RSIGEN: RECIPROCAL SELF-IMPROVEMENT FOR AGENTIC VISUAL GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Experience from agentic visual generation can improve the native understanding (agent) and generation roles of a unified multimodal backbone, but task success alone does not determine who should learn. Generator updates can also make action supervision stale. We introduce RSIgen, a task-centric reciprocal self-improvement framework linking learning allocation, supervision refresh, and recursive curriculum construction. Dynamic Responsibility Routing (DRR) uses matched interventions on evidence, inputs, and supported actions to select recipients and construct compatible training targets. After generator learning, RSIgen refreshes agent supervision by comparing tested actions' relative utility with branch inputs fixed; retained roles construct the next curriculum. Across BAGEL, Qwen-Image, and Wan-Image-7B, RSIgen improves AgentGen-Bench Full Overall9 by 3.2–3.3 points over a three-cycle SearchGen adaptation with action costs. Wan controls support allocation gains at matched role-level training exposure, refresh gains at a matched adaptation budget, and benefits from updated curriculum sources while recipients continue learning. Blind evaluation on 200 real-user requests yields pooled tie-adjusted preferences of 70.0% for task completion and 64.9% for visual quality against the same baseline.

1 INTRODUCTION

Agentic visual generation turns an image request into decisions about evidence acquisition, input construction, generation, and repair (Chen et al., 2026a; Zhang et al., 2026b). We study the *native-role setting*: the agent and generator are the understanding and generation roles of the same unified multimodal backbone. Task success depends on both the agent's orchestration and the generator's ability to realize the resulting inputs.

A useful action and a useful training target answer different questions. Consider an architecture poster that places buildings in the wrong chronological order. If correcting the layout instruction fixes the image while the evidence and generator stay unchanged, the correction can train the agent. A validated search-assisted image of the same buildings, paired with an evidence-free request, may instead teach stable historical knowledge to the generator. The final image alone does not identify the learning recipient or supply a compatible target.

Learning by either role reshapes its partner's next learning problem. If Direct succeeds on reserved uses of newly learned knowledge while Search remains reliable, an earlier search-required label may become stale. Shared architecture does not automatically recalibrate the agent's policy to the generator's updated capabilities. This *capability–policy calibration* aims at appropriate evidence use: volatile facts and instance-specific references retain external dependencies. Better agent inputs and evidence, in turn, change the generator's next learning opportunities (Figure 1).

SearchGen already teaches the generator and then recalibrates its reasoner (Wang et al., 2026), while GenEvolve distills contrasting trajectory experience into an evolving agent (Chen et al., 2026b). Building on these advances, we study three linked decisions: *Who should learn? Which actions remain useful after an update? What should the updated roles learn next?*

We introduce **RSIgen**, a framework for reciprocal self-improvement. Dynamic Responsibility Routing (DRR) uses matched interventions on evidence, inputs, and supported actions to select learning recipients and skill subtypes and construct compatible targets.

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

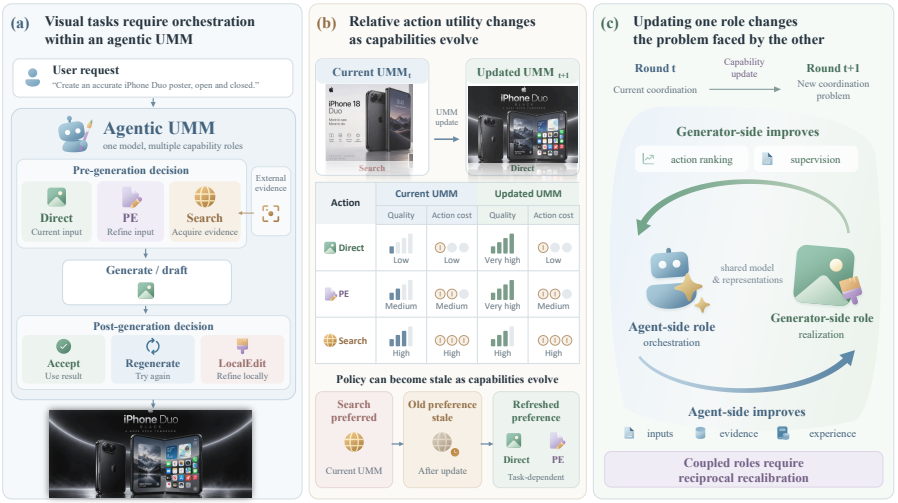

Figure 1: Reciprocal learning in agentic visual generation. (a) Agent orchestration and generator realization are roles within a unified multimodal model (UMM); PE denotes prompt enhancement. (b) Generator learning changes action utility. (c) Agent learning changes inputs and evidence for later curricula. Images and quality levels are illustrative.

After generator learning, RSIgen repeats affected action comparisons with the agent, branch inputs, evidence, evaluator, and image seeds fixed. These comparisons refresh supervision according to updated relative action utility across acquisition, input construction, stopping, and supported repair. The agent then learns with the updated generator fixed, and the retained roles construct the next curriculum.

Across BAGEL, Qwen-Image, and Wan-Image-7B, RSIgen improves AgentGen-Bench Full Overall9 by 3.2–3.3 points over SearchGen-3 + costs, our three-cycle SearchGen adaptation with action costs. Blind evaluation on 200 real-user requests yields pooled tie-adjusted preferences of 70.0% for task completion and 64.9% for visual quality against this baseline.

Our contributions, with mechanism studies on Wan, are:

- **Diagnosis-guided allocation:** distinguish execution choices from learning-recipient allocation and construct compatible targets, with gains at matched role-level training exposure.
- **Supervision refresh:** re-evaluate tested pre- and post-generation actions after generator learning, improving adaptation over current-generator on-policy learning at a shared total budget.
- **Reciprocal curriculum analysis:** establish the benefit of updated curriculum sources through controls that freeze those sources while keeping recipient learning active.

2 Related Work

Agentic visual generation and experience adaptation. Retrieval-augmented generation conditions synthesis on external evidence (Chen et al., 2022; Rombach et al., 2022); Gen-Searcher, ToolArtist, Qwen-Image-Agent, and Unify-Agent organize or learn its acquisition and use (Feng et al., 2026; Zhao et al., 2026; Zhang et al., 2026b; Chen et al., 2026a). SearchGen trains the generator with Diffusion-DPO, then adapts the reasoner with the updated generator (Wang et al., 2026). GenEvolve distills guidance extracted from contrasting trajectories (Chen et al., 2026b). RSIgen connects generator-dependent adaptation to diagnosis-guided allocation: matched interventions construct compatible targets for each role, and post-update re-probing refreshes acquisition and repair supervision. SearchGen-3 + costs extends the original single DPO–RFT pass to three cycles for our comparison.

Reciprocal multimodal learning. HermesFlow, UniRL, UniCorn, SUDER, and Ask, Solve, Generate couple understanding and generation through preferences, tasks, rewards, or alternating updates (Yang et al., 2025; Mao et al., 2025; Han et al., 2026a; Hong et al., 2025; Thawkar et al., 2026). Internal Gap revisits underutilized samples as capabilities improve (Han et al., 2026b),

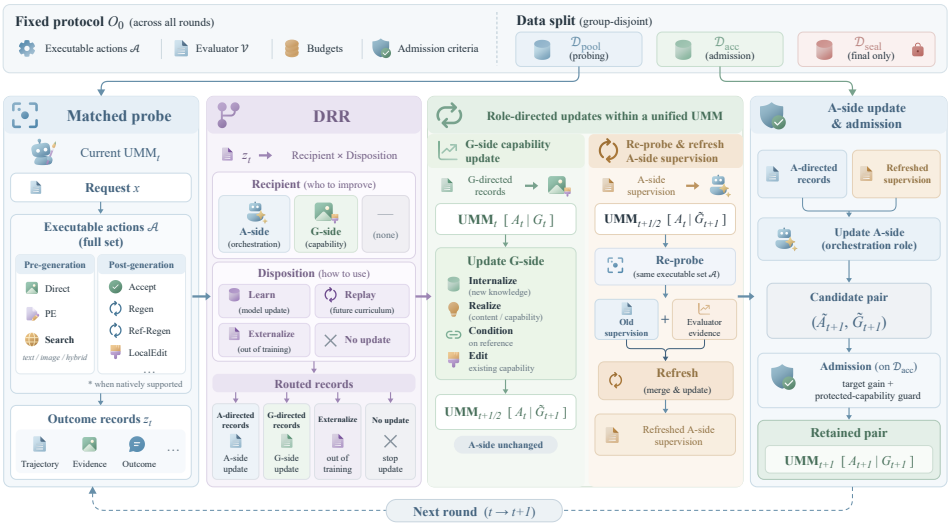

Figure 2: RSIgen's learning procedure. DRR compiles matched probes into typed curricula. Generator learning precedes re-probing and agent learning; retained roles construct the next curriculum. Pool, admission, and sealed evaluation data have separate roles. UMM denotes the unified model's understanding/orchestration and generation roles. Evaluator feedback and capability memory support training.

while Dreaming in Flow combines flow feedback with replay grounding (Hao et al., 2026). RSIgen couples the roles through the learning use of experience and the changing utility of agent actions. Its construction-source controls test whether updated peers improve later curricula while recipient learning remains active. Appendix I gives the full discussion, including optimization and feedback; Appendix J compares the mechanisms.

3 Task-Centric Reciprocal Self-Improvement

RSIgen links learning-recipient allocation, supervision refresh, and recursive curriculum construction under a fixed outer protocol (Figure 2).

3.1 Task, capability roles, and outcome rewards

At round t, $S_t = (A_t, G_t, M_t)$ comprises the native understanding/agent and generation roles of a unified multimodal backbone, plus outer-loop capability memory. Each update changes its recipient while holding the other role's effective function fixed. Parameters follow the backbone; Wan uses full-parameter updates to the trained role (Appendix B.3). For request x,

$$x \xrightarrow{A_t:\text{Think}} \text{decision} \xrightarrow{A_t:\text{Action}} (p, e) \xrightarrow{G_t} I \xrightarrow{\mathcal{E}_0} y. \tag{1}$$

The agent supplies prompt p and optional evidence/reference e; $\mathcal{E}_0$ scores the completed image. The agent observes images and tool results, but evaluator scores and M_t remain training-side resources.

Pre-generation actions are Direct, PromptEnhance, TextSearch, ImageSearch, and HybridSearch. Post-generation actions are accept, regenerate, reference-conditioned regenerate, and LocalEdit where natively supported. Conditioning and editing actions require native support.

Both RL stages use completed-image outcomes. With request-applicable checks $\mathcal{J}(x)$ and normalized weights w_j,

$$R_G(I, x, e) = \sum_{j \in \mathcal{J}(x)} w_j v_j(I, x, e) - \lambda_{\text{viol}} c_{\text{viol}}(I, x, e), \tag{2}$$

$$R_A(\tau, x) = R_G(I_{\text{final}}, x, e_{\text{final}}) - \lambda_s C_s(\tau) - \lambda_g C_g(\tau) - \lambda_l C_l(\tau). \tag{3}$$

Checks v_j evaluate image requirements; c_{viol} penalizes predefined violations. Factual/reference targets are anchored independently of agent-selected evidence. C_s counts searches, C_g generations/edits,



Figure 3: Schematic routing decisions. (a) Holding evidence and generator fixed, a corrected layout repairs a chronology error and yields an agent-construction candidate. (b) A module mismatch under checked input yields a generator-realization candidate when a compatible target is available. (c) Learning, externalization, and deferral depend on the dependency's properties and branch outcomes.

and C_l horizon-normalized decisions. Probe scores construct curricula; reference/KL regularization belongs to optimization.

The objective is held-out quality $Q(A, G)$ under training/inference budgets and a protected-capability guard. Protocol Ω_0 fixes actions, compilation, update stages, rewards, and admission. Pool $\mathcal{D}_{\text{pool}}$, admission $\mathcal{D}_{\text{acc}}$, and sealed evaluation $\mathcal{D}_{\text{seal}}$ remain source/request/artifact-group disjoint, including derived curricula and replay. Terminal reward alone cannot allocate learning; matched probes guide this decision.

3.2 Who should learn: diagnosis and curriculum compilation

Budgeted probes cover anchors, changed capability regions, uncertain comparisons, and under-covered requests. Matched interventions compare Direct, prompt enhancement, search modalities, Oracle evidence, and supported repairs:

$$q_t(x, a) = \frac{1}{K} \sum_{k=1}^{K} s(I_t^{(k)}(x, a), x), \tag{4}$$

where s scores the completed image. Comparisons fix the request, peer versions, evaluator, unaffected inputs, and image seeds. Each branch's retrieval artifacts are frozen for replay. Empty searches retain their outcomes and costs; unexecuted branches remain unmeasured.

Input checks validate request fidelity, evidence coverage, entity/reference binding, and interface compatibility. Oracle evidence is validated independently of image reward and serves diagnosis, not executable agent labels. Failed checks trigger reconstruction or no-update.

Dynamic Responsibility Routing (DRR) maps the versioned probe record z_t to a recipient, skill subtype, and disposition: learn, replay, externalize, or no-update (Figure 3; Table 5). Oracle repair of inadequate retrieval suggests agent acquisition; verified prompt corrections suggest agent construction. Residual failure under valid, requirement-complete inputs suggests generator realization. These are operational learning hypotheses, not unique causal identifications.

Successful search also offers a learning use: Direct failure on stable reusable knowledge with validated Search success and no material Oracle gain suggests generator internalization. Volatile facts and instance-specific references retain external dependencies. Refresh tests whether learning supports no-search supervision.

For opportunity $o = (x, z_t)$, let $d_{o,r}$ be a compatible input–target pair for recipient/subtype $r \in \mathcal{R}(o)$. DRR selects $\rho(o) \in \mathcal{R}(o)$ or no-update and compiles $\mathcal{C}_r = \{d_{o,r} : \rho(o) = r\}$. Generator SFT requires a validated admissible image or successful generation target. Pool expansion independently validates each added example's inputs, subtype, and target.

