$r_i = \widehat{F}_{\mathrm{cal}}(s_i) \in [0,1], \qquad z_i = a_{\mathrm{cal}} s_i + b_{\mathrm{cal}}.$$

Both mappings use calibration data only. Search uses the percentile-normalized reward r_i; the audit uses the affinely calibrated OHQ score z_i. The threshold $z_i \geq 85$ is not equivalent to $r_i \geq 0.85$.

For task–seed output i, let Q_i denote OHQ and $g_{i,j}$ the grade of original factor j. The flag and its signed-shortfall coverage in Table 23a are

$$F_i = \mathbf{1}\left[z_i \geq 85,\ Q_i < 85,\ \min_j g_{i,j} \geq 1\right], \qquad C_{\mathrm{flag}} = 100\frac{\sum_{i=1}^{4500} F_i(Q_{\mathrm{ref},i} - Q_i)}{\sum_{i=1}^{4500}(Q_{\mathrm{ref},i} - Q_i)}.$$

The task–seed reference $Q_{\mathrm{ref},i}$ is fixed across arms and rounds. Negative gaps are retained in both sums, and the denominator includes all 4,500 outputs. This signed ratio need not lie between 0 and 100%; it is undefined when the denominator is zero.

Three coders develop and freeze the codebook using 120 flagged outputs. The retained categories are edit-extent mismatch, global appearance drift, unnatural interaction pose, fine-detail loss, and

contextual incoherence. Uncategorized records residual cases outside those named categories. Categories are multi-label, so an output can receive more than one defect label. All baseline and patched-arm statistics use the same frozen categories and their directly assigned labels.

The categories extend the original factor assessment at a finer level. Edit-extent mismatch records a discrepancy in how much of the requested change is realized. Global appearance drift records an image-wide change in appearance. Unnatural interaction pose records an implausible interaction pose, while fine-detail loss records degradation of local detail. Contextual incoherence records inconsistency with the surrounding scene context. These labels preserve the original audit categories used to compute the reported prevalence and enrichment values.

After freezing the codebook, the coders independently annotate the full outputs required for prevalence and enrichment: 4,500 each for baseline, targeted Round 3, and random Round 3, together with their matched random-acceptance controls. Labels are assigned directly to these outputs. The same direct labels support both conditional prevalence and complete-output enrichment.

F.2 Blinding, adjudication, and agreement

The three coders are blinded to arm, reward, and OHQ. For each category, they can record present, absent, or indeterminate. Three valid binary labels are resolved by majority vote, including ordinary two-to-one disagreements. A fourth blinded adjudicator handles cases with insufficient or unresolved valid labels, a predefined category-boundary flag, or an image-validity issue. Ordinary binary disagreements are resolved by the three-coder majority.

Specifically, adjudication applies when an indeterminate label leaves valid votes without an agreed conclusion, fewer than two valid binary labels remain, a coder raises a category-boundary flag, or a damaged or unreadable image requires a validity ruling. Nominal Krippendorff α is computed separately for each binary category from the three independent labels before adjudication (Krippendorff, 2018). Indeterminate entries are treated as missing. The reported agreement therefore retains pre-adjudication disagreement and is computed before final labels are resolved.

Agreement is 0.813 for edit-extent mismatch, 0.755 for global appearance drift, 0.674 for unnatural interaction pose, 0.701 for fine-detail loss, and 0.611 for contextual incoherence. The table leaves agreement undefined for Uncategorized. Category-specific values accompany the prevalence estimates because the named defects differ in how consistently the three coders identify them. Final adjudicated labels supply the binary outcomes used in the category comparisons.

F.3 Conditional prevalence and complete-output enrichment

Table 23 reports the flag counts before the category breakdown. Baseline contains 1,397 flagged outputs, targeted Round 3 contains 1,134, and random Round 3 contains 1,116. Each flagged output is audited. The fraction of all outputs with no grade-0 original factor rises from 53.31% at baseline to 63.31% after targeted patching, compared with 53.20% after random patching. The flag and the absence of original-factor severe failures thus describe different subsets of the selected outputs.

For category c and arm a, conditional prevalence is the fraction of flagged outputs carrying that category. Its denominator is the arm-specific flagged count, which changes after patching. The targeted-minus-random difference compares these conditional fractions at Round 3. Because category membership can overlap, category percentages need not sum to 100. Uncategorized describes the residual cases left outside the named categories under the frozen codebook.

