DON'T OVERTHINK: WHEN REASONING BACKFIRES IN IMAGE EDITING AND HOW TO FIX IT

Anonymous authors
Paper under double-blind review

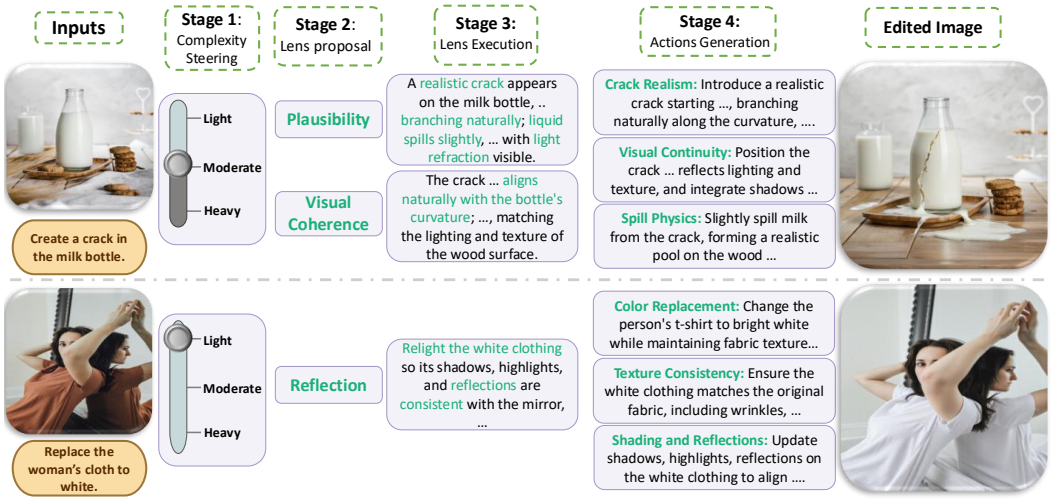


Figure 1: **Reasoning should be adaptive and structured, not uniform.** PRISM predicts how much reasoning each edit needs before committing to it. **(Top)** A crack in a milk bottle has physical consequences, so PRISM predicts *Moderate* complexity and runs two tailored lenses, Plausibility and Visual Coherence, which infer the spill and its reflections. **(Bottom)** A clothing recolor is direct, so a single *Light* Reflection lens keeps the mirror image consistent. Each reasoning chain is distilled into three unambiguous actions that drive a single generation pass.

ABSTRACT

Reasoning-augmented image editing operates under a prevailing assumption: more deliberation yields better edits. We show this assumption is fundamentally flawed. Through a diagnostic study of BAGEL, a state-of-the-art unified multimodal model, we identify three root pathologies of SFT-based reasoning: *complexity-agnostic outputs* that apply identical deliberation depth to trivial and complex edits alike; *adherence dependency*, where correct reasoning fails to translate into generation gains without sufficient adherence capacity in the generation expert; and *verbosity without substance*, where tripling the reasoning budget yields no quality improvement (GPT-Score ≈5.2), while structural directives at the same budget deliver a substantial gain to 6.1. These findings converge on a single conclusion: neither adaptive reasoning length nor structured templates alone resolves overthinking, both are *jointly necessary*. We introduce **PRISM** (**P**lanning with **R**easoning via **I**nstructed **S**tructured **M**odes), a compound adaptive-and-structured reasoning framework that decomposes pre-generation deliberation into three sequential stages, *complexity steering*, *tailored lens generation*, and *ordered lens execution*, distilled into unambiguous per-sample actions that drive a flow-matching generation expert. The understanding and generation experts are trained jointly via GRPO and Flow-GRPO with variance-guided early-window selection, without any external verifier or multi-turn inference. To power the framework, we introduce **REVISE-50K**, a dataset of 50,000 samples spanning 50 editing skills with graded complexity labels and disentangled hardness and verbosity axes. PRISM establishes a new state of the art on three reasoning-intensive benchmarks, **+13.0%** on KRIS, **+3.4%** on RISE, **+17.8%** on PICA, while preserving performance on direct editing tasks and surpassing multi-round external verifier systems at 20× lower inference cost.

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.

1 INTRODUCTION

Reasoning has become a first-class component of vision-language models. Chain-of-thought prompting (Wei et al., 2022) and large-scale reinforcement learning (Guo et al., 2025) have established that slow, deliberate thinking dramatically improves performance on complex tasks, and unified multimodal models (Deng et al., 2025; Xie et al., 2025; Wang et al., 2024) have carried this paradigm to image editing, where emitting reasoning traces before generation improves instruction following (Fu et al., 2024; Huang et al., 2024; Li et al., 2025a). The prevailing assumption is clear: the more the model deliberates, the better the edit.

We challenge this assumption with a striking finding: reasoning can actively hurt editing quality. Enabling BAGEL's chain-of-thought on direct, self-contained instructions, such as changing an object's color, *degrades* performance compared to the no-reasoning baseline (GEdit $5.35 \rightarrow 5.06$, Table 2), because explicit deliberation over an obvious edit wastes capacity and introduces noise; yet on reasoning-intensive instructions the same mechanism is indispensable (KRIS $56.21 \rightarrow 60.18$). Our diagnostic study (Section 3.2) identifies three root pathologies. *Complexity-agnostic reasoning*: the model applies identical deliberation depth to trivial and complex edits alike. *Adherence dependency*: reasoning only improves outcomes when the generation expert has sufficient capacity to faithfully execute the inferred plan; without this, even correct reasoning degrades output. *Verbosity without substance*: tripling the reasoning budget yields no quality gain (GPT-Score $\approx$5.2 across all length levels), while introducing structural directives at the same budget achieves a substantial jump to 6.1, what matters is not *how much* the model reasons, but *how it structures* its reasoning. Together, these observations reveal that neither adaptive reasoning length nor structured templates alone is the remedy; both are *jointly necessary*.

We introduce PRISM (Planning with Reasoning via Instructed Structured Modes), a compound adaptive-and-structured reasoning framework. PRISM decomposes reasoning into three sequential stages before committing to generation: *(i)* **complexity steering**, which predicts a per-sample budget (l lenses, n actions) from the input; *(ii)* **tailored lens generation**, which proposes sample-specific reasoning perspectives, guided by but not restricted to an initial vocabulary of nine failure-mode-specific lenses; and *(iii)* **ordered lens execution**, which builds sequential reasoning traces distilled into unambiguous tailored actions that drive a single-shot image generation. As confirmed by the lens ablation in Table 3, each subset benefits most when its matching lens is introduced, while additional lenses yield diminishing returns, motivating the adaptive routing illustrated in Figure 1. The understanding expert is trained via GRPO (Shao et al., 2024), after an SFT warm-start of complexity steering, and the generation expert via Flow-GRPO (Liu et al., 2025a) with our proposed *variance-guided early-window selection*, which concentrates gradient signal in the high-variance early denoising steps that most govern structural fidelity. To power this framework, we introduce **REVISE-50K** (**R**easoning **V**isual **I**nstruction **S**kills & **E**dits), 50,000 samples spanning 50 editing skills with graded complexity labels and deliberately disentangled hardness and verbosity axes.

PRISM sets a new state of the art on reasoning-intensive editing benchmarks. It improves over its BAGEL (Deng et al., 2025) backbone by **13.0%** on KRIS, **3.4%** on RISE, and **17.8%** on PICA, while *preserving* performance on direct editing benchmarks, precisely avoiding the overthinking failure it was designed to cure. PRISM surpasses EditThinker (Li et al., 2025a), which deploys an external 8B verifier across up to 8 iterative rounds, in a single forward pass at $20\times$ lower inference cost, and outperforms larger-scale systems such as Qwen-Image-Edit (Wu et al., 2025a), demonstrating that structured adaptive reasoning is a more effective lever than brute-force scale.

Our analysis shows that structure, not reinforcement learning alone, drives the reasoning gains. A free-form chain-of-thought trained with the identical GRPO setup, budget, and data barely improves KRIS over no reasoning (62.8 vs. 60.1), whereas structured reasoning lifts it to 73.4; adaptivity then removes the remaining overthinking penalty on direct edits (GEdit $6.37 \rightarrow 7.00$) without losing the KRIS gain. Adding components one at a time shows that Flow-GRPO, the reasoning framework, and REVISE-50K each contribute a separable share (KRIS $56.2 \rightarrow 60.1 \rightarrow 66.8 \rightarrow 73.2$). A human study with three annotators on 1,000 samples supports InternVL3.5-241B as the complexity teacher: prompted for a score, it agrees with the human consensus better than GPT-5 (QWK 0.608 vs. 0.521). Finally, the generated actions are rarely redundant (7%) or contradictory (4%), and contradictions concentrate on genuinely hard physical edits.

Our contributions are:

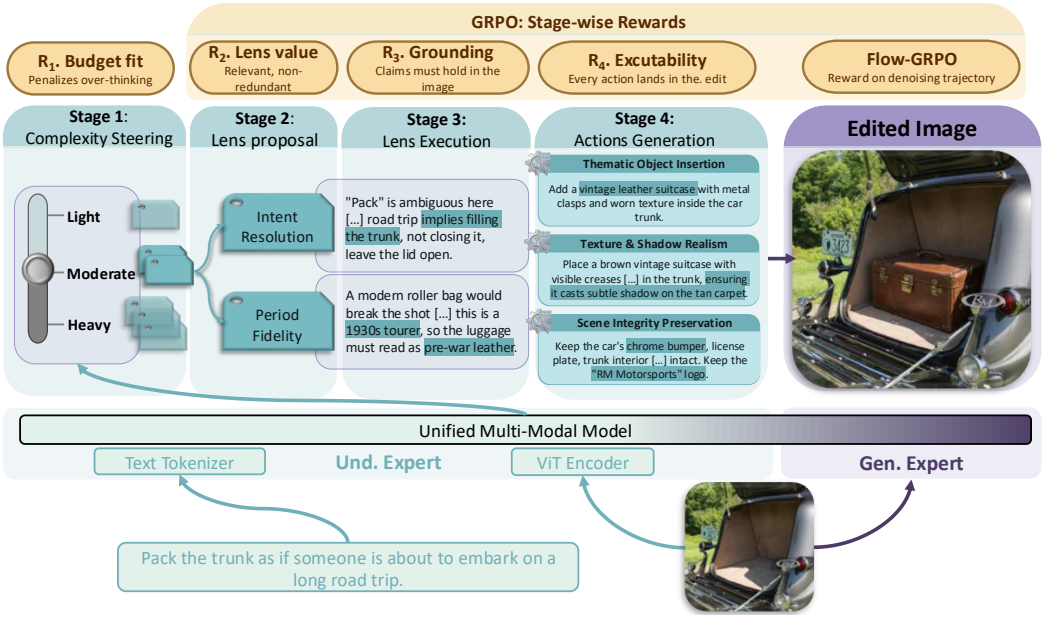


Figure 2: **Overview of PRISM.** The understanding expert steers complexity, proposes and executes tailored lenses, and distills actions for one-shot editing, trained with GRPO and Flow-GRPO.

- A **diagnostic study** of reasoning in instruction-based image editing, showing that overthinking is an active failure mode and that adaptivity and structure are jointly necessary to cure it.
- **PRISM**, an adaptive and structured reasoning framework that trains reasoning and generation within a single unified model via GRPO and Flow-GRPO, without external verifiers or multi-turn inference.
- **REVISE-50K**, a 50K-sample dataset spanning 50 editing skills with graded complexity labels, purpose-built for reasoning-adaptive RL training.
- **Analyses** that isolate structure from GRPO optimization, attribute the gains to each component, and validate complexity labels against human annotation.

2 Related Work

Slow and structured thinking. Chain-of-thought prompting (Wei et al., 2022) established explicit reasoning as a first-class training objective, and DeepSeek-R1 (Guo et al., 2025) scaled it through RL into deeply deliberative models. Structured variants organize these steps across trees (Yao et al., 2023) or multi-stage protocols (Xu et al., 2025), all sharing the premise that deeper reasoning monotonically improves performance. **Our paper challenges this premise in image editing**, showing that reasoning can actively hurt performance when misaligned with task complexity.

Adaptive reasoning. Prior work has shown that uniform deep reasoning can be inefficient. Methods such as Chen et al. (2024), DAST (Shen et al., 2025), and AdaptThink (Zhang et al., 2025) reduce overthinking in language models through preference optimization and adaptive thinking modes, while FAST (Xiao & Gan, 2025) and TON (Wang et al., 2025b) extend these ideas to vision language models using difficulty aware GRPO and selective reasoning. However, these approaches focus on perception and question answering tasks and do not directly apply to generative image editing. Generative editing differs in three key ways. First, the output space is continuous, with many valid edits and no single ground truth. Second, rewards rely on imperfect vision language model judges, making reward design critical. Third, adaptive reasoning alone is insufficient, as it tends to increase verbosity without improving quality. Our analysis (Section 3.2) shows that increasing reasoning length does not improve performance, whereas adding structural guidance yields significant gains.

We are the first to identify overthinking as a key failure mode in generative editing and to demonstrate the need for a combined adaptive and structured approach.

RL for visual generation and editing. DDPO (Black et al., 2024), DPOK (Fan et al., 2023), and Diffusion-DPO (Wallace et al., 2024) apply policy gradient and preference optimization to diffusion models. Flow-GRPO (Liu et al., 2025a) extends GRPO to flow-matching by converting the deterministic ODE to a stochastic SDE, later adopted for video generation (Xue et al., 2025) and image editing (Ahmadi et al., 2025). These works optimize *what to generate* but model reasoning and generation as disjoint; ours is the first to co-train both jointly within a single unified backbone.

Reasoning-augmented generation and editing. GoT (Fang et al., 2025), T2I-R1 (Jiang et al., 2025), and ShortCoTI (Gu et al., 2025) inject or refine CoT reasoning into image generation, while MGIE (Fu et al., 2024) and SmartEdit (Huang et al., 2024) use multimodal reasoning to improve instruction following in editing. Unified models, such as BAGEL (Deng et al., 2025; Chen et al., 2025; Wu et al., 2026a), have shown that interleaved thinking consistently improves generation and editing quality. None of these works study *when* reasoning harms image editing or examine the adaptive-versus-structured tradeoff that our analysis reveals.

3 Methodology

3.1 Preliminaries

We build on BAGEL (Deng et al., 2025), a decoder-only unified model (7B active / 14B total parameters) with a Mixture-of-Transformer-Experts (MoT) architecture comprising separate *understanding* and *generation* experts sharing self-attention at every layer over interleaved text and visual tokens. This decoupling motivates our dual training strategy: we apply **GRPO** (Shao et al., 2024) to the understanding expert to optimize instruction-following reasoning, and **Flow-GRPO** (Liu et al., 2025a) to the generation expert to optimize edit fidelity. GRPO is a critic-free RL algorithm that estimates normalized advantages from a group of G sampled responses, eliminating the value network of PPO, and optimizes a clipped surrogate objective under KL regularization (Guo et al., 2025; Yu et al., 2025). Flow-GRPO extends this to flow-matching models by converting their deterministic ODE into an equivalent SDE, introducing the sampling diversity required for group-relative advantage estimation. We further contribute *variance-guided early-window selection*, concentrating gradient signal in the high-variance early denoising steps where structural decisions most strongly determine edit fidelity, yielding faster convergence than the sliding-window schedule of MixGRPO (Li et al., 2025b). Full formulations and derivations are provided in Appendix C.1.

