DUPLEXBRIDGE: COORDINATING UNDERSTANDING AND GENERATION FOR UNIFIED IMAGE EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Complex image editing requires faithful task interpretation and accurate execution, but a failed edit does not reveal which responsibility failed. We present DuplexBridge, which uses a unified model's own understanding capability to enhance prompts for its own generation capability, without an external understanding module. A source-grounded bridge specifies targets, final-state changes, and preservation constraints. Understanding is supervised on fidelity to the original request without observing the generated image; generation is supervised on execution of the condition it actually receives. Reciprocal trust checks selectively admit global task feedback to each role while retaining independently valid local supervision. We introduce CE-Bench, a 1,200-task benchmark measuring joint satisfaction of task-derived change and preservation requirements. Across three model configurations, the full system improves complete compliance over direct editors by 21.75–25.75 percentage points, with gains also observed on KRIS-Bench, RISEBench, ImgEdit, and GEdit-Bench. From a common supervised Wan-Image initialization, gated joint training reaches 77.08% complete compliance, compared with 70.92% without gates and 75.00% when generation-local supervision instead checks the original task with both gates retained. These results support distinct local responsibilities and selective shared feedback for joint understanding–generation adaptation.

1 INTRODUCTION

Complex image editing must bind changes to the correct source content, infer final states, and preserve protected content. Setting only an analog clock to the digital clock's time plus 90 minutes requires all three.

Image-grounded interpretation and joint optimization already support complex editing (Fu et al., 2024; Huang et al., 2024). We study *native prompt enhancement* (PE) in unified models (Deng et al., 2025; Mao et al., 2026): their own understanding capability, U, constructs conditions for their own generation capability, G, without an external understanding module. When adapting both roles, what should supervise their local responsibilities, and when should they also learn from final task outcomes?

A task-level score alone does not distinguish faithful task expression from faithful execution. In figure 1(a), the digital reading 14:35 gives an analog target of 4:05. U may instead use the wrong reference and specify 11:50, which G faithfully renders. Alternatively, U specifies 4:05 correctly, but G changes the protected digital display. These failures motivate distinct checks of prompt fidelity, received-condition execution, and task outcome.

Executable planning and two-role adaptation have precedents (Zhou et al., 2026; Jiao et al., 2026). DARS routes planner–renderer updates using task-derived checklists and post-render planner diagnostics (Cao et al., 2026); DDA-Thinker separately evaluates plans and visual outcomes while keeping the editor frozen (Yang et al., 2026). DuplexBridge combines distinct local supervision references for both adapting roles with reciprocal checks that select only their additional global-feedback losses.

The bridge specifies source-bound targets, changes, and preservation (figure 1). G receives only the source and this condition, which also defines G-local execution checks.

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.

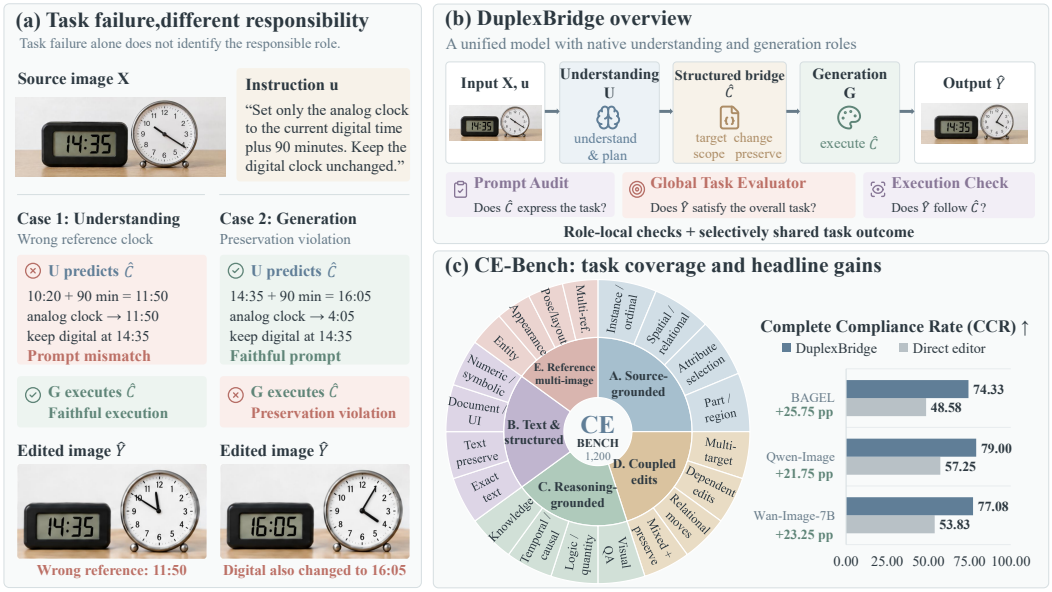


Figure 1: **DuplexBridge overview.** (a) Task failures with different local responsibilities. (b) Native U–G communication and three distinct checks. (c) CE-Bench coverage and full-system CCR gains, including PE, SFT, and RL. Primary sectors encode 300/240/240/240/180 tasks; outer subdivisions illustrate capabilities rather than subtype frequencies.

U-local checks compare the condition with the task without seeing the generated image. A separate global reward assesses the final edit against the task. Prompt trust gates this reward to G; execution trust gates it to U. Valid local supervision remains when either gate closes. Training alternates role updates; inference follows the native U-to-G path.

CE-Bench requires all task-derived change and preservation checks to pass; these requirements are independent of the predicted bridge. Full-system complete compliance rate (CCR) gains over direct editors span 21.75–25.75 points across three configurations, alongside public-benchmark improvements. From a common supervised Wan-Image initialization, full reaches 77.08% CCR versus 70.92% without gates and 75.00% with original-task G-local and both gates retained. Group-structure-matched random selection reaches 72.25%, supporting semantic selection in this setting.

Our contributions are:

1. A checkable bridge within a unified model that defines distinct local supervision targets for task expression and received-condition execution.
2. A joint learning rule that retains valid local supervision while selectively sharing global task feedback through reciprocal partner-trust gates.
3. CE-Bench, a 1,200-task evaluation of complete editing fulfillment, supported by public benchmarks and controlled comparisons of the feedback design.

2 Related Work

Understanding–generation coordination. MGIE and SmartEdit establish understanding-guided editing and coupled adaptation (Fu et al., 2024; Huang et al., 2024). BAGEL unifies understanding and generation, while Wan-Image develops structured editing conditions (Deng et al., 2025; Mao et al., 2026). Unified Thinker, ThinkGen, and ThinkRL-Edit study coordinated updates (Zhou et al., 2026; Jiao et al., 2026; Li et al., 2026). These precedents motivate our focus on what supervises each adapting role, beyond the choice to represent an edit as a plan or train both roles.

Supervision and feedback. PromptEnhancer assesses images against rewritten prompts; DDA-Thinker evaluates plans without observing generated images and also uses visual rewards. Both keep

generation frozen (Wang et al., 2026; Yang et al., 2026). DARS jointly adapts a planner and renderer using task-derived checklists and post-render planner diagnostics (Cao et al., 2026). DuplexBridge combines task-based U-local and received-condition G-local supervision with reciprocal eligibility for their additional global losses. Valid local learning remains available when global feedback is withheld.

Visual verification and evaluation. SLD verifies and corrects images at inference time (Wu et al., 2024). TIFA and Davidsonian Scene Graph motivate decomposed visual checking (Hu et al., 2023; Cho et al., 2024). CE-Bench evaluates complete fulfillment of task-derived changes and preservation, independently of the predicted bridge. Extended comparisons appear in sections O.1 and O.2.

3 A Structured Bridge for Understanding and Generation

3.1 Native roles and the forward path

Let X contain the source image and any task-provided reference images, with their roles identified, and let u be the editing instruction. DuplexBridge uses a unified model's own understanding role U_θ for prompt enhancement (PE) and its own generation role G_ϕ for editing:

$$(z_\theta, \widehat{C}) \sim \pi_\theta(\cdot \mid X, u; \psi), \qquad \widehat{Y} \sim G_\phi(\cdot \mid X, \widehat{C}; \psi). \tag{1}$$

Here z_θ is internal interpretation in a `<think>` block and $\widehat{C}$ is the public structured bridge. Trainable role parameters θ, ϕ are disjoint; fixed pretrained parameters ψ may include a shared backbone. No external understanding module constructs $\widehat{C}$. Generation receives X and $\widehat{C}$ in a fresh context, without the original instruction or U's internal reasoning.

3.2 Making the editing requirements explicit

U resolves target identities, reasoning-dependent answers, and terminal states before constructing $\widehat{C}$. The bridge binds changes to source entities, separates editable from protected content, and specifies preservation requirements and forbidden effects. Entailed requirements are made explicit without additional edits; requested text retains its exact characters. CE-v1 organizes these requirements as grouped JSON fields `goal`, `scope`, `ops`, `preserve`, and `avoid` (figure 2). The goal summarizes the operative fields without adding obligations.

The public condition also defines the execution checks:

$$D(\widehat{C}) = (D_\Delta, D_{\text{inv}}), \qquad Q_{\text{exec}} = \mathcal{A}(\widehat{C}), \tag{2}$$

where D_Δ and D_{inv} are disjoint change and invariant groups. A source resolver binds targets in X; fixed templates check applicable initial states before output requirements. Repeated obligations are deduplicated; overlapping preservation scopes are partitioned or deduplicated after semantic validation. Template enumeration is deterministic; source binding and visual judgment remain semantic operations.

The compiler uses the same condition as G and cannot recover omitted requirements from u or z_θ. A clearly specified edit can therefore be executable yet wrong for the user's request, motivating separate prompt and execution feedback. U is initialized on accepted teacher responses containing internal interpretation and public JSON before RL. Sections A, B and B.1 specify the schema, response format, and supervision templates.

4 Coordinating Learning with Local and Global Feedback

DuplexBridge separates three evaluations by the information they receive: prompt fidelity uses $(X, u, \widehat{C})$, execution uses $(X, \widehat{C}, \widehat{Y})$, and the global outcome uses $(X, u, \widehat{Y})$ (figure 3). Local rewards train each role's responsibility; partner trust determines when the joint outcome also supervises that role.

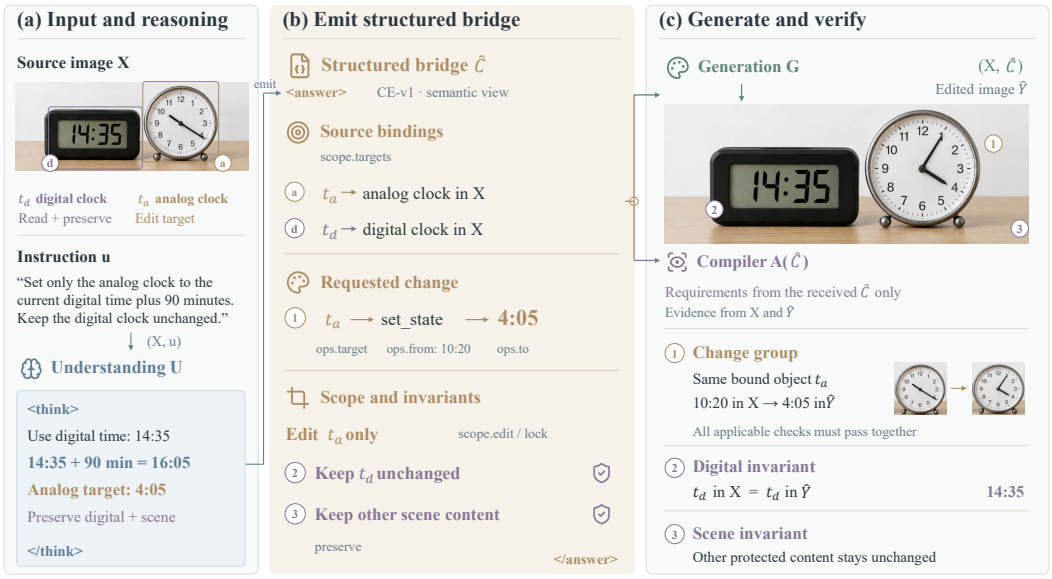


Figure 2: **From understanding to a checkable bridge.** U resolves the clock task and separates the editable analog clock from protected digital and scene content. Only the public condition enters G and the execution compiler; change and invariant checks use that same condition.

4.1 Role-local rewards

$$R_U^{\text{loc}} = F_U(X, u, \widehat{C}), \qquad R_G^{\text{loc}} = F_G(X, \widehat{C}, \widehat{Y}). \tag{3}$$

Understanding: prompt fidelity. A separately validated reference $C^\star$ supports format, requirement-level semantic precision/recall, and consistency/grounding judgments:

$$R_U^{\text{loc}} = s_C = (R_{\text{format}} + R_{\text{plan}} + R_{\text{clear}})/3. \tag{4}$$

The plan score uses maximum one-to-one semantic matching between predicted and verified requirements: target identity, operation, terminal value, and invariant scope must agree. With T matches among P predicted and R reference groups, $R_{\text{plan}} = 2T/(P+R)$. Format tests schema conformance; clarity checks contradictions and source bindings. This check never reads $\widehat{Y}$. Unresolved judgments abstain. The verified reference is distinct from teacher SFT targets (section E).

Generation: execution and local quality. For $Q_{\text{exec}} = \{Q_1, \ldots, Q_K\}$, each change group combines target grounding with its visual postconditions, while each distinct invariant forms another group. With visual judge V and check thresholds τ_q,

$$R_{\text{exec}} = \frac{1}{K} \sum_{k=1}^{K} \prod_{q \in Q_k} \mathbf{1}[V(X, \widehat{Y}, q) \geq \tau_q], \tag{5}$$

$$R_G^{\text{loc}} = w_\Delta R_{\text{exec},\Delta} + w_{\text{inv}} R_{\text{exec,inv}} + w_q R_{\text{quality}}. \tag{6}$$

The nonnegative weights sum to one; change and invariant scores average their respective groups. Quality excludes instruction compliance, and preservation is counted once. Invalid, empty, or unresolvable specifications abstain; low scores on valid checks remain learning signals (section A.3).

4.2 Global task feedback

A frozen pairwise reward model (RM) compares $\widehat{Y}$ with baseline edits from a fixed, nonempty set $\mathcal{R}_{X,u}$ under the original instruction and rubric i. It receives neither $\widehat{C}$ nor U's reasoning. Following the token-probability interface of RewardDance (Wu et al., 2025a),

$$R_{\text{global}} = F_{\text{global}}(X, u, \widehat{Y}; \mathcal{R}_{X,u}) = \frac{1}{|\mathcal{R}_{X,u}|} \sum_{Y_r \in \mathcal{R}_{X,u}} P_\omega(\texttt{Yes} \mid X, u, \widehat{Y}, Y_r, i). \tag{7}$$

Here `Yes` favors $\widehat{Y}$; its raw vocabulary probability gives a relative preference score in $[0, 1]$, not an absolute completion probability. References are independent of candidate bridges and shared across each rollout group. The direct-scoring RM distills a thinking teacher's judgment distribution and remains frozen during U/G training; its formulation and evaluation are in sections F and 6.4.

4.3 Bidirectional gates

Prompt trust controls G-global feedback; execution trust controls U-global feedback. Let c_{audit} indicate a confident prompt-fidelity judgment, v_C a passed prompt-trust check, and b_E, b_Q, b_F usable execution, quality, and global judgments. Define

$$\begin{aligned} m_{U\to G} &= c_{\text{audit}} v_C, & e_G^{\text{glob}} &= b_F m_{U\to G}, \\ m_{G\to U} &= b_E \mathbf{1}[R_{\text{exec}} \geq \tau_e], & e_U^{\text{glob}} &= b_F m_{G\to U}. \end{aligned} \tag{8}$$

Local eligibility is independent of these gates: $e_U^{\text{loc}} = c_{\text{audit}}$ and $e_G^{\text{loc}} = b_E b_Q$. Judgment usability differs from task success: a confidently identified failure can supply a low local reward. Gate-only ablations bypass semantic trust while retaining judgment-confidence and availability tests, so an unavailable check is never treated as a successful execution.