In the poster example, agent construction pairs the request, evidence, and original input with the executed correction. Internalization pairs an evidence-free student request with a validated assisted image:

$$(p^{\mathrm{ctx}}, e^*) \longrightarrow I^*, \qquad G : (p^{\mathrm{D}}, \varnothing) \longrightarrow I^*. \tag{5}$$

The student preserves the request and entity identity but excludes retrieved facts and descriptions. Reference-following records retain their conditions. Appendix B.2 gives all input–target forms. Generator curricula cover internalization, realization, reference following, and supported editing. Compatible flow generators use SFT/conditional flow matching and then Flow-GRPO for open-ended verifiable skills; internalization uses SFT only.

3.3 Which actions remain useful: post-generator refresh

Post-generator probes cover affected regions, uncertain comparisons, and anchors with the agent, branch inputs, evidence, source images, continuation, evaluator, and seeds fixed. On common tested executable branches $\mathcal{A}_x^P$, empirical terminal values define a tolerance-aware preferred set:

$$\begin{aligned} \widehat{V}_t(x, a) &= \frac{1}{K} \sum_{k=1}^{K} R_A(\tau_{t,k}^a, x), \\ \widehat{\mathcal{B}}_t^{\epsilon}(x) &= \left\{ a \in \mathcal{A}_x^P : \widehat{V}_t(x, a) \geq \max_{b \in \mathcal{A}_x^P} \widehat{V}_t(x, b) - \epsilon \right\}. \end{aligned} \tag{6}$$

Here $\tau_{t,k}^a$ forces branch a with the common continuation. Replacing G_t with $\widetilde{G}_{t+1}$ under matched conditions defines $\widehat{\mathcal{B}}_{t+1/2}^{\epsilon}$. Old supervision is *stale* when its action becomes invalid or materially inferior under the updated generator. A preferred-set change need not invalidate a near-optimal old label; ambiguous comparisons can remain unlabeled. Rankings depend on relative action improvement (Appendix B.4).

An internalization-based no-search target additionally requires evidence-free success on reserved new uses of the same knowledge unit, a small remaining Search–Direct gap, and preserved assisted success. Failure of both branches never licenses this target. Refreshed supervision covers search, query/reference selection, prompt construction, stopping, and supported repair. The agent learns from validated executed trajectories through SFT, then GRPO with the updated generator fixed.

3.4 What the roles learn next: recursive construction and admission

Writing $z_{t+1/2}$ for post-generator probes, an update attempt is

$$\begin{aligned} \widetilde{G}_{t+1} &= \mathcal{U}_G(G_t; \mathcal{C}_{A \to G}(z_t, M_t, \mathcal{D}_{\mathrm{pool}})), \\ \widetilde{A}_{t+1} &= \mathcal{U}_A(A_t; \mathcal{C}_{G \to A}(z_{t+1/2}, M_t, \mathcal{D}_{\mathrm{pool}}), \widetilde{G}_{t+1}). \end{aligned} \tag{7}$$

Here $G \to A$ denotes supervision compiled from generator behavior. Memory stores versioned outcomes for diagnosis, expansion, and replay. After admission, retained roles construct new experience through the fixed collector $\mathcal{P}_{\Omega_0}$:

$$z_{t+1} = \mathcal{P}_{\Omega_0}(A_{t+1}, G_{t+1}, M_{t+1}; \mathcal{D}_{\mathrm{pool}}) \longrightarrow \mathcal{C}_{A \to G}^{t+1} \longrightarrow \widetilde{G}_{t+2}. \tag{8}$$

These *construction peers* supply updated evidence, inputs, and outcomes for subsequent curricula. Source controls freeze them while recipient learning and current-partner RL remain active, testing their combined contribution to content, target validity/yield, and supervision freshness (Appendix C).

Admission compares target quality and protected generation/understanding capabilities with the retained state on identical held-out subsets, separate from curricula and rewards. Rejected attempts retain the state and consume their budget. Appendix D.6 specifies this local check; final evaluation compares retained endpoints with initialization and matched controls.

4 Experimental Setup

We evaluate BAGEL, Qwen-Image, and Wan-Image-7B on the 751-request AgentGen-Bench using its released Overall9 metric (Wang et al., 2026). Controls include agent-only adaptation, SearchGen-3 + costs, joint outcome RL, and uniform reciprocal learning. Within each backbone, comparisons share initialization, tools, resource allowances, corresponding seeds, and three update attempts.

Human evaluation compares RSIgen with SearchGen-3 + costs on 200 shared real-user requests, separate from the 200-request Test-mini diagnostic. Three independent blind ratings per pair assess task completion and visual quality. Wan mechanism studies test allocation at matched role exposure, refresh at matched adaptation budget, and changing curriculum sources at fixed training sizes. Appendices D–G provide metric definitions, training and evaluation settings, statistical units, and the complete human-evaluation protocol.

5 Results and Analysis

Table 1: Complete AgentGen-Bench results (0–100). NS/SI: NoSearch/Search-Intensive; Full: Full-slice Overall9. Colors mark best and second-best within each group. Public rows have model-dependent coverage; matched training comparisons appear in Appendix Table 8.

	Checklist		Visual Ref.		Image Quality		Overall9		
Model	NS	SI	NS	SI	NS	SI	NS	SI	Full
Commercial Models									
GPT-Image-2	**84.8**	**74.4**	**70.9**	**70.4**	**73.4**	**77.4**	**78.9**	**75.5**	**76.0**
Grok-Imagine-2.0-Low	83.0	73.1	68.7	68.3	72.5	76.9	76.9	74.9	75.2
Qwen-Image-3-Max	79.1	65.8	64.0	59.6	70.3	71.1	73.7	67.6	68.4
Nano Banana Pro	78.4	60.7	63.0	55.4	70.7	68.1	73.3	63.6	65.0
Qwen-Image-2	76.2	49.9	60.3	42.0	70.2	62.5	71.1	54.1	56.4
Generation Only									
Mage-Flow	**64.2**	23.0	**47.1**	16.2	**64.7**	**38.7**	**61.2**	**28.4**	**32.8**
Flux.2-Klein-9B	63.7	**23.8**	41.1	**16.7**	63.2	36.1	58.5	26.9	31.1
Flux.2-Klein-4B	55.8	20.1	35.3	12.7	59.7	33.0	52.3	23.9	27.7
Unified MLLMs									
SenseNova-U1	68.1	29.3	51.1	21.0	64.3	39.5	62.8	31.6	35.7
OmniGen2	49.1	16.4	30.6	9.5	56.4	29.2	47.4	20.4	24.0
Show-o2	32.7	8.1	19.9	3.4	47.2	30.2	34.4	17.5	19.8
BAGEL	51.5	18.7	32.1	14.2	56.2	30.4	49.1	22.2	25.8
BAGEL + RSIgen	63.4	36.4	45.6	27.4	61.4	43.3	60.4	36.8	39.9
Qwen-Image	74.9	25.0	60.1	17.7	69.2	37.9	70.0	28.6	34.1
Qwen-Image + RSIgen	**77.2**	**47.3**	**63.4**	**37.4**	**70.7**	**52.3**	**73.2**	**46.8**	**50.3**
Wan-Image-7B	65.3	22.3	49.3	16.3	65.1	37.2	62.3	27.4	32.1
Wan-Image-7B + RSIgen	70.4	44.3	55.4	35.4	67.2	50.2	67.3	44.4	47.5

5.1 Does the training method improve task quality?

RSIgen reaches AgentGen Full scores of 39.9, 50.3, and 47.5 on BAGEL, Qwen-Image, and Wan-Image-7B (Table 1). Matched training controls (Appendix Table 8) show gains over SearchGen-3 + costs of 3.2, 3.2, and 3.3 points; gains over uniform reciprocal learning are 3.7, 3.6, and 3.7. The improvement holds across all three backbone implementations.

Gains over SearchGen-3 + costs are larger on SI (3.4–3.6 points) than NS (0.9–2.2). RSIgen also exceeds joint outcome RL in each backbone.

5.2 DO HUMAN JUDGMENTS SUPPORT THE GAINS ON REAL-USER REQUESTS?

RSIgen receives more wins than losses on both dimensions for every backbone (Table 2). Tie-adjusted preferences range from 68.5% to 71.4% for task completion and 63.5% to 66.1% for visual quality. Pooled scores are 70.0% and 64.9%, respectively, supporting gains on the evaluated real-user requests.

Table 2: Blind human evaluation of RSIgen versus SearchGen-3 + costs on 200 shared real-user requests. W/T/L denotes RSIgen wins, ties, and losses; preference (Pref.) credits ties as half a win. Each backbone has 600 judgments per dimension; Overall pools 1,800, with no Unknown judgments. Appendix G gives the full breakdown.

		Task completion		Visual quality	
Backbone	Requests	W / T / L	Pref.↑	W / T / L	Pref.↑
BAGEL	200	312 / 198 / 90	68.5%	264 / 234 / 102	63.5%
Qwen-Image	200	324 / 194 / 82	70.2%	276 / 228 / 96	65.0%
Wan-Image-7B	200	336 / 185 / 79	**71.4%**	288 / 217 / 95	**66.1%**
Overall	200	972 / 577 / 251	**70.0%**	828 / 679 / 293	**64.9%**

Appendix G gives full judgment counts; Figure 6 shows five qualitative comparisons with complete prompts.

5.3 DOES DIAGNOSTIC CURRICULUM CONSTRUCTION IMPROVE LEARNING?

The Wan ablations (Table 11) test the updated roles and curriculum package. Agent-only and generator-only training reach 40.2 and 41.7, versus RSIgen's 47.5. Uniform reciprocal learning and failure-locus routing reach 43.8 and 44.1. Jointly updating both roles with diagnostic curricula improves quality over these recipes.

Allocation under matched role exposure. The allocation study matches role-level exposure across a 4,000-opportunity pool (Table 3): 1,600 agent items with 3.2M SFT/2.4M RL tokens, and 2,000 generator items with 4,000 SFT/7,200 RL image exposures. DRR reaches Full 47.4, gaining 1.4 over compatible-uniform (46.0) and 1.1 over failure-locus (46.3). Search/generation-edit calls fall from 1.29/1.37 to 1.21/1.32 relative to compatible-uniform. Thus the allocation gain persists at matched role-level training volume.

Table 3: Wan allocation comparison with matched role-level training exposure. Δ is relative to compatible-uniform. These results are separate from the curriculum-package ablations.

Strategy	Full ↑	Δ Full	Searches ↓	Gen./edits ↓
Compatible-uniform	46.0	0.0	1.29	1.37
Failure-locus	46.3	+0.3	1.27	1.36
DRR	**47.4**	**+1.4**	**1.21**	**1.32**

All arms use 4,000 opportunities. Agent: 1,600 training entries, 3.20M SFT tokens and 2.40M RL tokens. Generator: 2,000 entries, 4,000 SFT and 7,200 RL image presentations. Calls are per request.

Removing post-generator re-probing lowers Full from 47.5 to 45.0, a 2.5-point loss over the complete training trajectory. Removing capability memory or pool expansion lowers Full to 45.9 or 45.3. These component ablations complement the fixed-checkpoint continuation study.

What quality–tool-use tradeoff does the cost term select? The no-cost variant reaches 48.0 versus RSIgen's 47.5, with searches rising from 1.16 to 1.86 and generations/edits from 1.30 to 1.56 per request. The cost term trades 0.5 quality points for fewer inference calls. RSIgen also achieves higher quality with slightly fewer calls than SearchGen-3 + costs across all three backbones (Appendix D.5).

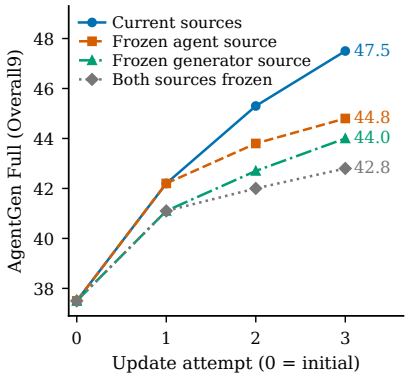

Figure 4: Wan curriculum-source controls on AgentGen-Bench Full. Recipients keep learning with current-partner RL while the named sources are frozen; exact scores appear in Table 15.

Table 4: Wan Stage-2 refresh at a matched total budget of 100, including preparation, rollout, and optimization. On-policy uses the current generator; Δ O9 is relative to this arm. Cost units are percentages of the common budget.

Metric	Stale	Current-G on-policy	Refresh
Quality and action selection			
Mini O9 ↑	47.2	49.1	**50.0**
Δ O9	-1.9	0.0	**+0.9**
Pre regret	5.40	4.00	**3.50**
Post regret	4.60	3.70	**3.20**
Adaptation cost			
Preparation	5	0	20
Rollout	65	74	55
Optimization	30	26	25
Total	100	100	100

5.4 Do action values change, and does refresh help adaptation?

Wan probes hold prompts, queries, evidence, seeds, and action packages fixed across generator versions; continuation fixes the updated rollout generator.

Generator updates change the relative utility of actions. Table 16A reports empirical preference shifts in 22.2% of pre-generation and 20.8% of post-generation decisions; old labels are stale in 17.0% and 15.7%. A preference shift changes the preferred actions, whereas staleness identifies an old supervision label that has become inappropriate.

Search-to-Direct accounts for 51.9% of pre-generation shifts; Regenerate-to-LocalEdit contributes 39.2% of post-generation shifts. Appendix Table 22 gives the complete transitions in retrieval and repair utility.