Table 1: **Verbosity optimization $\neq$ reasoning optimization.** We evaluate three different reasoning budgets using GPT-5.2 on KRIS. Scaling length alone yields no quality gain, while adding a lightweight structural directive improves quality, demonstrating that *structure*, not *length*, drives reasoning quality.

Variant	Avg. Words	GPT-Score
Light	21	5.24
Moderate	33	5.12
Heavy	59	5.26
Heavy + Structure	66	**6.10**

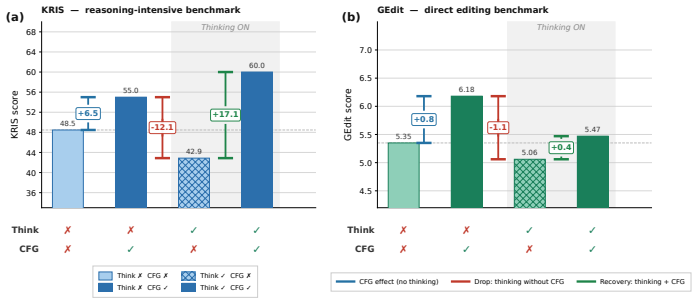


Figure 3: Thinking $\times$ CFG interaction on GEdit and KRIS. **Reasoning quality is gated by generation adherence.** We evaluate four combinations of thinking and CFG across a reasoning-intensive (KRIS) and a direct editing (GEdit) benchmark. Thinking alone degrades KRIS by 12.1 points, while enabling both thinking and CFG yields the best performance (+17.1), confirming that CFG acts as a proxy for adherence.

3.2 Reasoning Pathologies in BAGEL: Verbosity, Adherence, and Overthinking

We diagnose three distinct pathologies in BAGEL's SFT-based reasoning that collectively motivate our proposed solution.

Observation 1: Reasoning Backfires on Direct Benchmarks. Enabling BAGEL's chain-of-thought improves KRIS (Wu et al., 2025c) from 55.0 to 60.0, yet *hurts* GEdit (Liu et al., 2025b), dropping scores from 6.18 to 5.06. The culprit is not reasoning itself but *complexity-agnostic reasoning*: BAGEL applies uniform deliberation depth regardless of task complexity, generating the same elaborate chain-of-thought for a trivial color change as for a complex multi-object manipulation.

Observation 2: Reasoning Quality is Gated by Generation Adherence. As shown in Figure 3, enabling thinking *without* CFG degrades KRIS from 55.0 to 42.87, despite semantically correct reasoning; only with CFG does it recover to 60.0. This reveals a fundamental dependency: *reasoning quality is gated by generation adherence*, a model that cannot faithfully execute its own plan cannot benefit from planning. This directly motivates our use of Flow-GRPO on the generation expert to strengthen adherence as a prerequisite for effective reasoning.

Observation 3: Verbosity Optimization $\neq$ Reasoning Optimization. We train three LoRA variants under fixed budgets, *Light* (10–15 words), *Moderate* (20–30), *Heavy* (40–60), and evaluate reasoning quality via GPT-5.2 on KRIS. As shown in Table 1, all three yield virtually identical quality (GPT-Score ≈ 5.2) despite a $3\times$ length difference, confirming that more budget is a *verbosity optimization*, not a *reasoning optimization*. A fourth variant adds a structural directive at the Heavy budget, instructing the model to resolve ambiguities before reasoning; despite comparable length (avg. 66 words), quality jumps to 6.1, what matters is not *how much* to reason, but *how to structure* it. Prompts are in Appendix C.3.

Hypothesis and Motivation. These three observations converge on a single diagnosis: resolving overthinking requires a *compound* solution. Adaptive length without structure degenerates into verbosity (Obs. 3); rigid structure without adaptivity fails to discriminate task complexity (Obs. 1). These directions are *orthogonal but jointly necessary*, motivating the adaptive-and-structured design of PRISM (Section 3.3), trained jointly via GRPO on the textual expert and Flow-GRPO on the generation expert.

3.3 PRISM: Planning with Reasoning via Instructed Structured Modes

We introduce **PRISM** (**P**lanning with **R**easoning via **I**nstructed **S**tructured **M**odes), a compound reasoning framework built on BAGEL (Deng et al., 2025) whose core design principle is *structure-aware yet adaptive generation*: scaffolding that guides reasoning without fixing it to a rigid template, so the model adapts both *what* to reason about and *how* to act per sample. Given an input image I and instruction x, PRISM calls the unified model $\mathcal{M}$ sequentially through four stages (Figure 2): *(1)* complexity steering, *(2)* tailored lens generation, *(3)* ordered lens execution, and *(4)* tailored action generation, whose actions drive the flow-matching generation expert. The full pipeline is given in Algorithm 1 (Appendix C.2).

Stage 1: Adaptive Complexity Steering. Before reasoning, the model predicts a discrete complexity level $\hat{c} = \mathcal{M}(x, I; \rho_{\text{cpx}}) \in \{\texttt{light}, \texttt{moderate}, \texttt{heavy}\}$ directly from the input. A deterministic budget function β maps $\hat{c}$ to the number of lenses $l \in \{1, 2, 3\}$, with $n{=}3$ actions at every level, routing simple instructions to single-lens paths and complex ones to deeper multi-lens chains.

Stages 2–3: Tailored Lens Generation and Ordered Execution. A *lens* is a reasoning perspective the edit must be examined from. We seed the prompt with an initial vocabulary $\mathcal{L}$ of nine lenses derived from BAGEL's failure modes (Appendix C.4.1), but $\mathcal{L}$ only guides the model: it generates l *tailored* lenses for each input and is free to name its own,

$$\hat{L} = \big(\hat{l}_1, \ldots, \hat{l}_l\big) = \mathcal{M}(x,\ I,\ \mathcal{L};\ \rho_{\text{lens}}), \tag{1}$$

where each $\hat{l}_k$ is a free-form lens name and purpose, such as *Visual Coherence* or *Reflection* in Figure 1. The model makes wide use of this freedom: across our evaluation benchmarks, 86% of the generated lenses are not in $\mathcal{L}$. The lenses are then executed in order, each conditioned on all preceding traces so that later lenses refine rather than repeat earlier analysis:

$$\tau_k = \mathcal{M}\Big(x,\ I,\ \hat{L}_{1:k},\ \tau_{<k};\ \rho_{\hat{l}_k}\Big), \quad k = 1, \ldots, l, \quad \tau_{<1} \coloneqq \varnothing. \tag{2}$$

Stage 4: Tailored Action Generation. The reasoning traces $\{\tau_k\}_{k=1}^{l}$ are distilled into n *tailored* executable actions in a *single* forward pass, eliminating costly multi-turn inference:

$$\mathbf{A} = \big((t_1,\, d_1),\, \ldots,\, (t_n,\, d_n)\big) = \mathcal{M}\big(x,\; I,\; \{\tau_k\}_{k=1}^{l};\; \rho_{\text{act}}\big)\,, \tag{3}$$

where each action pairs a *tailored aspect title* t_j (e.g. `Subject Placement`) with an *unambiguous directive* d_j; unlike fixed roles (e.g. *high-level edit / identity preservation*), the model names its own aspect per input, which proves the most learnable action format (Appendix D.7). The generation expert executes all n actions simultaneously to produce the edited image $\hat{I}$ in one shot.

Training and reward signals. GRPO amplifies capabilities the base model already shows, but BAGEL predicts edit complexity poorly, so we first warm-start Stage 1 with SFT on offline complexity labels $\tilde{c}$ produced by InternVL3.5-241B-A28B (Wang et al., 2025c) (Appendix D.4). All four stages are then trained with GRPO, each with its own reward (Figure 2; formulations in Appendix C.5). R_1 scores $\hat{c}$ against $\tilde{c}$ (Eq. (8)). R_2–R_4 are scored by InternVL3.5-241B-A28B with stage-specific rubric prompts: R_2 rewards lenses that are relevant and non-redundant, R_3 rewards lens traces whose claims hold in the image, and R_4 rewards actions that are concrete and directly executable. The generation expert is trained with Flow-GRPO using EditReward (Wu et al., 2026c), an open-weight reward model trained on human edit preferences. Since scoring a vague instruction against an edited image yields an unreliable signal, we score the edit against each explicit action and average:

$$r_{\text{img}} = \frac{1}{n}\sum_{j=1}^{n} \mathcal{R}_{\text{img}}\Big(I,\; \hat{I},\; a_j\Big)\,. \tag{4}$$

All reward models are open-weight, so training relies on no proprietary system.

4 REVISE-50K: A Reasoning-Rich Dataset for Adaptive Image Editing

Existing image editing datasets (Brooks et al., 2023; Zhang et al., 2023; Wei et al., 2025; Ge et al., 2024) cover narrow task families, provide no signal about appropriate reasoning depth, and treat all instructions as equally demanding, the exact conditions that produce the overthinking failure mode we seek to cure. We therefore introduce **REVISE-50K** (**R**easoning **V**isual **I**nstruction **S**kills & **E**dits), a dataset of 50,000 multimodal samples purpose-built for GRPO-based training of reasoning-adaptive editing models (full statistics in Table 7). REVISE-50K is distinguished by three properties absent in prior work: (i) **breadth**: 50 fine-grained editing skills across 10 high-level categories (Figure 4), sourced from 74 heterogeneous datasets, with both levels roughly balanced to prevent distributional shortcuts; (ii) **disentangled instruction design**: hardness and verbosity are independently controlled, preventing the model from using surface cues as proxies for difficulty; and (iii) **graded reasoning annotations**: every sample carries a complexity label and reasoning trace that directly drive our adaptive reward (Section 3). Each skill is paired with semantically appropriate sources drawn from 74 datasets spanning natural photographs (e.g., COCO, FFHQ), domain-specific collections (e.g., WikiArt, DeepFashion), and procedurally generated stimuli (dice nets, matchstick puzzles, mazes). All instructions and reasoning annotations are generated by InternVL3.5-241B-A28B Wang et al. (2025c); even for source datasets that ship with existing instructions (e.g., OmniEdit), we discard and regenerate to enforce consistent style and difficulty distribution. Figure 4 provides overview of dataset creation and please refer to Appendix B for all the details and statistics.

5 Experiments

5.1 Experimental Setup

Implementation details. We build PRISM on BAGEL (Deng et al., 2025), the strongest open-source unified understanding and generation model, and fine-tune it on REVISE-50K using GRPO (Shao et al., 2024) with LoRA (Hu et al., 2022) (rank 128). Training runs for one epoch with learning rate $1{\times}10^{-5}$, batch size 1 per GPU, group size $G{=}8$, across 4 nodes of $8{\times}$A100 GPUs. The understanding and generation experts are trained jointly with stream-specific LoRA activation. To support a large group size, we train at $512{\times}512$ with 25 denoising steps, trading raw quality for

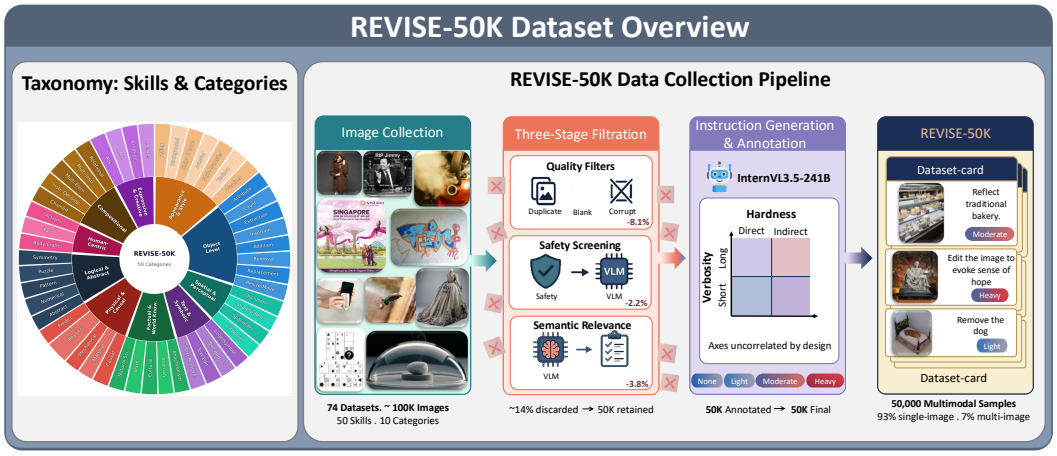


Figure 4: **Left:** Two-level taxonomy of REVISE-50K: 10 high-level categories (inner ring) and 50 fine-grained skills (outer ring), balanced across both levels. **Right:** Three-stage image filtration pipeline applied to ∼100K collected images before instruction generation; ∼14% of images are discarded in total.

Table 2: **Comparison with state-of-the-art image editing methods.** We report scores on two non-reasoning benchmarks (GEdit, ImgEdit) and three reasoning-intensive benchmarks (KRIS, RISE, PICABench-Superficial). Green gains are relative to Bagel-Think. * denotes our new robust evaluation protocol (Appendix D.5).

Method	Non-Reasoning Benchmarks		Reasoning-based Benchmarks		
	GEdit*	ImgEdit	KRIS	RISE	PICABench-Superficial
OmniGen2 (Wu et al., 2026a)	5.58	3.41	50.52	3.1	46.79
FLUX.1 Kontext (Black Forest Labs et al., 2025)	5.64	3.44	61.81	5.8	48.93
Qwen-Image-Edit (Wu et al., 2025a)	7.23	4.36	64.43	8.9	61.26
ReasonEdit (Yin et al., 2026)	6.86	4.36	61.57	3.1	48.23
ChronoEdit (Wu et al., 2026b)	7.12	4.42	70.96	1.5	58.67
Bagel (Deng et al., 2025)	5.35	3.2	56.21	6.1	45.07
Bagel-Think (Deng et al., 2025)	5.06	3.0	60.18	11.9	46.48
PRISM (ours)	**7.00** (+1.9)	**4.00** (+1.0)	**73.2** (+13.0)	**15.27** (+3.4)	**64.32** (+17.8)

efficiency; structured reasoning recovers this loss and surpasses the full-resolution baseline. We use standard GRPO for reasoning and Flow-GRPO (Liu et al., 2025a) with variance-guided early-window selection for generation. Unlike sliding-window methods such as MixGRPO (Li et al., 2025b) (Appendix D.3), our approach focuses gradients on informative steps, improving convergence and final reward (Figures 14, 15).

5.2 Comparison with State of the Art

Benchmarks.

We evaluate on two benchmark groups: *Reasoning-intensive benchmarks* (KRIS (Wu et al., 2025c), RISE (Zhao et al., 2025), PICA (Pu et al., 2026)), which require multi-step or physical reasoning, and *Standard editing benchmarks* (GEdit (Liu et al., 2025b), ImgEdit (Ye et al., 2025)) with direct, unambiguous instructions. This design tests both reasoning gains and the ability to avoid overthinking on simple tasks.