In figure 1(a), if G faithfully sets the analog clock to U's incorrect 11:50, G retains local supervision. Passed execution trust admits global feedback to U; failed prompt trust excludes G-global. When U correctly specifies 4:05 but G changes the protected digital clock, passed prompt trust admits G-global; U-global is excluded if execution trust fails. The mean threshold can still accept a missed change or preservation failure (section E.2).

4.4 Alternating updates with four loss components

Each role combines two separately normalized group-relative clipped losses:

$$\begin{aligned} \mathcal{L}_U &= \mathcal{L}_{\text{GR}}^{U,\text{loc}}(R_U^{\text{loc}}; e_U^{\text{loc}}) + \lambda_U \mathcal{L}_{\text{GR}}^{U,\text{glob}}(R_{\text{global}}; e_U^{\text{glob}}), \\ \mathcal{L}_G &= \mathcal{L}_{\text{GR}}^{G,\text{loc}}(R_G^{\text{loc}}; e_G^{\text{loc}}) + \lambda_G \mathcal{L}_{\text{GR}}^{G,\text{glob}}(R_{\text{global}}; e_G^{\text{glob}}). \end{aligned} \tag{9}$$

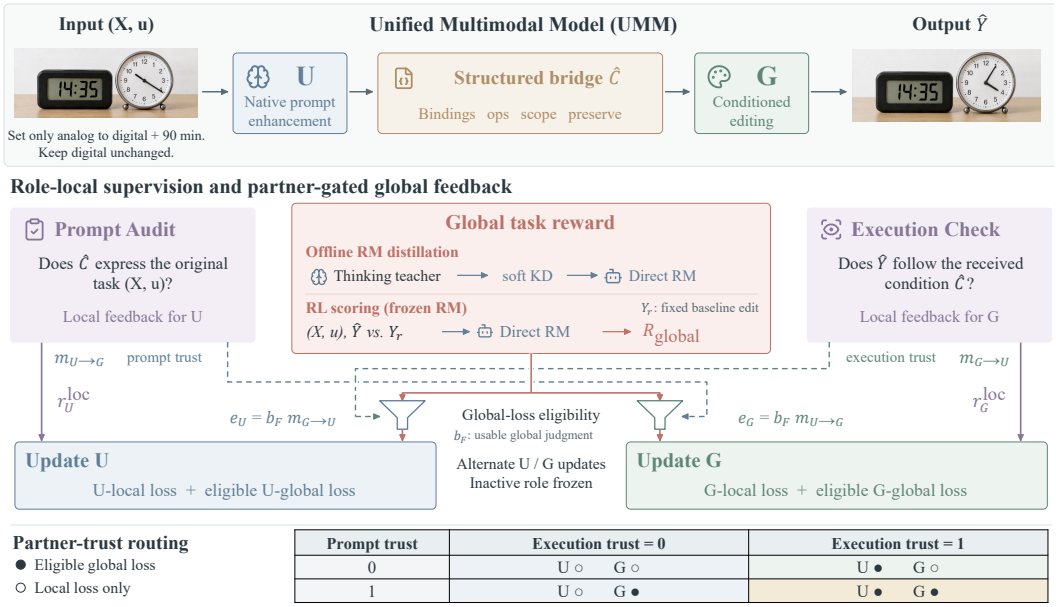


Figure 3: **Bidirectional partner-gated feedback.** Execution trust gates U-global; prompt trust gates G-global. Local feedback retains independent validity masks. The diagram abbreviates $e_U^{\text{glob}}, e_G^{\text{glob}}$ as e_U, e_G and local rewards as $r_U^{\text{loc}}, r_G^{\text{loc}}$. The routing matrix assumes usable judgments. Y_r is a baseline edit, distinct from task-provided reference images.

For U updates, freeze G, sample complete U responses, and generate several images per response. Average global scores over eligible images within each response, then weight eligible responses equally; a response with none can still contribute to U-local. For G updates, freeze U and compare stochastic image trajectories conditioned on one bridge.

Each component computes its own advantages on its eligible set and applies its own KL penalty. Excluded samples contribute neither its policy loss nor its KL; replacing their rewards by zero inside shared normalization is not equivalent. Groups with fewer than two eligible samples or zero reward variance are skipped for that component, including KL. Flow-policy ratios use stochastic trajectory likelihoods (Shao et al., 2024; Liu et al., 2025a); section E specifies aggregation and optimization. The gates provide selective feedback, not an unbiased estimate of unconditional return or counterfactual causal attribution (section E.4).

5 Experimental Setup

Models and comparisons. We evaluate BAGEL, Qwen-Image-Edit-2511 (Qwen-2511), and Wan-Image-7B, with mechanism ablations on Wan-Image. Each model supplies its own U and G for native PE. Qwen-Image labels in the figures denote Qwen-2511. Direct baselines edit the original instruction; full systems add native PE, structured supervised initialization, and joint RL. B_0 denotes the common supervised initialization before RL. U-only, G-only, local-only, and joint variants are independently trained from B_0, with common supervision and reference access. Unless varied explicitly, global feedback uses the reasoning-distilled RM. RM replacement experiments use equal update steps and rollout budgets. Detailed interface, training, and evaluation definitions appear in sections C, E and G.

CE-Bench. Our 1,200-task benchmark tests whether target binding, terminal-state changes, and preservation constraints are jointly satisfied. It covers source-grounded editing (300), text and structured content (240), reasoning-grounded editing (240), compositional editing (240), and reference-conditioned multi-image editing (180); section C.1 gives the taxonomy. Evaluation uses the original task, source, required reference images, and final output; requirements are independent of the model's predicted bridge. For item n, let a_{nk} denote a binary pass for requirement group k, and let P_n be the mean pass rate over its preservation groups. We report

$$\mathrm{AC} = \frac{1}{N}\sum_n \frac{1}{K_n}\sum_k a_{nk}, \qquad \mathrm{CCR} = \frac{1}{N}\sum_n \prod_k a_{nk}, \qquad \mathrm{Pres} = \frac{1}{N}\sum_n P_n. \tag{10}$$

All metrics are percentages and weight items equally; failed outputs remain in the denominator. CCR requires complete compliance with every change and preservation group. A change group includes target grounding and the required terminal state; each distinct preservation obligation counts once. Correctly changing the analog clock but altering the protected digital display fails CCR despite passing the change check. These task-derived groups differ from the predicted-condition groups used for G-local training.

Public evaluation. We report the KRIS-Bench composite, RISEBench-360 completion accuracy, ImgEdit's 1–5 score, and GEdit-Bench English Full-set G_O (0–10) (Wu et al., 2025b; Zhao et al., 2025; Ye et al., 2025; Liu et al., 2025b; Cao et al., 2026). Cross-system comparisons evaluate complete configurations; Wan-Image ablations test components within one configuration.

6 Experiments

6.1 End-to-end editing performance

DuplexBridge improves CE-Bench CCR by 25.75, 21.75, and 23.25 points on BAGEL, Qwen-2511, and Wan-Image, respectively, alongside preservation (table 1). These full-system gains include PE, supervised initialization, and RL. On public benchmarks (table 2), Wan-Image reaches 80.13/29.17 KRIS/RISE, gaining 19.29/16.11 points over its direct baseline. Qwen-2511 reaches 82.16/31.11, versus DARS's 80.72/27.50. These are complete-system comparisons; component controls follow. Task-specific text-to-image retraining is evaluated on GenEval2, R2I, WISE-V, and DPG in section L.

Table 1: **CE-Bench comparison.** A–E are category CCR (table 8); scores are percentages. Full systems include native PE, structured SFT, and gated RL. Dark/light pink marks the best/second-best values within each model group, including ties.

Model	A	B	C	D	E	AC ↑	Pres. ↑	CCR ↑
Commercial Models								
GPT-Image-1.5	**83.00**	**78.33**	72.92	82.50	68.89	93.47	94.12	77.83
Gemini-3-pro-image-preview	82.33	77.08	**81.67**	**84.17**	**72.78**	**94.08**	93.87	**80.08**
Seedream 4.5	80.33	77.50	62.92	72.50	58.89	91.36	**94.43**	71.50
Gemini-2.5-Flash-Image	73.00	69.58	64.17	70.83	59.44	89.67	90.72	68.08
Generation-only Editors								
FLUX.1-Kontext-dev (direct)	58.33	43.33	30.42	43.75	33.33	75.62	86.31	43.08
FLUX.2-klein-9B (direct)	**75.67**	64.58	47.08	60.00	**50.56**	**85.92**	92.87	**60.83**
LongCat-Image-Edit (direct)	74.33	**65.42**	**48.33**	**60.42**	47.22	85.74	**93.16**	60.50
Planner + Generator								
RePlan + Qwen-2511	70.33	58.75	52.92	59.58	49.44	84.62	90.41	59.25
PromptEnhancerV2 + Qwen-2511	76.33	68.75	60.83	66.67	56.11	88.62	91.26	66.75
ThinkRL-Edit (DARS config.)	76.00	64.58	65.42	70.42	57.78	89.06	91.84	67.75
EditThinker + Qwen-2511	77.33	67.08	63.33	70.00	60.56	89.48	92.18	68.50
DDA-Thinker-32B + Qwen-2511	**79.00**	71.25	70.83	75.42	64.44	91.17	93.10	72.92
DARS	**79.00**	**72.50**	**71.67**	**76.25**	**66.11**	**91.36**	**93.18**	**73.75**
Unified MLLMs								
UniREdit-BAGEL	72.00	57.08	55.00	62.92	47.78	85.68	89.34	60.17
UniReason 1.0	74.00	62.50	60.42	67.92	53.33	87.81	90.26	64.67
BAGEL	64.00	43.75	40.00	51.67	36.67	80.80	86.72	48.58
BAGEL + DuplexBridge	79.00	70.83	74.17	77.08	67.78	91.27	92.74	74.33
Qwen-2511	71.67	56.25	51.25	59.17	40.00	84.30	90.86	57.25
Qwen-2511 + DuplexBridge	82.00	78.33	**78.33**	**80.83**	**73.33**	**93.16**	**94.21**	**79.00**
Wan-Image-7B	68.33	52.08	46.67	56.67	37.78	82.70	88.47	53.83
Wan-Image-7B + DuplexBridge	**85.67**	**79.17**	77.08	67.08	**73.33**	92.31	93.86	77.08

Table 2: **Public editing benchmarks.** Higher is better; metric scales follow section 5. Dark/light pink marks the best/second-best value per column among displayed systems.

Method	KRIS ↑	RISE ↑	ImgEdit ↑	GEdit EN ↑
PromptEnhancerV2	71.54	18.90	4.18	7.59
DARS	80.72	27.50	4.39	7.86
BAGEL	56.21	5.80	3.45	6.40
+ DuplexBridge	76.48	26.11	4.04	7.41
Qwen-2511	64.42	17.50	4.36	6.97
+ DuplexBridge	**82.16**	**31.11**	**4.43**	**7.89**
Wan-Image-7B	60.84	13.06	4.09	6.88
+ DuplexBridge	80.13	29.17	4.36	7.74

Qualitative behavior. Figure 4 illustrates coupled requirements: a balance responds to added load, oil separates above water, and three cups reach a common size. Reference-guided garment replacement preserves the white trousers. Section K adds localized edits, spatial reasoning, image-grounded answers, and remaining answer-format errors across CE-Bench, KRIS-Bench, and RISEBench.

6.2 A STRUCTURED BRIDGE FOR JOINT ADAPTATION

Communication and supervision. On Wan-Image, native PE raises CCR from 53.83 to 56.67, and structured supervised initialization reaches $B_0 = 59.42$ (table 23). Grouped key–value and JSON interfaces reach 59.00 and 59.42, versus 56.67 and 57.42 for flat counterparts (table 10). From common B_0, ungated joint RL exceeds U-only and G-only variants (table 3). Local-only, ungated joint, and gated joint training reach 68.25, 70.92, and 77.08, supporting local supervision with selectively shared feedback.

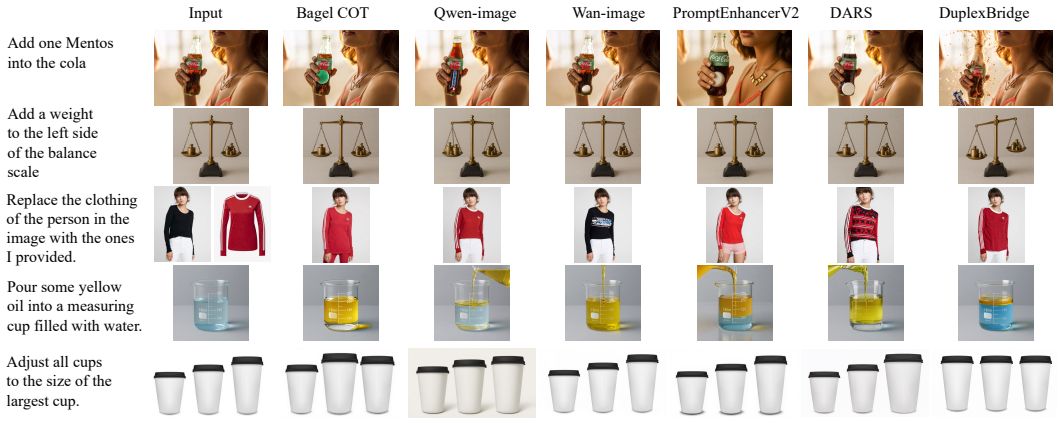


Figure 4: **CE-Bench qualitative comparisons.** The tasks combine requested changes with physical consequences, reference fidelity, and preservation. Rows show Mentos–cola interaction, added load, garment replacement, oil–water separation, and cup-size equalization.

Table 3: **Coordination and feedback controls on Wan-Image.** CE-Bench CCR (%). Left: role/reward variants from common B_0. Right: feedback alternatives. Dark/light pink marks the best/second-best CCR per panel. Full metrics: tables 19, 22 and 23.

Training variant	CCR
Supervised initialization (B_0)	59.42
U-only RL	66.67
G-only RL	67.50
Joint, local only	68.25
Joint, ungated global	70.92
Joint, gated global (full)	**77.08**

Feedback variant	CCR
Prompt trust only	74.17
Execution trust only	73.33
Group-structure-matched random	72.25
Original-task G-local	75.00
Received-condition + both gates	**77.08**

Execution-supervision reference. With both gates retained, received-condition G-local reaches 77.08 CCR versus 75.00 for original-task G-local (table 3), supporting alignment with G's input. U-local and global feedback retain the original task reference.

6.3 Selecting reliable global feedback

Both prompt and execution trust contribute: using either alone reaches 74.17 and 73.33 CCR, respectively, compared with 77.08 for both. Semantic gating exceeds group-structure-matched random selection by 4.83 points (77.08 versus 72.25). Matching accepted counts alone can fragment prompt groups and alter valid group-relative updates. Group matching better constrains this difference, but the contrast does not uniquely separate selection quality from optimizer exposure (section H.1). Soft semantic weighting reaches 76.33 (table 22); both selection variants improve on ungated feedback at 70.92. From the displayed endpoints, gating improves CCR by 3.33 points with a flat interface and 6.16 with the structured bridge (table 26), a pattern consistent with complementary communication and feedback selection.

On 6,000 fixed rollouts (table 25), the wrong-prompt rate among eligible G-global samples falls from 18.59% to 2.90%; the unexecuted-prompt rate among eligible U-global samples falls from 27.41% to 4.60%. Labels combine external-VLM and human judgments. These conditional rates describe selection on fixed outputs; the retrained endpoints measure editing performance.

Joint fulfillment across requirement loads. Full has higher CCR point estimates at all four observed requirement loads, with gains of 5.33, 6.39, 8.06, and 3.33 points for $K = 3, 5, 7, 9$. The $K = 9$ interval includes zero. Overall, change-complete but preservation-failing outputs fall from 9.50% to 6.42%. The component results vary by load: change all-pass declines at $K = 9$. Section M presents the paired intervals, joint outcomes, and category composition of each load stratum.

6.4 DIRECT REWARD SCORING FOR SHARED FEEDBACK

Table 4: **Reward quality, cost, and downstream editing.** HPA is human preference agreement. Latency is per pairwise RM call, relative to base scoring; gated Wan-Image comparisons share update and rollout budgets. HPA/CCR are percentages. Dark/light pink marks column-wise best/second-best values, including ties.