Refreshed labels improve adaptation to the updated generator. Stale-label and refresh arms share agent initialization, requests, rollout generator, and realized training compute. Refresh reduces pre/post regret from 5.37/4.60 to 3.30/3.00. Three-stage Mini Overall9 rises from 47.37 to 50.20, with paired request-cluster 95% interval [1.8, 3.9] for the gain. Action agreement and eligible-state LocalEdit precision/recall also improve (Appendix Table 17).

Refresh also improves on ordinary current-generator adaptation. At the matched Stage-2 budget (Table 4), refresh reaches Mini O9 50.0 versus on-policy learning's 49.1 and stale supervision's 47.2. Refresh lowers pre/post regret from 4.0/3.7 to 3.5/3.2 relative to on-policy learning. This advantage includes preparation cost within the common 100-unit budget.

5.5 Do updated peers improve the next curriculum?

Source controls retain recipient learning and current-partner RL (Figure 4). Current sources reach 47.5 on Wan, compared with 42.8 when both sources remain at initialization. Freezing only the agent gives 44.8; freezing only the generator gives 44.0. Updating both sources yields a 4.7-point advantage over dual freezing.

Current-source gains diminish across attempts (4.7, 3.1, and 2.2 points). Separation from frozen-source controls reflects changes in both curriculum content and valid-target yield.

Current construction sources improve usable curriculum yield. Across three attempts, current sources yield 5,580/7,480 valid agent/generator targets versus 4,430/6,740 under dual freezing, with 1,950 versus zero refreshed agent targets. All arms train on 5,400/6,600 items (Appendix Table 25); quality improves alongside yield and freshness at fixed training sizes.

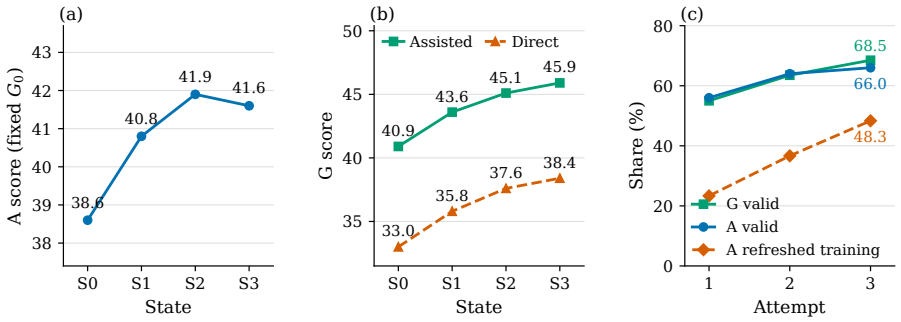

Figure 5: Wan role performance and curriculum validity. Evaluation uses 200 requests and four seeds per request/state; agent evaluation fixes G_0. Details appear in Appendix F.

Role and curriculum trajectories. Figure 5 shows agent quality with G_0 rising from 38.6 to 41.9 at S_2, then declining to 41.6. Assisted/direct generator quality increases from 40.9/33.0 to 45.9/38.4. Thus generator improvement coexists with a nonmonotonic fixed-partner agent trajectory.

Generator/agent target validity rises from 55.0%/56.0% to 68.5%/66.0%, while newly valid counts decline. Refreshed supervision grows from 23.3% to 48.3% of the fixed agent curriculum. Selection shifts toward construction and realization (Appendix F).

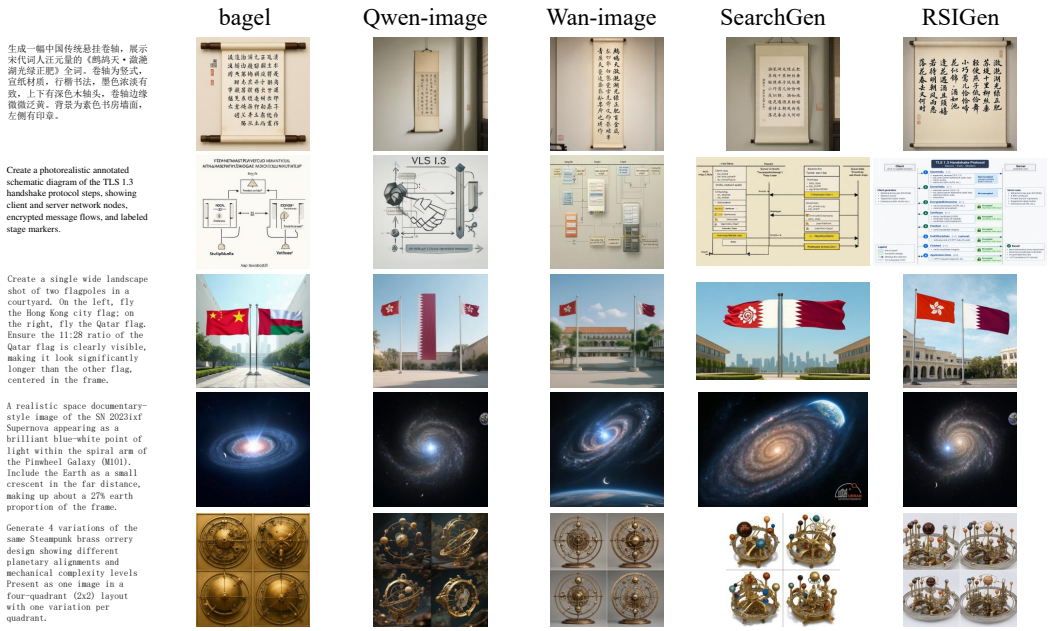

Figure 6: Five-request comparison with complete prompts at left and BAGEL, Qwen-Image, Wan-Image, SearchGen, and RSIgen outputs in columns. Table 8 reports aggregate comparisons.

6 Conclusion

RSIgen connects diagnosis-guided learning allocation, supervision refresh, and reciprocal curriculum construction. It improves AgentGen-Bench quality across three backbones, with complementary human preferences on 200 real-user requests. Wan controls show allocation and refresh gains at matched exposure and adaptation budget, while source controls connect recursive improvement to target validity and supervision freshness at fixed curriculum sizes. The role trajectories highlight the importance of adapting coordination as capabilities change. Limitations are discussed in Appendix A.

Reproducibility Statement

The supplementary material contains experimental data, metric definitions, and scripts that generate the numerical tables and plots, including the aggregate human-evaluation counts and preference calculation. Appendix B.3 specifies the role-specific training sequence and target construction; Appendix D describes the comparison methods; Appendix E defines the action comparisons and statistical estimators; and Appendix F gives the Wan role evaluations, curriculum measurements, and optimization settings. The LaTeX sources include the original qualitative outputs and showcase.

Ethics Statement

Retrieved references can contain copyrighted or identifying material; their use and redistribution require appropriate rights. Factual and identity-sensitive requests require checks beyond visual plausibility. RSIgen retains explicit external evidence for volatile information and instance-specific material. Evaluation references remain distinct from the retrieval and training inputs defined by the protocol.

AI Use Statement

Generative AI tools were used to assist with literature exploration and summarization, language editing and polishing, suggestions on manuscript organization and presentation, and the design and refinement of scientific figures. All AI-assisted outputs were critically reviewed, verified, and revised by the authors before being incorporated into the paper. The authors take full responsibility for the scientific content, claims, analyses, and presentation of the final manuscript.

References

Shuang Chen, Quanxin Shou, Hangting Chen, Yucheng Zhou, Kaituo Feng, Wenbo Hu, Yi-Fan Zhang, Yunlong Lin, Wenxuan Huang, Mingyang Song, Dasen Dai, Bolin Jiang, Manyuan Zhang, Shi-Xue Zhang, Zhengkai Jiang, Lucas Wang, Zhao Zhong, Yu Cheng, and Nanyun Peng. Unify-agent: A unified multimodal agent for world-grounded image synthesis, 2026a. URL https://arxiv.org/abs/2603.29620. arXiv:2603.29620; bibliographic record retrieved 2026-09-08.

Sixiang Chen, Zhaohu Xing, Tian Ye, Xinyu Geng, Yunlong Lin, Jianyu Lai, Xuanhua He, Fuxiang Zhai, Jialin Gao, and Lei Zhu. Genevolve: Self-evolving image generation agents via tool-orchestrated visual experience distillation, 2026b. URL https://arxiv.org/abs/2605.21605. Metadata verified against arXiv on 2026-09-19.

Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W. Cohen. Re-imagen: Retrieval-augmented text-to-image generator, 2022. URL https://arxiv.org/abs/2209.14491. arXiv:2209.14491; bibliographic record retrieved 2026-09-07.

Zixiang Chen, Yihe Deng, Huizhuo Yuan, Kaixuan Ji, and Quanquan Gu. Self-play fine-tuning converts weak language models to strong language models, 2024. URL https://arxiv.org/abs/2401.01335. arXiv:2401.01335; bibliographic record retrieved 2026-09-07.

Kaituo Feng, Manyuan Zhang, Shuang Chen, Yunlong Lin, Kaixuan Fan, Yilei Jiang, Hongyu Li, Dian Zheng, Chenyang Wang, and Xiangyu Yue. Gen-searcher: Reinforcing agentic search for image generation, 2026. URL https://arxiv.org/abs/2603.28767. arXiv:2603.28767; bibliographic record retrieved 2026-09-06.

Jakob Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. Counterfactual multi-agent policy gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. doi: 10.1609/aaai.v32i1.11794. URL https://ojs.aaai.org/index.php/AAAI/article/view/11794.

Ruiyan Han, Zhen Fang, XinYu Sun, Yuchen Ma, Ziheng Wang, Yu Zeng, Zehui Chen, Lin Chen, Wenxuan Huang, Wei-Jie Xu, Yi Cao, and Feng Zhao. Unicorn: Towards self-improving unified

multimodal models through self-generated supervision, 2026a. URL https://arxiv.org/abs/2601.03193. arXiv:2601.03193; bibliographic record retrieved 2026-09-06.

Yujin Han, Hao Chen, Andi Han, Zhiheng Wang, Xinyu Liu, Yingya Zhang, Shiwei Zhang, and Difan Zou. Turning internal gap into self-improvement: Promoting the generation-understanding unification in MLLMs. In *International Conference on Learning Representations*, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/dceda5883d3223231bef121cc8da2d61-Abstract-Conference.html. Preprint arXiv:2507.16663.

Ke Hao, Yuanzhi Liang, Tingxi Chen, Rui Li, Haibin Huang, Chi Zhang, Yun Gu, and Xuelong Li. Dreaming in flow: Generative grounding feedback for self-evolving unified multimodal models, 2026. URL https://arxiv.org/abs/2609.08282.

Jun He, Junyan Ye, Zilong Huang, Dongzhi Jiang, Chenjue Zhang, Leqi Zhu, Renrui Zhang, Xiang Zhang, and Weijia Li. Mind-brush: Integrating agentic cognitive search and reasoning into image generation, 2026a. URL https://arxiv.org/abs/2602.01756. Metadata verified against arXiv on 2026-09-19.

Zefeng He, Siyuan Huang, Xiaoye Qu, Yafu Li, Tong Zhu, Yu Cheng, and Yang Yang. Gems: Agent-native multimodal generation with memory and skills, 2026b. URL https://arxiv.org/abs/2603.28088. Metadata verified against arXiv on 2026-09-19.

Jixiang Hong, Yiran Zhang, Guanzhong Wang, Yi Liu, Ji-Rong Wen, and Rui Yan. SUDER: Self-improving unified large multimodal models for understanding and generation with dual self-rewards, 2025. URL https://arxiv.org/abs/2506.07963.

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, 2025. URL https://arxiv.org/abs/2505.00703. arXiv:2505.00703; bibliographic record retrieved 2026-09-06.

Weiyang Jin, Yuwei Niu, Jiaqi Liao, Chengqi Duan, Aoxue Li, Shenghua Gao, and Xihui Liu. SRUM: Fine-grained self-rewarding for unified multimodal models, 2025. URL https://arxiv.org/abs/2510.12784.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2022. URL https://arxiv.org/abs/2210.02747. arXiv:2210.02747; bibliographic record retrieved 2026-09-07.

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, 2025. URL https://arxiv.org/abs/2505.05470. arXiv:2505.05470; bibliographic record retrieved 2026-09-06.

Jie Liu, Zilyu Ye, Linxiao Yuan, Shenhan Zhu, Yu Gao, Jie Wu, Kunchang Li, Xionghui Wang, Xiaonan Nie, Weilin Huang, and Wanli Ouyang. Unigrpo: Unified policy optimization for reasoning-driven visual generation, 2026. URL https://arxiv.org/abs/2603.23500. arXiv:2603.23500; bibliographic record retrieved 2026-09-06.

Weijia Mao, Zhenheng Yang, and Mike Zheng Shou. Unirl: Self-improving unified multimodal models via supervised and reinforcement learning, 2025. URL https://arxiv.org/abs/2505.23380. arXiv:2505.23380; bibliographic record retrieved 2026-09-06.

Jiadong Pan, Liang Li, Yuxin Peng, Yu-Ming Tang, Shuohuan Wang, Yu Sun, Hua Wu, Qingming Huang, and Haifeng Wang. Learning to generate via understanding: Understanding-driven intrinsic rewarding for unified multimodal models, 2026. URL https://arxiv.org/abs/2603.06043. arXiv:2603.06043; bibliographic record retrieved 2026-09-06.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2023. URL https://arxiv.org/abs/2305.18290. arXiv:2305.18290; bibliographic record retrieved 2026-09-07.

Robin Rombach, Andreas Blattmann, and Björn Ommer. Text-guided synthesis of artistic images with retrieval-augmented diffusion models, 2022. URL https://arxiv.org/abs/2207.13038. arXiv:2207.13038; bibliographic record retrieved 2026-09-07.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635. PMLR, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

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, 2024. URL https://arxiv.org/abs/2402.03300. arXiv:2402.03300; bibliographic record retrieved 2026-09-07.

Zhenchen Tang, Songlin Yang, Zichuan Wang, Bo Peng, Yang Li, Beibei Dong, and Jing Dong. Endogenous reprompting: Self-evolving cognitive alignment for unified multimodal models, 2026. URL https://arxiv.org/abs/2601.20305.