Results. As shown in Table 2, the fairest comparison is against our own backbone, BAGEL (Deng et al., 2025), which isolates the pure contribution of the PRISM framework. PRISM improves over BAGEL by **13.0%** on KRIS, **3.4%** on RISE, and **17.8%** on PICA, establishing a new state of the art on all three reasoning-intensive benchmarks. Crucially, PRISM also outperforms stronger

Table 3: **Lens Specificity.** Each row adds one lens: Dis Disambiguation, Gnd Grounding, Pla Plausibility, Act Action.

	Disambiguation (KRIS)				
	Rule-based Reasoning	Part Completion	Overall	Grounding	Plausibility
Act (zero-shot)	5.80	6.10	6.40	5.70	5.31
Act	7.25	7.50	7.99	6.40	7.10
Dis Act	7.80	8.60	8.43	6.50	6.50
Dis Gnd Act	7.67	8.60	8.38	**6.90**	7.00
Dis Gnd Pla Act	7.73	**9.02**	8.61	6.50	7.50
Adaptive + Structure Act	**8.12**	8.92	**8.72**	6.73	**7.82**

Table 4: **Source of the gains.** Components added one at a time; rows 2–3 train on OmniEdit.

	GEdit	KRIS
BAGEL	5.35	56.2
+ Flow-GRPO	6.94	60.1
+ Adaptive & structured	**7.08**	66.8
OmniEdit→REVISE-50K	7.00	**73.2**

Table 5: **Structure, not GRPO, drives the gain.** All reasoning variants use the same GRPO setup, budget, and data.

Reasoning	Adapt.	Struct.	GEdit	KRIS
None (Flow-GRPO only)	–	–	6.94	60.1
Free-form CoT	✗	✗	6.25	62.8
Structured, always Heavy	✗	✓	6.37	**73.4**
PRISM	✓	✓	**7.00**	73.2

baselines that use larger models, higher resolutions, and larger-scale training data, such as Qwen-Image-Edit (Wu et al., 2025a), demonstrating that structured adaptive reasoning is a more effective lever than brute-force scale.

On the standard editing benchmarks, BAGEL's own thinking variant (BAGEL-Think) *degrades* performance relative to the base model, an instance of the overthinking pathology we characterize in Section 3.2. PRISM avoids this failure: by routing simple instructions to low-depth reasoning paths (Section 5.3), it preserves performance on GEdit and ImgEdit while achieving large gains on reasoning-intensive tasks.

5.3 Analysis

We now dissect PRISM: which lenses matter, where its gains come from, and whether they stem from structure or from GRPO itself. The action-format ablation is in Appendix D.7.

Lens specificity. Table 3 reports scores on three aspect-stratified evaluation subsets (Appendix D.2) as reasoning lenses are added sequentially. Two consistent patterns emerge. First, performance improves as additional lenses are introduced, confirming that structured reasoning is beneficial. Second, and more importantly, the *largest* gain on each subset occurs precisely when the lens targeting that subset's focal aspect is added, e.g., the Disambiguation lens on the disambiguation subset. Once the targeted lens is included, further lenses yield diminishing returns and can even slightly degrade performance, because irrelevant reasoning introduces noise into the action-generation stage. These two observations jointly motivate the adaptive design: since we cannot apply all lenses to every sample, and the beneficial lenses vary by instruction, the model must determine per sample which lenses to activate and at what depth, as described in Section 3.3. Indeed, letting the model select its own lenses per sample (last row) achieves the best overall score (8.72) and the best results on rule-based reasoning and plausibility, while remaining close to the best fixed configuration on the other two columns.

Source of the gains. Table 4 adds each component to BAGEL in turn, under identical training settings. Flow-GRPO on the generation expert alone gives the largest jump on direct editing (GEdit $5.35 \rightarrow 6.94$) but only a moderate one on KRIS ($+3.9$), addressing the adherence bottleneck of Observation 2. Adding our adaptive and structured reasoning, still on OmniEdit, lifts KRIS by a

further 6.7 points while keeping GEdit stable. Replacing OmniEdit with REVISE-50K adds another 6.4 points on KRIS, with GEdit unchanged within judge noise, so the dataset is a distinct driver rather than a proxy for the method (Appendix D.6).

Structure, not GRPO, drives the reasoning gain. To separate the structured pipeline from GRPO itself, Table 5 trains a free-form chain-of-thought baseline with the same GRPO setup, budget, and data. It reaches only 62.8 on KRIS, barely above no reasoning (60.1), and lowers GEdit to 6.25. Adding structure lifts KRIS to 73.4 under identical training, so the reasoning gain comes from the structured pipeline, not from GRPO alone. Applied uniformly, however, structure still overthinks direct edits (GEdit 6.37); adding adaptivity recovers GEdit to 7.00 while keeping KRIS at 73.2, the only configuration that is strong on both.

Complexity steering (Stage 1). Complexity is ordinal, so we measure agreement with quadratic-weighted Cohen's κ (QWK) and validate the labels against humans: three annotators each rated the same 1,000 samples (200 per benchmark) on a 1–10 scale, with hidden repeats to measure test-retest consistency. Humans agree with themselves at QWK 0.777 but with each other at only 0.458, so the task has a low ceiling, and we use the median of three as the consensus label. Against this consensus, InternVL3.5-241B prompted for a score agrees better than GPT-5 (0.608 vs. 0.521, paired bootstrap $p<0.001$), reaching 78% of the human test-retest ceiling; GPT-5 is highly self-consistent (0.828) but consistently differs from humans, which supports InternVL3.5-241B as the complexity teacher. For the policy, GRPO alone learns complexity poorly from BAGEL: against the teacher's labels on all five benchmarks (n=3,870), an SFT-trained Stage 1 reaches QWK 0.605 versus 0.569 for GRPO-only training, and leads on four of five benchmarks against GPT-5, which motivates the SFT warm-start. Subsequent GRPO training of the later stages does not erode Stage 1: at the deployed checkpoint its agreement is statistically indistinguishable from the SFT model on both references.

Action quality. Using GPT-5.2 to flag problematic action sets across all five benchmarks, only 7% of samples contain a redundant action and 4% a contradictory one. Redundancy arises mostly on direct benchmarks such as GEdit, when a `light` sample receives three actions and a separate "keep unchanged" action restates what the main action already preserves; the generation expert simply executes the overlap once. Contradictions concentrate on PICA, where the model misses a physical consequence, e.g. removing the water from a cup while also keeping a floating ball in place; these cases coincide with low PICA scores, marking genuinely hard physical reasoning rather than a systematic flaw.

6 Conclusion

We showed that explicit reasoning is not uniformly beneficial for instruction-based image editing. On BAGEL, chain-of-thought improves reasoning-intensive edits but degrades direct ones, and our diagnostic study traces this overthinking to three pathologies: complexity-agnostic reasoning, a dependence on generation adherence, and verbosity without substance. Together they imply that adaptive reasoning depth and structured reasoning are jointly necessary. We introduced **P**lanning with **R**easoning via **I**nstructed **S**tructured **M**odes, which predicts each edit's complexity, generates and executes tailored reasoning lenses, and distills them into unambiguous actions for a single generation pass, trained within one unified model with GRPO and Flow-GRPO; and REVISE-50K, a 50K-sample dataset with graded complexity labels across 50 editing skills. PRISM sets a new state of the art on KRIS, RISE, and PICA while improving direct editing, and our analysis attributes the reasoning gain to structure rather than to reinforcement learning alone, with adaptivity removing the overthinking penalty on simple edits.

Limitations and Future Work. PRISM inherits the limits of its components: its quality is bounded by the base model and by the automatic reward models used in training, and several of our evaluations rely on VLM judges. Edit complexity is itself hard to label: human annotators agree with one another only moderately (QWK 0.458), which caps how precisely any router can be trained and validated. Finally, our reasoning is purely textual; extending PRISM to stronger unified backbones and to visual reasoning traces, where the model sketches intermediate images before editing, are natural next steps.

AI USE STATEMENT

We used generative AI tools as assistants, under author direction and on material whose substance the authors had already fixed. For the manuscript, their use was confined to drafting and polishing: redrafting passages from author-supplied content and findings, correcting grammar and typography, condensing sections to the page limit, and checking bibliography entries against their original sources. For the experimental code, we used an AI coding assistant to help implement and debug parts of the training and evaluation pipeline (reward computation, batched multi-stage inference, and evaluation and plotting scripts); every such change was specified, reviewed, and tested by the authors.

We did not use generative AI to generate research ideas or to design experiments, and all results come from the authors' own runs. Separately, AI models are part of our data pipeline and method, and each use is described where it occurs: we use InternVL3.5-241B-A28B to collect our data, generating the editing instructions, reasoning annotations, and complexity labels of REVISE-50K (Appendix B), and as the judge that scores the reasoning stages during GRPO training (Section 3.3); GPT-5.2 serves as an automatic evaluation judge (Appendices C.3 and D.5). No AI system is an author. We have reviewed all AI-assisted text and code and take full responsibility for the final content of this work, including every claim, number, and artifact.

ETHICS STATEMENT

This work studies when explicit reasoning helps or harms instruction-based image editing. REVISE-50K is assembled from 74 publicly available image datasets (Appendix B.2), and the aspect-stratified evaluation subsets use images from Pexels (Appendix D.2); each source is used under its own licence. All instructions and reasoning annotations are generated automatically. Before any instruction is generated, every image passes a dual-consensus NSFW screen that removed 2.2% of the collected images (Appendix B.4). Our human evaluations involve annotators in three studies: a blinded pairwise preference study comparing PRISM against BAGEL-Think and Qwen-Image-Edit on 200 GEdit and 200 KRIS samples; complexity annotation of 1,000 benchmark samples, each rated independently by three annotators (Section 5.3); and a blinded comparison of GEdit-R with the original GEdit protocol on 300 GEdit samples. In every study, annotators judge only public benchmark images and instructions, together with model outputs in the preference studies, where outputs are shown without model identities; complexity annotators never see any model's prediction. All annotators gave informed consent and were paid for their work by the authors' institution.

Three risks are worth stating plainly. First, a stronger instruction-following editor lowers the cost of producing misleading or manipulated imagery, and some source collections contain human faces (e.g., FFHQ); PRISM is no exception, and we would not deploy it without provenance or watermarking safeguards. Second, the instructions, reasoning traces, complexity labels, and lens rankings in REVISE-50K are produced by a single large VLM, so the dataset, and any model trained on it, inherits that model's cultural, factual, and visual biases; this matters most for the knowledge-heavy skills where the "correct" edit depends on world knowledge. Third, our training rewards and several of our evaluations rely on automatic VLM judges; a gain under such a judge is evidence of agreement with that judge, not a guarantee that the edit is correct or appropriate.

We have tried to report negative outcomes as carefully as positive ones. The paper is built on one: enabling reasoning *degrades* BAGEL on direct editing benchmarks (Section 3.2, Table 2). We also report that tripling the reasoning budget yields no quality gain (Table 1), that free-form actions barely improve under GRPO training (Table 10), and that adding lenses beyond the one matched to a subset gives diminishing and sometimes negative returns (Table 3).

REPRODUCIBILITY STATEMENT

Section 3.3 specifies PRISM in full: complexity steering and the budget function β, Eq. (1) tailored lens generation, Eq. (2) ordered lens execution, Eq. (3) action generation, and Eq. (4) the image reward. Algorithm 1 (Appendix C.2) gives the complete inference and training loop, including the SFT warm-start of Stage 1. The stage rewards R_1–R_4 are specified in Appendix C.5, and the GRPO and Flow-GRPO objectives in Appendix C.1. Every prompt is reproduced verbatim: the

verbosity-ablation training and judge prompts (Appendix C.3), the lens prompts (Appendix C.4.2), and the action-generation and action-assessment prompts (Appendix C.6).

Section 5.1 states the training configuration: BAGEL as the backbone, LoRA at rank 128, learning rate 1×10^{-5}, batch size 1 per GPU, group size $G{=}8$, one epoch on 4 nodes of $8\times$A100 GPUs, and training at 512×512 with 25 denoising steps. Flow-GRPO injects SDE noise only in denoising steps 24–17; the variance analysis that selects this window is in Appendix D.3. The reasoning reward model is InternVL3.5-241B-A28B and the image reward model is EditReward, both open-weight; the complexity labels used for the Stage-1 warm-start and reward are produced in one offline pass with InternVL3.5-241B-A28B (Appendix D.4). The construction of REVISE-50K, including sources, instruction generation, every filtration threshold, and the complexity annotation protocol, is described in Appendix B.

For evaluation, we follow the released protocol of each benchmark, except for GEdit, where we use the robust GEdit-R protocol of Appendix D.5. The aspect-stratified subsets and the search queries used to build them are described in Appendix D.2 and Table 8. One caveat matters for exact replication: GEdit-R and the reasoning-quality scores use GPT-5.2, a closed model served through an API, so re-running these judges later may not reproduce our scores exactly even with identical outputs.

Our code, the REVISE-50K dataset, the trained checkpoints, and the training configuration for every run reported here will be made publicly available upon acceptance.

REFERENCES

Saba Ahmadi, Rabiul Awal, Ankur Sikarwar, Amirhossein Kazemnejad, Ge Ya Luo, Juan A. Rodriguez, Sai Rajeswar, Siva Reddy, Christopher Pal, Benno Krojer, and Aishwarya Agrawal. The promise of RL for autoregressive image editing. In *Advances in Neural Information Processing Systems*, volume 38, pp. 130232–130267, 2025.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, 2024.

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

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. InstructPix2Pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18392–18402, 2023.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, et al. Do not think that much for 2+ 3=? on the overthinking of o1-like llms. *arXiv preprint arXiv:2412.21187*, 2024.

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

Falconsai. NSFW image detection. `https://huggingface.co/Falconsai/nsfw_image_detection`, 2023.

Ying Fan, Olivia Watkins, Yuqing Du, Hao Liu, Moonkyung Ryu, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, Kangwook Lee, and Kimin Lee. DPOK: Reinforcement learning for fine-tuning text-to-image diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 79858–79885, 2023.

Rongyao Fang, Chengqi Duan, Kun Wang, Linjiang Huang, Hao Li, Shilin Yan, Hao Tian, Xingyu Zeng, Rui Zhao, Jifeng Dai, et al. Got: Unleashing reasoning capability of multimodal large language model for visual generation and editing. *arXiv preprint arXiv:2503.10639*, 2025.

Tsu-Jui Fu, Wenze Hu, Xianzhi Du, William Yang Wang, Yinfei Yang, and Zhe Gan. Guiding instruction-based image editing via multimodal large language models. In *International Conference on Learning Representations*, 2024.

Yuying Ge, Sijie Zhao, Jinguo Zhu, Yixiao Ge, Kun Yi, Lin Song, Chen Li, Xiaohan Ding, and Ying Shan. SEED-X: Multimodal models with unified multi-granularity comprehension and generation. *arXiv preprint arXiv:2404.14396*, 2024.

Zeqi Gu, Markos Georgopoulos, Xiaoliang Dai, Marjan Ghazvininejad, Chu Wang, Felix Juefei-Xu, Kunpeng Li, Yujun Shi, Zecheng He, Zijian He, Jiawei Zhou, Abe Davis, and Jialiang Wang. Improving chain-of-thought efficiency for autoregressive image generation. *arXiv preprint arXiv:2510.05593*, 2025.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025. doi: 10.1038/s41586-025-09422-z.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Jingcheng Hu, Yinmin Zhang, Qi Han, Daxin Jiang, Xiangyu Zhang, and Heung-Yeung Shum. Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base model. In *Advances in Neural Information Processing Systems*, volume 38, pp. 180122–180145, 2025.

Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, and Ying Shan. SmartEdit: Exploring complex instruction-based image editing with multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8362–8371, 2024.

Dongzhi Jiang, Ziyu Guo, Renrui Zhang, Zhuofan Zong, Hao Li, Le Zhuo, Shilin Yan, Pheng-Ann Heng, and Hongsheng Li. T2I-R1: Reinforcing image generation with collaborative semantic-level

and token-level CoT. In *Advances in Neural Information Processing Systems*, volume 38, pp. 44581–44615, 2025.

Dohwan Ko, Jinyoung Park, Seoung Choi, Sanghyeok Lee, Seohyun Lee, and Hyunwoo J. Kim. MoE-GRPO: Optimizing mixture-of-experts via reinforcement learning in vision-language models. *arXiv preprint arXiv:2603.24984*, 2026.

Hongyu Li, Manyuan Zhang, Dian Zheng, Ziyu Guo, Yimeng Jia, Kaituo Feng, Hao Yu, Yexin Liu, Yan Feng, Peng Pei, Xunliang Cai, Linjiang Huang, Hongsheng Li, and Si Liu. EditThinker: Unlocking iterative reasoning for any image editor. *arXiv preprint arXiv:2512.05965*, 2025a.

Junzhe Li, Yutao Cui, Tao Huang, Weijie Kong, Yiming Cheng, Chuxuan Zeng, Yinping Ma, Chun Fan, Miles Yang, Zhao Zhong, and Liefeng Bo. MixGRPO: Unlocking flow-based GRPO efficiency with mixed ODE-SDE. *arXiv preprint arXiv:2507.21802*, 2025b.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training flow matching models via online RL. In *Advances in Neural Information Processing Systems*, volume 38, pp. 45650–45685, 2025a.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, et al. Step1X-Edit: A practical framework for general image editing. *arXiv preprint arXiv:2504.17761*, 2025b.

Yiyang Ma, Xingchao Liu, Xiaokang Chen, Wen Liu, Chengyue Wu, Zhiyu Wu, Zizheng Pan, Zhenda Xie, Haowei Zhang, Xingkai Yu, Liang Zhao, Yisong Wang, Jiaying Liu, and Chong Ruan. JanusFlow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7739–7751, 2025.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

Yuandong Pu, Le Zhuo, Songhao Han, Jinbo Xing, Kaiwen Zhu, Shuo Cao, Bin Fu, Si Liu, Hongsheng Li, Yu Qiao, Wenlong Zhang, Xi Chen, and Yihao Liu. PICABench: How far are we from physically realistic image editing? In *International Conference on Learning Representations*, 2026.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Yi Shen, Jian Zhang, Jieyun Huang, Shuming Shi, Wenjing Zhang, Jiangze Yan, Ning Wang, Kai Wang, Zhaoxiang Liu, and Shiguo Lian. DAST: Difficulty-adaptive slow-thinking for large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 2322–2331, 2025.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8228–8238, 2024.

Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. VL-Rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 34852–34878, 2025a.

Jiaqi Wang, Kevin Qinghong Lin, James Cheng, and Mike Zheng Shou. Think or not? Selective reasoning via reinforcement learning for vision-language models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 123247–123285, 2025b.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025c.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

Cong Wei, Zheyang Xiong, Weiming Ren, Xinrun Du, Ge Zhang, and Wenhu Chen. OmniEdit: Building image editing generalist models through specialist supervision. In *International Conference on Learning Representations*, 2025.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022.

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

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, and Ping Luo. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12966–12977, 2025b.

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, et al. OmniGen2: Towards instruction-aligned multimodal generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026a.

Jay Zhangjie Wu, Xuanchi Ren, Tianchang Shen, Tianshi Cao, Kai He, Yifan Lu, Ruiyuan Gao, Enze Xie, Shiyi Lan, Jose M. Alvarez, Jun Gao, Sanja Fidler, Zian Wang, and Huan Ling. ChronoEdit: Towards temporal reasoning for image editing and world simulation. In *International Conference on Learning Representations*, 2026b.

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*, 2026c.

Yongliang Wu, Zonghui Li, Xinting Hu, Xinyu Ye, Xianfang Zeng, Gang Yu, Wenbo Zhu, Bernt Schiele, Ming-Hsuan Yang, and Xu Yang. KRIS-Bench: Benchmarking next-level intelligent image editing models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 174202–174248, 2025c.

Wenyi Xiao and Leilei Gan. Fast-slow thinking GRPO for large vision-language model reasoning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 190068–190098, 2025.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. In *International Conference on Learning Representations*, 2025.

Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. LLaVA-CoT: Let vision language models reason step-by-step. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2087–2098, 2025.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. DanceGRPO: Unleashing GRPO on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 11809–11822, 2023.

Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. ImgEdit: A unified image editing dataset and benchmark. In *Advances in Neural Information Processing Systems*, volume 38, pp. 146002–146032, 2025.

Fukun Yin, Shiyu Liu, Yucheng Han, Zhibo Wang, Peng Xing, Rui Wang, Wei Cheng, Yingming Wang, Aojie Li, Zixin Yin, Pengtao Chen, Xiangyu Zhang, Daxin Jiang, Xianfang Zeng, and Gang Yu. ReasonEdit: Towards reasoning-enhanced image editing models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, et al. DAPO: An open-source LLM reinforcement learning system at scale. In *Advances in Neural Information Processing Systems*, volume 38, pp. 125532–125554, 2025.

Jiajie Zhang, Nianyi Lin, Lei Hou, Ling Feng, and Juanzi Li. AdaptThink: Reasoning models can learn when to think. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3716–3730, 2025.

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, pp. 31428–31449, 2023.

Xiangyu Zhao, Peiyuan Zhang, Kexian Tang, Xiaorong Zhu, Hao Li, Wenhao Chai, Zicheng Zhang, Renqiu Xia, Guangtao Zhai, Junchi Yan, Hua Yang, Xue Yang, and Haodong Duan. Envisioning beyond the pixels: Benchmarking reasoning-informed visual editing. In *Advances in Neural Information Processing Systems*, volume 38, pp. 192865–192904, 2025.

A Extended Related Work

Slow and structured thinking in language models. Chain-of-thought prompting (Wei et al., 2022) demonstrated that eliciting intermediate reasoning steps dramatically improves language model performance on multi-step problems, establishing explicit reasoning as a first-class training objective. Scaling this paradigm through reinforcement learning, DeepSeek-R1 (Guo et al., 2025) produces models that reason extensively before answering, a strategy highly effective on symbolic and mathematical tasks. Structured variants further organize these steps: Tree of Thoughts (Yao et al., 2023) enables deliberate tree-structured search over partial solutions, while LLaVA-CoT (Xu et al., 2025) adapts structured multi-stage reasoning to vision-language models via a four-step protocol (Summary $\rightarrow$ Caption $\rightarrow$ Reasoning $\rightarrow$ Conclusion). These works collectively embed a foundational premise: reasoning is universally beneficial and depth monotonically improves performance. **Our paper challenges this premise directly in the context of image editing**, showing empirically that reasoning can actively hurt performance when misaligned with task complexity.

Adaptive reasoning: when and how much to think. Recognizing that uniform deep reasoning can be wasteful or counterproductive, a growing body of work explores adaptive thinking. Chen et al. (2024) identify the overthinking phenomenon in o1-style LLMs and train models to prefer shorter correct responses via preference optimization, achieving length reduction without accuracy loss. DAST (Shen et al., 2025) refines this with a difficulty-aware token length budget tied to sampling accuracy for offline preference optimization. AdaptThink (Zhang et al., 2025) treats thinking-mode selection as the policy itself, training models to choose between thinking and no-thinking per query. In the vision-language domain, FAST (Xiao & Gan, 2025) introduces difficulty-aware GRPO that dynamically rewards concise reasoning for easy inputs and detailed reasoning for hard ones, achieving over 10% accuracy gains with up to 67% fewer tokens on visual reasoning QA. TON (Wang et al., 2025b) similarly trains VLMs to selectively skip reasoning for easy queries via two-stage SFT and GRPO.

Despite these advances, all prior adaptive reasoning works share a critical limitation: they are designed exclusively for *perception or reasoning QA tasks*, where the reward signal reduces to a binary or categorical correctness check against a ground-truth answer. Image editing breaks this assumption along three dimensions. First, the output space is continuous, and infinitely many valid edits exist for any instruction. Second, rewards must be computed by an imperfect VLM judge, whose own robustness upper-bounds the achievable policy quality. Third, and most critically, we show empirically (Section 3.2) that adaptive reasoning length *alone* is insufficient: without structural scaffolding, variable-length reasoning degenerates into a verbosity optimization, and our ablation demonstrates that tripling output length yields no quality improvement (GPT-Score $\approx$5.2 across all budget levels), while introducing structural directives at the same budget achieves a substantial gain to 6.1. This failure mode cannot arise in classification-style tasks where the reasoning chain is never consumed by a downstream generative model. **We are the first to characterize overthinking as a distinct failure mode in generative image editing, and to show that resolving it requires a compound adaptive-and-structured solution.**

Verifier-augmented editing. ReasonEdit (Yin et al., 2026) and EditThinker (Li et al., 2025a) achieve strong reasoning-guided editing through external verification loops. EditThinker runs an 8B verifier for up to 8 iterative rounds, incurring up to $8\times$ inference overhead and preventing joint optimization of reasoning and generation.

Reinforcement learning for visual generation and editing. RL has been extensively applied to align visual generation with non-differentiable reward signals. DDPO (Black et al., 2024) casts denoising as a multi-step MDP and applies policy gradient fine-tuning to diffusion models. DPOK (Fan et al., 2023) introduces online policy gradient with KL regularization for text-to-image diffusion. Diffusion-DPO (Wallace et al., 2024) adapts direct preference optimization to diffusion via the ELBO. Flow-GRPO (Liu et al., 2025a) brings GRPO to flow-matching models by converting their deterministic ODE to an equivalent SDE that enables group-relative advantage estimation; this approach has been extended to video generation (Xue et al., 2025) and autoregressive image editing (Ahmadi et al., 2025).

This body of work establishes RL as a principled tool for visual quality optimization. However, all of these methods treat reasoning and generation as entirely disjoint processes, optimizing *what to generate* without modeling *how the model should think before generating*. Our work is the first to co-train reasoning and generation within a single unified backbone via complementary RL objectives, GRPO on the understanding expert and Flow-GRPO on the generation expert, enabling end-to-end joint optimization of both processes under verifiable reward signals.

Reasoning-augmented image generation and editing. A parallel thread incorporates explicit reasoning chains into generation and editing. In generation, GoT (Fang et al., 2025) introduces a generation chain-of-thought where the model first produces semantic-spatial reasoning with coordinates, which then conditions a diffusion generator. T2I-R1 (Jiang et al., 2025) defines dual-level CoT, semantic planning plus token-level pixel generation, optimized via BiCoT-GRPO. In editing, MGIE (Fu et al., 2024) learns to derive expressive instructions from terse commands, while SmartEdit (Huang et al., 2024) introduces perception-aware pre-training to handle complex indirect instructions. BAGEL (Deng et al., 2025), our backbone, has already shown that allowing a unified model to emit intermediate "thinking" before generation consistently improves both generation and editing quality.

Closest to our motivation, ShortCoTI (Gu et al., 2025) identifies *visual overthinking* in autoregressive T2I generation and uses RL with a difficulty-scaled length reward to shorten reasoning prompts by 54% while maintaining or slightly improving image quality. However, ShortCoTI controls only *how much* the model reasons, not how the reasoning is structured, and operates in text-to-image generation, a fundamentally different setting from instruction-based image editing, where source image and instruction are jointly processed and the edit must precisely preserve unrelated regions. Our analysis shows that in editing, adapting reasoning length alone is insufficient (Section 3.2); adaptivity and structure are jointly necessary. The remaining works assume, implicitly or explicitly, that richer reasoning is always preferable, a premise our analysis systematically refutes.

Verifier-augmented editing. The most closely related works are ReasonEdit (Yin et al., 2026) and EditThinker (Li et al., 2025a), which introduce external verification loops to enable reasoning-guided image editing. EditThinker employs an 8B external verifier that inspects each generated edit, resolves ambiguities, and issues corrective instructions for up to 8 iterative rounds, achieving strong performance at the cost of substantial inference overhead (up to $8\times$ the base editing cost per sample). This multi-model, multi-turn architecture also prevents joint optimization: reasoning and generation remain decoupled across separate models, trained independently.

We pose a direct question: can a *single* unified model, without external verifiers or additional parameters, match this performance in a single forward pass? Our method answers affirmatively. By internalizing adaptive structured reasoning directly into the policy model via compound GRPO training, simultaneously optimizing the understanding expert for reasoning quality and the generation expert for edit fidelity, we match or exceed EditThinker's performance while operating at $20\times$ lower inference cost.

B REVISE-50K: Extended Details

Existing image editing datasets (Brooks et al., 2023; Zhang et al., 2023; Wei et al., 2025; Ge et al., 2024) cover narrow task families, provide no signal about appropriate reasoning depth, and treat all instructions as equally demanding, the exact conditions that produce the overthinking failure mode we seek to cure. We therefore introduce **REVISE-50K** (**R**easoning **V**isual **I**nstruction **S**kills & **E**dits), a dataset of 50,000 multimodal samples purpose-built for GRPO-based training of reasoning-adaptive editing models (full statistics in Table 7). REVISE-50K is distinguished by three properties absent in prior work: (i) **breadth**: 50 fine-grained editing skills across 10 high-level categories, sourced from 74 heterogeneous datasets; (ii) **disentangled instruction design**: hardness and verbosity are independently controlled, preventing the model from using surface cues as proxies for difficulty; and (iii) **graded reasoning annotations**: every sample carries a complexity label and reasoning trace that directly drive our adaptive reward (Section 3).

B.1 Taxonomy: Skills and Categories

REVISE-50K is organised into a two-level taxonomy of 10 high-level categories and 50 fine-grained skills, visualised in Figure 4. Representative skills, such as anachronism detection, symmetry-pattern fixing, multi-instruction execution, and mechanical-state prediction, illustrate that the taxonomy deliberately covers edits requiring genuine visual reasoning, not merely pixel-level manipulation. Both taxonomy levels are roughly balanced, ensuring that training gains cannot be attributed to distributional shortcuts in any single category or skill. Qualitative samples spanning diverse skills and complexity levels are shown in Figures 9 and 10 (Appendix B.8).

B.2 Image Collection and Curation

Collection. Each skill is paired with semantically appropriate sources drawn from 74 datasets spanning natural photographs (e.g., COCO, FFHQ), domain-specific collections (e.g., WikiArt, DeepFashion), and procedurally generated stimuli (dice nets, matchstick puzzles, mazes); Figure 7 shows the per-category source distribution. We collect 2$\times$ more images than needed ($\sim$100K) and retain 50K after filtration, ensuring no skill is under-represented after attrition. Skills that inherently require multiple inputs, such as body-clothing transfer, object insertion/extraction, and future-frame prediction, are collected as paired image sets; 93% of final samples are single-image, and 7% contain 2–4 images.