Reward model	HPA ↑	Rel. latency ↓	Gated CCR ↑
Base direct (M0)	77.6	**1.0×**	74.25
Explicit thinking (M1)	**82.3**	6.0×	**77.50**
Distilled direct (M5)	81.7	**1.0×**	77.08

Distillation raises human preference agreement (HPA) from 77.6% to 81.7%, near explicit thinking's 82.3%, at direct scoring latency (table 4). It raises gated CCR from 74.25 to 77.08 under equal update and rollout budgets. Explicit thinking reaches 77.50 at six times the pairwise RM-call latency. This comparison replaces only the frozen global evaluator; the local checks and feedback-routing rule remain the same. Section F.4 gives additional reward variants, probabilistic scores, and external evaluations.

6.5 HUMAN EVALUATION OF FINAL EDITS

Table 5: **Final-output human evaluation.** On 200 CE-Bench tasks, N denotes human-decidable outputs per method. CCR and agreement are percentages; bias is automatic minus human CCR in points. Paired comparisons use 194 common tasks.

Method	N	Human	Auto.	Bias	Agreement
Direct	195	53.33	54.36	+1.03	88.72
Ungated	196	69.39	70.92	+1.53	91.33
Full	197	76.14	76.65	+0.51	93.40

On the 194 tasks with resolved human labels for both full and ungated (table 5), human CCR increases from 69.07% to 75.77%: a 6.70-point gain (95% interval [1.03, 12.37]). The automatic gain on these same tasks is 6.19 points; automatic minus human gain is -0.52 points $[-3.61, 2.58]$, an interval that includes zero.

Table 5 uses each method's decidable outputs, whereas the paired gain uses their common subset. This explains the full method's 76.14% per-method CCR versus 75.77% in the paired comparison.

Full's net instruction preference over ungated is 17.77 points [8.63, 26.90], with ties in the denominator. Net preservation and quality preferences are 0.51 and -2.03 points; both intervals include zero, so these comparisons do not establish quality non-inferiority. Section N gives full counts, unresolved-label bounds, direct-baseline comparisons, and the paired category-stratified bootstrap protocol.

7 CONCLUSION

DuplexBridge connects a unified model's own understanding and generation through a checkable editing condition. Local supervision distinguishes task fidelity from received-condition execution, while reciprocal trust gates select additional global feedback and retain valid local learning. Across three configurations, complete compliance improves over direct editors; controlled Wan-Image comparisons support the supervision references and semantic selection, with gains also supported by human evaluation. These findings support separating local responsibilities when jointly adapting understanding and generation.

Reproducibility Statement

The appendix specifies the CE-v1 schema, prompt templates, local checks, four-component optimization, reward distillation, and benchmark metrics (sections A to C, E and F.2).

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

Haoxiang Cao, Jiajiong Cao, Xuanpu Zhang, Changqian Yu, and Chaoqun Wang. DARS: Dual-level credit assignment RL with structured reasoning for instruction-based image editing, 2026. URL https://arxiv.org/abs/2608.20161.

Kaijie Chen, Zihao Lin, Zhiyang Xu, Ying Shen, Yuguang Yao, Joy Rimchala, Jiaxin Zhang, and Lifu Huang. R2I-Bench: Benchmarking reasoning-driven text-to-image generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 12595–12630, 2025. doi: 10.18653/v1/2025.emnlp-main.636. URL https://aclanthology.org/2025.emnlp-main.636/.

Jaemin Cho, Yushi Hu, Jason Baldridge, Roopal Garg, Peter Anderson, Ranjay Krishna, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/446905abfec9db8856f7f7465bbd5552-Abstract-Conference.html.

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

Haiwen Diao, Jiahao Wang, Chenjing Ding, Hanming Deng, Jiangnan Chen, Ruixi Zhang, Ruohui Wang, Wenwen Tong, Xiangyu Fan, Yubo Wang, Yue Zhu, Yuwei Niu, Zhengqi Bai, Zhiqian Lin, Zhitao Yang, Zhongang Cai, Bo Yang, Chen Feng, Chengguang Lv, Guangjia Liu, Guanlin Wang, Hanyu Zhang, Haojia Yu, Hongcan Xiao, Hongli Wang, Huan Wu, Huaping Zhong, Jian Fang, Jianan Fan, Jiaqi Li, Jiefan Lu, Jing Zuo, Jingcheng Ni, Junxiang Xu, Linjun Dai, Mutian Xu, Peishen Yan, Penghao Wu, Ruijie Mao, Ruisi Wang, Shihao Bai, Shuang Yang, Shuya Yang, Shuyan Zheng, Silei Wu, Siying Li, Tao Chu, Tianbo Zhong, Tongxi Zhou, Weichao Luo, Weichen Fan, Wenhao Jia, Wenjie Gao, Xiangli Kong, Yan Li, Yang Yong, Zimo Wen, Zixuan Qian, Wenxiu Sun, Ruihao Gong, Quan Wang, Lewei Lu, Lei Yang, Ziwei Liu, and Dahua Lin. SenseNova-U1.5: Towards native unified visual intelligence, 2026. URL https://arxiv.org/abs/2609.11929.

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.

Tsu-Jui Fu, Wenze Hu, Xianzhi Du, William Yang Wang, Yinfei Yang, and Zhe Gan. Guiding instruction-based image editing via multimodal large language models. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f0e91b1314fa5eabf1d7ef6d1561ecec-Abstract-Conference.html.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. ELLA: Equip diffusion models with LLM for enhanced semantic alignment, 2024. URL https://arxiv.org/abs/2403.05135.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A. Smith. TIFA: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/papers/Hu_TIFA_Accurate_and_Interpretable_Text-to-Image_Faithfulness_Evaluation_with_Question_Answering_ICCV_2023_paper.pdf.

Yushi Hu, Reyhane Askari-Hemmat, Melissa Hall, Emily Dinan, Luke Zettlemoyer, and Marjan Ghazvininejad. Multimodal RewardBench 2: Evaluating Omni Reward Models for Interleaved Text and Image. *arXiv preprint arXiv:2512.16899*, 2025. doi: 10.48550/ARXIV.2512.16899. URL https://arxiv.org/abs/2512.16899.

Zican Hu, Xuyang Hu, Yiming Liu, Zuwei Long, Wei Liu, Yunzhuo Hao, Jiawei Gu, Linjie Li, Yu Cheng, Zhenhong Sun, Weibo Gu, Xing Sun, and Zhi Wang. Bridging interleaved multimodal reasoning as a unified decision process, 2026. URL https://arxiv.org/abs/2607.03748.

Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, and Ying Shan. SmartEdit: Exploring complex instruction-based image editing with multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8362–8371, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Huang_SmartEdit_Exploring_Complex_Instruction-based_Image_Editing_with_Multimodal_Large_Language_CVPR_2024_paper.html.

Zhangqi Jiang, Zheng Sun, Xianfang Zeng, Yufeng Yang, Xuanyang Zhang, Yongliang Wu, Wei Cheng, Gang Yu, Xu Yang, and Bihan Wen. Geditbench v2: A human-aligned benchmark for general image editing, 2026. URL https://arxiv.org/abs/2603.28547.

Siyu Jiao, Yiheng Lin, Yujie Zhong, Qi She, Wei Zhou, Xiaohan Lan, Zilong Huang, Fei Yu, Yingchen Yu, Yunqing Zhao, Yao Zhao, and Yunchao Wei. ThinkGen: Generalized thinking for visual generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14713–14723, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Jiao_ThinkGen_Generalized_Thinking_for_Visual_Generation_CVPR_2026_paper.html.

Xin Jin, Huanqia Cai, Zhen Li, Zechao Zhan, Dengyang Jiang, Aiming Hao, Yuming Jiang, Xiangpeng Yang, Chunle Guo, Peng Gao, Ming-Ming Cheng, and Steven C. H. Hoi. Z-Reward: Beyond Scalar Rewards by Internalizing Reasoning into Score Distributions. *arXiv preprint arXiv:2606.09076*, 2026. doi: 10.48550/ARXIV.2606.09076. URL https://arxiv.org/abs/2606.09076.

Amita Kamath, Kai-Wei Chang, Ranjay Krishna, Luke Zettlemoyer, Yushi Hu, and Marjan Ghazvininejad. Geneval 2: Addressing benchmark drift in text-to-image evaluation. *arXiv preprint arXiv:2512.16853*, 2025. URL https://arxiv.org/abs/2512.16853.

Hengjia Li, Liming Jiang, Qing Yan, Yizhi Song, Hao Kang, Zichuan Liu, Xin Lu, Boxi Wu, and Deng Cai. Thinkrl-edit: Thinking in reinforcement learning for reasoning-centric image editing, 2026. URL https://arxiv.org/abs/2601.03467.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training flow matching models via online RL. In *Advances in Neural Information Processing Systems*, volume 38, pp. 45650–45685, 2025a. doi: 10.52202/085713-1362. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/3a10c46572628d58cb44fb705f25cbbf-Abstract-Conference.html.

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.

Shiyu Liu, Yucheng Han, Peng Xing, Fukun Yin, Rui Wang, Wei Cheng, Jiaqi Liao, Yingming Wang, Honghao Fu, Chunrui Han, Guopeng Li, Yuang Peng, Quan Sun, Jingwei Wu, Yan Cai, Zheng Ge, Ranchen Ming, Lei Xia, Xianfang Zeng, Yibo Zhu, Binxing Jiao, Xiangyu Zhang, Gang Yu, and Daxin Jiang. Step1x-edit: A practical framework for general image editing, 2025b. URL https://arxiv.org/abs/2504.17761.

Chaojie Mao, Chen-Wei Xie, Chongyang Zhong, Haoyou Deng, Jiaxing Zhao, Jie Xiao, Jinbo Xing, Jingfeng Zhang, Jingren Zhou, Jingyi Zhang, Jun Dan, Kai Zhu, Kang Zhao, Keyu Yan, Minghui Chen, Pandeng Li, Shuangle Chen, Tong Shen, Yu Liu, Yue Jiang, Yulin Pan, Yuxiang Tuo, Zeyinzi Jiang, Zhen Han, Ang Wang, Bang Zhang, Baole Ai, Bin Wen, Boang Feng, Feiwu Yu, Gang Wang, Haiming Zhao, He Kang, Jianjing Xiang, Jianyuan Zeng, Jinkai Wang, Junjie Zhou, Ke Sun, Linqian Wu, Pei Gong, Pingyu Wu, Ruiwen Wu, Tongtong Su, Wenmeng Zhou, Wenting Shen, Wenyuan Yu, Xianjun Xu, Xiaoming Huang, Xiejie Shen, Xin Xu, Yan Kou, Yangyu Lv, Yifan Zhai, Yitong Huang, Yun Zheng, Yuntao Hong, Zhe Zhang, and Zhicheng Zhang. Wan-image: Pushing the boundaries of generative visual intelligence, 2026. URL https://arxiv.org/abs/2604.19858.

Yuwei Niu, Munan Ning, Mengren Zheng, Weiyang Jin, Bin Lin, Peng Jin, Jiaqi Liao, Chaoran Feng, Fanqing Meng, Kun-Peng Ning, Bin Zhu, and Li Yuan. WISE: World knowledge-informed semantic evaluation for text-to-image generation. In *Forty-third International Conference on Machine Learning*, 2026. URL https://github.com/PKU-YuanGroup/WISE.

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.

Linqing Wang, Zhiyong Xu, Ximing Xing, Yiji Cheng, Zhiyuan Zhao, Donghao Li, Tiankai Hang, Zhenxi Li, Jiale Tao, Qixun Wang, Ruihuang Li, Comi Chen, Xin Li, Mingrui Wu, Xinchi Deng, Shuyang Gu, Chunyu Wang, and Qinglin Lu. Promptenhancer: Taming your rewriter for text-to-image generation via fine-grained reward. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14895–14904, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Wang_PromptEnhancer_Taming_Your_Rewriter_for_Text-to-Image_Generation_via_Fine-Grained_Reward_CVPR_2026_paper.html.

Yibin Wang, Zhimin Li, Yuhang Zang, Chunyu Wang, Qinglin Lu, Cheng Jin, and Jiaqi Wang. Unified Multimodal Chain-of-Thought Reward Model through Reinforcement Fine-Tuning. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5315. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/e95e9f0c127aa1cfa2628adb2f3cb107-Abstract-Conference.html.

Jie Wu, Yu Gao, Zilyu Ye, Ming Li, Liang Li, Hanzhong Guo, Jie Liu, Zeyue Xue, Xiaoxia Hou, Wei Liu, Yan Zeng, and Weilin Huang. RewardDance: Reward Scaling in Visual Generation. *arXiv preprint arXiv:2509.08826*, 2025a. doi: 10.48550/ARXIV.2509.08826. URL https://arxiv.org/abs/2509.08826.

Keming Wu, Sicong Jiang, Max Ku, Ping Nie, Minghao Liu, and Wenhu Chen. Editreward: A human-aligned reward model for instruction-guided image editing. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/4d68af3056842229f8b356d686aec914-Abstract-Conference.html.

Tsung-Han Wu, Long Lian, Joseph E. Gonzalez, Boyi Li, and Trevor Darrell. Self-correcting LLM-controlled diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6327–6336, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Wu_Self-correcting_LLM-controlled_Diffusion_Models_CVPR_2024_paper.html.

Yongliang Wu, Zonghui Li, Xinting Hu, Xinyu Ye, Xianfang Zeng, Gang Yu, Wenbo Zhu, Bernt Schiele, Ming-Hsuan Yang, and Xu Yang. KRIS-Bench: Benchmarking next-level intelligent image editing models. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025b. doi: 10.52202/085713-5242. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/e619b285582fb12f4c3de3a507b8b99c-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.

Hanqing Yang, Qiang Zhou, Yongchao Du, Sashuai Zhou, Zhibin Wang, Jun Song, Tiezheng Ge, Cheng Yu, and Bo Zheng. Dda-thinker: Decoupled dual-atomic reinforcement learning for reasoning-driven image editing, 2026. URL https://arxiv.org/abs/2604.25477.

Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4387. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/bec783f119b969772c35243b1c801090-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.

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, 2026a. URL https://arxiv.org/abs/2608.04436.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. *arXiv preprint arXiv:2601.18734*, 2026b. doi: 10.48550/ARXIV.2601.18734. URL https://arxiv.org/abs/2601.18734.

Xiangyu Zhao, Peiyuan Zhang, Kexian Tang, Xiaorong Zhu, Hao Li, Wenhao Chai, Zicheng Zhang, Renqiu Xia, Guangtao Zhai, Junchi Yan, Hua Yang, Xue Yang, and Haodong Duan. Envisioning beyond the pixels: Benchmarking reasoning-informed visual editing. In *Advances in Neural Information Processing Systems*, volume 38. Curran Associates, Inc., 2025. doi: 10.52202/085713-5791. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/fe79898dcf078ec54b6feeea10ebb751-Abstract-Datasets_and_Benchmarks_Track.html. Datasets and Benchmarks Track.

Dian Zheng, Manyuan Zhang, Hongyu Li, Hongbo Liu, Kai Zou, Kaituo Feng, and Hongsheng Li. Uni-Edit: Intelligent editing is a general task for unified model tuning. *arXiv preprint arXiv:2605.21487*, 2026. URL https://arxiv.org/abs/2605.21487.

Sashuai Zhou, Qiang Zhou, Jijin Hu, Hanqing Yang, Yue Cao, Junpeng Ma, Yinchao Ma, Jun Song, Tiezheng Ge, Cheng Yu, Bo Zheng, and Zhou Zhao. Unified thinker: A general reasoning core for image generation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10597–10613. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.484. URL https://aclanthology.org/2026.acl-long.484/.