Ritesh Thawkar, Shravan Venkatraman, Omkar Thawakar, Abdelrahman Shaker, Fahad Khan, Hisham Cholakkal, Salman Khan, and Rao Muhammad Anwer. Ask, solve, generate: Self-evolving unified multimodal understanding and generation via self-consistency rewards, 2026. URL https://arxiv.org/abs/2606.27376. arXiv:2606.27376; bibliographic record retrieved 2026-09-06.

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, 2023. URL https://arxiv.org/abs/2311.12908. arXiv:2311.12908; bibliographic record retrieved 2026-09-07.

Fu-Yun Wang, Han Zhang, Michael Gharbi, Hongsheng Li, and Taesung Park. Unirl-zero: Reinforcement learning on unified models with joint language model and diffusion model experts, 2025a. URL https://arxiv.org/abs/2510.17937. arXiv:2510.17937; bibliographic record retrieved 2026-09-06.

Haozhe Wang, Weijia Feng, Jinpeng Yu, Che Liu, Ping Nie, Fangzhen Lin, Jiaming Liu, Ruihua Huang, Jimmy Lin, Wenhu Chen, and Cong Wei. Search beyond what can be taught: Evolving the knowledge boundary in agentic visual generation, 2026. URL https://arxiv.org/abs/2607.05382. arXiv:2607.05382; bibliographic record retrieved 2026-09-06.

Yibin Wang, Yuhang Zang, Hao Li, Cheng Jin, and Jiaqi Wang. Unified reward model for multimodal understanding and generation, 2025b. URL https://arxiv.org/abs/2503.05236. arXiv:2503.05236; bibliographic record retrieved 2026-09-06.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025. URL https://arxiv.org/abs/2508.02324. arXiv:2508.02324; bibliographic record retrieved 2026-09-08.

Ling Yang, Xinchen Zhang, Ye Tian, Shiyi Zhang, Chenming Shang, Minghao Xu, Wentao Zhang, and Bin Cui. HermesFlow: Seamlessly closing the gap in multimodal understanding and generation. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2082. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/59d4e18a60490b9ed9913f3be2b14839-Abstract-Conference.html.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models, 2024. URL https://arxiv.org/abs/2401.10020. arXiv:2401.10020; bibliographic record retrieved 2026-09-07.

Jenny Zhang, Shengran Hu, Cong Lu, Robert Lange, and Jeff Clune. Darwin godel machine: Open-ended evolution of self-improving agents, 2025. URL https://arxiv.org/abs/2505.22954. arXiv:2505.22954; bibliographic record retrieved 2026-09-06.

Jenny Zhang, Bingchen Zhao, Wannan Yang, Jakob Foerster, Jeff Clune, Minqi Jiang, Sam Devlin, and Tatiana Shavrina. Hyperagents, 2026a. URL https://arxiv.org/abs/2603.19461. arXiv:2603.19461; bibliographic record retrieved 2026-09-06.

Zekai Zhang, Jiahao Li, Jie Zhang, Kaiyuan Gao, Kun Yan, Lihan Jiang, Ningyuan Tang, Shengming Yin, Tianhe Wu, Xiaoyue Chen, Xiao Xu, Yan Shu, Yanran Zhang, Yixian Xu, Yuxiang Chen, Zhendong Wang, Zihao Liu, Zikai Zhou, Huishuai Zhang, Dongyan Zhao, and Chenfei Wu. Qwen-image-agent: Bridging the context gap in real-world image generation, 2026b. URL https://arxiv.org/abs/2606.26907. arXiv:2606.26907; bibliographic record retrieved 2026-09-08.

Jiahao Zhao, Xiaomin Yu, Zhongxiang Sun, Fengwei Teng, Chengwei Qin, Xiaobin Hu, Jun Xu, and Shuicheng Yan. Toolartist: Tool-using unified multimodal models for agentic image generation, 2026. URL https://arxiv.org/abs/2608.04436. arXiv:2608.04436; bibliographic record retrieved 2026-09-06.

SUPPLEMENTARY MATERIAL

Appendix A discusses limitations; Appendices B–C specify curriculum construction and source controls; Appendices D–F provide detailed experiments. Human judgments and qualitative outputs appear in Appendices G–H, followed by extended related work and methodological positioning.

A LIMITATIONS

Current agentic visual generation systems, whether built from separate understanding and generation models or a unified model, often understand task requirements more reliably than they can realize them in generated images. RSIgen uses this capability asymmetry to improve collaboration between understanding and generation, translating stronger understanding into more effective supervision and better system-level performance. Nevertheless, the extent of further improvement remains constrained by the generative capabilities of the underlying models. Larger advances therefore also depend on improving these models' ability to translate understood requirements into faithful visual outputs.

B CURRICULUM CONSTRUCTION AND UPDATE PROTOCOL

B.1 INPUTS, EVIDENCE, AND ROUTING PRECEDENCE

Protocol Ω_0 fixes the action schema, task constraints, evaluator, curriculum compiler, update stages, replay policy, resource caps, and admission rule. Each probe stores the request/source group, peer versions, realized inputs, supported branches, terminal checks, and recipient/subtype. Capability memory retains these records for diagnosis and replay.

Input and Oracle checks. Input validity requires consistency with the request, coverage of the tested facts or visual attributes, correct entity/reference binding, and checkpoint-interface compatibility. Oracle is a reference intervention validated against these requirements independently of image reward. It tests evidence adequacy within the specified requirements. Failed or unresolved input checks trigger reconstruction or no-update. Oracle is confined to diagnosis; executable agent labels use the supported task-time actions.

Precedence. The routing order separates an upstream input problem from a downstream realization problem:

1. Check the request, prompt, evidence, and interface. Compare only branches with valid inputs for their intended intervention.
2. If Oracle repairs inadequate Search evidence, construct an agent-acquisition opportunity. If an executed prompt or condition correction repairs the outcome with the remaining inputs fixed, construct an agent-construction opportunity.
3. If Direct fails on stable reusable knowledge while validated Search succeeds and approximately matches Oracle, consider generator internalization. Require admissible target use and a knowledge-specific failure before compiling this subtype.
4. If failure persists with requirement-complete valid input, consider generator rendering, reference following, or supported editing. Admit the opportunity only when a compatible image target is available.
5. Route regressions to replay of the original compatible recipient/subtype. Preserve volatile or instance-specific dependencies through external access, and defer unresolved or mixed diagnoses.

Fixed run-level criteria for success, improvement, and approximate equivalence determine curriculum admission. Routing uses action interventions to select learning hypotheses, whose utility is evaluated through the resulting training curricula. Table 3 compares allocation strategies with matched role-level item, token, and image exposure.

Table 5: From probe outcomes to typed curricula. A route is admitted only when its inputs and training target are valid for the recipient. Oracle serves diagnosis and is excluded from executable agent labels.

Probe pattern	Recipient/subtype	Compiled training object
Oracle repairs inadequate retrieval	Agent: acquisition	Valid query and evidence-selection trajectory
Corrected input repairs the output	Agent: construction	Executed corrected prompt or condition
Search repairs a stable-knowledge failure	Generator: internalization	Evidence-free prompt and validated assisted image
Failure persists with sufficient valid input	Generator: realization	Compatible rendering, reference, or edit pair
Previously mastered capability regresses	Original recipient/replay	Validated prior target
Volatile dependency or unresolved diagnosis	Externalize or no-update	Retained access or further probes

B.2 From a diagnosis to an input–target pair

Table 6 illustrates the compiler's input–target forms. Agent targets come from executed, validated trajectories. Tool observations remain in the context, while SFT predicts only agent-produced outputs.

Table 6: Illustrative curriculum records. x is the request, h is the permitted history, e is validated evidence, and I^* is an admissible checked image target. Each target has its own validity check after pool expansion.

Recipient/subtype	Model input	Training target and construction
Agent: acquisition	(x, h): named spacecraft request and allowed search history	Executed query, chosen evidence/reference, and subsequent tool actions whose evidence passes request-level checks.
Agent: construction	$(x, h, e, p_{\text{old}})$: poster request, correct chronology evidence, and reversed layout	The executed corrected prompt p_{fixed} specifying the requested panel order, or the corresponding action trajectory.
Generator: internalization	$(p^{\text{D}}, \varnothing)$: "Draw [named spacecraft] above the Moon," without retrieved module descriptions	I^* from validated evidence-assisted generation; the student receives the original entity request without the teaching evidence.
Generator: realization	(p, e): complete layout and conditions; source image/mask for a supported edit	A compatible image I^* from an admissible image source or a validated successful generation candidate. Required references remain present.

Generator SFT requires a separately validated admissible image or a successful generation candidate. A failed output can supply the diagnosis, but target compilation still requires such an image. Reference-following and editing pairs retain their native conditions. Internalization removes the teacher's evidence from the student input, including retrieved facts embedded in a rewritten prompt.

Expansion retrieves related pool requests from a diagnosed capability neighborhood. Each addition receives its own input checks, subtype assignment, and target validation. Internalization reserves new paraphrases or compositions of the same knowledge unit in separate source/request/artifact groups for curriculum and refresh decisions. These uses remain separate from final evaluation.

B.3 Optimization and refresh

Table 7 summarizes the common optimization schedule. The three evaluated backbones are BAGEL, Qwen-Image, and Wan-Image-7B, with initialization followed by three update attempts. The stage order is shared, while native conditioning and edit interfaces determine which records and actions are compatible with each checkpoint.

For a compatible flow generator, supervised target imitation uses conditional flow matching (Lipman et al., 2022). With noise z_0, target-image latent z_1, and interpolation time u,

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{z_0, z_1, u} \left\| v_\theta((1-u)z_0 + uz_1, u \mid p, e) - (z_1 - z_0) \right\|_2^2 . \tag{9}$$

Table 7: Common update schedule. Role isolation is part of the update rule: the nonrecipient's effective function is held fixed during each stage. Internalization uses the supervised generator stage.

Stage	Training signal	Inputs/curriculum	Fixed partner
Generator SFT	Conditional target imitation	Validated image targets by subtype	Agent
Generator RL	Flow-GRPO, terminal-image reward	Compatible open-ended verifiable skills	Agent
Re-probe	Matched branch outcomes	Changed regions, comparisons, and anchors	Agent and branch inputs
Agent SFT	Executed action/trajectory targets	Post-generator supervision	Updated generator
Agent RL	GRPO, terminal quality and costs	Rollouts with the updated partner	Updated generator
Admission	Target quality and protected capabilities	Held-out admission subsets	Retained-state comparator

Flow-GRPO optimizes compatible verifiable generation skills with completed-image rewards (Liu et al., 2025). Agent GRPO uses group-relative advantages from Eq. (3); its likelihood mask contains only agent-produced tokens, excluding tool observations and image outputs. Reference-policy regularization remains separate from the reward.

The Wan experiments use full-parameter training. At each optimization stage, the nonrecipient's effective function, including its conditioning path, stays fixed. During re-probing, the agent, prompts, references, latent randomness, sampler, continuation, and evaluator stay fixed to compare generator versions.

B.4 Relative action improvement and label refresh

Equation (6) uses the agent's terminal reward, including trajectory costs. Oracle and unsupported actions are absent from the executable comparison set. Retrieval may differ across search modalities, but each branch's artifacts stay unchanged across the old and updated generators. A completed search with empty or unhelpful evidence retains its fallback generation and search cost. This is distinct from an infrastructure failure that prevents observing the branch.

The effect of a generator update on rankings depends on differential gains. Define $\gamma_t(x, a) = \widehat{V}_{t+1/2}(x, a) - \widehat{V}_t(x, a)$. Then

$$\begin{aligned}\widehat{V}_{t+1/2}(x, a) - \widehat{V}_{t+1/2}(x, b) = \widehat{V}_t(x, a) - \widehat{V}_t(x, b) \\ + \gamma_t(x, a) - \gamma_t(x, b).\end{aligned} \tag{10}$$

Equal gains preserve the pairwise difference. A preferred set can change while an old label remains competitive, distinguishing an action shift from label staleness. Task success is evaluated separately from the relative ranking of branches.

An internalization-based no-search target requires evidence-free success on reserved uses, a small residual Search–Direct gap, and preserved verified-Search success. Thus the rule distinguishes stronger Direct generation from degraded Search. Other refreshed targets cover queries, reference selection, prompt construction, stopping, and compatible repair; ambiguous comparisons receive further probes.

Algorithm 1 One reciprocal update attempt under Ω_0

Require: Retained state (A_t, G_t, M_t) and admissible pool $\mathcal{D}_{\text{pool}}$
1: Collect matched capability probes within the diagnosis budget
2: Apply input checks; route recipients and dispositions
3: Reserve transfer uses; compile and validate typed generator targets
4: Hold A_t fixed; update G_t by SFT and applicable Flow-GRPO
5: Re-probe affected regions and anchors with $\widetilde{G}_{t+1}$
6: Compile post-generator agent supervision
7: Hold $\widetilde{G}_{t+1}$ fixed; update A_t by SFT and GRPO
8: Compare target quality and protected capabilities with the retained state
9: Admit the candidate if both checks pass; otherwise retain (A_t, G_t)
10: Account for diagnosis, training, failed calls, and rejected attempts
11: Use the retained pair and permitted memory to construct new experience

C Recursive Construction-Source Controls

Each arm separates the trainable recipients from read-only curriculum-construction sources. Freeze-A uses A_0 wherever the source agent constructs evidence or inputs. Freeze-G uses G_0 for curriculum-side outcomes, including post-generator diagnosis. Their combination fixes both sources; the full arm uses the current sources. Recipient learning continues in every arm, and agent RL uses its current updated generator. These controls therefore test recursive curriculum construction while retaining current-partner policy learning.