Three-stage filtration. Raw images pass through the pipeline in Figure 4 before any instruction is generated. Stage 1 applies five deterministic quality filters (deduplication, near-duplicate removal, blank-image rejection, extreme-aspect-ratio pruning, corrupt-file detection), removing 8.1% of images in total. Stage 2 screens for unsafe content via a dual-consensus NSFW policy (NudeNet detector + Falconsai ViT classifier), flagging 2.2% of images while substantially suppressing false positives. Stage 3, the highest-impact stage at 3.8%, verifies semantic relevance with a skill-specific VLM prompt, catching images that pass quality and safety checks but are misaligned with their assigned skill (e.g., a price tag inadvertently classified as math content). Across all stages, $\sim$14% of collected images are discarded; full per-filter statistics are in Appendix B.4.

B.3 Instruction Generation and Reasoning Annotation

All instructions and reasoning annotations are generated by InternVL3.5-241B-A28B Wang et al. (2025c), the largest open-weight model in the InternVL3.5 family. We chose an open-weight model deliberately: any user of REVISE-50K can fully replicate or extend the pipeline without proprietary API access. Even for source datasets that ship with existing instructions (e.g., OmniEdit), we discard them and regenerate to enforce consistent style and difficulty distribution.

Instructions are controlled along two independent axes: **hardness** (direct vs. indirect phrasing, sampled at $\sim$60% indirect by default) and **verbosity** (8–20 vs. 20–50 words, sampled 50/50). The critical property is that these axes are uncorrelated, so a two-word instruction can be deeply indirect while a fifty-word one can be entirely explicit, ensuring the model cannot exploit surface features as a proxy for difficulty (Figure 5b). Each sample is further enriched with a graded complexity label (None / Light / Moderate / Heavy) and a reasoning trace jointly assessed from the instruction and image content; these labels directly drive our complexity-conditioned reward during GRPO training. The full annotation protocol, four-level complexity definitions, and per-category breakdowns are provided in Appendix B.5.

B.4 Image Filtration Pipeline

We apply a three-stage filtration pipeline to the $\sim$100K collected images before any instruction is generated. The pipeline is designed to be conservative: any image that fails a stage is discarded, never repaired.

Stage 1: Hard Quality Filters. Five deterministic filters are applied in sequence:

F1. Exact Deduplication. We compute the MD5 digest of every image and remove exact duplicates *cross-skill*, ensuring the same image cannot appear in multiple skills. Removed **2.3%** of collected images.

F2. Near-Duplicate Removal. A 64-bit perceptual hash (pHash) is computed per image; pairs with Hamming distance ≤ 8 bits are flagged. Candidate pairs are found efficiently via a band-index using the LSH pigeonhole guarantee ($n_{\text{bands}} = 9$), ensuring *zero false negatives* by construction. Flagged an additional **3.1%** of images, predominantly cross-dataset pairs sharing the same underlying photograph.

F3. Blank and Near-Blank Images. Images are converted to grayscale; those with pixel variance $\sigma^2 < 20.0$ are rejected. This threshold captures solid-colour images and heavily compressed near-blanks (typical photographic content has $\sigma^2 > 200$). Removed **0.9%** of images.

F4. Extreme Aspect Ratios. Images satisfying $\max(W/H,\ H/W) \geq 4.0$ are discarded. Such near-1D images are incompatible with standard VLM input resolutions and produce degenerate crops. Removed **1.4%** of images.

F5. Corrupt File Detection. Rather than relying on the metadata-only `PIL.verify()`, we force a full pixel decode (`img.load()` + NumPy array construction) to catch truncated or malformed files that pass header validation. Corrupt files accounted for **0.4%** of collected images.

Stage 1 together removed **8.1%** of all collected images.

Stage 2: Content Safety Screening. We apply two complementary NSFW classifiers and remove any image flagged by *both*:

- **NudeNet (detection-based).** A bounding-box detector localising five explicit body-part classes (`FEMALE_BREAST_EXPOSED`, `MALE/FEMALE_GENITALIA_EXPOSED`, `BUTTOCKS_EXPOSED`, `ANUS_EXPOSED`). An image is flagged if any detection exceeds confidence $\tau = 0.5$.
- **Falconsai ViT (classification-based).** A Vision Transformer fine-tuned for binary NSFW/normal classification (Falconsai, 2023). Images with NSFW probability ≥ 0.5 are flagged.

Requiring consensus from both classifiers, one localisation-based and one classification-based, substantially reduces false positives while maintaining high recall of genuinely harmful content. Dual-consensus flagging identified **2.2%** of images as NSFW, a substantially lower false-positive rate than either classifier applied independently.

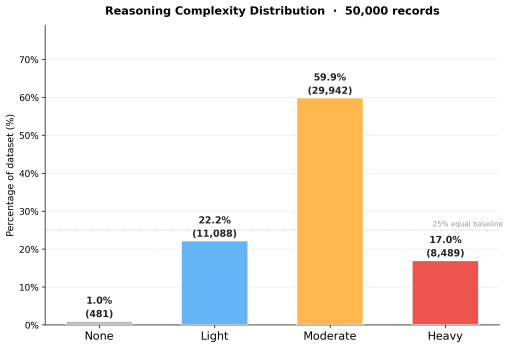
Stage 3: Semantic Relevance Verification. Automated sampling can yield images that are formally valid but semantically misaligned with their assigned skill (*e.g.*, a grocery price tag inadvertently counted as "math" content). We address this with a skill-specific Yes/No prompt answered by InternVL3.5-241B-A28B (Wang et al., 2025c) served via vLLM + Ray. For each of the 50 skills we author a tailored prompt that defines the minimal visual requirements: the *advertisement* prompt accepts both existing ads and clean surfaces suitable for ad placement, while the *math* prompt requires a clearly readable mathematical element as the primary visual subject. This stage rejected **3.8%** of images, the highest single-stage removal rate, underscoring the necessity of VLM-based verification over purely metadata-driven sampling.

Summary. Table 6 summarises removal rates per stage. In total, $\sim$**14%** of collected images were discarded before the final 50K samples were selected to balance skill coverage.

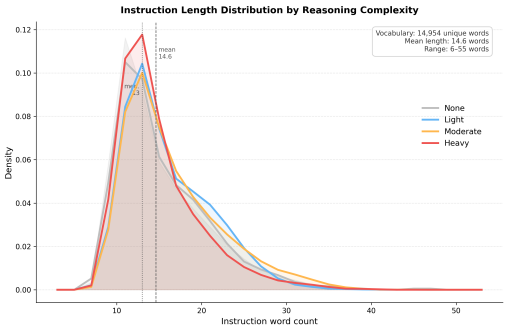
Resolution distribution. Figure 5d shows the distribution of input resolution (shorter side) after filtering. Approximately **77%** of samples have $\min(W, H) \geq 512$ pixels, ensuring high visual fidelity for most training examples. We deliberately retain the remaining **23%** at lower resolution to promote robustness and generalisation to real-world low-quality inputs.

Table 6: Per-stage image removal rates across the filtration pipeline.

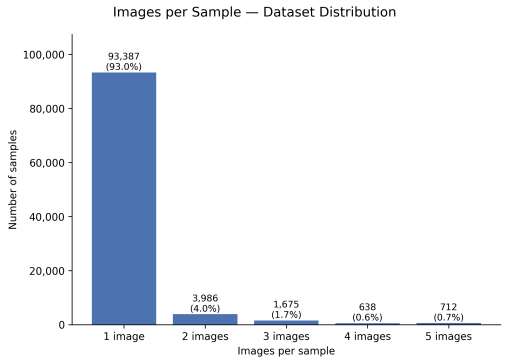
Stage	Filter	Removed
Stage 1	Exact deduplication	2.3%
	Near-duplicate (pHash)	3.1%
	Blank / near-blank	0.9%
	Extreme aspect ratio	1.4%
	Corrupt file	0.4%
Stage 2	Dual-consensus NSFW	2.2%
Stage 3	Semantic relevance (VLM)	3.8%
Total		**≈14%**



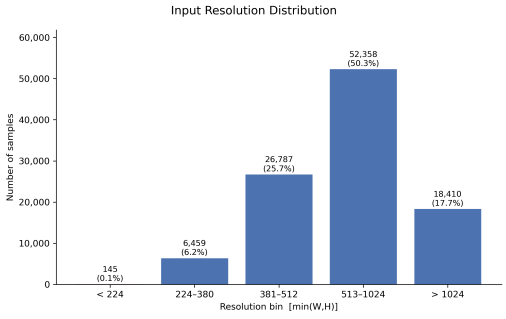
(a) Reasoning complexity distribution.



(b) Instruction length distribution.



(c) Images per sample.



(d) Input resolution distribution.

Figure 5: REVISE-50K dataset statistics: reasoning complexity label distribution, instruction length distribution, images-per-sample histogram, and input resolution distribution after filtering.

B.5 Reasoning Complexity Annotation Protocol

Every sample in REVISE-50K is enriched with three annotation fields produced by InternVL3.5 (Wang et al., 2025c): a complexity label (None / Light / Moderate / Heavy), a step-by-step reasoning trace at the calibrated depth, and a refined instruction, i.e., a clean, unambiguous 10–30-word rewrite of the original that removes vagueness without changing intent.

Complexity is assessed jointly from the instruction text and the image content. The four levels are defined as follows:

- **None**: the instruction is entirely self-contained; no visual inspection is required. Reasoning is left empty.

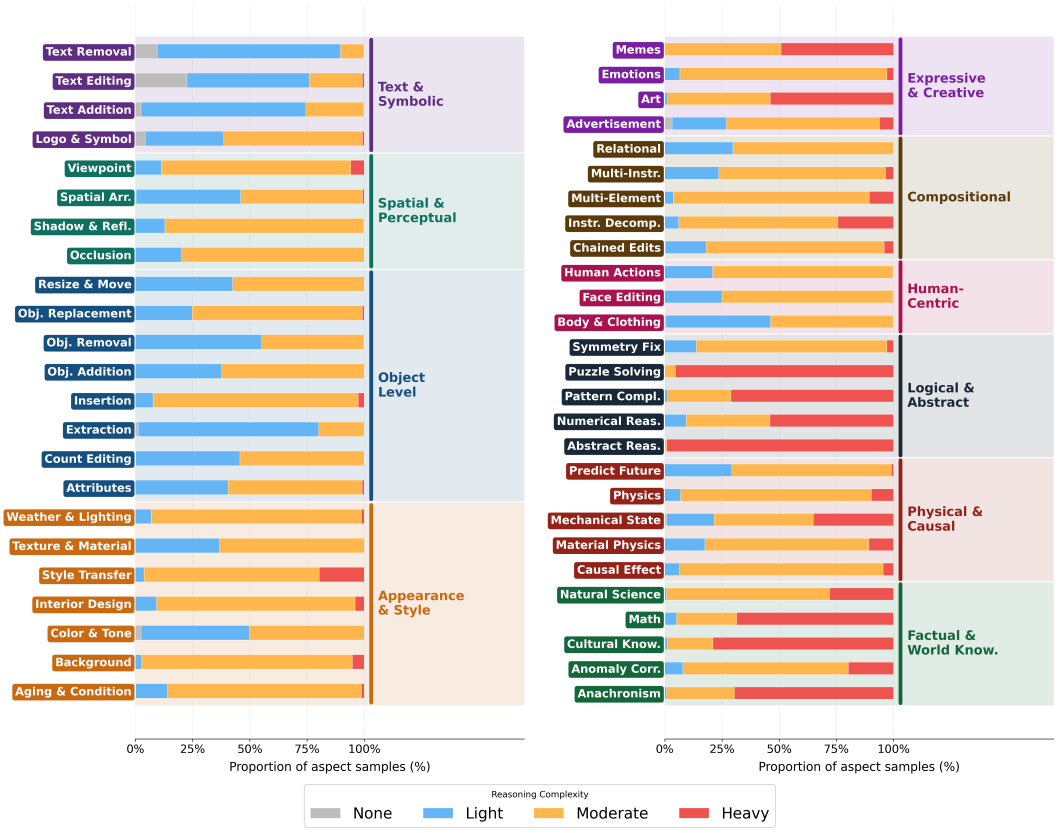


Figure 6: Reasoning complexity distribution broken down by the 10 high-level categories. Logical & Abstract and Factual & World Knowledge are dominated by Heavy samples; Text & Symbolic and Object-Level Edits skew toward Light.

- **LIGHT**: the edit is direct and unambiguous, but a brief glance at the image is needed to locate the target. Reasoning is capped at one sentence ($\leq$20 words).
- **MODERATE**: mild ambiguity is present, or the scene requires contextual understanding (e.g., identifying the correct object among similar ones, spatial reasoning, or indirect phrasing). Reasoning comprises 2 sentences (40–50 words total).
- **HEAVY**: the instruction demands significant reasoning from one or more of: cultural or historical knowledge, mathematical or numerical inference, physical or causal reasoning, pattern or puzzle solving, anomaly detection, multi-step logic, or domain-specific concepts. Reasoning comprises 3–5 sentences (50–70 words total).

The overall distribution is approximately 1% None, 22% Light, 60% Moderate, and 17% Heavy (Figure 5a). The distribution is intentionally skewed toward non-trivial reasoning: the 60% Moderate fraction reflects the real-world prevalence of indirectly phrased requests, while the 17% Heavy fraction covers edits that require domain knowledge beyond simple visual lookup.

B.6 PER-CATEGORY REASONING COMPLEXITY BREAKDOWN

Figure 6 shows the reasoning complexity distribution broken down by category. Logical & Abstract Reasoning and Factual & World Knowledge are dominated by Heavy samples, as these categories inherently require cultural knowledge, pattern solving, or multi-step logic. Text & Symbolic and Object-Level Edits skew toward Light, reflecting their more operationally direct nature. Importantly, even the "easiest" categories contain substantial Moderate fractions, because we sample $\sim$60% indirect instructions per skill, introducing tricky phrasings even for conceptually simple edits.

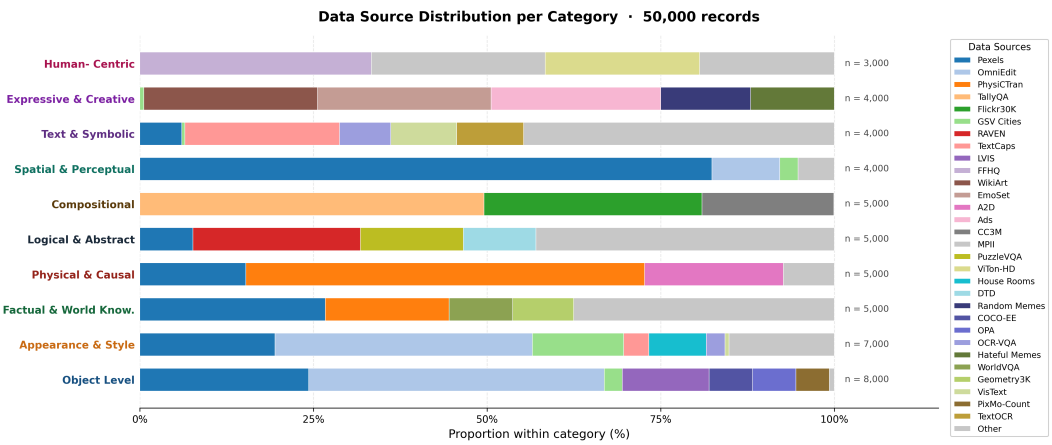


Figure 7: Per-category source dataset distribution across the 10 high-level categories in REVISE-50K. No single source dominates any category, ensuring visual diversity within each skill group.