A CONTRACT SCHEMA AND COMPILATION

A.1 RUNTIME LAYOUT AND AN ILLUSTRATIVE INSTANCE

CE-v1 is the public JSON realization of the structured bridge, shared by offline teacher-generated targets and understanding-role predictions, with `goal`, `scope`, `ops`, `preserve`, and `avoid` fields. The following illustrative contract instantiates the clock task in figures 1 and 2.

```
{
  "goal": "Set analog to 4:05; keep digital clock unchanged.",
  "scope": {
    "mode": "local",
    "targets": [
      {"id": "t_a", "kind": "entity",
       "selector": "analog clock reading 10:20 in the source"},
      {"id": "t_d", "kind": "entity",
       "selector": "digital clock reading 14:35 in the source"},
      {"id": "t_s", "kind": "region",
       "selector": "all source content outside both clocks"}
    ],
    "edit": ["t_a"], "lock": ["t_d", "t_s"]
  },
  "ops": [{
    "id": "o1", "type": "set_state", "target": "t_a",
    "from": ["10:20"], "to": ["4:05"], "loc": "source position",
    "keep": ["dial markings", "case", "position"],
    "avoid": ["changing non-hand parts of the analog clock"]
  }],
  "preserve": ["digital clock: 14:35 and unchanged appearance",
               "all source content outside both clocks"],
  "avoid": ["additional edits"]
}
```

The source analog clock reads 10:20 and the digital clock reads 14:35. The instruction sets only the analog clock to the digital time plus 90 minutes, while leaving the digital clock unchanged. Resolving $14{:}35 + 90$ minutes gives $16{:}05$, displayed as $4{:}05$ on the analog dial. The public contract records this terminal state rather than passing the arithmetic instruction to the editor. The digital clock supplies source evidence and remains protected; reading its time does not make it an edit target. Its lock and explicit preservation clause map to the same invariant, while `t_s` excludes both clocks to avoid overlapping scene coverage. The analog clock contributes one change group; its retained attributes and the other protected content contribute deduplicated invariants. Private planning, rewards, expected judge answers, hidden benchmark IDs, and critiques are excluded from this JSON. Global and op-local preservation are interpreted with their respective scopes, then grouped without double counting. The `goal` summary must be entailed by the operative fields; goal-only obligations or conflicts fail the semantic coverage check rather than disappearing from execution checks. This entailment test belongs to pre-compilation semantic validation. A confirmed failure sets $b_E = v_C = 0$; a confident prompt check can still supply U-local supervision. An unresolved test abstains from execution scoring ($b_E = 0$).

A.2 CE-V1 TYPE PROFILE AND SUPPORTED COVERAGE

The following types define the public interface used by both prompt templates in section B. System-instruction compression does not alter these types or the shared response format.

```
Contract  := {goal: String, scope: Scope, ops: Operation[1..],
              preserve: String[], avoid: String[]}
Scope     := {mode: "local" | "global", targets: Target[1..],
              edit: ID[], lock: ID[]}
Target    := {id: ID, kind: "entity" | "region", selector: String}
Operation := {id: ID, type: Op, target: ID, from: String[],
              to: String[], loc: String,
              keep: String[], avoid: String[]}
Op        := recolor | replace | insert | remove | move
           | transform_text | set_state
ID        := nonempty string, unique within its target
             or operation list
```

All target references must resolve within `scope.targets`; `edit` and `lock` are disjoint ID sets, and operation targets belong to `edit`. Semantic overlap between differently described regions still requires semantic checking. The profile fixes roles and list structure but not symbolic visual meanings. Selectors refer to X and may combine category, attributes, ordinal order, and explicit spatial relations. A selector must resolve to one instance, or a determinate set expanded to individual source handles. Ambiguous sets are rejected. Visual evidence handles record source regions and are reused across dependent checks; source–output matching uses the source identity, not a new output-relative selector. The editor can still misground the same description; sharing a specification does not guarantee shared visual interpretation.

For insertion, the target record identifies the source placement region, `loc` refines placement, `to` describes the new entity, and `from=[]`. Removal requires a resolvable source target and `to=[]`, whose operation-specific meaning is absence, not a missing obligation. Move requires a source location and a destination in `to`. Recolor, replacement, text transformation, and state change require applicable source and terminal descriptions. A text transformation records the required final text, not just an instruction to reason about it. `set_state` is limited to observable terminal states, not unspecified creativity or hidden physical causes. Inter-edit dependencies are represented by jointly required final states, not a low-level rendering program. The private field `edit_sequence` describes semantic planning; order sensitivity of the serialized `ops` array is an empirical control, not assumed invariance.

The raw-JSON condition preserves accepted JSON rather than routinely sorting, renaming, or deduplicating its contents. Explicit serialization controls operate on separate copies of the editor condition. Semantic duplicate/contradiction checks prepare a grouping map for the compiler without changing that condition; uncertain equivalence fails validation. Each reference requirement separately records its supporting instruction phrase or default rule as annotation metadata. Unsupported operations, missing applicable states, contradictions, uncheckable requirements, and unresolved targets are reported on unfiltered predictions.

A.3 FIELD-TO-CHECK COMPILER

The frozen compiler maps each valid change to a group with the following dependency order: source target/region resolution, applicable source-state verification, then the operation-specific output requirement. Invariant groups reuse source evidence where relevant. Templates refer to source evidence handles, not independently re-resolved output selectors. For predicted contracts its sole requirement source is the extracted public JSON $C_{\text{seen}} \equiv \widehat{C}$, not the private plan or original instruction. Each group is associated with the contributing fields, such as `ops[0].to[0]`, `ops[0].keep[0]`, and `preserve[0]`. Duplicate clauses map to a single group so that repeated wording does not increase its reward weight.

Resolver state and execution order. The compiler state is a map from clause-local IDs to typed records $H = (\texttt{kind}, \texttt{source_region}, \texttt{members}, \texttt{status})$, where kind is entity, region, or

Field	Instantiated check	Evidence stage
scope.targets; ops.target/loc	resolve the specified instance or region	prerequisite in X
ops.from	verify the applicable source state	prerequisite in X
ops.type/to	test final state, destination, or absence	postcondition in (X, Y)
scope.edit/lock	compare protected entities and regions	paired preservation
ops.keep; preserve	test retained attributes at the specified scope	paired preservation
ops.avoid; avoid	check absence of forbidden effects at that scope	output or paired evidence

Table 6: Template families and their evidence dependencies. Template enumeration is deterministic; source grounding and visual assessment remain semantic operations.

new-entity specification, members identifies a resolved source set, and status is resolved, ambiguous, or absent. A frozen source resolver $\mathcal{R}_X$ creates these records using X and C only, never the edited output or reference answers. The handles are internal evidence state, not hidden benchmark IDs or extra generator-facing instructions. Source resolution is an automatic semantic operation; normalization and template enumeration are deterministic. The visual judge associates source handles with output evidence according to the frozen operation-specific rule.

```
check_contract(X, C_seen, Y):
  C = C_seen                              # never raw/private plan
  parse and structurally validate C; on failure return INVALID, 0
  H = source_resolver(X, C)               # semantic, may abstain
  check contradictions and goal/operative-field entailment
  if a violation is confirmed return INVALID, 0
  if entailment is unresolved return ABSTAIN, undefined
  record unresolved/ambiguous source bindings in H
  Q = instantiate_fixed_templates(C)   # fixed; no new facts
  associate each group with its contributing fields
  bind prerequisite checks in Q to H
  for each requirement group:
    evaluate prerequisites in X
    if prerequisite fails/is unresolved:
      group = false; downstream checks = unresolved
    else:
      evaluate operation-specific postconditions in (X, Y)
      group = conjunction of all applicable subchecks
  return groups, diagnostic trace, mean(groups)
```

Before aggregation, semantic overlap among invariants is resolved by the validation stage, not by silently adding a learned rewrite to the template compiler. The same procedure compiles predicted contracts for local execution; a public checklist uses the same strict grouping policy but not this contract parser.

For every group, all applicable prerequisites and postconditions must pass. A failed or unresolved prerequisite makes that group false once and leaves dependent subchecks unresolved in the diagnostic record. It does not create multiple negative reward atoms. This conjunction is used in local execution training, predicted-contract diagnostics, and public scoring; their *requirement sources* differ, not the reduction rule. Malformed, empty, unsupported, or confirmed-contradictory contracts receive zero execution and an invalid flag. Unresolved source bindings retain per-group diagnostics; both cases exclude the local execution loss. Global-loss eligibility follows Equation 8, with prompt validity assessed separately by the check. Canonical benchmark items require at least one change and one preservation group. Overlapping default and explicit preservation requirements are partitioned or deduplicated before counting; uncertain overlaps are adjudicated before benchmark freezing or rejected for a predicted contract.

Operation/example	Source prerequisite	Bound output requirement
recolor clock case blue	selected case resolves; source color matches	same case has the required blue color
replace clock by calendar	source clock/region resolves	requested calendar replaces it in that region
insert clock on wall	source wall/placement region resolves	new clock is present at that placement
remove selected clock	source clock resolves	that source instance is absent
move clock beside lamp	source clock and lamp resolve	same clock is at the specified destination
transform clock label	label region and source string resolve	that region contains the required final text
set analog clock to 4:05	analog clock resolves and reads 10:20	same clock displays 4:05

Table 7: Illustrative requirements for every supported operation. The state-change row follows figures 1 and 2; the other rows describe independent hypothetical tasks. Each example also carries separately deduplicated preservation requirements.

B Offline Supervision and Understanding-Role Prompts

B.1 Shared response protocol and direct supervision

The single-source templates below apply to A–D tasks. For these tasks, the offline teacher and understanding role receive the same source image X and original instruction u, and follow the same response protocol. Category E additionally uses the source/reference image roles defined in section C.1. Only the system instruction differs: s_T spells out the annotation rules in detail, whereas s_S states them compactly. Both instructions prescribe the same response schema. For a successful teacher completion, write

$$r_T = \text{serialize}\big(\texttt{<think>}, z_T, \texttt{</think>}, C_T\big), \tag{11}$$

where z_T contains the six private fields below and C_T is a CE-v1 JSON contract. The accepted *complete teacher response* is the understanding role's SFT target:

$$\mathcal{D}_{\text{SFT}} = \{(X, u, r_T) : F(X, u, r_T) = 1\}, \tag{12}$$

$$\mathcal{L}_{\text{SFT}}(\theta) = -\mathbb{E}_{\mathcal{D}_{\text{SFT}}} \sum_{t=1}^{|r_T|} \log p_\theta(r_{T,t} \mid s_S, X, u, r_{T,<t}). \tag{13}$$

Here F is the offline acceptance filter, and $r_{T,t}$ denotes a response token. The full accepted response is used unchanged: the private fields, including `contract_budget`, their delimiters, and the final JSON all receive supervision. We neither project the trace into fewer fields nor train on JSON alone. System, image, and user-input tokens provide context and are excluded from the response loss.

The six fields are `prompt_intent_text`, `image_evidence`, `scope_binding`, `edit_sequence`, `preserve_risks`, and `contract_budget`, in that order. The first five establish the semantic plan; the sixth records its target/operation inventory and intended JSON length without introducing new task facts. This private budget record is part of the shared annotation and prediction protocol, not an additional public interface or a hard decoding guarantee.

Teacher annotations C_T are training targets, not automatically correct benchmark oracles. Independently validated C^* and schema-blind Q_{public} have the distinct evaluation and post-training roles defined in the main text. Teacher synthesis and acceptance use only the designated training inputs; held-out reference annotations are never used to construct a test-time understanding-role response.

B.2 DETAILED TEACHER SYSTEM INSTRUCTION

SOURCE_IMAGE denotes the actual image in the multimodal message, not a caption or filename used as visual evidence. ORIGINAL_INSTRUCTION is the unmodified task text. The host instantiates the same length parameters in both system templates. Within an actual completion, each numbered private field occupies one line; the budget specification wraps only for readability below.

Listing 1: Detailed teacher system instruction for offline annotation. Its complete accepted response supervises the understanding role.

```
You are the offline teacher annotator for source-relative image editing.
Use only the supplied source image and original user instruction
as task evidence. Do not assume a separate image caption.
Infer a required answer or final state from that evidence when
the instruction calls for reasoning. Do not invent missing facts.

Your output has a private planning block followed by one JSON
object. Accepted complete responses supervise the student planner.
The student follows this same output protocol. At runtime, only
its validated JSON reaches the image editor and execution compiler.

PRIVATE OUTPUT FORMAT
Produce <think>, exactly six field lines in the order below,
then </think>. Each field occupies one line; use semicolons within
a line. Do not add a seventh field.
1. prompt_intent_text: requested changes and exact requested text.
2. image_evidence: task-relevant visible source facts.
3. scope_binding: source targets, locations, edit/lock boundaries.
4. edit_sequence: required user-level semantic edits and final states.
5. preserve_risks: protected content and forbidden side effects.
6. contract_budget: the budget record defined below.

The first five fields establish all task semantics. The sixth
only compiles those facts into a bounded interface plan. It must
not introduce objects, edits, text, attributes, regions, styles,
or constraints absent from the established plan.

CONTRACT BUDGET
Use this record on the sixth field line, replacing angle-bracket
metavariables with concrete values:
schema=goal_scope_ops_preserve_avoid;
target_count=<T>;op_count=<O>;
target_ids=[<all target IDs>];op_ids=[<all operation IDs>];
edit=[<editable target IDs>];lock=[<protected target IDs>];
text_ops=[<operation IDs that write text>];
answer_start=goal;answer_chars={{MIN_CHARS}}-{{MAX_CHARS}};
close=ops_array_then_preserve;minimal=yes;generator_readable=yes
T counts every record in scope.targets, including protected ones.
O counts user-level semantic operations, not rendering refinements.
The budget is a construction plan, not a decoder guarantee.

PUBLIC JSON: CE-v1 PROFILE
After </think>, emit exactly one JSON object and no Markdown fence,
comment, explanation, private trace, or extra top-level field.
Use the top-level order goal, scope, ops, preserve, avoid.

goal: a concise summary consistent with the operative fields.
It must not contain an obligation absent from those fields.
scope: an object with exactly mode, targets, edit, lock.
  mode: "local" or "global".
  targets: a nonempty array of records with exactly id, kind, selector.
    id: a unique nonempty string in the target list.
    kind: "entity" or "region".
    selector: a source-relative description resolving the target.
```

```
  edit: IDs of targets allowed to change.
  lock: IDs of targets protected against change.
  edit and lock must be disjoint; all IDs must exist in targets.
ops: a nonempty array of operation records, each containing:
  id: a unique nonempty operation ID.
  type: recolor, replace, insert, remove, move, transform_text,
        or set_state.
  target: one target ID from scope.edit.
  from: a list of applicable observable source descriptions.
  to: a list of required observable final descriptions.
  loc: a source-relative location or placement description.
  keep: a list of attributes to preserve for this operation.
  avoid: a list of forbidden effects for this operation.
preserve: a list of scene-level preservation requirements.
avoid: a list of scene-level forbidden effects.
Description arrays and ID arrays contain strings; targets and ops
contain the records defined above. Other leaf values, including
goal, selector, and loc, are strings. No extra keys are allowed.

For insert, target a source placement region, use from=[],
and describe the new entity in to. For remove, identify the
source instance and use to=[] to mean its required absence.
Other operations require nonempty from and to lists.
For move, to includes the destination; retain source identity.
For transform_text, to includes the concrete final text.
For set_state, specify a visually checkable terminal state.
Every target and operation must match the budgeted IDs and counts.
Rendering, lighting, shadows, blending, texture, and integration
details stay inside the corresponding semantic operation; they
must not create additional operations.

PRESERVATION AND EXACT TEXT
Preserve non-target source content unless the instruction permits
its change. Keep op-local and scene-level constraints at their
respective scopes. Do not add aesthetic changes not requested.
When the instruction specifies an exact string to appear in the
image, retain its case, spacing, punctuation, and characters in
the applicable text operation. Do not paraphrase that string.
Use valid JSON escaping; decoded string content must stay exact.
For text derived by translation, counting, or another requested
transformation, place the concrete answer established in the
private plan in to; do not replace it with a reasoning directive.

OUTPUT CHECK
Check JSON structure, budget counts and references, literal
coverage, goal consistency, and preservation scope before output.
Prefer a concise complete contract within the configured character
budget. Never omit a required fact or shorten an exact string just
to meet the length target. If a faithful concrete contract cannot
be constructed, output only CONSTRUCTION_FAILED. The host treats
that sentinel as a failed attempt, not a generator-facing contract.
```