Source opportunities, retrieval snapshots, compiler, optimization stages, and resource caps define the matched comparison. Each arm derives its memory, targets, and scores from its designated construction sources. Under common randomness, full and freeze-A start with source agent A_0 and share the first attempt. Subsequent differences measure the combined contribution of curriculum content, target validity, construction yield, and supervision freshness.

D Task Controls and Metric Details

D.1 Experimental design and statistical units

Models and comparisons. We evaluate BAGEL, Qwen-Image, and Wan-Image-7B, with mechanism studies on Wan. Qwen-Image uses Qwen2.5-VL as its condition encoder (Wu et al., 2025); Wan uses full-parameter updates to the trained role. Within each backbone, comparisons share initialization, tools, resource allowances, corresponding seeds, and three update attempts. Controls include agent-only adaptation, SearchGen-3 + costs, joint outcome RL, and uniform reciprocal learning (Appendices B.3 and D).

Mechanism controls. The action-refresh diagnostic uses 200 held-out `testmini` requests, separate from training, tuning, and checkpoint selection. Old/new generator comparisons fix evidence, action packages, post-generation states, and paired image seeds across three sequential stages; unsuccessful searches retain their costs. Wan allocation controls match role-level items and SFT/RL exposure, and Stage-2 adaptation matches total preparation, rollout, and optimization budget. Source-control arms use equal training-item counts; separate role evaluations use 200 requests and four image seeds per state. Appendices E and F give the protocols.

Statistical and resource units. Main results aggregate multiple training seeds. The original continuation diagnostic uses 10,000 paired request-cluster bootstrap replicates, preserving stages and arms; 95% intervals quantify request variation conditional on checkpoints and evidence. Sequential stages measure progression; image seeds measure generation sampling. Inference searches and generations/edits are counted per request. The original two-arm continuation matches realized tokens/training compute; Stage-2 adaptation matches total budget.

D.2 NATIVE-ROLE TRAINING CONTROLS

Table 9 consolidates the training recipes. Agentic initialization is the initial tool-using pair. The Gen-Searcher adaptation fixes the generator and adopts the same completed-image outcome and trajectory-cost contract as RSIgen, whereas the original method also uses final-text feedback (Feng et al., 2026). SearchGen-3 + costs retains Diffusion-DPO followed by positive-advantage rejection finetuning (Wang et al., 2026). Its three cycles are a local extension of SearchGen's published DPO–RFT pass. Joint outcome RL trains both roles from terminal outcomes. Uniform reciprocal learning shares RSIgen's role-specific optimization stages and post-generator refresh, while changing curriculum allocation.

Table 8: Core training comparisons on AgentGen-Bench Full (Overall9 global mean). Each column compares methods within one backbone. Scores aggregate multiple training seeds; Appendix D gives the recipes and NS/SI strata.

Method	BAGEL	Qwen-Image	Wan-Image-7B
Agentic initialization	29.3	40.0	37.5
A-only / Gen-Searcher adapt.	33.0	43.6	40.2
SearchGen-3 + costs	36.7	47.1	44.2
Joint outcome RL	35.6	46.3	43.3
Uniform reciprocal	36.2	46.7	43.8
RSIgen	**39.9**	**50.3**	**47.5**

Table 9: Training recipes for the within-backbone controls. A is the orchestration role and G the generation role. Trained comparisons span three update attempts; SearchGen-3 uses three DPO–RFT cycles. Initialization, interfaces, resource allowances, and corresponding seeds are common within each backbone.

Method	Updated	Optimization and curriculum
Agentic initialization	–	Initial tool-using pair before adaptation.
A-only / Gen-Searcher adaptation	A	Agent adaptation with completed-image outcome feedback and trajectory costs; G stays fixed.
SearchGen-3 + costs	A, G	Diffusion-DPO and positive-advantage rejection finetuning, repeated for three cycles, with trajectory costs.
Joint outcome RL	A, G	Terminal-outcome RL without DRR's alternating recipient-specific SFT curricula.
Uniform reciprocal	A, G	RSIgen's optimization stages and refresh, with uniform reciprocal curriculum allocation.
RSIgen	A, G	DRR recipient-specific curricula, generator SFT and applicable Flow-GRPO, agent trajectory SFT and GRPO, and post-generator refresh.

Uniform and failure-locus controls test the curriculum package, including selected targets and recipient exposure; the one-sided controls test which roles learn. Direct and prompt enhancement provide untrained inference references. The continuation study in Appendix E changes the old versus refreshed supervision bank while matching the updated rollout generator and realized training resources.

Table 10: Within-backbone AgentGen-Bench control results. Full: full-benchmark Overall9 global mean; NS/SI: AgentGen strata.

Method	NS	SI	Full
BAGEL			
Agentic initialization	54.2	25.4	29.3
A-only / Gen-Searcher adapt.	56.3	29.4	33.0
SearchGen-3 + costs	58.2	33.4	36.7
Joint outcome RL	57.6	32.2	35.6
Uniform reciprocal	58.6	32.8	36.2
RSIgen	**60.4**	**36.8**	**39.9**
Qwen-Image			
Agentic initialization	70.6	35.3	40.0
A-only / Gen-Searcher adapt.	71.2	39.4	43.6
SearchGen-3 + costs	72.3	43.2	47.1
Joint outcome RL	71.7	42.4	46.3
Uniform reciprocal	72.1	42.8	46.7
RSIgen	**73.2**	**46.8**	**50.3**
Wan-Image-7B			
Direct	62.3	27.4	32.1
Prompt enhancement	64.1	31.2	35.6
Agentic initialization	64.3	33.4	37.5
A-only / Gen-Searcher adapt.	65.2	36.3	40.2
SearchGen-3 + costs	66.2	40.8	44.2
Joint outcome RL	65.7	39.8	43.3
Uniform reciprocal	66.3	40.3	43.8
RSIgen	**67.3**	**44.4**	**47.5**

D.3 Component ablations

Table 11: Wan ablations on AgentGen-Bench. Full measures quality; S and G/E count inference searches and generations/edits per request.

Variant	Full ↑	S	G/E
Full RSIgen	47.5	**1.16**	**1.30**
A-only updates	40.2	1.59	1.45
G-only updates	41.7	1.76	1.51
Uniform reciprocal	43.8	1.41	1.42
Failure-locus-only routing	44.1	1.51	1.44
No post-G re-probe	45.0	1.53	1.39
No capability memory	45.9	1.31	1.36
No full-pool expansion	45.3	1.29	1.35
No trajectory costs	**48.0**	1.86	1.56

D.4 AgentGen aggregation

We follow the released AgentGen-Bench dataset and SearchGen evaluator. The original SearchGen paper uses the earlier benchmark name SearchGen-Bench. Full is the benchmark's global mean and the primary AgentGen statistic, computed over each model's scored coverage.[1] For example, Nano Banana Pro's published Full score is 65.0 with 738/751 scored requests, whereas nominal-count weighting of its rounded NS/SI scores (73.3/63.6) gives 64.9. Scored coverage and rounding both affect this calculation. GPT-Image-2 covers 718/751, while the published BAGEL and Qwen-Image rows cover 751/751.

Overall9 includes Checklist, Rubric, Prompt Faithfulness, Image Quality, Text Rendering, AI Naturalness, Composition, Physical Plausibility and Visual Reference; it excludes Text Reference. Applicable denominators can differ, so the mean of displayed component averages need not equal Overall9.

[1] Component scores and scored coverage follow the SearchGen project results and the benchmark release.

Benchmark evaluation references define the scoring side of the protocol, separately from runtime retrieval inputs and training targets.

Table 12: NS component scores. Ch: Checklist; Ru: Rubric; Pr: Prompt Faithfulness; IQ: Image Quality; Tx: Text Rendering; Nat: AI Naturalness; Co: Composition; Ph: Physical Plausibility; VR: Visual Reference; O9: Overall9.

Model	Ch	Ru	Pr	IQ	Tx	Nat	Co	Ph	VR	O9
Commercial Models										
GPT-Image-2	**84.8**	**82.7**	**79.3**	**73.4**	**92.3**	**67.6**	**86.5**	**81.2**	**70.9**	**78.9**
Grok-Imagine-2.0-Low	83.0	80.0	75.9	72.5	85.5	66.3	84.0	80.8	68.7	76.9
Qwen-Image-3-Max	79.1	76.2	69.8	70.3	86.2	63.9	82.5	78.3	64.0	73.7
Nano Banana Pro	78.4	76.3	70.2	70.7	85.6	65.2	80.3	77.1	63.0	73.3
Qwen-Image-2	76.2	73.1	67.5	70.2	79.5	63.8	79.2	74.2	60.3	71.1
Generation Only										
Mage-Flow	**64.2**	**61.8**	**54.5**	**64.7**	**51.2**	**56.3**	**71.8**	**67.4**	**47.1**	**61.2**
Flux.2-Klein-9B	63.7	60.7	49.5	63.2	30.0	53.5	71.0	67.0	41.1	58.5
Flux.2-Klein-4B	55.8	52.6	42.3	59.7	16.1	49.8	67.7	60.1	35.3	52.3
Unified MLLMs										
SenseNova-U1	68.1	65.2	58.5	64.3	58.3	56.2	72.2	65.1	51.1	62.8
OmniGen2	49.1	46.1	33.5	56.4	7.8	47.3	61.4	60.3	30.6	47.4
Show-o2	32.7	30.9	21.0	47.2	2.4	38.0	49.8	40.9	19.9	34.4
BAGEL	51.5	49.2	37.7	56.2	16.1	48.8	62.8	58.5	32.1	49.1
BAGEL + RSIgen	63.4	62.1	58.3	61.4	50.6	60.2	68.4	73.2	45.6	60.4
Qwen-Image	74.9	72.0	65.5	69.2	61.1	61.7	77.0	78.6	60.1	70.0
Qwen-Image + RSIgen	**77.2**	**76.1**	**72.4**	**70.7**	**68.4**	**69.2**	**80.1**	**81.3**	**63.4**	**73.2**
Wan-Image-7B	65.3	63.2	57.4	65.1	54.3	57.2	73.4	75.1	49.3	62.3
Wan-Image-7B + RSIgen	70.4	69.2	65.3	67.2	60.4	63.2	76.1	75.4	55.4	67.3

Table 13: SI component scores, using the same metric abbreviations as the NS table.

Model	Ch	Ru	Pr	IQ	Tx	Nat	Co	Ph	VR	O9
Commercial Models										
GPT-Image-2	**74.4**	**73.7**	**72.2**	**77.4**	77.7	68.9	**83.9**	84.9	**70.4**	**75.5**
Grok-Imagine-2.0-Low	73.1	72.5	70.7	76.9	**79.9**	**69.0**	83.5	**85.6**	68.3	74.9
Qwen-Image-3-Max	65.8	65.3	62.8	71.1	67.9	62.3	77.0	78.6	59.6	67.6
Nano Banana Pro	60.7	60.1	56.4	68.1	61.3	60.0	72.7	78.1	55.4	63.6
Qwen-Image-2	49.9	49.6	44.1	62.5	50.1	54.3	66.2	71.4	42.0	54.1
Generation Only										
Mage-Flow	23.0	21.7	**15.7**	**38.7**	**9.7**	**35.0**	**40.0**	**50.2**	16.2	**28.4**
Flux.2-Klein-9B	**23.8**	**22.4**	15.1	36.1	5.8	29.9	38.3	47.7	**16.7**	26.9
Flux.2-Klein-4B	20.1	18.6	11.5	33.0	3.4	28.1	35.7	44.8	12.7	23.9
Unified MLLMs										
SenseNova-U1	29.3	27.6	21.6	39.5	15.8	32.2	42.5	48.4	21.0	31.6
OmniGen2	16.4	14.9	8.8	29.2	2.5	24.3	31.4	40.8	9.5	20.4
Show-o2	8.1	8.0	4.2	30.2	0.7	26.5	34.3	31.7	3.4	17.5
BAGEL	18.7	17.7	12.4	30.4	2.2	25.0	32.4	38.1	14.2	22.2
BAGEL + RSIgen	36.4	35.2	31.4	43.3	29.5	37.2	44.3	44.8	27.4	36.8
Qwen-Image	25.0	23.8	17.4	37.9	9.2	29.9	41.0	45.5	17.7	28.6
Qwen-Image + RSIgen	**47.3**	**46.2**	**42.4**	**52.3**	**39.4**	**45.2**	**55.1**	**55.7**	**37.4**	**46.8**
Wan-Image-7B	22.3	21.2	15.4	37.2	10.3	30.2	41.3	51.2	16.3	27.4
Wan-Image-7B + RSIgen	44.3	43.2	39.4	50.2	36.5	43.2	53.4	53.2	35.4	44.4

D.5 Quality and inference-call accounting

Table 14: Inference-call averages for SearchGen-3 + costs and RSIgen. Counts charge search and generation/edit calls; different calls can have different latency and compute costs.

Backbone	Method	Full	Searches	Generations/edits
BAGEL	SearchGen-3 + costs	36.7	1.21	1.32
BAGEL	RSIgen	**39.9**	**1.18**	**1.31**
Qwen-Image	SearchGen-3 + costs	47.1	1.21	1.31
Qwen-Image	RSIgen	**50.3**	**1.13**	**1.28**
Wan-Image-7B	SearchGen-3 + costs	44.2	1.26	1.33
Wan-Image-7B	RSIgen	**47.5**	**1.16**	**1.30**

The no-cost Wan variant improves quality by 0.5 points while using 60.3% more searches and 20.0% more generations/edits, calculated from the displayed call averages. These per-request counts measure inference tool use.