B.7 Dataset Diversity Analysis

Source diversity. Figure 7 confirms that no single source dataset dominates any of the 10 categories, ensuring that category-level competence cannot be attributed to a single corpus's visual style or content bias. The 74 sources span three provenance types: natural photographs (e.g., COCO, LVIS, Flickr30K, GSV-Cities), domain-specific benchmarks (e.g., DeepFashion, FFHQ, WikiArt, OmniEdit), and procedurally generated stimuli (e.g., dice nets, matchstick puzzles, mazes, nonograms).

Skill balance. Each of the 50 skills contributes approximately 1,000 samples to the final 50K dataset ($\pm$10%, depending on skill-specific filtration attrition). The sunburst chart in Figure 4 confirms that outer-ring categories are similarly balanced, preventing the model from over-specialising in any one competency domain.

Instruction diversity. The 50K instructions span $\sim$**15,000 unique vocabulary tokens**. Figure 8 shows a complexity-stratified word cloud: HEAVY-reasoning instructions cluster around *complete* and *pattern* (visual logic); MODERATE instructions around *adjust* and *transform*; LIGHT instructions around *change* and *color*. This natural correlation between word choice and task demand confirms that the annotation model correctly calibrates complexity, rather than assigning labels arbitrarily.

Table 7: REVISE-50K full dataset statistics.

Property	**Value**
Total samples	50,000
Number of skills	50
Number of categories	10
Image sources	74
Single-image samples	93%
Multi-image samples	7% (2–4 images)
Resolution $\geq$512 px	77%
Unique vocabulary tokens	$\sim$15,000
Reasoning complexity distribution	
NONE	$\sim$1%
LIGHT	$\sim$22%
MODERATE	$\sim$60%
HEAVY	$\sim$17%
Instruction generation model	InternVL3.5-241B-A28B
Reasoning annotation model	InternVL3.5-241B-A28B
Semantic verification model	InternVL3.5-241B-A28B



Figure 8: Complexity-stratified word cloud of the 50K instructions. HEAVY-reasoning instructions cluster around reasoning-intensive terms (*complete*, *pattern*); MODERATE around *adjust* and *transform*; LIGHT around *change* and *color*.

B.8 QUALITATIVE DATASET SAMPLES

Figures 9 and 10 show representative samples from REVISE-50K spanning diverse skill categories, instruction phrasings, and complexity levels.

C METHODOLOGY: ADDITIONAL DETAILS

C.1 PRELIMINARIES

Unified Multimodal Models. Recent work has shown that jointly training multimodal understanding and generation within a single model yields strong emergent capabilities (Xie et al., 2025; Ma et al., 2025; Wang et al., 2024; Wu et al., 2025b). Among these, BAGEL (Deng et al., 2025) is an open-source, decoder-only foundation model with 7B active parameters (14B total), pretrained on trillions of interleaved text, image, video, and web tokens. Architecturally, BAGEL adopts a Mixture-of-Transformer-Experts (MoT) design with two modality-specific transformer experts, one dedicated to multimodal *understanding* and one to multimodal *generation*, operating over a shared token sequence via shared self-attention at every layer, enabling bottleneck-free cross-modal interaction. For text token prediction, BAGEL follows the standard next-token prediction paradigm; for visual token prediction, it adopts Rectified Flow (Lipman et al., 2023) over noised VAE tokens. This clean decoupling of the two experts is what directly motivates our training strategy: we apply GRPO to the understanding/reasoning expert to improve instruction fidelity, and Flow-GRPO to the generation expert to optimize visual editing quality, both under verifiable reward signals. We use BAGEL as our backbone and baseline, aiming to enhance its reasoning capabilities and generation quality simultaneously.

Group Relative Policy Optimization (GRPO). GRPO (Shao et al., 2024) is a critic-free reinforcement learning algorithm that eliminates the value network of PPO (Schulman et al., 2017) by estimating advantages directly from a sampled group of responses. Given a query q, the policy π_θ samples G responses $\{o_i\}_{i=1}^{G}$, each assigned a scalar reward $r_i = R(q, o_i)$. The normalized group advantage is:

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^{G})}{\text{std}(\{r_j\}_{j=1}^{G})}. \tag{5}$$



Figure 9: **REVISE-50K samples (Part I).**

The policy is updated by maximizing a clipped surrogate objective with a KL regularization penalty:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{q,\{o_i\}}\left[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\Big(\rho_{i,t}\,\hat{A}_i,\ \text{clip}(\rho_{i,t},1{-}\varepsilon,1{+}\varepsilon)\,\hat{A}_i\Big) - \beta\,\mathbb{D}_{\text{KL}}[\pi_\theta \,\|\, \pi_{\text{ref}}]\right], \tag{6}$$



Figure 10: **REVISE-50K samples (Part II).**

where $\rho_{i,t} = \pi_\theta(o_{i,t} \mid q, o_{i,<t}) / \pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})$ is the token-level importance sampling ratio, ε is the clipping threshold, and β weights the KL penalty against the frozen reference policy π_{ref}. GRPO was central to DeepSeek-R1 (Guo et al., 2025) and has since been adopted widely for post-training

LLMs (Yu et al., 2025; Hu et al., 2025) and VLMs (Wang et al., 2025a; Ko et al., 2026). In this work, we apply GRPO to BAGEL's understanding expert to optimize its reasoning over editing instructions.

Flow-GRPO. Applying GRPO directly to flow-matching models is non-trivial: their standard sampling follows a *deterministic* ODE, which prevents the group diversity required for advantage estimation. Flow-GRPO (Liu et al., 2025a) resolves this by converting the ODE into an equivalent SDE that preserves the original model's marginal distribution at all timesteps:

$$\underbrace{dx = v_\theta(x,t)\,dt}_{\text{ODE (GRPO)}} \longrightarrow \underbrace{dx = \big[v_\theta(x,t) - \tfrac{1}{2}g(t)^2\nabla_x \log p_t(x)\big]dt + g(t)\,dW_t}_{\text{SDE (Flow-GRPO)}}, \tag{7}$$

where v_θ is the learned velocity field, $g(t)$ is the diffusion coefficient, and dW_t is a Wiener process. This stochastic perturbation introduces the sampling diversity needed for group-relative advantage estimation, while a *denoising reduction* strategy reduces the number of training denoising steps to improve sampling efficiency without degrading inference quality. Flow-GRPO achieves strong gains on compositional generation and visual text rendering tasks (Liu et al., 2025a), and has been extended to image editing (Ahmadi et al., 2025) and video generation (Xue et al., 2025). We apply Flow-GRPO to BAGEL's generation expert, optimizing editing adherence, structural preservation, and perceptual quality under multi-aspect reward signals. Beyond the standard formulation, we introduce *variance-guided early-window selection*: rather than uniformly sampling denoising timesteps, we concentrate gradient signal in the early timestep windows of highest generation variance, where the model's decisions have the greatest impact on structural edit fidelity. As shown in Appendix D.3, this yields faster convergence and a higher final reward compared to the sliding-window schedule of MixGRPO (Li et al., 2025b).

C.2 PRISM: Full Inference & Training Pipeline

Algorithm 1 PRISM: Inference & Training

Require: $\mathcal{M}$, I, x, $\mathcal{L}$, $\tilde{c}$
Ensure: Edited image $\hat{I}$
(Train, once) warm-start Stage 1 of $\mathcal{M}$ with SFT on $(x, I, \tilde{c})$
$\hat{c} \leftarrow \mathcal{M}(x,\, I;\, \rho_{\text{cpx}})$ ▷ **S1:** predict complexity
$(l,\, n) \leftarrow \beta(\hat{c})$ ▷ $l \in \{1,2,3\}$ lenses, n=3 actions
$\hat{L} \leftarrow \mathcal{M}(x,\, I,\, \mathcal{L};\, \rho_{\text{lens}})$ ▷ **S2:** generate l tailored lenses (guided by $\mathcal{L}$)
for $k = 1, \ldots, l$ **do**
  $\tau_k \leftarrow \mathcal{M}\Big(x,\, I,\, \hat{L}_{1:k},\, \tau_{<k};\, \rho_{\hat{l}_k}\Big)$ ▷ **S3:** execute lens ($\tau_{<1}$=$\varnothing$)
end for
$\mathbf{A} \leftarrow \mathcal{M}\big(x,\, I,\, \{\tau_k\}_{k=1}^{l};\, \rho_{\text{act}}\big)$ ▷ **S4:** generate n tailored (t_j, d_j) actions
$\hat{I} \leftarrow \mathcal{M}(\mathbf{A},\, I)$ ▷ single-shot edit by the generation expert
(Train) R_1 (Eq. (8)), R_2–R_4 (InternVL3.5-241B judge), r_{img} (Eq. (4))
(Train) Update $\mathcal{M}$ via GRPO (Eq. (6)) and Flow-GRPO

C.3 Verbosity Ablation: Training and Evaluation Prompts

Training Prompts. Each variant is trained via LoRA on BAGEL's textual stream. The system prompt is held constant; only the reasoning budget directive changes.

Prompt 1: Light (10–15 words)

Role: You are an image-editing assistant.

Goal: Given an image and an editing instruction, output a LIGHT, direct reasoning in 10-15 words total.

Requirements:

- No thinking out loud. One crisp statement only.
- Mention: target object, exact change, and key preservation constraint.

Output format (strict):

Reasoning: <10-15 words>

Example:

Instruction: "Add a spotlight on the singer"
Reasoning: Add a warm spotlight above the singer, illuminating face and upper body, stage unchanged.

Give me the output given this input instruction:

Prompt 2: Moderate (20–30 words)

Role: You are an image-editing assistant.

Goal: Given an image and an editing instruction, output a MODERATE reasoning. Total word budget: 20-30 words. Spend 10-15 words thinking out loud first -- those words count against your budget.

Output format (strict):

[Thinking out-loud]: <10-15 words>
[Reasoning]: <15-20 words>

Example:

Instruction: "Add a spotlight on the singer"
[Thinking out-loud]: The singer needs a dramatic spotlight to stand out on stage.
[Reasoning]: Add a warm-white spotlight from above, illuminating the singer's face and upper body, keeping stage background dark and pose unchanged.

Give me the output given this input instruction:

Prompt 3: Heavy (40–60 words)

Role: You are an image-editing assistant.

Goal: Given an image and an editing instruction, output a HEAVY, detailed reasoning. Total word budget: 40-60 words.

Output format (strict):

Reasoning: <40-60 words>

Example:

Instruction: "Convert the bus station into an electric one"
Reasoning: Station has diesel buses and old signage. Need visible EV infrastructure without changing the layout or passengers. Replace diesel buses with electric models, add EV charging pillars, cables, and digital screens. Apply green/blue accent signage while preserving the original layout, scale, platform, passengers, and surrounding buildings unchanged.

Give me the output given this input instruction:

Prompt 4: Heavy + Structure (40–60 words)

Role: You are an image-editing assistant.

Goal: Given an image and an editing instruction, output a HEAVY, detailed reasoning. Total word budget: 40-60 words. Spend up to 30 words thinking out loud -- those words count against your budget. Use the remaining words for your final answer.

Output format (strict):

[Thinking out-loud]: <20-30 words>
[Reasoning]: <20-40 words>

Example:

Instruction: "Convert the bus station into an electric one"
[Thinking out-loud]: Station has diesel buses and old signage. Need visible EV infrastructure without changing the layout or passengers.
[Reasoning]: Replace diesel buses with electric models, add EV charging pillars, cables, and digital screens. Apply green/blue accent signage while preserving the original layout, scale, platform, passengers, and surrounding buildings unchanged.

Give me the output given this input instruction:

Evaluation Prompt. All variants are evaluated using the following GPT-5.2 judge prompt. The model's generated reasoning is substituted in alongside the source image(s) and editing instruction.

Prompt 5: GPT-5.2 Verbosity Evaluation

You are an expert evaluator for visual reasoning tasks. Given an input image (or multiple images), an editing/reasoning instruction, and a generated reasoning response, evaluate the quality of the generated reasoning.

Score the response on a scale of 0-10 based on:

1. **Correctness:** Does the reasoning arrive at the correct answer?
2. **Coherence:** Is the reasoning logically sound and well-structured?
3. **Completeness:** Does it address all aspects of the instruction?
4. **Disambiguation:** Does the reasoning correctly identify and resolve any ambiguities in the instruction or image (e.g. multiple objects, unclear references, puzzle, math, world-knowledge, etc.)?

Provide a brief justification, then end with exactly:
Score: <integer between 0 and 10>

C.4 PRISM Lens Pool: Derivation and Definitions

C.4.1 Failure-Mode Taxonomy of the BAGEL Baseline

To ground the design of $\mathcal{L}$, we conduct a systematic failure analysis of the BAGEL baseline prior to training. Specifically, we draw a random 1K sample from REVISE-50K, run BAGEL on each, and collect low-scoring outputs (as judged by our automated metric) as failure candidates. We then sample a stratified subset of these failures across instruction types and evaluate each with InternVL3.5-14B prompted to classify the primary failure mode into one of nine categories. Figure 11 shows the resulting distribution; each identified failure cluster maps directly to one lens in $\mathcal{L}$:

Disambiguation	**Grounding**	**Plausibility**
Instruction ambiguity	Wrong object selected	Physically implausible result
Photorealism	**FailureMode**	**Decomposition**
Texture / photo artifact	Common model failure	Missed sub-operation
Consistency	**Causality**	**Temporal**
Style / background inconsistency	Wrong causal inference	Temporal / sequential error

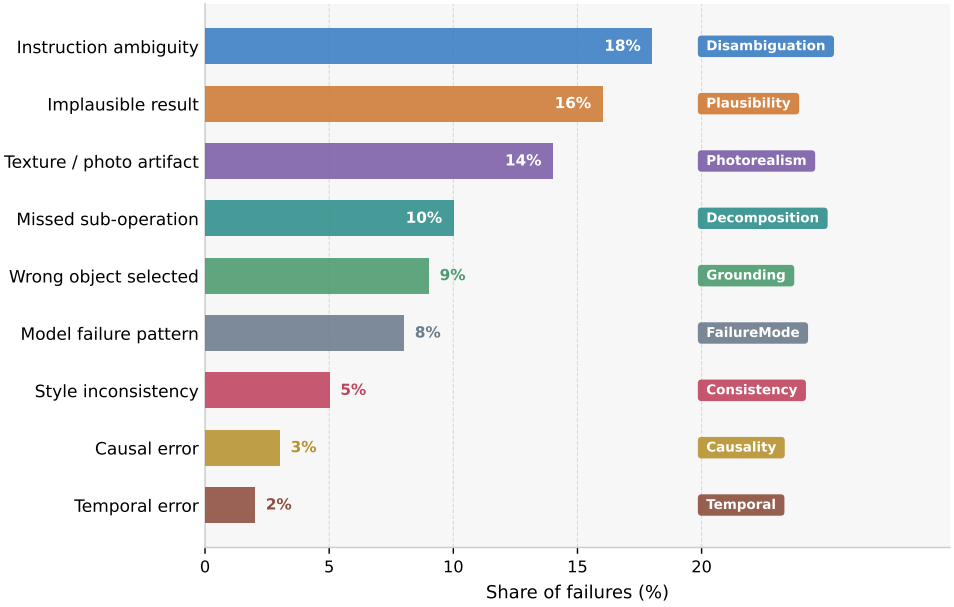


Figure 11: **Failure-mode distribution from BAGEL baseline analysis.** Each bar shows the percentage of failures attributable to that cluster, measured over low-scoring outputs from a random 1K sample of REVISE-50K. Colored badges indicate the lens in $\mathcal{L}$ to which each cluster maps.