B.3 COMPACT UNDERSTANDING-ROLE INSTRUCTION AND SHARED USER MESSAGE

Listing 2: Compact understanding-role system instruction. It requires the same six private fields and CE-v1 JSON output as the teacher.

```
Plan the requested image edit using only the source image and
original instruction. Resolve required reasoning; do not invent
facts. Return the shared teacher/student output format:

<think>
1. prompt_intent_text: requested edits and exact requested text
2. image_evidence: task-relevant visible source facts
3. scope_binding: source targets, locations, edit/lock boundaries
4. edit_sequence: semantic edits and resolved final states
5. preserve_risks: protected content and forbidden effects
6. contract_budget: the concrete one-line budget record below
</think>
Then emit one CE-v1 JSON object, with no fence or extra prose.
Use exactly these six private field lines in this order.
The sixth only compiles facts established in the preceding five.

Budget record (fill angle-bracket values; keep on one line):
schema=goal_scope_ops_preserve_avoid;target_count=<T>;op_count=<O>;
target_ids=[<IDs>];op_ids=[<IDs>];edit=[<IDs>];lock=[<IDs>];
text_ops=[<text-writing op IDs>];answer_start=goal;
answer_chars={{MIN_CHARS}}-{{MAX_CHARS}};
close=ops_array_then_preserve;minimal=yes;generator_readable=yes
T counts all scope.targets, including protected targets; O counts
semantic operations, not rendering refinements. Match the record
to the JSON. This plan is not a hard decoding guarantee.

CE-v1 keys, in order: goal, scope, ops, preserve, avoid.
goal: a short summary; no unique or contradictory obligation.
scope: {mode, targets, edit, lock}; mode is local or global.
targets: nonempty array of {id, kind, selector}; kind is entity or
region; selector is source-relative. edit/lock are disjoint ID
arrays referencing targets. IDs are nonempty strings, unique
within their respective target or operation list.
ops: nonempty array of {id, type, target, from, to, loc, keep, avoid}.
target references scope.edit; loc is source-relative. type is
recolor, replace, insert, remove, move, transform_text, or set_state.
from/to/keep/avoid and global preserve/avoid are string arrays;
other leaf values are strings. No extra keys.
For insert, target a placement region and set from=[]; to describes
the new entity. For remove, to=[] means the source target's absence.
Other operations need nonempty from/to. A move keeps source identity
and states its destination; text edits state the concrete final text;
set_state specifies a visually checkable terminal state.

Preserve non-target content unless change is permitted. Keep local
and global constraints scoped correctly. Do not split rendering,
lighting, blending, or texture into extra operations or add unasked
aesthetic changes. Preserve exact requested strings after JSON
decoding; derived answers must be concrete, not reasoning directives.
Check structure, references, budget, literal coverage, and goal/scope
consistency. Be concise without dropping requirements to meet length.
If no faithful concrete contract is possible, return only
CONSTRUCTION_FAILED. The private block never reaches the editor.
```

Listing 3: Single-source user-message template shared by the teacher and understanding role.

```
Source image (shared teacher/student input):
{{SOURCE_IMAGE}}

Original instruction:
{{ORIGINAL_INSTRUCTION}}

Requirements:
- Use the source image and original instruction as task evidence.
- Produce the six private fields in the specified order.
- The budget must only compile the preceding semantic plan.
- Output the budgeted targets/ops and all preservation constraints.
- Keep rendering refinements inside their semantic operation.
- Preserve every explicitly required output string exactly.
- After the private block, return only the CE-v1 JSON contract.
- Follow the system-defined failure sentinel if construction fails.
```

The compact instruction reduces the system-prompt text while retaining the same prescribed response content. System and input tokens remain distinct from private-trace and JSON-output tokens.

B.4 Offline acceptance and SFT data construction

The filter checks response extraction, field order, schema and reference validity, budget consistency, exact-text retention, source grounding, semantic coverage, and conflicts. Structural checks are deterministic; semantic acceptance remains fallible; structural validity alone does not establish semantic correctness. The primary SFT path accepts or rejects a teacher completion, without rewriting its contents.

Listing 4: Offline synthesis and direct teacher-response supervision. This is data-construction logic, not the runtime editor interface.

```
synthesize_training_example(X, u, max_attempts):
  for attempt in 1..max_attempts:
    r_T = teacher(s_T, shared_user_message(X, u))
    extract six-field trace z_T and CE-v1 contract C_T
    if sentinel, ambiguous extraction, or invalid response format:
      record rejection; continue
    if schema, references, or budget checks fail:
      record rejection; continue
    if literal, grounding, or semantic coverage checks fail:
      record rejection; continue
    retain (X, u, r_T)                # exact complete response
    train understanding role on r_T conditioned on s_S and the same (X, u
    )
    return ACCEPT
  return REJECT
```

A rejected sample may be regenerated by the teacher under a predeclared retry limit; only a newly accepted complete response can become the primary SFT target. There is no automatic contract repair between teacher generation and supervised target storage. In particular, modifying a JSON object while retaining its original private plan would break direct response supervision and can invalidate the plan–contract agreement.

Literal and length conventions. Extract explicitly requested output literals from u under a frozen rule and associate each occurrence with its intended text operation. Compare decoded literal values by their UTF-8 bytes without case folding, trimming, or Unicode normalization; decode JSON escapes first. Repeated literals on different targets need distinct coverage associations. Translation, arithmetic, or other derived answers require semantic checks: a literal checksum cannot validate reasoning. Contract literal retention and OCR/human image-text accuracy are separate metrics. The

configured character length counts Unicode code points in the JSON string, including formatting but excluding the private block. Neither model may remove a required fact or shorten an exact literal merely to fit that target.

B.5 Understanding-role inference and information boundaries

At inference, the understanding role receives s_S and the shared (X, u) message and emits the same response structure as its teacher targets. The host separates z_θ from $\widehat{C}$ without rewriting the accepted JSON; only $(X, \widehat{C})$ enters the raw-JSON editor. The execution compiler reads that identical $\widehat{C}$. The six private fields, system/user planner messages, teacher annotations, and reference answers are not concatenated to the editor or used to fill missing contract requirements. This is an information boundary, not a change in the understanding role's learned output format.

Malformed responses and the failure sentinel count as failed attempts. The protocol excludes automatic repair of predicted conditions; exhausted test items remain in the evaluation denominator. The identity $C_{\text{seen}} \equiv \widehat{C}$ specifies that G and the execution compiler consume the same public condition. A natural-language rendering is a separate interface variant and must retain the condition's requirements to support the same execution target.

C CE-Bench and Configuration Definitions

C.1 Taxonomy and task allocation

CE-Bench organizes 1,200 tasks by their primary source of difficulty: source grounding, exact symbolic content, reasoning, coupled composition, and cross-image reference use. The five mutually exclusive categories have 300/240/240/240/180 items, respectively (table 8). The four secondary categories within each primary category describe its capability coverage. CE-v1 in section A names the prompt schema, not a benchmark version.

Category boundaries. A tests which visible source entity or region should change. A3 uses directly visible attributes; selecting a target through logic, knowledge, or computation belongs to C. A2 covers directly stated spatial relations, whereas C2 requires inference or comparison to resolve the target. B tests exact text, symbols, numbers, and structured content, including preservation of non-target text and layout. A supplied replacement value belongs to B; a value that must first be calculated belongs to C. C covers both question-to-edit and inferred-state tasks, grouping them by the reasoning needed to derive the target or terminal state rather than by interrogative wording. D tests complete execution of multiple principal changes, including dependent edits, relational rearrangements, and mixed operations. E tests the use of reference images while respecting their roles and preserving source content.

Mutually exclusive primary labels. Assign exactly one primary label using precedence E>D>C>B>A:

1. E if solving the task requires at least one additional reference image.
2. Otherwise D if at least two principal output changes or coupled editing obligations must be jointly fulfilled. Ordinary preservation obligations alone do not turn a single-change task into D.
3. Otherwise C if the target choice or terminal state must be inferred rather than being directly specified and visibly grounded.
4. Otherwise B if success primarily depends on exact text, symbolic, or document content.
5. Otherwise A for source-grounded editing.

Preservation accompanies every category. For example, transferring a jacket from another image to the second person in the source is E even though it also requires ordinal grounding and identity preservation.

Multi-image inputs and reference roles. Each task has input $(X_{\text{src}}, \mathcal{I}_{\text{ref}}, u)$ and requires one edited source image Y. A–D have $\mathcal{I}_{\text{ref}} = \varnothing$; E uses task-provided reference images, with multi-reference tasks in E4. References supply the entity, appearance, pose, or layout requested by u; the source determines the editable scene and protected content. Stable image identifiers bind each

Table 8: **CE-Bench taxonomy.** Primary categories A–E contain 300/240/240/240/180 items (25%/20%/20%/20%/15%). Secondary rows describe capability coverage within each primary category.

ID	Category
A. Source-Grounded Editing (300)	
A1	Instance / Ordinal Grounding
A2	Spatial / Relational Grounding
A3	Attribute-Conditioned Selection
A4	Part / Region Grounding
B. Text & Structured-Content Editing (240)	
B1	Exact Text Editing
B2	Multi-Text & Text Preservation
B3	Document / UI Layout Editing
B4	Numeric / Symbolic / Structured Visual Editing
C. Reasoning-Grounded Editing (240)	
C1	Visual Question-to-Edit
C2	Logical / Spatial / Quantitative Reasoning
C3	Temporal / Causal / Physical Inference
C4	Knowledge / Conceptual / Procedural Reasoning
D. Coupled & Compositional Editing (240)	
D1	Multi-Target Composition
D2	Dependent / Coupled Edits
D3	Relational Rearrangement
D4	Mixed Operations + Preservation
E. Reference-Conditioned Multi-Image Editing (180)	
E1	Entity / Subject Transfer
E2	Attribute / Appearance Transfer
E3	Pose / Layout / Structural Transfer
E4	Multi-Reference Composition & Reasoning
Total: 1,200	

referenced entity or attribute to its input image. The evaluation protocol requires the editor and evaluator to receive all task input images with these roles preserved. A reference jacket must not transfer its wearer's identity when source identity is protected; a pose reference must not replace source clothing unless requested.

E1–E3 distinguish the information supplied by a reference: entity identity, appearance, or structure. E4 requires joint use of at least two references, including selection among references or transfer of different attributes from different images. These task-provided images are distinct from the baseline *reference edits* used by the pairwise reward model: reference edits are comparison outputs, not extra user conditioning images. A comparison under this protocol requires identical reference availability and image ordering for each task, with source/reference roles preserved by model-specific conditioning paths.

Orthogonal capability tags. Primary labels determine aggregate scoring; nonexclusive tags retain cross-cutting capabilities. Tags cover grounding (ordinal, spatial, part), reasoning (visual, logical, spatial, quantitative, temporal, causal, physical, factual, conceptual, procedural), operations, reference use, and preservation constraints. For example, an E task can carry ordinal-grounding and causal-reasoning tags without being counted again in A or C.

Aggregation. AC, CCR, and Pres. retain the item-level definitions in equation (10). For any such metric M, the overall item mean is

$$\begin{aligned} M &= \sum_{c\in\{A,B,C,D,E\}} \frac{N_c}{1200} M_c, \qquad M_c = \frac{1}{N_c} \sum_{n\in\mathcal{D}_c} m_n, \\ &= 0.25M_A + 0.20M_B + 0.20M_C + 0.20M_D + 0.15M_E. \end{aligned} \tag{14}$$

This aggregation uses category-size weights, not an equal-weight category macro-average. Failed outputs remain in their assigned category and denominator. Nonexclusive tags support conditional analyses without duplicating items in the overall metric. The A–E columns of the reported tables follow this taxonomy and retain the stated category sizes.

C.2 Output-based annotation and compliance metrics

The annotation protocol derives public obligations directly from $(X_{\mathrm{src}}, \mathcal{I}_{\mathrm{ref}}, u)$, with no task references for A–D, without requiring a submitted prompt schema. Requirement groups specify the requested output before model predictions are assessed. A separate reference $C^\star$ supports prompt checks; the global evaluator scores the source image, request, task-provided references when present, and final output directly. Teacher responses instead supply SFT targets. These are distinct roles: a fluent teacher output is not automatically an evaluation oracle.

One change group contains target grounding, the operation, and its required terminal state; grounding is a prerequisite, not a separately counted success. An invariant group binds a protected region/entity and property. Overlapping default and explicit preservation constraints are deduplicated. Every applicable subcheck must pass; an unresolved prerequisite fails its group once. AC, CCR, and Pres. follow equation (10), with failed outputs retained. For each item, $\mathcal{G}_n^{\mathrm{pres}}$ is the deduplicated subset of preservation/invariant groups among the K_n total groups; p_{nj} is the same binary judgment as a_{nj} on this subset, not a separate evaluator. Canonical items contain at least one preservation group, so P_n is defined. Pres. first averages these judgments within each item and then averages P_n equally over all N items. It is not a pooled, requirement-weighted average. For example, two passing preservation groups out of three give $P_n = 2/3$; that item has the same $1/N$ weight as an item with eight preservation groups. The formulas yield fractions in $[0, 1]$; tables multiply them by 100. Class A–E scores in table 1 are class-specific CCR under table 8, not atomic scores. Their overall CCR uses the actual category counts in equation (14). Because CCR requires all groups to pass, class differences can reflect both task difficulty and the number of requirements. Category CCR therefore measures complete task fulfillment rather than difficulty independently of requirement count.

C.3 Configuration categories and the B0 baseline

A direct editor receives the original instruction without an explicit PE policy. Planner + Generator uses an external understanding model to produce conditioning for an editor. In each DuplexBridge

Table 9: **Wan-Image communication and role controls.** Left: SFT interface variants. Right: separately trained role variants share B_0. The two experiments report CCR (%); rows across panels are not paired. Full metric tables: tables 10 and 23.

Interface (SFT)	CCR
Fact-matched prose	57.00
Flat key-value	56.67
Flat JSON	57.42
Grouped key-value	59.00
CE-v1 grouped JSON	59.42

Training configuration	CCR
B0: matched SFT	59.42
U-only RL	66.67
G-only RL	67.50
Joint RL, ungated	70.92
DuplexBridge, full	77.08

configuration, the named unified model uses its own understanding capability to enhance the prompt for its own generation capability. No external understanding module is used in this forward path. BAGEL, Qwen-2511, and Wan-Image-7B each follow this native U-to-G setup; mechanism controls use Wan-Image-7B throughout.

B0 uses the same native U-to-G path with a common supervised warm-start, before DuplexBridge RL. The understanding policy is initialized with the complete teacher response, including private planning and public CE-v1 fields; the generator starts from a common initialization across the RL controls. Direct/no-PE and native-PE/no-CE-v1-SFT are separate Wan-Image controls. This separates the effects of PE, supervised initialization, and subsequent RL. Qwen-2511 denotes Qwen-Image-Edit-2511 throughout, including its direct-editor, B_0, and DuplexBridge configurations.

D Information-Matched Interfaces and Check Construction

Table 10: Wan-Image interface ablations at SFT. Rows compare prose, flat and grouped structures, and CE-v1 variants without the redundant goal or with shuffled fields. AC and CCR are percentages.

Interface	CE AC	CE CCR
Fact-matched prose	84.38	57.00
Flat key-value	84.16	56.67
Flat JSON	84.55	57.42
Grouped key-value	85.09	59.00
CE-v1 grouped JSON	85.28	59.42
CE-v1 without redundant goal	85.17	59.17
CE-v1 shuffled fields	85.02	58.92

The SFT-stage interface comparison specifies matched facts, training data, training budget, and prediction source. Fact-matched prose gives 57.00 CCR; flat key–value and flat JSON give 56.67 and 57.42; grouped key–value and grouped JSON give 59.00 and 59.42. Removing a redundant goal gives 59.17 and shuffled fields give 58.92. The no-goal condition preserves information only if the summary contains no unique obligation. The intended factor is organization and serialization of the same predicted content. Grouping raises CCR in both syntaxes: 56.67 to 59.00 for key–value and 57.42 to 59.42 for JSON. The grouped-syntax difference is smaller (59.00 versus 59.42), consistent with organization mattering more than JSON syntax. These endpoint comparisons do not isolate token length or truncation effects. The SFT comparison is distinct from the structure–gating RL comparison in table 26.