D.6 Protected-capability admission check

Retention serves as a regression guard during RSI. For each candidate $\widetilde{S}_{t+1}$, let $\mathcal{D}^{G}_{\mathrm{protect},t}$ and $\mathcal{D}^{A}_{\mathrm{protect},t}$ be held-out generation and understanding subsets. Candidate and retained components are evaluated on identical items and settings within the attempt. For protected capability c, the guard is

$$\Delta_{c,t} = Q_{c,t}(\widetilde{S}_{t+1}) - Q_{c,t}(S_t) \geq -\epsilon_c, \qquad \text{for every protected } c. \tag{11}$$

Here $Q_{c,t}$ evaluates the relevant component: generation checks keep raw prompts fixed without agent assistance, and understanding checks invoke the agent's native understanding role without generation. Admission also requires the target-quality criterion. Protected items do not enter curriculum construction, training rewards, or the sealed final evaluation. The comparator is the current retained state S_t, and rejection leaves that state unchanged.

The tolerance ϵ_c applies locally to the protected subset at each admitted update, relative to the current retained state. Successive accepted steps can accumulate their permitted losses.

D.7 Curriculum-source trajectories

Table 15 gives the exact Wan scores plotted in Figure 4. Each frozen-source control keeps recipient optimization and current-partner RL active; it changes which peer produces the curriculum. The common initialization makes the subsequent separation interpretable as the contribution of using updated construction sources over the fixed sequence of attempts.

Table 15: Wan-Image-7B curriculum-source controls on AgentGen Full. Columns are retained states after fixed update attempts.

Curriculum source	Initial	Attempt 1	Attempt 2	Attempt 3
Current construction peers	37.50	**42.20**	**45.30**	**47.50**
Freeze construction A0	37.50	**42.20**	43.80	44.80
Freeze construction G0	37.50	41.10	42.70	44.00
Freeze both construction peers	37.50	41.10	42.00	42.80

E Frozen-Evidence Action-Refresh Diagnostic

The diagnostic separates the utility of fixed action branches from the policy's response to updated supervision. The generator pairs are (G_0, G_1), (G_1, G_2), and (G_2, G_3); G_{j+1} is the post-generator-update checkpoint, corresponding to the half-step partner in Section 3. The evaluation uses Wan-Image-7B and the fixed Test-mini population described in Section 4.

Table 16: Action refresh on Wan-Image-7B Test-mini. Panel A summarizes empirical branch-preference changes over 200 requests and 200 canonical post-states per stage. Panel B compares policies using the updated generator; regret is in reward units and Mini O9 measures completed-image quality. Auxiliary policy metrics and paired request-cluster 95% intervals appear in Tables 17 and 23.

A. Action changes after generator updates			
Decision	Evaluated	Preference shift	Stale old labels
Pre-generation	200	22.2%	17.0%
Post-generation	200	20.8%	15.7%

B. Matched continuation with refreshed supervision			
Policy	Pre regret ↓	Post regret ↓	Mini O9 ↑
Old A_j	7.63	6.10	44.77
Continue w/ stale labels	5.37	4.60	47.37
RSIgen refresh	**3.30**	**3.00**	**50.20**

E.1 Acquisition and matched replay

For each stage and request x_i, the frozen agent A_j constructs an enhanced prompt and canonical text and image queries, q_i^T and q_i^I. Queries use the request and policy context, separately from benchmark reference annotations and evaluator feedback. SearchGen-Corpus-1M supplies historical executed search sessions: a covered query can replay a valid session, while an uncovered query uses live acquisition before replay.

The stage fixes acquired search results, selected evidence, references, and action realizations before comparing the two generators. Direct uses the original prompt; PromptEnhance uses the frozen enhanced prompt; TextSearch and ImageSearch use their respective evidence E_i^T and E_i^I; HybridSearch uses exactly $E_i^T + E_i^I$ without an additional query. Old/new comparisons share prompts, evidence order, image seeds, inference settings, continuation rules, and evaluation. Evidence can change between stages but stays fixed within each comparison.

A completed search with no usable evidence remains a valid executed action. The fallback supplies the original prompt when neither modality yields usable evidence; HybridSearch retains any usable modality. Every branch retains its generated outcome and search-call cost under Eq. (3), including both modality calls for HybridSearch. Replay charges the logical cost of the action even when evidence acquisition is shared. API or evaluator failures after retries are infrastructure failures, separately from empty or unhelpful evidence. Tables 19 and 20 distinguish evaluated decisions from acquisition outcomes.

E.2 Canonical post-generation states

Each stage uses one canonical state per request, $s_i = (x_i, p_i, E_i, I_i, h_i)$. A fixed source seed produces I_i through Direct under G_j, and the state retains its prompt, history, and stage evidence. Both generator checkpoints and both continuation arms receive the same source image and action packages. Thus the fixed-state comparison tests the utility and selection of repair actions without changing the initial image.

The post-action set contains Accept, Regenerate, ReferenceRegenerate, and LocalEdit where eligible. Accept returns I_i; Regenerate uses the fixed prompt; ReferenceRegenerate uses I_i as its frozen image condition through the native interface. LocalEdit uses a fixed localized target, instruction, preserve constraints, mask, and reference package. Eligibility depends on a locally identifiable repair target and support by both compared checkpoints. An ineligible edit is removed from the action set while the state remains in the evaluation. Both post-generation regret and agreement use all evaluated states; LocalEdit precision and recall use eligible states.

Accept's image quality is invariant to the generator checkpoint. Consequently, a shift toward Accept reflects a change in the relative benefit of intervention. Sunk state-construction costs are common to all post-actions, and incremental costs are charged once. End-to-end Mini O9 evaluates the separate complete policy workflow, including its chosen queries and induced state distribution. Newly emitted

queries use live acquisition followed by immutable caching; identical queries reuse their artifacts across arms.

E.3 Action values and empirical comparison statistics

For a decision u and shared executable action set $\mathcal{A}(u)$, a finite-probe action bank has values

$$\widehat{V}_j(u,a) = \frac{1}{K}\sum_{k=1}^{K} R_A(\tau_{j,k}^{u,a}, x), \qquad \widehat{V}_j^*(u) = \max_{a\in\mathcal{A}(u)} \widehat{V}_j(u,a), \tag{12}$$

using paired image seeds for old/new-generator comparisons. Equation (6) defines tolerance-aware choices for supervision construction. The diagnostic reports empirical action shifts and stale labels from finite-probe branch comparisons: a shift changes the preferred actions, while a stale-label event invalidates the old supervision action under the updated generator.

Let $P_{j+1}(u,a)$ be the updated-generator preference-membership indicator from the diagnostic, F_j its pre-generation shift population, U_j^{pre} and U_j^{post} the evaluated decision populations, and $L_j \subseteq U_j^{\mathrm{post}}$ the LocalEdit-eligible states. For policy π selecting action $a_\pi(u)$,

$$\mathrm{PreTrack}_j = \frac{\sum_{u\in F_j} P_{j+1}(u, a_\pi(u))}{|F_j|}, \tag{13}$$

$$\mathrm{Regret}_{j,d} = \frac{\sum_{u\in U_j^d} [\widehat{V}_{j+1}^*(u) - \widehat{V}_{j+1}(u, a_\pi(u))]}{|U_j^d|}, \quad d \in \{\mathrm{pre}, \mathrm{post}\}, \tag{14}$$

$$\mathrm{PostAgree}_j = \frac{\sum_{u\in U_j^{\mathrm{post}}} P_{j+1}(u, a_\pi(u))}{|U_j^{\mathrm{post}}|}, \tag{15}$$

$$\mathrm{LEPrec}_j = \frac{\sum_{u\in L_j} \mathbf{1}[a_\pi(u) = \mathrm{LE}] P_{j+1}(u, \mathrm{LE})}{\sum_{u\in L_j} \mathbf{1}[a_\pi(u) = \mathrm{LE}]}, \tag{16}$$

$$\mathrm{LERecall}_j = \frac{\sum_{u\in L_j} \mathbf{1}[a_\pi(u) = \mathrm{LE}] P_{j+1}(u, \mathrm{LE})}{\sum_{u\in L_j} P_{j+1}(u, \mathrm{LE})}. \tag{17}$$

Empty conditional denominators are undefined. Regret evaluates action selection in the reward units of Eq. (3). Finite-sample maximization can bias absolute regret upward; in a paired policy contrast on the same value bank, the maximum cancels:

$$\widehat{\mathrm{Regret}}(a_{\mathrm{refresh}}) - \widehat{\mathrm{Regret}}(a_{\mathrm{stale}}) = \widehat{V}(a_{\mathrm{stale}}) - \widehat{V}(a_{\mathrm{refresh}}). \tag{18}$$

The paired difference therefore compares the values of the selected actions directly, with uncertainty conditional on the shared bank and action-selection procedure.

E.4 Continuation comparison

The policies are the initial A_j, old-label continuation A_{stale}, and refreshed-label continuation A_{refresh}. The continuation arms share initialization, training requests, the updated rollout generator, and realized token/training-compute budgets. They differ in which generator's branch comparisons supply action supervision. The initial agent provides the starting-policy reference.

Table 16 reports regret and completed-image quality, and Table 17 gives the complementary alignment metrics. Mini O9 applies the official Overall9 aggregation to the complete 200-request workflow. The per-stage policy results are in Table 18. Display precision follows each summary statistic; rounding stage-level values before averaging can differ slightly from subtracting the displayed summary means.

E.5 Coverage and transition directions

There are 200 evaluated pre-generation requests and 200 canonical post-generation states in each stage. Pre-generation shift counts are 49, 44, and 40, totaling 133; stale-label counts are 38, 34, and 30. Post-generation shift counts are 45, 42, and 38, totaling 125; stale-label counts are 34, 31, and 29. The headline shift and stale rates are equal-weight stage means.

Table 17: Auxiliary policy-adaptation metrics (percent). Pre Track conditions on pre-generation shift cases; Post Agree uses all evaluated post-states. LocalEdit precision and recall condition on eligible states. All entries are equal-weight stage summaries.

Policy	Pre Track ↑	Post Agree ↑	LE Precision ↑	LE Recall ↑
Old A_j	34.0	67.3	62.6	49.5
Continue w/ stale labels	52.9	75.1	72.7	61.4
RSIgen refresh	**78.2**	**83.4**	**83.8**	**76.3**

Table 18: Per-stage policy adaptation. Track conditions on pre-generation shift cases; Mini O9 evaluates the complete 200-request workflow.

Stage	Policy	Pre Track (%)	Mini O9
1	Old A_j	33.1	44.60
1	Continue w/ stale labels	51.2	47.10
1	RSIgen refresh	**77.1**	**49.90**
2	Old A_j	35.0	44.90
2	Continue w/ stale labels	53.6	47.40
2	RSIgen refresh	**79.4**	**50.30**
3	Old A_j	34.0	44.80
3	Continue w/ stale labels	54.0	47.60
3	RSIgen refresh	**78.0**	**50.40**

Across 600 stage–request observations, text search produces usable evidence in 438 cases and no usable evidence in 162 (27.0%); image search has 396 and 204 (34.0%), respectively. Both modalities have zero infrastructure failures after retries. LocalEdit is eligible for 126, 121, and 116 states, or 63.0%, 60.5%, and 58.0% of the stage populations. Empty retrieval outcomes remain included in action values, and edit ineligibility leaves the state in the other post-action metrics.

Transition shares pool events: Search-to-Direct contributes $69/133 = 51.9\%$ of pre-generation shifts and Regenerate-to-LocalEdit contributes $49/125 = 39.2\%$ of post-generation shifts. Search-ModeSwitch combines changes among text, image, and hybrid retrieval; Accept-to-Repair and Repair-to-Accept group admissible repair actions. These pooled shares describe action-transition directions; the headline shift and stale-label rates average the per-stage proportions.

Table 19: Per-stage matched coverage and empirical action changes. Rates divide by 200 evaluated decisions per row.

Phase	Stage	Evaluated	Shift count	Shift (%)	Stale count	Stale (%)
Pre	1	200	49	24.5	38	19.0
Pre	2	200	44	22.0	34	17.0
Pre	3	200	40	20.0	30	15.0
Post	1	200	45	22.5	34	17.0
Post	2	200	42	21.0	31	15.5
Post	3	200	38	19.0	29	14.5

E.6 Request-cluster uncertainty

The bootstrap uses $B = 10{,}000$ paired request-cluster replicates. Each replicate samples the 200 request IDs with replacement and keeps all stages, matched branches, and policy arms for each sampled request together. Conditional metrics recompute their stage-specific numerators and denominators, followed by the equal-weight stage mean. Refresh-minus-stale differences are computed inside each paired resample. Table 23 reports the percentile 95% intervals. This procedure measures request-level variation conditional on the checkpoints and frozen evidence, separately from variation across training runs.

Table 20: Search acquisition outcomes pooled across 600 stage–request pairs. Searches with no usable evidence remain valid, cost-bearing trajectories; infra counts API or evaluator failures after retries.

Type	Executed	Usable evidence	No usable evidence	No usable (%)	Infra
Text	600	438	162	27.0	0
Image	600	396	204	34.0	0

Table 21: Canonical post-state population and LocalEdit eligibility. An ineligible state retains the other post-actions and contributes to Post Agree and Post Regret.

Stage	Total states	LE eligible	Eligible (%)
1	200	126	63.0
2	200	121	60.5
3	200	116	58.0

Table 22: Grouped action transitions pooled across stages. Shares divide by 133 pre-generation or 125 post-generation shifts. SearchModeSwitch includes switches involving HybridSearch; Repair groups admissible repair actions.

Phase	Transition	Count	Share (%)
Pre	Search→Direct	69	51.9
Pre	Direct→Search	15	11.3
Pre	SearchModeSwitch	24	18.0
Pre	Search→PromptEnhance	13	9.8
Pre	PromptEnhance→Search	12	9.0
Post	Regenerate→LocalEdit	49	39.2
Post	ReferenceRegenerate→LocalEdit	21	16.8
Post	LocalEdit→Regenerate	16	12.8
Post	Accept→Repair	18	14.4
Post	Repair→Accept	21	16.8

Table 23: Paired request-cluster bootstrap 95% intervals (10,000 replicates), conditional on trained checkpoints and frozen evidence. The first four rows are rate intervals; remaining rows are refresh-minus-stale differences.