C.4.2 Lens Definitions and Stage Prompts

Each lens $l_k \in \mathcal{L}$ is applied via a dedicated stage prompt ρ_{l_k} that focuses the understanding expert on a specific reasoning dimension. We present the three prompts currently implemented; the remaining six follow the same structural template and will be released with the code.

Prompt 6: Photorealism Lens

```
You are an expert image editor focused on photorealism.  Given an
image and an editing instruction (with optional additional context),
your sole job is to specify the fine-grained visual details needed to
make the edit look high-quality, natural and believable.

Don't repeat what is already mentioned in the input or the reasoning.
Add new valuable details related to plausibility and photorealism.
Generate 1-2 sentences.  Maximum 50 words.

Example:
  Instruction:  "Add a beard to the man."
  Output:  The beard should match the man's existing hair color and
grow
           naturally from his jawline, sideburns, and upper lip --
following
           the contours of his face shape rather than sitting flat on
top of
           the skin.  The beard's edges must be soft and tapered, not
sharply
           cut, and subtle skin show-through should be visible.

Now, analyze this instruction:
```

Prompt 7: Grounding Lens

```
You are an expert image editor specializing in spatial understanding.
Given an image and an editing instruction (with optional additional
context), your sole job is to precisely locate the target objects or
region.

Your output should:
  - Be concise.  1-2 sentences.  Maximum 50 words.
  - Describe the target's position in the image (e.g. top-left,
behind the X).
  - Describe distinguishing visual properties:  color, size, shape,
texture,
    and relationship to surrounding objects.

Example 1:
  Instruction:  "Detect and correct the anomaly in this image."
  Output:  The target is the upper steering wheel occupying roughly
the
           top-center third of the image, identical in shape and
color to the
           real one below but with no hands on it and slightly
misaligned.

Example 2:
  Instruction:  "Add a groom next to the bride."
  Output:  The bride is standing slightly right of center, facing
left, in a
           white dress against a floral arch backdrop.  Accordingly,
the groom
           should be placed directly to her left, close enough that
their arms
           naturally touch, at the same scale and ground level as
her.

Now, analyze this instruction:
```

Prompt 8: Disambiguation Lens

```
You are an expert image editor specializing in reasoning.  Given
images and an editing instruction, your sole job is to reason and
resolve any ambiguity in the instruction, to make the edit direct and
clear.

Your output should:
  - Be 2-3 sentences maximum (100 words maximum).
  - Interpret any vague terms by thinking out loud first then reach
the final
    solution.
  - If the instruction involves a math problem, puzzle, physics,
chemistry, or
    world knowledge, solve it, mentioning the reasoning steps, and
state
    the answer clearly at the end.
  - Reason thoroughly about the input instruction, then end with one
crisp
    sentence stating the concrete edit to be made.

Example 1:
  Instruction:  "Detect and correct the anomaly in this image."
  Output:  [Reasoning] The anomaly here is a duplicated steering
wheel.  The
           upper one is a ghost copy floating above the real one
being gripped
           by the driver.
           [Refined Edit] Remove the upper duplicate steering wheel.

Example 2:
  Instruction:  "Change the weather in this image to match December
in Australia."
  Output:  [Reasoning] December in Australia falls in the middle of
summer, not
           winter.  Accordingly, this means bright sunshine, clear
blue skies,
           and warm golden light, not snow or frost.
           [Refined Edit] Convert the weather to a sunny summer one.

Output format (strict):
  [Reasoning] <str>
  [Refined Edit] <concise str>

Now, analyze this instruction:
```

C.5 Reward Formulations and Prompts

C.5.1 Complexity Reward

The complexity reward is an exact-match signal with partial credit for a valid-but-wrong prediction:

$$r_{\text{cpx}}(\hat{c}) = \begin{cases} 1.0 & \hat{c} = \tilde{c} \\ 0.2 & \hat{c} \neq \tilde{c},\ \hat{c} \in \mathcal{C} \\ 0.0 & \hat{c} \notin \mathcal{C} \end{cases} \tag{8}$$

C.5.2 Stage 2–4 Rewards

Stages 2–4 produce free-form text with no single reference answer, so each is scored by InternVL3.5-241B-A28B (Wang et al., 2025c) with a stage-specific rubric prompt that sees the source image(s), the instruction, and the preceding stages' outputs.

- **R_2, lens value (Stage 2):** the proposed lenses are relevant to this edit and non-redundant with one another.

- **R_3, grounding (Stage 3):** the claims made in each lens trace hold in the image, rather than describing content that is not there.
- **R_4, executability (Stage 4):** each action is concrete, self-contained, and directly executable by the generation expert.

The rubrics for action quality follow the criteria of the assessment prompt in Appendix C.6.1.

C.6 Actions Generation and Assessment Prompts

C.6.1 Actions Quality Assessment

The following prompt is used by GPT-5.2 to evaluate the quality of the action items generated by the PRISM understanding expert, given the source image(s), editing instruction, and the model's reasoning context.

Prompt 9: GPT-5.2 Action Quality Assessment

Role: `You are an Image-Edit Reasoning Evaluation Assistant.`

Task: `Assess the quality of the generated action items for an image editing task. The model receives input images, an editing instruction, and prior reasoning context, then outputs a set of action items specifying how to execute the edit.`

Evaluation Criteria:

```
  - Each action item should be grounded in the image content, not
generic.
  - Each action item should be useful, relevant and correct.
  - If the input instruction is vague or complex and needs reasoning,
the actions should
    resolve ambiguity and act as direct refined editing instructions.
  - If the instruction is indirect and requires reasoning, the action
should not reason
    out loud.  Instead it should give the direct edit to apply after
reasoning offline.
  - If the instruction is tricky and the action tries to refine it
but is logically wrong
    and against the intention of the original instruction, the score
should be very
    low.  (Zero if the action is totally wrong w.r.t. the input
instructions.)
  - Each action must be self-contained.
  - Each action item should be concise and directly executable.
```

Scoring Rubric (0–10):

```
  8–10 Excellent:  the actions fully satisfy the criteria and are
                   sound and match the intention of the input
instruction.
  5–7 Good:  Mostly satisfies the criteria with minor gaps or slight
vagueness.
  1–4 Fair:  Missing key specifics or incorrectly grounded, or not
useful
             and doesn't add new information, just redundant.
  0:  Does not attempt the task, is off-topic, or is not in English.

Provide a brief justification, then end with exactly:
  Score:  <integer between 0 and 10>
```

C.6.2 Actions Generation

The following three prompts are used by the PRISM policy model to generate action items, one prompt per action mode. N denotes the number of action items, which is determined by the complexity level predicted by the understanding expert.

Prompt 10: Freeform Action Generation

```
You are an expert image editor specializing in execution planning.
Given images, an editing instruction, and reasoning context from
prior analysis, your sole job is to produce exactly N concrete action
item(s) that together fully specify how to perform the edit.
```

Rules:

```
  - Each action item must be concise.  1 sentence only.  (20 words
per action)
  - Each action must cover a different aspect of the edit.  Do not
repeat the
    same aspect across actions.
  - Each action should be self-contained.
  - Ground every item in the prior reasoning context where relevant.
  - It is a must to follow the output format exactly.
```

Output format:

```
  [Action 1] <concise str>
  [Action 2] <concise str>
  ...
  [Action N] <concise str>

Now, generate the action items for this instruction:
```

Prompt 11: Tailored Action Generation

```
You are an expert image editor specializing in execution planning.
Given images, an editing instruction, and reasoning context from
prior analysis, your sole job is to produce exactly N concrete action
items that together fully specify how to perform the edit.

Unlike fixed templates, you must identify the N most important and
distinct execution aspects for THIS specific edit, then write one
precise, directly executable action per aspect.  Name each aspect
concisely (2-4 words) and use that name as the bracket label.

What makes a good set of aspects:
  - Together they must fully cover the edit end-to-end with zero
overlap.
  - Each aspect targets a dimension that genuinely matters for this
edit (e.g.,
    the core transformation, style or medium consistency,
identity/structure
    preservation, lighting or color match, background continuity,
realism
    constraints -- choose whichever N are most relevant).
  - An aspect that does not add meaningful, non-obvious guidance
should be
    replaced by one that does.

Example (Instruction:  "Convert the young girl to an old lady in this
oil painting"):
  [Age Realism] Transform the young girl into an elderly woman by
adding deep
                facial wrinkles, sagging skin along the jaw and neck,
white hair,
                and age spots on all exposed skin.
  [Painting Style] Preserve the oil-painting style, color palette,
and loose
                   brushstroke character of the original.
  [Structural Preservation] Keep the subject's pose, clothing, and
the entire
                             background unchanged.

Rules:
  - Each action must be concise:  1 sentence only (25 words max per
action).
  - No redundancy -- each aspect must contribute something distinct.
  - Ground every action in the actual image content and prior
reasoning context.
  - The bracket label must be the real aspect name (2-4 words), not a
numbered
    placeholder.
  - It is a must to follow the output format exactly.

Output format:
  [<Aspect Name>] <concise str>
  [<Aspect Name>] <concise str>
  ...  (repeated N times)

Now, generate the action items for this instruction:
```

Prompt 12: Structured Action Generation (4 Fixed Roles)

```
You are an expert image editor specializing in execution planning.
Given images, an editing instruction, and reasoning context from
prior analysis, your sole job is to produce exactly 4 concrete action
item(s) that together fully specify how to perform the edit.

Each action item must be focused, non-redundant, and directly
executable:
  1. [High-Level Edit]: Provide a single, self-contained directive that
captures the full intent of the edit. This should be holistic,
covering all needed changes in one crisp statement, leveraging all
prior reasoning context.
  2. [Fine-Grained Details]: Specify the exact low-level visual changes
needed to execute the edit with high quality. Be precise: describe
textures, proportions, colors, transitions, and subtle adjustments
required to make the result look realistic and polished.
  3. [Identity Preservation]: List the objects, regions, and visual
properties that must remain completely unchanged. State them as
explicit keep directives (e.g. 'Keep the subject's face, hair color,
and background lighting intact').
  4. [Execution Guardrails]: List the concrete failure modes and quality
risks specific to this edit. State each as an explicit keep or
maintain directive (e.g. 'Keep edges clean without halo artifacts').
Be specific to this edit, not generic.

Rules:
  - Each action item must be concise. 1 sentence only. (20 words
per action)
  - Do not repeat information across action items.
  - Each action should be self-contained.
  - Ground every item in the prior reasoning context where relevant.
  - It is a must to follow the output format exactly.

Output format:
  [Action 1 - High-Level Edit] <concise str>
  [Action 2 - Fine-Grained Details] <concise str>
  [Action 3 - Identity Preservation] <concise str>
  [Action 4 - Execution Guardrails] <concise str>

Now, generate the action items for this instruction:
```

D Experiments: Additional Details

D.1 Benchmark Complexity Profiles

Figure 12 shows the reasoning-complexity distribution across all five evaluation benchmarks used in our experiments, as assigned by our automated annotator. The profiles reveal substantial variation in difficulty that directly shapes how different methods are expected to rank. RISE and KRIS are heavily skewed toward harder instructions, with 56% and 35% Heavy respectively, consistent with their design focus on spatial reasoning and semantic disambiguation, making them the most demanding settings for any model that skips structured thinking. GEdit and ImgEdit present a more balanced profile dominated by Moderate samples ($\sim$61% and 49%), reflecting general-purpose editing instructions that span the full difficulty range. PICA sits in between, with a Moderate-dominant distribution (58%) and a meaningful Heavy fraction (21%). These profiles explain why improvements due to PRISM are most pronounced on KRIS and RISE: the higher the concentration of Heavy instructions, the greater the benefit of activating the correct reasoning lens before executing the edit.

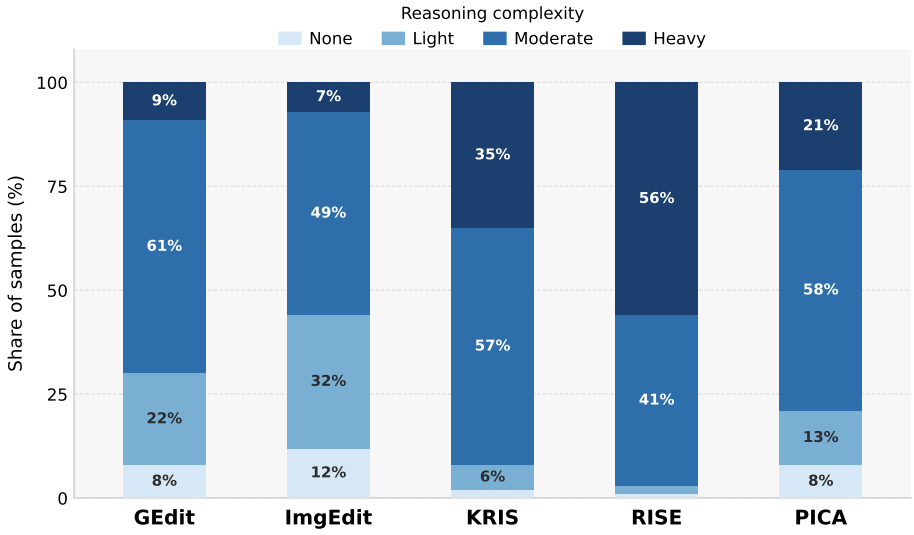


Figure 12: **Reasoning-complexity distribution across five evaluation benchmarks.** Stacked bars show the proportion of samples assigned each complexity level by our automated annotator. RISE and KRIS are heavily skewed toward harder instructions (56% and 35% *Heavy*, respectively), consistent with their focus on spatial reasoning and semantic disambiguation. ImgEdit contains the largest share of lighter instructions, reflecting its broader scope and lower average edit complexity.