All construction conditions retain the same global evaluator and prompt checks and use the same source contract. Source fidelity measures coverage of source obligations with correct target, scope, state, and polarity; original-task recall instead uses the original task inventory. Duplication measures redundant checks. These three diagnostics use all rubric attempts. Human agreement is measured on an adjudicated content-matched subset. It assesses execution-check construction, rather than human alignment of CE-Bench final-image scoring. Source fidelity and task recall use the full set of construction attempts, whereas this agreement is conditional on matched content.

Table 11: Wan-Image execution-check construction. Fidelity, task recall, and duplicates use all rubric attempts. Human agreement is conditional on an adjudicated content-matched subset. The final column is CCR after separate retraining, not fixed-output rescoring.

Construction	Fidelity	Recall	Dup.	Human agr.	Retrained CCR
Same-contract prose rubric	95.11	88.73	4.28	87.14	74.92
Same-contract atomic reconstruction	95.46	89.12	3.17	87.36	75.58
Compiled execution checks	98.21	91.68	0.73	87.92	77.08

The compiled checks give 98.21 source fidelity, 91.68 original-task recall, and 0.73 duplicates, compared with 95.46, 89.12, and 3.17 for atomic reconstruction. Human agreement changes from 87.36 to 87.92 on matched content. The CCR column comes from separate retraining with each construction method: 74.92 for prose, 75.58 for atomic reconstruction, and 77.08 for compilation. It is not a rescore of a single frozen output set. Higher retained-specification coverage does not itself guarantee accurate visual judgments. All three constructions take requirements from the received contract. A separate reference-source ablation uses the original task for G-local and reaches 75.00 CCR, compared with 77.08 for received-condition supervision (table 19). The construction comparison and reference-source comparison therefore address different questions.

E Bidirectional Training and Reward Details

E.1 Roles, initialization, and supervision boundary

Native PE refers to the model's own understanding-to-generation path. Teacher responses and reward evaluators provide training supervision; they do not replace the model's understanding role when constructing its editing condition. The update formulation separates active understanding parameters θ, active generation parameters ϕ, and any shared frozen parameters ψ. The inactive role remains unchanged during its partner's update; role calls rebuild context and clear private caches before generation. Role isolation additionally requires that updating θ leave $G_\phi(\cdot \mid X, C; \psi)$ unchanged for fixed X, C, and conversely for U when updating ϕ. Adapter activation in a reused conditioning encoder must respect this requirement; clearing the KV cache alone does not ensure it. All detailed training-direction and mechanism controls in this appendix use Wan-Image-7B. B0 includes native PE and a common SFT warm-start, before RL.

Enhancer SFT uses the entire unchanged accepted teacher response: the six-line private block, delimiters, and CE-v1 JSON. Only the system instruction is shortened (section B.1); final JSON extraction is an inference handoff, not a training-target projection. θ_0 denotes this SFT initialization and ϕ_0 the common generator initialization. All matched RL controls share these states and access to the same verified references, local checks, execution checks, and the same continuous global evaluator. Gate-only controls retain all four loss components. No annotated reference condition $C^\star$ enters inference.

E.2 Local prompt check and reward confidence

Local reward components lie in $[0, 1]$. $R_{\text{format}} \in \{0, 1\}$ is exact schema conformance. For R_{plan}, a frozen semantic matcher builds a bipartite graph between predicted and independently verified reference requirements. Edges require compatible target identity, operation, terminal value, and invariant scope. Maximum one-to-one matching gives T matches; with P predicted and R reference groups, $R_{\text{plan}} = 2T/(P + R)$. Extra clauses reduce precision, missing clauses reduce recall, and uncertain matching abstains. Invalid predictions receive zero. R_{clear} averages a no-contradiction judgment and the resolved-source-binding fraction, with malformed/empty predictions scored zero. Thus $R_U^{\text{loc}} = s_C = (R_{\text{format}} + R_{\text{plan}} + R_{\text{clear}})/3$ depends on the prompt, not the sampled output image.

G's local reward in equation (6) combines changes and invariants from Q_{exec} with quality excluding instruction compliance. Preservation is not duplicated. Accepted execution specifications require nonempty change and invariant sets. Undefined means are excluded, not silently renormalized. A

semantically wrong but checkable instruction can therefore receive a high local execution score without a high global outcome score.

Judgment reliability and score magnitude are separate quantities. Let z_C, z_E, z_Q denote reliability estimates for the prompt, execution, and quality judgments, with any estimator and threshold fixed using independent development labels. Let r_E indicate resolvable execution groups and r_F valid images, a nonempty fixed reference set, the declared judgment boundary, and finite global scores:

$$c_{\text{audit}} = \mathbf{1}[C^\star \text{ available}]\mathbf{1}[z_C \geq \tau_{aC}], \tag{15}$$

$$v_C = \mathbf{1}[R_{\text{format}} = 1]\mathbf{1}[R_{\text{plan}} \geq \tau_p]\mathbf{1}[R_{\text{clear}} \geq \tau_c]\mathbf{1}[\text{required targets resolved}], \tag{16}$$

$$b_E = \mathbf{1}[z_E \geq \tau_{aE}]r_E, \qquad b_Q = \mathbf{1}[z_Q \geq \tau_{aQ}], \qquad b_F = r_F, \tag{17}$$

$$h_E = \mathbf{1}[R_{\text{exec}} \geq \tau_e], \qquad m_{U\to G} = c_{\text{audit}} v_C, \qquad m_{G\to U} = b_E h_E, \tag{18}$$

$$e_U^{\text{loc}} = c_{\text{audit}}, \qquad e_G^{\text{loc}} = b_E b_Q, \qquad e_U^{\text{glob}} = b_F m_{G\to U}, \qquad e_G^{\text{glob}} = b_F m_{U\to G}. \tag{19}$$

Malformed, empty, unsupported, or contradictory prompts have $b_E = v_C = 0$. A confidently identified invalid prompt can retain local check eligibility. A wrong but checkable prompt retains G-local supervision; passed execution trust also admits U-global feedback. A correct prompt that passes prompt trust retains G-global feedback when execution fails; failed execution trust excludes U-global feedback. An invalid global judgment blocks both global terms, not the local terms. Missing $C^\star$ blocks its dependent check terms, but does not by itself invalidate $F_{\text{global}}(X, u, \widehat{Y}; \mathcal{R}_{X,u})$. A score near 0.5 can reflect similar candidates; it is not an automatic abstention criterion. The composite G-local reward requires execution and quality to be usable; this condition does not test correctness with respect to u.

Verified $C^\star$ is part of the supervision budget for U-local and prompt-trusted G-global, whereas G-local and execution-trusted U-global need no such reference. Matched controls must therefore share reference access, including its task coverage. Accepted SFT targets C_T do not automatically qualify as verified references.

Execution threshold and critical requirements. The gate uses the average group score R_{exec}, not the change-only score. With $K-1$ passing groups and one failed group, $R_{\text{exec}} = (K-1)/K$; if this meets τ_e, the gate can accept despite a missed change or preservation failure. Thus the digital-clock violation in figure 1(a) closes U-global only if execution trust fails. Grouping avoids counting one obligation repeatedly but does not prevent this dilution by numerous distinct invariants. Execution trust is an average-score criterion rather than an all-critical-requirements criterion.

E.3 Global evaluator and score semantics

$F_{\text{global}}(X, u, \widehat{Y}; \mathcal{R}_{X,u})$ compares the final image with fixed baseline edits under the original request. Its criteria include task fulfillment, protected source content, and visual quality. The rubric, image ordering, reference-generation checkpoint and seeds, and score convention are fixed before RL. Baseline reference edits depend on (X, u) and are independent of the candidate contract. The same reference set is used for every candidate in a task group, so comparisons within the group share a common reference.

The raw-token convention in equation (7) gives a bounded, relative preference score. Changing reference difficulty changes its meaning; a value of 0.5 is not a calibrated 50% chance of completing the task. The evaluator does not read $\widehat{C}$, U's private reasoning, or verified contract $C^\star$. $C^\star$ is a prompt-check reference, distinct from reward reference images $\mathcal{R}_{X,u}$. Binary execution passes and public CCR retain their own definitions. Section F describes the RM and its dedicated evaluation protocol.

E.4 Candidate aggregation and conditional global feedback

For fixed (X, u), draw J complete responses under behavior policy θ_{old}, then L images for each valid candidate under one frozen generator revision. Fix $\mathcal{R}_{X,u}$ before sampling the J responses and reuse it for every candidate and image. Invalid responses may retain local check eligibility but have

no image evidence; unused image slots are logged, not silently replaced. For candidate j, let

$$n_j = \sum_{\ell=1}^{L} e^{\text{glob}}_{U,j\ell}, \qquad \bar{e}_j = \mathbf{1}[n_j > 0], \qquad \overline{R}^{\text{glob}}_j = \frac{\sum_\ell e^{\text{glob}}_{U,j\ell} R_{\text{global},j\ell}}{n_j}. \tag{20}$$

The mean is defined only for $n_j > 0$. Eligible prompt candidates have equal weight, not weight proportional to the number of accepted images. $R^{\text{loc}}_{U,j}$ and $e^{\text{loc}}_{U,j}$ are independent of the sampled images. No arbitrary zero enters an ineligible component's reward distribution.

For a fixed prompt, unconditional downstream value would be

$$V_\phi(X, u, C; \mathcal{R}_{X,u}) = \mathbb{E}_{Y \sim G_\phi(\cdot|X,C)}[F_{\text{global}}(X, u, Y; \mathcal{R}_{X,u})]. \tag{21}$$

The gated mean instead uses the accepted subset:

$$V^{\text{accepted}}_\phi(X, u, C; \mathcal{R}_{X,u}) = \mathbb{E}[F_{\text{global}}(X, u, Y; \mathcal{R}_{X,u}) \mid e^{\text{glob}}_U = 1, X, u, C]. \tag{22}$$

Continuous scoring distinguishes graded outcomes within that subset; it does not eliminate selection by the execution threshold or reward prompts directly for a higher probability of passing that threshold. Finite rollout counts also affect which candidates have a defined mean. Partner adaptation is therefore described through the actual gated objective, not an assertion that it maximizes equation (21) without selection bias.

Finite-sample selection. Fix (X, u, C), $\mathcal{R}_{X,u}$, the generator, and evaluator configuration, and let $p(C) = \Pr(e^{\text{glob}}_U = 1 \mid X, u, C)$. Assume conditionally i.i.d. image–judgment pairs, integrable rewards, and $\mathbb{E}_{C \sim \pi_{\theta_{\text{old}}}}[a_L(C)] > 0$. Then

$$a_L(C) = \Pr(n_j > 0 \mid C) = 1 - (1 - p(C))^L, \qquad q_L(C) = \frac{\pi_{\theta_{\text{old}}}(C \mid X, u) a_L(C)}{\mathbb{E}_{C' \sim \pi_{\theta_{\text{old}}}}[a_L(C')]}. \tag{23}$$

Here $\pi(C \mid X, u)$ marginalizes the internal interpretation, and q_L is the distribution of an individually sampled candidate conditioned on having an accepted image, before group-size and reward-variance filtering. For $p(C) > 0$, the accepted-image mean is unbiased for $V^{\text{accepted}}_\phi(X, u, C)$ conditional on $n_j > 0$ under these sampling assumptions. This fact does not make the U-global policy update an unbiased gradient of either an unconditional return or a normalized selective-risk objective: its candidate population, group-relative advantages, and clipping also change the update. Equal weighting among eligible candidates does not undo $a_L(C)$.

For $0 < p(C) < 1$, the ordinary law of total expectation gives

$$V_\phi(C) = p(C) V^{\text{accepted}}_\phi(C) + (1 - p(C)) V^{\text{rejected}}_\phi(C), \tag{24}$$

with $(X, u, \mathcal{R}_{X,u})$ suppressed throughout this selection analysis. Consequently, ranking prompts by accepted return can reverse their unconditional ranking. As an illustrative calculation, a prompt with acceptance 0.1, accepted return 1, and rejected return 0 has unconditional value 0.1; another with acceptance 0.9, accepted return 0.8, and rejected return 0 has value 0.72. The accepted-return ordering favors the first. Acceptance coverage and unconditional task outcome therefore remain distinct from return conditional on acceptance.

Generator updates freeze U, fix one sampled prompt, and compare L stochastic image trajectories against the same $\mathcal{R}_{X,u}$. G-local and G-global form separate eligible image groups. Neither requires high execution. A failed prompt check excludes only G-global, not local execution learning. Groups are regenerated after partner updates rather than pooling behavior-policy revisions.

E.5 Four separately normalized group-relative losses

For $t \in \{U\text{loc}, U\text{glob}, G\text{loc}, G\text{glob}\}$ in a rollout group, let $I_t = \{i : e_{ti} = 1\}$, using candidate masks for U and image masks for G. Compute μ_t, σ_t only over I_t:

$$A_{ti} = \begin{cases} (R_{ti} - \mu_t)/(\sigma_t + \epsilon), & i \in I_t,\ |I_t| \geq 2,\ \sigma_t > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{25}$$

Groups with fewer than two eligible samples or zero variance are skipped for that component, including KL. For a non-skipped group,

$$\mathcal{L}_t = \frac{1}{|I_t|} \sum_{i \in I_t} \left[-\min\{\rho_{ti} A_{ti}, \mathrm{clip}(\rho_{ti}, 1-\varepsilon, 1+\varepsilon) A_{ti}\} + \beta_t \mathcal{D}_{ti} \right]. \tag{26}$$

ρ is the current/behavior likelihood ratio and $\mathcal{D}$ the frozen-reference policy penalty. Each role combines its local loss with a global loss weighted by λ_U or λ_G, as in equation (9). The complete U response is the action. The U surrogate uses text-policy likelihoods, and the G surrogate uses stochastic-trajectory likelihoods for flow generation (Liu et al., 2025a), rather than likelihoods of deterministic ODE outputs. A sample excluded from one loss may remain in its local loss. Positive raw reward does not imply a positive centered advantage. The gates do not establish a causal failure label. Rewards, masks, and group normalization statistics are held fixed during optimization of their sampled batch; gradients pass through the active policy surrogate, not through judge calls or discrete acceptance decisions.

Training protocol summary. The following pseudocode summarizes alternating updates and the separate eligibility rules for local and global losses.

```
freeze RM, score convention, rubric, and reference sets
initialize U and G from their common B0 states
repeat according to the configured alternation schedule:
  U phase: freeze G; snapshot the U behavior policy
    fix cached reference images for each task
    sample J complete U responses
    check each response; extract each valid public condition C
    sample L images from the frozen G conditioned on (X, C)
    compute local checks, original-task scores, and four masks
    average U-global scores over accepted images for each C
    build separate U-local and U-global eligible candidate groups
    normalize and optimize each non-skipped loss, with its own KL
  G phase: freeze U; snapshot the G behavior policy
    fix each task's reference images; sample a condition C from U
    sample L stochastic G trajectories conditioned on (X, C)
    compute local checks, original-task scores, and four masks
    build separate G-local and G-global eligible image groups
    normalize and optimize each non-skipped loss, with its own KL
  discard the sampled groups before changing the partner policy
```

F Reasoning-Distilled Reward: Formulation and Evaluation

The reasoning-distilled RM supplies the global reward in the reported DuplexBridge experiments. This section documents its formulation and the dedicated RM evaluation.