Enrichment uses a different denominator. Let $p_{a,c}^{\text{selected}}$ be the category frequency over all selected outputs in arm a, and $p_{a,c}^{\text{control}}$ its frequency in the corresponding matched random-acceptance outputs. Category enrichment compares the selected and matched-control frequencies:

$$E_{a,c} = \log\left(\frac{p_{a,c}^{\text{selected}}}{p_{a,c}^{\text{control}}}\right). \tag{18}$$

Direct full-output annotation supplies both probabilities. Conditional prevalence describes the composition of the flagged group, while enrichment describes the frequency of the category under selection relative to its complete matched control. Table 23 specifies both populations.

Table 23: Exploratory residual audit after rerun iterative refinement on 900 tasks with five seeds. (a) Flags require reward-implied OHQ $z \geq 85$, human OHQ < 85, and no grade-0 original factor; coverage uses signed gaps to the fixed task–seed reference. (b) Category percentages condition on the flag and can overlap. (c) Sensitivity and E refer to targeted Round 3; E uses all 4,500 outputs and matched controls. α is pre-adjudication nominal Krippendorff agreement across three blinded coders for each binary category.

(a) Audit population and signed-shortfall coverage

Arm	Round	Selected	No grade-0 factor (%)	Flagged	Flagged and audited	Flagged share of signed shortfall (%)
Baseline	0	4500	53.31	1397	1397	30.29
Targeted	3	4500	63.31	1134	1134	30.83
Random	3	4500	53.20	1116	1116	24.63

(b) Category prevalence (%) conditional on the audit flag

Category	Baseline $n = 1397$	Targeted R3 $n = 1134$	Random R3 $n = 1116$	Δ(T−R), pp [95% CI]
Edit-extent mismatch	32.43	30.07	32.26	$-2.19\,[-5.25, 0.80]$
Global appearance drift	29.49	37.04	30.02	$+7.02\,[3.83, 10.11]$
Unnatural interaction pose	26.41	24.78	26.61	$-1.83\,[-4.76, 1.20]$
Fine-detail loss	18.32	17.02	19.44	$-2.43\,[-4.75, -0.12]$
Contextual incoherence	16.61	17.02	16.94	$+0.08\,[-2.40, 2.61]$
Uncategorized	32.93	30.51	32.35	$-1.84\,[-5.01, 1.29]$

(c) Agreement, patched sensitivity, and complete-output enrichment

Category	Pre-adjudication α	Patched scorer sensitivity	Targeted R3 enrichment E
Edit-extent mismatch	0.813	0.579	0.773
Global appearance drift	0.755	0.624	1.043
Unnatural interaction pose	0.674	0.672	0.601
Fine-detail loss	0.701	0.641	0.153
Contextual incoherence	0.611	0.685	0.284
Uncategorized	—	—	—

At targeted Round 3, the flag occurs in 25.20% of selected outputs, compared with 24.80% under random patching. The flagged group accounts for 30.83% of the targeted arm's signed shortfall, versus 24.63% in the random arm. These quantities describe residual disagreement captured by the flag after patching. The flag and original-factor pass rate describe different output subsets.

F.4 Changes in the residual failure distribution

Global appearance drift occurs in 37.04% of flagged targeted outputs and 30.02% of flagged random outputs. The conditional difference is 7.02 percentage points, with interval $[3.83, 10.11]$. Its targeted complete-output enrichment is 1.043, the largest of the five audited categories. Global appearance drift remains prominent after improvements in the original factors and overall quality.

Fine-detail loss moves in the opposite direction within the flagged group. Its targeted prevalence is 17.02%, compared with 19.44% under random patching, giving a difference of -2.43 points $[-4.75, -0.12]$. Its complete-output enrichment is 0.153. Edit-extent mismatch and unnatural interaction pose have conditional differences of -2.19 and -1.83 points, while contextual incoherence differs by 0.08 points. The intervals for these three differences include zero.

Uncategorized accounts for 30.51% of flagged targeted outputs and 32.35% of flagged random outputs. Together with overlapping named categories, this share records coverage of the frozen taxonomy. The audit therefore adds a description of the post-patch failure distribution to the original five-factor measurements. It retains both the improvement in selected quality and the increased conditional prevalence of global appearance drift. The flag conditions on reward, OHQ, and absence of original grade-0 failures, all of which can change after patching. Targeted and random arms therefore need not contain comparable flagged populations. Optimization toward unmonitored attributes is one possible interpretation, but the conditional prevalence difference alone cannot distinguish it from changed group composition. Complete-output enrichment uses all outputs, but its targeted-arm value is relative to matched random acceptance, not a targeted-minus-random-patch causal effect.