Metric	Lower	Upper
Pre shift (%)	18.1	26.5
Pre stale (%)	13.4	20.9
Post shift (%)	16.9	25.1
Post stale (%)	12.3	19.3
Δ Pre Track (pp)	18.00	32.00
Δ Pre Regret	-2.60	-1.60
Δ Post Agree (pp)	5.00	12.00
Δ Post Regret	-2.00	-1.20
Δ LE Precision (pp)	7.00	16.00
Δ LE Recall (pp)	10.00	20.00
Δ Mini O9	1.80	3.90

F WAN ROLE PERFORMANCE AND CURRICULUM MEASUREMENTS

These studies measure role performance, construction yield, and training composition over three attempts. Construction-source arms use current or initial peer versions while recipients continue learning (Appendix C).

F.1 PERFORMANCE UNDER SPECIFIED PARTNER AND INPUT CONDITIONS

Table 24 evaluates four retained states on 200 requests with four image seeds per request/state. The agent uses G_0 throughout; generator scores cover direct and evidence-assisted generation. Each is a quality readout under its stated condition, reported separately from system Full.

Figure 5 summarizes the trajectories. States follow update attempts; curriculum statistics refer to the corresponding pre-G and post-G stages. Both generator scores increase across the states. Agent quality with G_0 rises from 38.6 to 41.9 at S_2, then declines to 41.6 at S_3, revealing a nonmonotonic fixed-partner trajectory alongside improving system quality.

Table 24: Wan role evaluations on 200 requests with four image seeds per request and state. Agent performance uses the fixed initial generator. Generator scores cover assisted and direct evaluation. Image seeds are generation replicates.

State	A with fixed G_0	G: assisted	G: direct
S_0	38.6	40.9	33.0
S_1	40.8	43.6	35.8
S_2	**41.9**	45.1	37.6
S_3	41.6	**45.9**	**38.4**

F.2 CONSTRUCTION YIELD AND TRAINING COMPOSITION

Each source arm considers 2,000 requests per phase: 4,000 generator-target candidates before the G update and 3,000 agent-target candidates afterward. For candidate count C, valid count V, and newly valid count N, validity is V/C. Cross-attempt totals sum target entries; requests are counted separately within each phase.

Tables 26 and 27 distinguish construction from selected training records. Training records partition into new, refreshed, and existing use; existing use further partitions into replay and first use of previously available targets:

$$\text{TrainTotal} = \text{NewTrain} + \text{RefreshedTrain} + \text{ExistingUse},$$
$$\text{ExistingUse} = \text{Replay} + \text{ExistingFirstUse}.$$

Refreshed-request counts measure requests represented by the refreshed training records.

Every arm uses 1,800 agent and 2,200 generator records per attempt, totaling 5,400 and 6,600. With current sources, agent/generator construction validity rises from 56.0%/55.0% to 66.0%/68.5%; refreshed agent supervision grows from $420/1800 = 23.3\%$ to $870/1800 = 48.3\%$. Newly valid generator/agent targets decline from 1,500/1,080 to 1,180/650 across the same attempts. Thus curriculum validity and freshness improve within fixed training sizes even as fewer targets become newly valid.

F.3 SKILL COMPOSITION OF THE CURRENT-SOURCE CURRICULUM

Table 28 partitions selected records by skill: acquisition/construction for A, and internalization/realization/reference following/supported editing for G. Their per-attempt totals are 1,800 and 2,200. Skill and new/refreshed/existing-use categories describe two separate partitions of the same training records.

Agent construction grows from 1,020 to 1,170 records while acquisition falls from 780 to 630. Generator realization grows from 780 to 1,000 while internalization falls from 920 to 650. These quantities measure selected curriculum composition.

Table 25: Wan curriculum construction and supervision changes across the four source controls. Counts sum target entries across three attempts. Valid and new-valid counts describe construction; refreshed counts describe updated targets actually used for training.

	Agent curricula			Generator curricula			
Source	Valid	New valid	Refreshed	Valid	New valid	Refreshed	Full
Current	5,580	2,660	1,950	7,480	4,140	970	47.5
Freeze-A	5,280	2,560	1,660	7,060	3,820	790	44.8
Freeze-G	4,690	2,270	400	6,960	3,680	680	44.0
Freeze-both	4,430	2,120	0	6,740	3,540	570	42.8

Each arm uses 5,400 agent and 6,600 generator training entries over the three attempts. Repeated optimization of an entry contributes to exposure, not to these entry counts.

Table 26: Per-attempt curriculum construction on Wan. Each phase diagnoses 2,000 requests. G curricula are constructed before the generator update; A curricula follow post-G re-probing. C: candidate targets; V: valid targets; N: newly valid targets. Existing targets reused without reconstruction are outside the construction denominator.

Source	Attempt	Role	C	V	N	V/C (%)
Current	1	G	4,000	2,200	1,500	55.00
Current	1	A	3,000	1,680	1,080	56.00
Current	2	G	4,000	2,540	1,460	63.50
Current	2	A	3,000	1,920	930	64.00
Current	3	G	4,000	2,740	1,180	68.50
Current	3	A	3,000	1,980	650	66.00
Freeze-A	1	G	4,000	2,200	1,500	55.00
Freeze-A	1	A	3,000	1,680	1,080	56.00
Freeze-A	2	G	4,000	2,390	1,280	59.75
Freeze-A	2	A	3,000	1,820	860	60.67
Freeze-A	3	G	4,000	2,470	1,040	61.75
Freeze-A	3	A	3,000	1,780	620	59.33
Freeze-G	1	G	4,000	2,200	1,500	55.00
Freeze-G	1	A	3,000	1,510	950	50.33
Freeze-G	2	G	4,000	2,320	1,230	58.00
Freeze-G	2	A	3,000	1,610	780	53.67
Freeze-G	3	G	4,000	2,440	950	61.00
Freeze-G	3	A	3,000	1,570	540	52.33
Freeze-both	1	G	4,000	2,200	1,500	55.00
Freeze-both	1	A	3,000	1,510	950	50.33
Freeze-both	2	G	4,000	2,250	1,180	56.25
Freeze-both	2	A	3,000	1,500	720	50.00
Freeze-both	3	G	4,000	2,290	860	57.25
Freeze-both	3	A	3,000	1,420	450	47.33

Table 27: Training-entry composition by source, attempt, and role. Total equals new plus refreshed plus existing-use entries. Replay is a subset of existing use; the remaining existing entries are previously valid targets selected for training for the first time. Refreshed requests count requests represented by the refreshed entries.

Source	Attempt	Role	New	Refreshed	Existing	Replay	Total	Refr. req.
Current	1	G	1,300	150	750	500	2,200	120
Current	1	A	900	420	480	320	1,800	340
Current	2	G	1,320	300	580	460	2,200	240
Current	2	A	780	660	360	280	1,800	520
Current	3	G	1,060	520	620	500	2,200	410
Current	3	A	540	870	390	320	1,800	680
Freeze-A	1	G	1,300	150	750	500	2,200	120
Freeze-A	1	A	900	420	480	320	1,800	340
Freeze-A	2	G	1,150	260	790	600	2,200	205
Freeze-A	2	A	700	560	540	400	1,800	445
Freeze-A	3	G	920	380	900	700	2,200	300
Freeze-A	3	A	520	680	600	460	1,800	540
Freeze-G	1	G	1,300	150	750	500	2,200	120
Freeze-G	1	A	800	0	1,000	650	1,800	0
Freeze-G	2	G	1,100	220	880	660	2,200	175
Freeze-G	2	A	650	160	990	700	1,800	125
Freeze-G	3	G	840	310	1,050	800	2,200	245
Freeze-G	3	A	440	240	1,120	800	1,800	190
Freeze-both	1	G	1,300	150	750	500	2,200	120
Freeze-both	1	A	800	0	1,000	650	1,800	0
Freeze-both	2	G	1,040	180	980	720	2,200	140
Freeze-both	2	A	600	0	1,200	850	1,800	0
Freeze-both	3	G	750	240	1,210	880	2,200	190
Freeze-both	3	A	360	0	1,440	1,000	1,800	0

Table 28: Current-source Wan training curricula by recipient subtype. Columns correspond to the curricula consumed by each update attempt; each attempt uses 1,800 agent and 2,200 generator entries.

Role	Subtype	Attempt 1	Attempt 2	Attempt 3
A	acquisition	780	680	630
A	construction	1,020	1,120	1,170
G	internalization	920	780	650
G	realization	780	900	1,000
G	reference-following	320	350	370
G	supported-edit	180	170	180

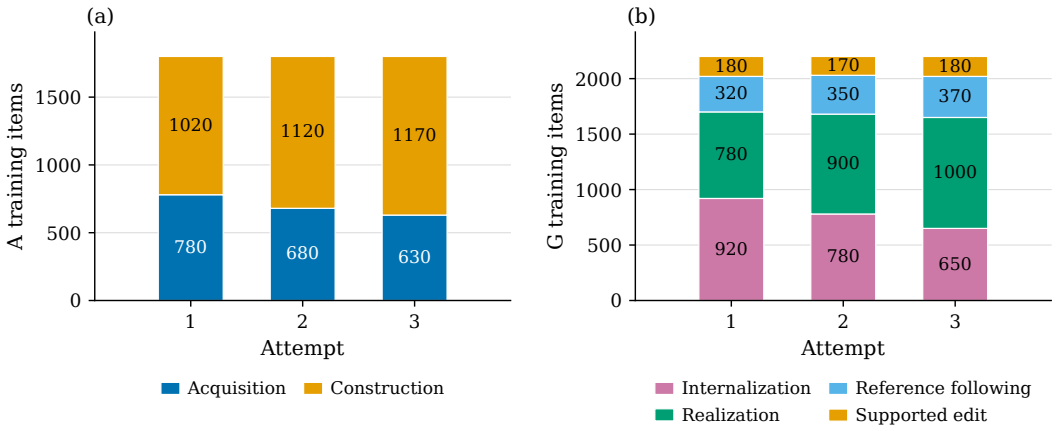

Figure 7: Current-source training curricula across three Wan update attempts. Each bar counts selected training items, partitioned by recipient subtype; agent and generator totals remain 1,800 and 2,200 per attempt. Replay is included in the relevant subtype. Candidate counts and optimization exposures use separate units; Appendix F gives the exact construction, training, and subtype counts.

F.4 Settings for the additional Wan studies

Wan uses full-parameter updates to the trained role. Table 29 gives learning rates, rollout-group sizes, and effective SFT batch size. Role evaluation uses four image seeds per request/state; action probing uses eight images per action/version.

The allocation comparison uses 4,000 opportunities and selects 1,600 agent and 2,000 generator records per strategy. Matched exposure is 3.2 million agent SFT tokens, 2.4 million agent RL tokens, 4,000 generator SFT image presentations, and 7,200 generator RL image presentations. These are the allocation study's training volumes.

For Stage-2 adaptation, preparation, rollout, and optimization use normalized cost $c = 100C/B_{\text{adapt}}$, where C is consumed cost and B_{adapt} the common single-stage budget. One unit is 1% of that budget; each arm spends 100 units at $j = 2$. The original continuation diagnostic reports its three-stage mean separately.

Table 29: Training and evaluation settings for the supplementary Wan experiments. Candidate counts describe construction, while the batch and rollout groups describe optimization.

Setting	Value	Unit
Parameter update	Full-parameter training	–
Role-evaluation requests	200	requests
Role-evaluation image seeds	4	images/request/state
Diagnostic requests per phase	2,000	requests
G candidate targets	4,000	targets/attempt
A candidate targets	3,000	targets/attempt
Agent SFT learning rate	1×10^{-5}	–
Agent GRPO learning rate	1×10^{-6}	–
Generator SFT learning rate	2×10^{-5}	–
Generator Flow-GRPO learning rate	1×10^{-5}	–
Agent GRPO group size	8	trajectories/prompt
Generator Flow-GRPO group size	24	images/prompt
Generation resolution	512	pixels
Effective SFT batch	32	samples/step
Action-probe images K	8	images/action/version

G HUMAN EVALUATION ON REAL-USER REQUESTS

The study compares RSIgen with SearchGen-3 + costs on 200 real-user agentic image-generation requests, separate from the AgentGen-Bench Test-mini diagnostic. The same requests are evaluated with BAGEL, Qwen-Image, and Wan-Image-7B. Three raters independently assess each pair under blind conditions on task completion and visual quality. Each dimension therefore contains 600 judgments per backbone and 1,800 pooled judgments, all on the same 200 requests.

Table 30 reports the complete counts, percentages, and tie-adjusted preferences. Wins and losses indicate preferences for RSIgen and the baseline, respectively. Ties receive half credit, so the pooled task-completion preference is $100(972 + 577/2)/1800 = 70.0\%$, and visual-quality preference is $100(828 + 679/2)/1800 = 64.9\%$, rounded to one decimal place. These differ from the raw win proportions of 54.0% and 46.0%. Unknown judgments are zero in both dimensions for every backbone.

The pooled preferences equal the unrounded, equally weighted backbone averages because each backbone contributes the same number of judgments. They are descriptive summaries of the evaluated sample; repeated judgments and reuse of requests across backbones do not increase the number of distinct requests.

Table 30: Complete human-evaluation results. Both panels use the same 200 real-user requests across all three backbones, with three independent blind ratings per image pair. Judgment counts are per dimension. Overall reuses these 200 requests and pools 1,800 judgments; it does not count 600 distinct requests. W/T/L is from RSIgen's perspective. Preference is $100(W + 0.5T)/(W + T + L)$; Unknown is zero throughout.