F.1 Decision probabilities and deployment interface

Let ℓ_v be the logit for vocabulary token v at the judgment boundary b. For a fixed logit temperature $T_{\mathrm{logit}} > 0$, define

$$P_v = \frac{\exp(\ell_v / T_{\mathrm{logit}})}{\sum_{w \in \mathcal{V}} \exp(\ell_w / T_{\mathrm{logit}})}, \qquad r_{\mathrm{raw}} = P_{\texttt{Yes}}, \qquad r_{\mathrm{bin}} = \frac{P_{\texttt{Yes}}}{P_{\texttt{Yes}} + P_{\texttt{No}}}. \tag{27}$$

Logit temperature is distinct from teacher trace sampling temperature. Raw and binary scores are different: probabilities $P_{\texttt{Yes}} = 0.20$ and $P_{\texttt{No}} = 0.30$ give $r_{\mathrm{raw}} = 0.20$ and $r_{\mathrm{bin}} = 0.40$. Changing the convention changes both the scalar reward and the compatible distillation target. The raw-token formulation retains other-token mass, while binary normalization conditions on the two judgments.

A three-bucket representation retains the probability mass needed to define raw-token scores:

$$\begin{aligned} \mathbf{p}_\omega &= (P_{\texttt{Yes}}, P_{\texttt{No}}, 1 - P_{\texttt{Yes}} - P_{\texttt{No}}), \\ \mathbf{q}_T &= (Q_{\texttt{Yes}}, Q_{\texttt{No}}, 1 - Q_{\texttt{Yes}} - Q_{\texttt{No}}). \end{aligned} \tag{28}$$

The third bucket aggregates other-token mass without introducing a new reward head. Full-vocabulary matching also retains this mass; binary matching instead uses $(r_{\text{bin}}, 1 - r_{\text{bin}})$. These are distinct distillation objectives.

The one-token formulation assumes that the exact rendered `Yes` and `No` answers each correspond to one tokenizer token at b, including whitespace conventions. For a multi-token answer, the corresponding quantity is a sequence probability rather than the single-token probability used here. No answer token is included in the scored prefix. Teacher reasoning must finish at a valid boundary before logits are collected. The direct student scores that boundary after prompt/visual prefill and never decodes or receives the teacher's reasoning.

F.2 Distillation objective and its single-step interpretation

For pairwise input $x = (X, u, Y_A, Y_B, i)$, i is the shared rubric and `Yes` denotes a preference for candidate A. A frozen thinking teacher and a direct student define

$$\begin{aligned} z \sim \pi_{\bar{\omega}}(\cdot \mid x, m_T), \quad Q_T(v \mid x, z) = \pi_{\bar{\omega}}(v \mid x, m_T, z, b), \\ P_\omega(v \mid x) = \pi_\omega(v \mid x, m_S, b). \end{aligned} \tag{29}$$

The modes m_T, m_S select thinking and direct scoring. The judgment boundary b follows the teacher's reasoning and precedes its answer token. The student receives neither the teacher trace z nor its cached states. Let $\mathbf{q}_T$ and $\mathbf{p}_\omega$ denote teacher and student judgment distributions on the same outcome space. Soft judgment distillation minimizes

$$\mathcal{L}_{\text{KD}} = \mathbb{E}_{x,z} D_{\text{KL}}(\text{sg}[\mathbf{q}_T(x, z)] \| \mathbf{p}_\omega(x)) \,. \tag{30}$$

Here sg stops gradients through the teacher target. Only the student is updated; the objective places no loss on teacher reasoning tokens. Standard prefix-level OPSD matches conditional distributions on student rollouts (Zhao et al., 2026b). With a single judgment token and no student answer in the teacher prefix, the only scored prefix is empty. Sampling a `Yes`/`No` answer adds no trajectory-matching benefit when both logits are already known. Collecting image pairs from the current generator is generator-side on-policy data collection, distinct from RM prefix-level on-policy distillation.

For $\bar{\mathbf{q}}_T(x) = \mathbb{E}_z[\mathbf{q}_T(x, z)]$, forward KL obeys

$$\mathbb{E}_z D_{\text{KL}}(\mathbf{q}_T(x, z) \| \mathbf{p}_\omega(x)) = D_{\text{KL}}(\bar{\mathbf{q}}_T(x) \| \mathbf{p}_\omega(x)) + \mathbb{E}_z D_{\text{KL}}(\mathbf{q}_T(x, z) \| \bar{\mathbf{q}}_T(x)). \tag{31}$$

To see this, add and subtract $\log \bar{q}_{T,v}$ in $\sum_v \mathbb{E}_z[q_{T,v} \log(q_{T,v}/p_{\omega,v})]$ and use $\mathbb{E}_z q_{T,v} = \bar{q}_{T,v}$. The final term is independent of student parameters for a frozen teacher. Thus the target is the teacher's marginal judgment distribution, not its internal reasoning algorithm or a calibration guarantee. The identity assumes finite KL on the target support.

Averaging K_z trace distributions gives a multi-trace target. Under independent, identically sampled traces and no outcome-based selection, one newly sampled trace estimates the expected gradient, and averaging reduces target-estimation variance. Repeated training on one cached trace instead optimizes its fixed empirical target. Filtering malformed traces conditions the target on validity rather than the unconditional trace distribution.

Training labels never enter the teacher prompt that produces soft targets. No sampled-token REINFORCE, GRPO, or importance sampling is needed for this differentiable KL.

F.3 Inputs and use during training

The pair input contains the visual task input X (source and any task-provided reference images), original request, two edited candidates, and a common rubric. The rubric asks whether A better fulfills the request while preserving protected source content and visual quality. Both modes see identical candidate order and rubric. Generator identity, predicted contracts, private U reasoning, human preference labels, and test annotations are excluded. Teacher traces are produced from these inputs; they do not carry privileged verified task facts.

The difficult evaluation subset covers wrong targets, near-miss text, preservation damage, omitted coupled edits, and visually attractive but task-incorrect outputs. These categories test whether reward judgments follow the editing request rather than overall visual appeal.

The distillation objective matches teacher judgment distributions at the student's direct scoring boundary. After distillation, the reward model stays frozen during U/G RL. Baseline reference edits are shared within each task group and are independent of candidate conditions. The reward model supplies only the global feedback; prompt-fidelity checks and received-condition execution checks retain their separate roles.

Schematic prompts. The shared input below defines the scoring task for both modes. Bracketed fields denote visual and textual inputs or the position at which logits are read.

```
Source image: [X]
Original edit request: [u]
Candidate A: [Y_A]
Candidate B: [Y_B]
Criterion: Compare fulfillment of the original request, preservation
of protected source content, and visual quality under the fixed rubric.
Question: Is A preferred to B? Yes means A; No means B.

Teacher mode: Analyze the comparison, finish the reasoning block,
then give the judgment. [Read logits before the judgment token.]
Student mode: Give the judgment directly, without an explanation.
[Read logits at the direct judgment boundary.]
```

The rubric compares task fulfillment, preservation, and visual quality. Human labels and teacher answers are excluded from its input fields. Teacher reasoning may itself state a conclusion before b; confident terminal logits can then reflect repetition rather than correct visual analysis. Independent labels, order swaps, and difficult-pair evaluation test judgment reliability; high confidence or low multi-trace variance alone does not validate a target.

F.4 Reward evaluation and controls

The evaluation covers six RM variants, difficult-pair categories, external reward benchmarks, five downstream configurations, and relative latency. M0 is base direct scoring and M1 explicit thinking. M2 continues preference SFT; M3 uses decision-then-reason training and direct deployment. M4 distills hard judgments and M5 soft judgment distributions. The main global-reward experiments use M5. The pairwise evaluation covers overall preference agreement and a difficult-pair subset.

Table 12: **Reward model comparison.** HPA and hard-subset accuracy are percentages. Latency measures one pairwise RM call end to end, normalized to M0; M1 uses online reasoning, while the other rows score directly. Full diagnostics: tables 13 and 15.

RM	HPA ↑	Hard ↑	NLL ↓	Brier ↓	Rel. latency ↓
M0: Base direct	77.6	66.5	0.489	0.158	1.0×
M1: Explicit think	82.3	75.8	0.377	0.120	6.0×
M2: Preference SFT	78.9	69.2	0.439	0.142	1.0×
M3: Decision–reason	79.9	71.2	0.409	0.133	1.0×
M4: Hard KD	80.4	72.3	0.501	0.154	1.0×
M5: Soft KD	81.7	74.8	0.385	0.122	1.0×

Judgment quality and reliability. Table 13 reports HPA, difficult-pair accuracy, NLL, Brier score, SwapErr, and A/B gap. The binary metric specification uses $s = P_{\texttt{Yes}}/(P_{\texttt{Yes}} + P_{\texttt{No}})$:

$$\begin{aligned} \mathrm{NLL} &= -\frac{1}{N}\sum_n [y_n \log s_n + (1-y_n)\log(1-s_n)], \\ \mathrm{Brier} &= \frac{1}{N}\sum_n (s_n - y_n)^2, \qquad \mathrm{SwapErr} = \frac{1}{N}\sum_n |s(x_n) + s(x_n^{\mathrm{swap}}) - 1|. \end{aligned} \tag{32}$$

HPA measures agreement with human preference. The A/B gap is $100|\mathrm{Acc}_A - \mathrm{Acc}_B|$ percentage points, comparing accuracy with the preferred candidate in the two input positions. NLL, Brier, and SwapErr characterize probability quality and sensitivity to candidate order.

Table 13: **Reward-model judgment quality.** HPA/Hard are accuracies (%); A/B gap is in percentage points. NLL/Brier/SwapErr follow equation (32).

RM	HPA ↑	Hard ↑	NLL ↓	Brier ↓	SwapErr ↓	A/B gap ↓
M0: Base direct	77.6	66.5	0.489	0.158	0.058	2.2
M1: Explicit think	82.3	75.8	0.377	0.120	0.045	1.3
M2: Preference SFT	78.9	69.2	0.439	0.142	0.051	1.0
M3: Decision–reason	79.9	71.2	0.409	0.133	0.046	0.8
M4: Hard KD	80.4	72.3	0.501	0.154	0.062	1.6
M5: Soft KD	81.7	74.8	0.385	0.122	0.041	0.6

Table 14: **Difficult-pair accuracy (%).** Scores are shown for five error categories and the reported overall.

Category	M0	M1	M4	M5
Wrong target or ordinal	70.0	78.3	75.8	78.3
Near-miss text	60.0	72.5	68.3	70.8
Preservation damage	69.2	76.7	73.3	78.3
Coupled-edit omission	61.7	73.3	68.3	70.8
Visually good, task wrong	71.7	78.3	75.8	75.8
Reported overall	66.5	75.8	72.3	74.8

Soft KD raises hard-subset accuracy from 66.5% to 74.8%, compared with 75.8% for explicit thinking. Relative to base direct, gains span all five reported categories, including near-miss text (60.0% to 70.8%) and visually attractive but incorrect edits (71.7% to 75.8%). M5 matches M1 on target/ordinal pairs and scores higher on preservation damage (78.3 versus 76.7), while M1 remains higher on text, coupled omissions, and attractive-but-wrong pairs. The table also reports the overall difficult-pair score.

Online scoring latency. Table 15 reports normalized latency of $1.0\times$ for the five direct scorers and $6.0\times$ for explicit thinking. Only M1 performs online reasoning. Latency covers the end-to-end online inference of one pairwise RM call, normalized to base direct scoring.

Table 15: **Relative RM latency.** Latency measures end-to-end online inference for a single pairwise RM call, normalized to M0. These ratios describe RM calls, rather than whole-training speedup.

RM	Relative latency ↓	Online thinking
M0	1.0×	No
M1	6.0×	Yes
M2	1.0×	No
M3	1.0×	No
M4	1.0×	No
M5	1.0×	No

External reward benchmarks. Table 16 keeps MMRB2 image-editing pair accuracy (Hu et al., 2025) separate from EditReward-Bench candidate-count-dependent ranking scores (Wu et al., 2026). M5 improves MMRB2 from 66.2 to 70.6, and EditReward $K = 2, 3, 4$ from 57.2/35.9/10.8 to 61.0/43.5/15.8. M1 remains higher on all four reported external metrics.

Table 16: **External RM evaluation.** MMRB2 editing pair accuracy and EditReward-Bench $K = 2, 3, 4$ ranking scores are reported separately (%).

RM	MMRB2 edit pair acc. ↑	EditReward $K = 2$ ↑	EditReward $K = 3$ ↑	EditReward $K = 4$ ↑
M0	66.2	57.2	35.9	10.8
M1	71.8	61.8	44.9	16.6
M2	67.1	58.3	37.8	11.9
M3	68.6	59.6	40.4	13.3
M4	69.3	60.1	41.5	13.8
M5	70.6	61.0	43.5	15.8

Downstream reward–gate interaction. Table 17 reports all five configurations. D0–D3 cross base/distilled direct reward with ungated/gated learning; D4 uses explicit thinking with gates. Relative to base direct, M5 raises CCR by 1.84 points without gates and 2.83 with gates. Gating adds 5.17 points under M0 and 6.16 under M5; the descriptive interaction is $6.16 - 5.17 = 0.99$ points. This reward–gate interaction differs from the structure–gate interaction of 2.83 points in section H.1. M1 with gates reaches 77.50 CCR, 0.42 above M5, while M5 has slightly higher preservation (93.86 versus 93.73). D0–D4 use equal update steps and rollout budgets.

Table 17: **Reward × gate comparisons on Wan-Image.** D0–D3 form the 2×2 comparison; D4 uses explicit thinking. D0–D4 use equal update steps and rollout budgets. Quality metrics are percentages; differences are percentage points.

Equal update steps and rollout budgets

ID / RM	Gates	CCR ↑	AC ↑	Pres. ↑
D0 / M0	Off	69.08	88.91	91.35
D1 / M0	On	74.25	91.28	93.12
D2 / M5	Off	70.92	89.72	91.82
D3 / M5	On	77.08	92.31	93.86
D4 / M1	On	77.50	92.42	93.73

Descriptive CCR contrast (pp)	Ungated	Gated	Interaction
M5 minus M0	1.84	2.83	0.99
Gated M1 minus gated M5			0.42

Teacher traces. The $K_z = 2, 4$ teacher-trace variants reach 81.8% and 81.9% HPA (table 18). The 0.1-point difference is descriptive and does not establish a multi-trace scaling advantage.

Table 18: **Teacher-trace comparison.** HPA is reported in percent. The deployed M5 is the reference configuration; K_z denotes the number of sampled teacher traces in the two variants.

Teacher-trace setting	HPA ↑
Deployed M5	81.7
$K_z = 2$	81.8
$K_z = 4$	81.9

Additional mechanism checks. The group-structure-matched random gate reaches 72.25 CCR, versus 71.33 for the count-matched control and 77.08 for semantic gating. Using the original task as G's local reference gives 75.00, versus 77.08 for the received-condition configuration (table 19). These controls examine selection granularity and local supervision reference, respectively. They support the use of semantic selection and received-condition supervision in this setting; they do not identify a unique causal decomposition of the gains.

Table 19: **Additional mechanism endpoints on Wan-Image.** CCR is a percentage; the final column gives full minus control (percentage points). Count-matched and group-structure-matched random gates are separate controls. The G-local comparison changes the reference for local supervision from the original task to the received condition. These endpoints compare training variants.

Condition	CCR ↑	Full minus control
Count-matched random gate	71.33	5.75
Group-structure-matched random gate	72.25	4.83
Original-task G-local source	75.00	2.08
Full: semantic gates + received-condition G-local	77.08	—

G Supervised Initialization Controls

The main tables compare unadapted baselines with full DuplexBridge; the supervised B_0 checkpoint uses native PE and structured SFT, before RL. tables 20 and 21 retain all three supervised checkpoints. From B_0 to full DuplexBridge, CE CCR increases by 17.91, 16.92, and 17.66 percentage points for BAGEL, Qwen-2511, and Wan-Image-7B. All three configurations improve the four public metrics and each RISE category over their respective B_0 (table 29). Warm-start improvements are not uniform: Qwen preservation falls from 90.86 before adaptation to 90.14 at B_0, then reaches 94.21 with DuplexBridge. Its ImgEdit score likewise falls from 4.36 before adaptation to 4.28 at B_0, then reaches 4.43 with DuplexBridge.

Table 20: Supervised initialization controls on CE-Bench. Qwen-2511 denotes Qwen-Image-Edit-2511. A–E follow table 8. B_0 is native PE plus structured SFT, before RL. Main-table baseline/Full comparisons include the complete method; these initialization controls separate supervised adaptation from subsequent RL. Dark/light pink mark the best/second-best value in each column, including ties.