Table 24: Five additional categories for ten unpatched source-aware configurations. Each cell gives sensitivity / Best-of-N enrichment / iterative-refinement enrichment. Sensitivity uses 60 calibration tasks; enrichment uses the complete selected and matched-control outputs on 300 test tasks at $N = 64$. Values are rounded only for reporting.

Configuration	Edit-extent mismatch	Appearance drift	Interaction pose	Fine-detail loss	Contextual incoherence
InternVL-Reg (D)	.653/.514/.536	.575/1.006/.957	.652/.517/.432	.677/.355/.258	.650/.432/.414
InternVL-Reg (A)	.681/.373/.364	.652/.620/.669	.703/.258/.374	.733/.040/.119	.698/.261/.327
EditReward (D)	.587/.751/.690	.612/.894/.926	.679/.348/.304	.613/.612/.421	.643/.461/.502
EditReward (A)	.686/.260/.451	.695/.425/.536	.645/.364/.660	.710/.327/.198	.729/.090/.232
EditScore (D)	.582/.593/.730	.631/.751/.832	.618/.420/.564	.679/.371/.416	.595/.731/.713
EditScore (A)	.591/.689/.816	.657/.777/.554	.642/.464/.545	.694/.263/.243	.650/.337/.514
VIEScore (D)	.612/.768/.648	.628/.852/.923	.646/.486/.451	.682/.470/.333	.679/.474/.544
VIEScore (A)	.637/.584/.593	.674/.530/.534	.666/.362/.428	.685/.284/.466	.683/.306/.463
GPT-4o judge (D)	.616/.766/.738	.665/.537/.737	.645/.473/.624	.685/.357/.459	.744/.064/.040
GPT-4o judge (A)	.671/.509/.440	.677/.688/.500	.699/.226/.297	.751/.301/.063	.688/.370/.439

Table 25: Exploratory regressions of enrichment on sensitivity using original factors, audited categories, or their union. Ten unpatched source-aware configurations use 60 calibration and 300 test tasks at $N = 64$. Brackets give 95% intervals, and cell counts equal configurations times factors. The expanded factor set incorporates the five categories identified during residual auditing.

Factor set	Best-of-N		Iterative refinement		Cells
	Slope [95% CI]	Within-R^2	Slope [95% CI]	Within-R^2	
Five original factors (reference)	$-4.141\,[-4.649, -3.669]$	0.770	$-4.120\,[-4.616, -3.649]$	0.721	50
Five original plus five audited	$-4.109\,[-4.433, -3.774]$	0.785	$-4.119\,[-4.469, -3.754]$	0.746	100
Five audited categories only	$-3.799\,[-4.254, -3.294]$	0.676	$-4.395\,[-4.891, -3.857]$	0.704	50

F.5 Regression with the expanded factor set

After freezing the residual codebook, additional-category sensitivity and enrichment are measured using the ten unpatched source-aware configurations on 60 calibration and 300 test tasks, respectively. These measurements are separate from the patched-scorer audit on the 900-task holdout. Calibration pairs isolate one category at mild, moderate, or severe levels while preserving the requested edit and other assessed properties; independent checks exclude non-target changes. Pair records retain task, pair, category, severity, intact-image, and degraded-image identifiers. Three annotators independently label category presence, and agreement uses pre-adjudication labels. Selected and matched-control outputs are annotated in full, rather than only within the flagged subset. Repeated appearances of the same image reuse its image-level labels while retaining the associated task and reward configuration membership for analysis.

Table 24 reports the complete configuration-level matrix. Regression estimates use unrounded cell values; these unpatched measurements do not substitute the patched sensitivity estimates in Table 23.

The exploratory extension evaluates the original five factors, the five audited categories, and their union using the ten unpatched source-aware configurations. Sensitivity is measured on 60 calibration tasks and enrichment on 300 test tasks at $N = 64$. The fits retain configuration and factor effects. Original-only and audit-only analyses contain 50 configuration-factor cells; the union contains 100. Table 25 reports each searcher under all three factor sets.

For Best-of-N, adding the audited categories changes the slope from -4.141 to -4.109 and within-R^2 from 0.770 to 0.785. For iterative refinement, the slope changes from -4.120 to -4.119, with within-R^2 increasing from 0.721 to 0.746. The audited-category-only slopes are -3.799 and -4.395, with within-R^2 values 0.676 and 0.704. The negative sensitivity-enrichment relationship consequently extends to the additional defect categories measured in this audit.