Backbone	Requests	Judgments	W / T / L	W / T / L (%)	Pref.↑	Unknown
Task completion						
BAGEL	200	600	312 / 198 / 90	52.0 / 33.0 / 15.0	68.5%	0
Qwen-Image	200	600	324 / 194 / 82	54.0 / 32.3 / 13.7	70.2%	0
Wan-Image-7B	200	600	336 / 185 / 79	56.0 / 30.8 / 13.2	**71.4%**	0
Overall	200	1,800	972 / 577 / 251	54.0 / 32.1 / 13.9	**70.0%**	0
Visual quality						
BAGEL	200	600	264 / 234 / 102	44.0 / 39.0 / 17.0	63.5%	0
Qwen-Image	200	600	276 / 228 / 96	46.0 / 38.0 / 16.0	65.0%	0
Wan-Image-7B	200	600	288 / 217 / 95	48.0 / 36.2 / 15.8	**66.1%**	0
Overall	200	1,800	828 / 679 / 293	46.0 / 37.7 / 16.3	**64.9%**	0

H Additional Visual Examples and Measured Branch Comparisons

Figure 6 in the main paper compares five requests with complete prompts: calligraphy, TLS 1.3, flags, astronomy, and coordinated design variations. RSIgen separates client/server columns and labels protocol stages, and renders four orrery designs in a consistent grid. The remaining rows illustrate text, entity appearance, and scene-composition requirements. The numerical cases below illustrate changing Direct quality and editing value; they are distinct from these five visual requests.

H.1 Two measured branch comparisons

Table 31 gives two numerical comparisons of generator versions. In Case 1, Direct quality rises from 0.38 to 0.75 while Search changes from 0.82 to 0.83, shrinking the gap from 0.44 to 0.08. On this request, Direct approaches assisted quality while assisted performance is preserved.

Case 2 compares post-generation quality minus incremental action cost. Regenerate rises from 0.62 to 0.64 and LocalEdit from 0.54 to 0.77, making LocalEdit the highest-valued listed action under the updated generator. Accept remains at 0.52; reference-conditioned regeneration rises from 0.59 to 0.64. Differential improvements therefore change the action ranking.

Table 31: Measured action outcomes for two numerical branch examples. Case 1 compares Direct and Search quality; the Search–Direct quality gap decreases from 0.44 to 0.08. Case 2 compares post-generation actions and reports quality minus incremental action cost as the action value.

Case	Action	Old quality	New quality	Cost	Old value	New value
Case1	Direct	0.38	0.75	–	–	–
Case1	Search	0.82	0.83	–	–	–
Case2	Accept	0.52	0.52	0.00	0.52	0.52
Case2	Regenerate	0.68	0.70	0.06	0.62	0.64
Case2	Reference regenerate	0.67	0.72	0.08	0.59	0.64
Case2	Local edit	0.57	0.80	0.03	0.54	0.77

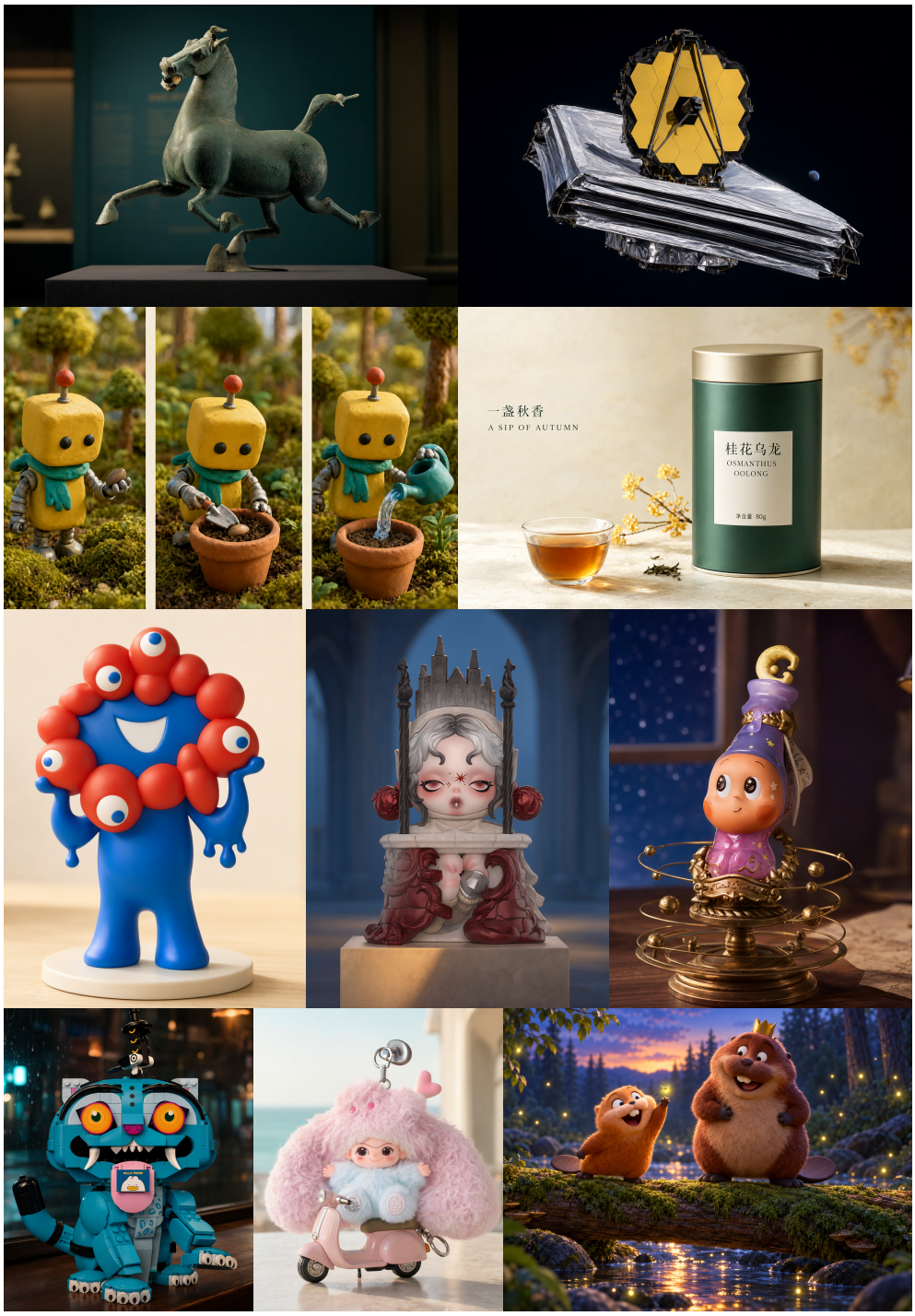

Figure 8: Additional RSIgen visual examples, including object depiction, sequential character scenes, product presentation, and stylized character designs.

I Extended Related Work

Agentic visual generation. Retrieval-augmented generation uses external information to address knowledge and reference requirements. Re-Imagen conditions synthesis on retrieved image–text pairs (Chen et al., 2022), while retrieval-augmented diffusion uses retrieved image embeddings (Rombach et al., 2022). Agentic systems additionally decide how to acquire and use that information: Gen-Searcher trains search decisions with a frozen generator and image/text feedback (Feng et al., 2026); ToolArtist learns native multimodal tool use; Qwen-Image-Agent organizes planning, search, and memory; and Unify-Agent jointly supervises textual trajectories and image generation (Zhao et al., 2026; Zhang et al., 2026b; Chen et al., 2026a). These approaches establish the task process in which RSIgen operates. Our focus is how experience from that process should train different capabilities, and how its action supervision changes when the generator also learns.

Experience reuse and generator-dependent adaptation. SearchGen is the closest co-training precedent. It trains the generator with Diffusion-DPO and then recalibrates the reasoner through positive-advantage rejection finetuning using the updated generator (Wang et al., 2026). Generator-dependent adaptation is therefore already part of its design. RSIgen further separates the learning uses of experience through matched diagnosis, constructs recipient-specific targets, and explicitly revisits acquisition and repair supervision after the generator update. Our SearchGen-3 + costs baseline is a local three-cycle extension of the original single DPO–RFT pass, so the comparison includes repeated adaptation in both systems.

GenEvolve extracts structured experience from contrasting same-request trajectories and supplies it to a privileged teacher for token-level distillation (Chen et al., 2026b). Such experience can improve search strategy, reference selection, and prompt construction. RSIgen instead asks how a validated experience should be assigned to agent or generator training and when its action labels should be reconsidered. These are differences in supervision construction; neither the use of experience nor iteration alone defines our contribution.

Reciprocal multimodal self-improvement. Several methods couple understanding and generation through preferences, rewards, or generated tasks. HermesFlow aligns cross-task preferences; UniRL and UniCorn use generated experience to construct multimodal learning tasks (Yang et al., 2025; Mao et al., 2025; Han et al., 2026a). SUDER introduces dual self-rewards, while Ask, Solve, Generate alternates role-specific updates (Hong et al., 2025; Thawkar et al., 2026). Turning Internal Gap into Self-Improvement revisits previously underutilized samples as understanding and generation improve (Han et al., 2026b). Dreaming in Flow combines flow-level feedback with visual replay grounding (Hao et al., 2026). RSIgen locates reciprocity at the curriculum and action-supervision level: a generator update changes useful agent decisions, and an agent update changes the experience available for later generator learning. The construction-source controls test this second direction while keeping the learning recipients active. Appendix J compares these mechanisms.

Optimization, feedback, and scope. Flow matching, DPO, Diffusion-DPO, and GRPO supply the underlying optimization tools (Lipman et al., 2022; Rafailov et al., 2023; Wallace et al., 2023; Shao et al., 2024). Flow-GRPO, T2I-R1, UniGRPO, and UniRL-Zero use outcome-driven optimization for generation or multimodal reasoning (Liu et al., 2025; Jiang et al., 2025; Liu et al., 2026; Wang et al., 2025a). RSIgen uses completed-image rewards and trajectory costs for RL, with matched probes selecting curricula and validated trajectories providing action SFT. Understanding-based rewards provide feedback for multimodal learning (Pan et al., 2026; Wang et al., 2025b; Jin et al., 2025). RSIgen distinguishes training feedback from final evaluation. Counterfactual advantages such as COMA assign credit within policy optimization (Foerster et al., 2018); DRR assigns compatible learning targets before optimization. Model-generated supervision also underlies iterative learning in SPIN and self-rewarding models (Chen et al., 2024; Yuan et al., 2024). In contrast to systems that alter agent implementations or improvement mechanisms (Zhang et al., 2025; 2026a), RSIgen updates capability roles within a fixed outer protocol.

J Positioning Reciprocal Curriculum Construction

Table 32 relates RSIgen to prior supervision mechanisms. RSIgen connects diagnosis-guided learning allocation, post-update action supervision, and subsequent curriculum construction. Table 8 evaluates task quality against training baselines.

Table 32: Closest methodological precedents and the scope of the RSIgen comparison. Prior work already establishes the ingredients in the middle column. The last column specifies a distinct evaluation question, not a blanket absence claim about that work.

Precedent	Established mechanism	Additional question in RSIgen
SearchGen (Wang et al., 2026)	Generator Diffusion-DPO followed by reasoner recalibration; knowledge-boundary adaptation	Does intervention-informed recipient/subtype allocation help when current peers construct later curricula?
Gen-Searcher (Feng et al., 2026)	Search-agent SFT/GRPO with a frozen generator and image/text feedback	How should action supervision change when the generator also learns?
ToolArtist; Unify-Agent (Zhao et al., 2026; Chen et al., 2026a)	Native tool use; joint textual and image supervision	Which validated failures should train acquisition, construction, internalization, or realization?
GenEvolve (Chen et al., 2026b)	Contrastive trajectory experience, privileged teacher, token-level distillation	Does updating the generator require revising the agent's old action labels?
HermesFlow; SUDER; ASG (Yang et al., 2025; Hong et al., 2025; Thawkar et al., 2026)	Joint preference alignment, dual self-rewards, and (in ASG) alternating role-adapter updates	How do matched action/input probes assign the recipient in a tool-using system?
Internal Gap; UniCorn (Han et al., 2026b;a)	Capability-dependent sample reuse and self-generated multimodal curricula	Which policy labels become stale after a peer update, under fixed action costs?
Dreaming in Flow (Hao et al., 2026)	Flow-level feedback and dream replay grounding jointly improve understanding/generation	Can terminal-action comparisons guide learning allocation and subsequent search/repair choices?

Other relevant distinctions. SRUM freezes its understanding evaluator and trains generation with cached global–local rewards, whereas Endogenous Reprompting freezes the generator and learns evaluation and reprompting through a two-stage protocol (Jin et al., 2025; Tang et al., 2026). GEMS and Mind-Brush broaden agentic generation with memory, skills, search, and reasoning (He et al., 2026b;a). RSIgen uses such capabilities to form recipient-specific learning opportunities and revisits their action labels after generator updates. DAgger addresses the data distribution induced by a changing policy (Ross et al., 2011); RSIgen additionally considers changing action utility induced by a changing generator. Both motivate re-evaluation as the system changes.

How the controls map to the method. The original uniform and failure-locus comparisons test the DRR curriculum package, including which opportunities and compatible targets reach each role. Table 3 compares allocation with equal role-level training exposure. The continuation diagnostic holds the rollout generator current while changing action supervision; Table 4 additionally compares ordinary on-policy adaptation at a shared stage-2 total budget. Construction-source freezes change the peers supplying curriculum evidence while keeping the learning recipients active, and Table 25 measures the corresponding target yield and refreshed supervision. These interventions address learning allocation, adaptation, and subsequent curriculum construction.

For the Wan endpoint, the descriptive difference-in-differences between agent and generator construction sources is

$$47.5 - 44.8 - 44.0 + 42.8 = 1.5. \tag{19}$$

This four-endpoint contrast summarizes the combined content and target-yield effects of construction sources. The primary comparison is the 4.7-point gap between current and jointly frozen sources.