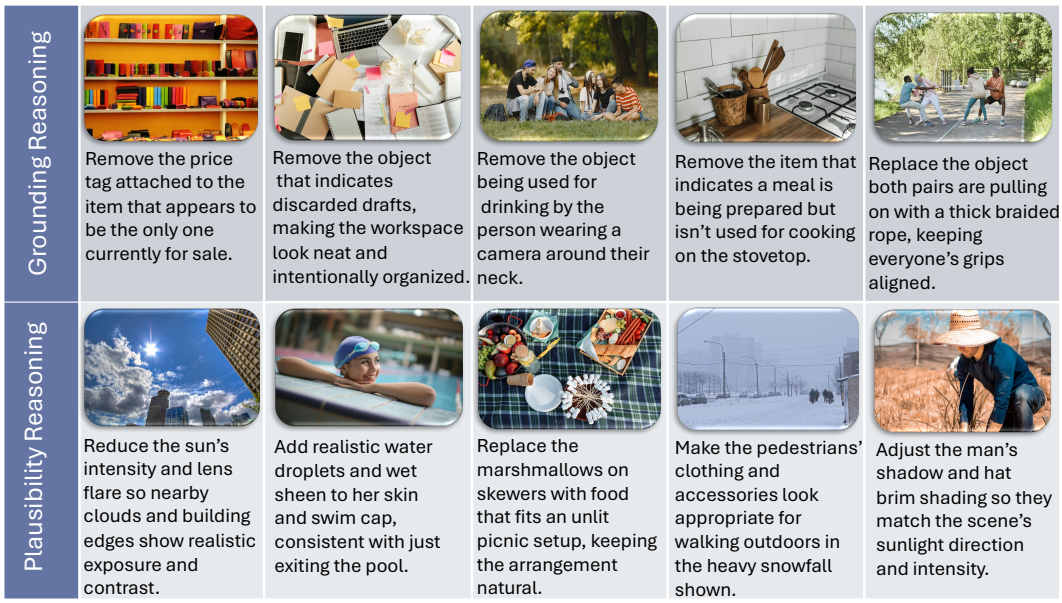


Figure 13: **Samples for the Grounding and Plausibility subsets.**

D.2 Aspect-Stratified Evaluation Subsets

The stratified evaluation subsets used in Table 3 are designed to isolate the contribution of each reasoning lens to overall edit quality. We require subsets that each concentrate on a single aspect: (i) logical reasoning and disambiguation, (ii) spatial grounding and localization, and (iii) physical plausibility.

For logical reasoning and disambiguation we draw samples directly from the KRIS benchmark (Wu et al., 2025c), which provides semantically ambiguous edit instructions that demand multi-step

inference to resolve intent. No sufficiently focused public benchmark exists for the remaining two aspects, so we construct two purpose-built evaluation subsets.

Image retrieval. We retrieve candidate images from Pexels,[1] a large-scale creative-commons photography repository, using aspect-specific search queries listed in Table 8. Each retrieved image is then paired with an edit instruction generated by GPT-5.2, prompted to produce an instruction that (a) targets the focal aspect without making it lexically explicit, and (b) cannot be executed correctly without activating the corresponding reasoning lens. This design ensures that a model that skips structured reasoning will either select the wrong edit target or produce a physically implausible result. We collect 200 image–instruction pairs per aspect.

Qualitative illustration. Representative examples are shown in Figure 13. As the figure illustrates, instructions are intentionally indirect. For example, a grounding-subset instruction may ask the model to "apply the change to whoever is looking directly at the camera" rather than specifying a region, forcing explicit subject localization before any edit can be attempted. Similarly, a plausibility-subset instruction may request a material substitution that must respect structural load-bearing properties, requiring the model to reason about physical feasibility before committing to an edit.

Table 8: **Pexels search queries for the grounding and plausibility evaluation subsets.** Two to three representative queries are shown per category; the full set comprises 40 grounding queries across 4 categories and 47 plausibility queries across 6 categories.

Aspect	Category	Representative queries
Grounding	Indoor multi-object	cluttered living room furniture; kitchen counter utensils; office desk laptop books
	Outdoor multi-object	outdoor café terrace umbrellas; street market vendors; farmer market produce display
	Foreground vs. background	portrait crowd blurred background; single tree forest stand-out; fountain plaza people
	Group / relational	group people different actions park; sports team formation; family gathering outdoor
Plausibility	Time-of-day / lighting	golden hour long shadows portrait; noon harsh sunlight urban; night city lights dark sky
	Seasonal / weather	winter snowy outdoor street; summer sunny hot beach; rainy day wet street umbrella
	Shadow & lighting physics	directional sunlight outdoor shadow; window light indoor pattern; tree casting long shadow
	Food & meal context	breakfast table morning coffee; dinner candles restaurant; outdoor picnic midday spread
	People in environment	person winter coat outdoor snowy; hiker mountain trail gear; swimmer pool summer sunny
	Architectural consistency	mediterranean villa sunny courtyard; nordic cabin winter forest; tropical resort palm trees

D.3 Variance-Guided Window Selection for Flow-GRPO

The image-generation stream of PRISM adopts Flow-GRPO (Liu et al., 2025a), which enables policy-gradient optimization of flow-matching models by converting the deterministic generation ODE into an equivalent SDE. A natural extension, MixGRPO (Li et al., 2025b), improves sample efficiency by confining this stochastic perturbation to a *sliding window* over the denoising trajectory rather than applying it uniformly across all steps. We build on this direction but ask a more precise question: *which windows carry the most reward-relevant information?*

Reward-variance analysis. To answer this, we conduct a controlled per-window variance study. We run BAGEL (Deng et al., 2025) in inference mode over 200 samples, generating $G{=}8$ outputs

[1] https://www.pexels.com

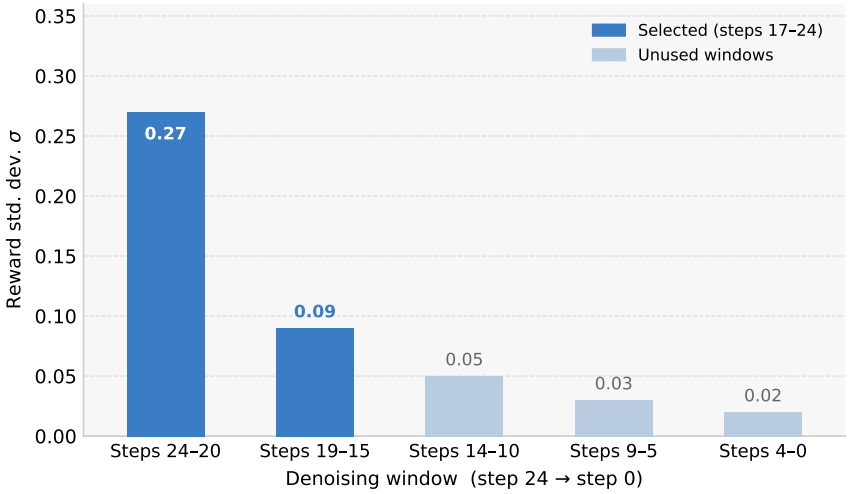


Figure 14: **Reward standard deviation per denoising window.** For each of the five non-overlapping windows spanning the 25-step denoising trajectory, we enable SDE noise injection only within that window and measure the standard deviation of rewards across the generation group (G=8), averaged over 200 samples and five independent runs. Variance is heavily concentrated in the early high-noise steps (20–24), motivating our choice to restrict Flow-GRPO training to steps 17–24 (shaded region).

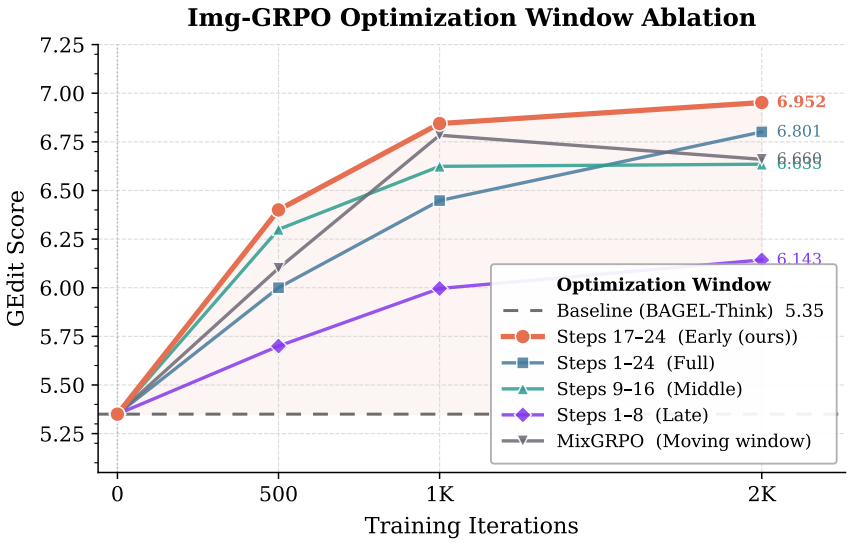


Figure 15: **Img-GRPO optimization window ablation on GEdit.** Each curve corresponds to a fixed denoising-step window used during Flow-GRPO training; the dashed line is the frozen BAGEL-Think baseline. Optimizing only the early denoising steps (17–24) yields the fastest convergence and highest final score, confirming that the generation reward signal is concentrated in the early diffusion stages.

per sample. For each of the five non-overlapping windows spanning the 25-step denoising trajectory (steps 24–20, 19–15, 14–10, 9–5, and 4–0), we enable SDE noise injection *exclusively within that window* while keeping all remaining steps deterministic (ODE), thereby isolating each window's individual contribution to reward variation. We measure the standard deviation of rewards across the G outputs within each group and average this per-window statistic over all 200 samples and five independent runs.

The results are shown in Figure 14. Reward variance is overwhelmingly concentrated in the earliest denoising steps: window 24–20 (the highest-noise regime, where global structure and composition

are determined) attains σ=0.27, an order of magnitude larger than the final refinement window (steps 4–0, σ=0.02). This monotonic decay confirms that early windows are the primary locus of reward-relevant variation, while the later low-noise steps contribute negligible gradient signal.

Early-window training. Motivated by this analysis, we fix the SDE window to steps 24–17, the region of highest reward variance, and revert all remaining steps to the deterministic ODE, eliminating the overhead of MixGRPO's progressive window schedule. As Figure 15 shows, this targeted strategy achieves both faster convergence and higher final performance than (i) full-trajectory Flow-GRPO, which dilutes optimization signal uniformly across all 25 steps, and (ii) MixGRPO's sliding window, which continues to invest in low-variance windows that yield no measurable gain. Crucially, this improvement is not merely empirical: it follows directly from the variance analysis, which identifies steps 24–17 as the only region where the model's denoising policy meaningfully influences the final reward.

D.4 Automated Labeling Protocol

Complexity labels $\tilde{c}$ are collected in a single offline labeling pass over the training set using InternVL3.5-241B-A28B (Wang et al., 2025c), served via vLLM + Ray. This is analogous to offline preference dataset collection before RLHF training (Ouyang et al., 2022): a one-time cost that eliminates the need to invoke a large VLM for Stage 1 during online training. The labels serve twice: as SFT targets for the Stage-1 warm-start, and as the reference for the Stage-1 reward R_1 (Eq. (8)).

Complexity labeling. InternVL3.5 is prompted to assign a difficulty label $\tilde{c} \in \{\texttt{light}, \texttt{moderate}, \texttt{heavy}\}$ to each (x, I) pair based on the number and nature of reasoning steps required, following the same four-level taxonomy defined in Appendix B.5 (with NONE merged into LIGHT for training).

D.5 GEdit Robust Evaluation Protocol

The original GEdit benchmark (Liu et al., 2025b) evaluates image edits along two dimensions: *instruction adherence* and *perceptual quality*. While effective for measuring whether an edit was executed, this two-aspect design harbours a systematic blind spot: models that apply an aggressively high classifier-free guidance (CFG) scale can inflate both adherence and quality scores while silently destroying content unrelated to the edit, a failure mode that a two-aspect score cannot penalise.

Our extensions. We introduce two modifications to obtain a more robust evaluation signal.

(1) Identity preservation as an explicit third aspect. We add a dedicated *identity preservation* score that quantifies how faithfully the unedited regions of the image are retained. This directly targets the CFG-exploitation artefact: a model that achieves high adherence by over-editing the background is now penalised, whereas it would be rewarded under the original protocol.

(2) Stronger VLM judge. We replace the original judge model with GPT-5.2, which provides more reliable and fine-grained assessments across all three aspects.

Aggregation. Given per-aspect scores s_{adh}, s_{qual}, and s_{pres}, we aggregate them via the geometric mean:

$$\text{GEdit-R} \;=\; \left(s_{\text{adh}} \cdot s_{\text{qual}} \cdot s_{\text{pres}}\right)^{1/3}. \tag{9}$$

The geometric mean preserves GEdit's original design principle that all aspects are complementary: an outlier-high score on one dimension cannot compensate for a deficiency in another. A model must perform well across all three to achieve a high GEdit-R score.

D.6 REVISE-50K as a Training Foundation

Data quality and diversity are critical to the effectiveness of the PRISM framework: the GRPO reward signal can only guide the model toward structured reasoning if the training distribution exposes it to the full spectrum of reasoning challenges encountered at test time. To isolate the contribution of REVISE-50K, we train PRISM under identical hyperparameters on two data sources and evaluate

Table 9: **REVISE-50K importance.** PRISM trained on REVISE-50K versus the same framework trained on OmniEdit (Wei et al., 2025). Best in **bold**.

Method	Data	GEdit	KRIS
BAGEL-Think	—	5.06	60.18
PRISM	OmniEdit	**7.08**	66.84
	REVISE-50K	7.00	**73.21**

on GEdit and KRIS (Table 9). When trained on OmniEdit (Wei et al., 2025), a balanced corpus dominated by direct, unambiguous instructions, PRISM already improves over BAGEL-Think by **+2.02** on GEdit and **+6.66** points on KRIS, confirming that the framework itself is beneficial even without a reasoning-focused training distribution. Replacing OmniEdit with REVISE-50K, whose 50K samples span 50 skills with $\sim$77% of instructions requiring non-trivial reasoning, keeps GEdit on par (7.00 vs. 7.08, within the judge's noise of about 0.1) and raises KRIS by a further **+6.4** points, to **+13.0** over BAGEL-Think. The gain is concentrated on KRIS, the most reasoning-intensive benchmark, demonstrating that REVISE-50K's deliberate coverage of indirect, multi-step instructions is the key ingredient that unlocks the full potential of adaptive lens generation and structured action generation.

D.7 Action Format Ablation

Table 10: **Action format ablation.** GPT-5.2 score of the generated actions at initialization (0) and after 1K GRPO steps; best in **bold**.

Format	Iter.	Score
Freeform	0	5.94
	1K	5.97
Structured	0	7.48
	1K	7.69
Tailored	0	6.48
	1K	**7.99**

Beyond adaptive lens generation, the structure of the generated action plan itself significantly affects edit quality. Table 10 compares three action formats, each producing exactly 3 actions, at initialization and after 1K training steps. *Freeform* actions impose no structural constraint, leaving the model with an unconstrained output space that proves difficult to optimize via GRPO: scores remain nearly flat across training, improving by only 0.03 points. *Structured* actions fix the roles to high-level edit, fine-grained details, and identity preservation; this rigid template provides a strong prior at initialization but limits the model's ability to adapt to each input, leaving modest headroom for improvement. *Tailored* actions instruct the model to first define an aspect title and purpose for each action before generating it, providing structural scaffolding while preserving per-sample flexibility. Despite a lower zero-shot starting point, tailored actions achieve the highest score of **7.99** after training, demonstrating that structure-aware yet adaptive generation is the most learnable and highest-performing configuration. Our final setup therefore uses tailored actions driven by the sample-specific reasoning lenses.