Model	A	B	C	D	E	AC	Pres.	CCR
BAGEL (B0)	68.67	49.58	47.92	62.92	47.78	83.91	88.63	56.42
Qwen-2511 (B0)	**73.00**	**61.25**	**55.00**	**64.58**	**51.11**	**86.62**	**90.14**	**62.08**
Wan-Image-7B (B0)	70.00	57.92	52.50	62.08	49.44	85.28	89.21	59.42

Table 21: Supervised initialization controls on public benchmarks. Qwen-2511 denotes Qwen-Image-Edit-2511. B_0 adds native PE and structured SFT to the unadapted baselines in table 28, before RL. Dark/light pink mark the best/second-best value in each column, including ties.

Model	KRIS	RISE	ImgEdit	GEdit EN
BAGEL (B0)	64.23	16.94	3.74	6.93
Qwen-2511 (B0)	**70.36**	**21.94**	**4.28**	**7.28**
Wan-Image-7B (B0)	66.82	18.06	4.16	7.09

H Wan-Image Role, Reward, and Global-Gate Ablations

H.1 Additional local-only and structure–gating controls

The additional controls in table 26 use Wan-Image-7B with the same data, initialization, and training budget. Flat retains the same task facts and check rules as the structured condition, removing only grouping. Thus both interfaces support local verification and semantic gates; structured grouping is the specified factor, not access to extra reference requirements. Both interfaces preserve the requested content; the comparison concerns its grouping within the condition.

Dual-role local-only retains $\mathcal{L}_{\mathrm{GR}}^{U,\mathrm{loc}}$ and $\mathcal{L}_{\mathrm{GR}}^{G,\mathrm{loc}}$, their validity masks, and alternating updates; it omits both global losses and their KL contributions. Other conditions match the full method. This reward

Table 22: Wan-Image global-reward gate ablations. The first four rows vary the two semantic trust decisions, with local supervision and reward-validity conditions retained. Oracle gates use extra annotations.

Gate	CE CCR	KRIS	RISE
Neither global gate	70.92	76.27	26.11
Prompt trust only	74.17	78.62	28.06
Execution trust only	73.33	78.14	27.50
Both global gates	77.08	80.13	29.17
Count-matched random	71.33	76.48	26.11
Soft semantic weighting	76.33	79.76	28.89
Annotation-assisted oracle gates	78.67	81.07	30.00

Table 23: **Wan-Image-7B training-direction ablation.** All RL variants start at B0 with common supervision and reference access. U-only and G-only are independently trained; UT/DARS-style rows transplant mechanisms into Wan-Image, not native-system reproductions.

Variant	U/G RL	CE AC	CE CCR	Pres.	KRIS	RISE
Direct, no PE	–/–	82.70	53.83	88.47	60.84	13.06
Native PE, no CE-v1 SFT	–/–	83.86	56.67	88.92	63.41	15.83
B0: matched SFT	–/–	85.28	59.42	89.21	66.82	18.06
U-only RL	✓/–	87.96	66.67	88.74	73.28	24.44
G-only RL	–/✓	88.61	67.50	92.28	71.94	21.94
Joint RL, ungated	✓/✓	89.72	70.92	91.82	76.27	26.11
UT-style staged	✓/✓	90.38	72.67	92.14	77.39	26.94
DARS-style routing	✓/✓	90.94	74.08	92.57	78.68	27.78
DuplexBridge, full	✓/✓	92.31	77.08	93.86	80.13	29.17

ablation differs from U-only and G-only, which control the updated role. The local-only result is 68.25, versus 59.42 at B0, 70.92 with ungated global feedback, and 77.08 with gated global feedback. Differences here use the displayed aggregates.

Let r_{sg} denote CCR for structure $s \in \{0,1\}$ and gating $g \in \{0,1\}$. The reported cells are $(r_{00}, r_{01}, r_{10}, r_{11}) = (67.42, 70.75, 70.92, 77.08)$. The interaction contrast is

$$\Delta_{\mathrm{SG}} = (r_{11} - r_{10}) - (r_{01} - r_{00}) = (6.16) - (3.33) = 2.83 \text{ points}. \tag{33}$$

The additive prediction from the Flat/ungated cell is $67.42 + 3.50 + 3.33 = 74.25$, compared with the reported joint value 77.08. This contrast summarizes the observed endpoint interaction between structure and gating; it does not identify a causal pathway through verifier quality. The 1.75-point checkpoint cross-pair contrast uses a different experiment.

The four main gate settings vary the prompt-trust gate for G-global and the execution-trust gate for U-global while retaining both local losses and their validity conditions. Removing a semantic trust decision bypasses v_C or h_E, while retaining the corresponding judgment-confidence and availability tests. In particular, bypassing the execution-success threshold does not replace an unavailable execution judgment with a valid reward. Soft weighting and count-matched random routing are separate controls.

The trained endpoints in table 22 and the frozen-pool diagnostics in table 25 have different populations. The latter applies all selectors to the same 6,000 rollouts. G-global and U-global coverage divide accepted counts by 6,000; wrong-prompt and unexecuted-prompt error rates divide by the respective accepted counts. These are not module-wide training fractions. For G-global, $c_{\mathrm{audit}} v_C$ is common to the images under one prompt: semantic trust selects a whole prompt group before image-level b_F validity. A random image mask with the same total accepted count can instead fragment groups, alter eligible group sizes, and cause more groups to be skipped. For U-global, image selection also determines which candidates have $n_j > 0$. Thus accepted-count matching does not establish matching of non-skipped groups, effective updates, or the KL terms retained by equation (26). The group-structure-matched random control gives 72.25 CCR, versus 77.08 for semantic gating (table 19); the count-matched control gives 71.33. These controls compare selection policies under different matching constraints and are reported separately. Their endpoint differences

Table 24: Wan-Image training-direction results on general editing. The full method is slightly below the DARS-style transplant on ImgEdit (4.36 versus 4.37), while GEdit EN Full-set G_O is 7.74 versus 7.71.

Variant	ImgEdit	GEdit EN
Direct, no PE	4.09	6.88
Native PE, no CE-v1 SFT	4.11	6.97
B0: matched SFT	4.16	7.09
U-only RL	4.17	7.21
G-only RL	4.31	7.52
Joint RL, ungated	4.34	7.58
UT-style staged	4.32	7.60
DARS-style routing	4.37	7.71
DuplexBridge, full	4.36	7.74

Table 25: Global-feedback selection on one frozen pool of 6,000 rollouts. Coverage uses all rollouts; errors use the corresponding global-eligible subset, not all module training. Wrong-prompt and unexecuted-prompt labels are assigned using an external VLM together with human annotation. These fixed-pool diagnostics differ from retrained endpoint scores.

Gate	G-global cov.	Wrong / G-global	U-global cov.	Unexec. / U-global
Neither semantic gate	93.60	18.59	91.40	27.41
Prompt trust only	71.20	2.90	91.40	27.41
Execution trust only	93.60	18.59	64.80	4.60
Both trust gates	71.20	2.90	64.80	4.60
Count-matched random	71.20	18.12	64.80	27.11

do not decompose the contributions of sample selection and optimizer exposure. Wrong-prompt and unexecuted-prompt labels are assigned using an external VLM together with human annotation. The diagnostic rates are computed from these labels within each selector's accepted subset of the fixed pool. Oracle gates use additional adjudicated annotations and yield 78.67 CCR, 81.07 KRIS, and 30.00 RISE. They are diagnostics, not a guaranteed upper bound.

I Wan-Image Checkpoint Cross-Pairing

U0/G0 are the shared B0 states; Ut/Gt are checkpoints from the same full run. They differ from the independently trained U-only/G-only conditions. The CCR difference-in-differences is

$$77.08 - 65.92 - 68.83 + 59.42 = 1.75 \quad \text{percentage points.} \tag{34}$$

It describes checkpoint-specific partner dependence. KRIS and RISE also improve under the learned pair, as reported in table 27. The cross-pair statistic does not isolate architecture sharing or provide a causal decomposition.

J Public Benchmark Comparisons and Protocols

J.1 Comparison scope

Table 28 compares each complete DuplexBridge configuration with its direct editor. The DARS row in table 2 is a separately trained complete system (Cao et al., 2026); the Wan-Image UT-style and DARS-style rows in table 23 transplant training choices into a common Wan-Image setup.

J.2 Native metric boundaries

RISEBench-360 has 85 temporal, 90 causal, 100 spatial, and 85 logical tasks. Overall accuracy weights these task counts; it is not the unweighted mean of the four category percentages or the continuous visual-dimension scores.

Table 26: **Wan-Image structure, routing, and reward decomposition.** Reported CE CCR (%). Left: structure × global routing; right: training decomposition. RL variants share B_0, rather than forming a sequential curriculum. Contrasts use displayed values; see section H.1.

Interface	Ungated	Gated
Flat	67.42	70.75
Structured	70.92	77.08
Interaction: +2.83 points		

Training configuration	CCR
Direct (no PE)	53.83
+ Native PE	56.67
+ Structured SFT (B_0)	59.42
Dual-role local-only	68.25
Local + ungated global	70.92
Local + gated global	77.08

Table 27: Wan-Image checkpoint cross-pairing. U0/G0 are B0; Ut/Gt come from the same full training run, not the separately trained U-only/G-only conditions.

Pair	CE CCR	KRIS	RISE
U0 G0	59.42	66.82	18.06
Ut G0	65.92	72.48	22.22
U0 Gt	68.83	73.94	24.17
Ut Gt	77.08	80.13	29.17

KRIS evaluates knowledge-dependent editing with a composite, not complete success accuracy (Wu et al., 2025b). ImgEdit uses its native 1–5 scoring scale (Ye et al., 2025). For GEdit-Bench-EN, we use the English Full-set G_O column defined in DARS Table S5, distinct from the intersection of valid outputs across methods. It uses the native 0–10 scale. GEditBench v2 uses a different pairwise/Elo evaluation (Jiang et al., 2026); its scores and any v1/v2 subsets are not interchangeable. No cross-benchmark average is reported.

Table 28: **Four public editing benchmarks.** Qwen-2511 denotes Qwen-Image-Edit-2511. Bare names are unadapted baselines. KRIS is a composite; RISE is accuracy; ImgEdit uses 1–5; GEdit is English Full-set G_O (0–10). Dark/light pink mark the best/second-best value in each column, including ties.

Method	KRIS	RISE	ImgEdit	GEdit EN
BAGEL	56.21	5.80	3.45	6.40
BAGEL + DuplexBridge	76.48	26.11	4.04	7.41
Qwen-2511	64.42	17.50	4.36	6.97
Qwen-2511 + DuplexBridge	**82.16**	**31.11**	**4.43**	**7.89**
Wan-Image-7B	60.84	13.06	4.09	6.88
Wan-Image-7B + DuplexBridge	80.13	29.17	4.36	7.74

Table 29: Native UMM configurations on RISEBench-360. Qwen-2511 denotes Qwen-Image-Edit-2511. Overall is task-weighted across 85 temporal, 90 causal, 100 spatial, and 85 logical items. Dark/light pink mark the best/second-best value in each column, including ties.

Method	Temporal	Causal	Spatial	Logical	Overall
BAGEL (B0)	14.12	23.33	23.00	5.88	16.94
BAGEL + DuplexBridge	24.71	33.33	31.00	14.12	26.11
Qwen-2511 (B0)	21.18	26.67	29.00	9.41	21.94
Qwen-2511 + DuplexBridge	**31.76**	**38.89**	**34.00**	**18.82**	**31.11**
Wan-Image-7B (B0)	16.47	23.33	24.00	7.06	18.06
Wan-Image-7B + DuplexBridge	29.41	35.56	**34.00**	16.47	29.17

K Additional Qualitative Results

Figures 5 and 6 compare final outputs on KRIS-Bench and RISEBench. Figure 7 complements the main-text CE-Bench examples in figure 4 with localized edits, text replacement and translation, and reference-guided composition.

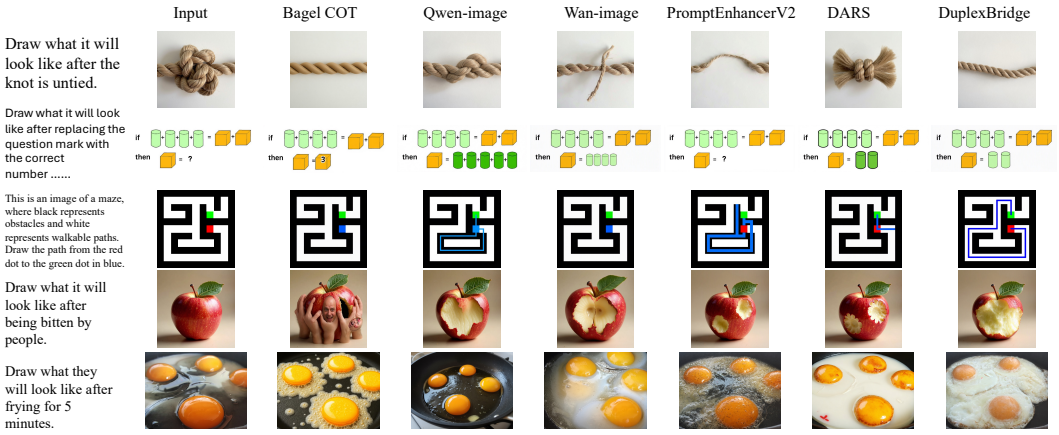


Figure 5: **Selected qualitative comparisons on KRIS-Bench.** Rows cover untying a knot, a visual equivalence relation, maze navigation, a bitten apple, and cooking progression. The maze requires a connected path through the white corridor; the physical transformations also require recognizable source content.

KRIS-Bench. DuplexBridge draws a connected blue maze route through the white corridor between the red and green endpoints. The untied rope and opaque egg whites show physical state changes. In the equivalence example, however, it outputs two cylinder icons instead of the requested number, exposing a distinction between semantic interpretation and answer-format compliance.

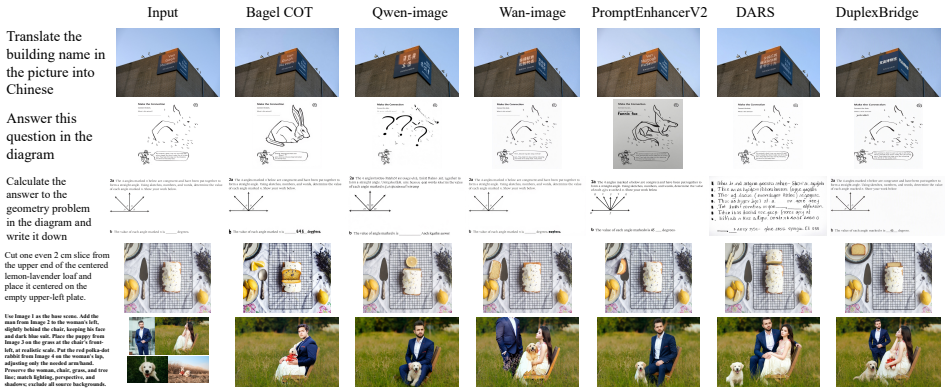


Figure 6: **Selected qualitative comparisons on RISEBench.** Rows cover sign translation, image-grounded question answering, geometric reasoning, spatially specified slicing, and composition from multiple references. The geometry example requires an answer within the existing diagram; the final row combines entity placement with preservation of the base scene.

RISEBench. DuplexBridge inserts "Jackrabbit" and "45" while retaining the worksheet diagrams, places a sliced piece on the upper-left plate, and combines a man, puppy, and red-patterned toy rabbit with the woman's base scene. These examples illustrate image-grounded answers, placement, and preservation alongside the quantitative results in table 2.



Figure 7: **Additional CE-Bench showcase.** Arrows indicate the transition from source images to edited outputs; auxiliary images provide visual references for flower replacement and dog insertion. Examples include sign-text replacement and translation, localized color changes, a traffic-light state change, and object appearance edits. The paired images show the edited regions alongside the surrounding scene.

L TEXT-TO-IMAGE ADAPTATION

Task-specific adaptation. DuplexBridge-T2I adapts the framework through task-specific retraining with a redesigned structured prompt interface. The adapted unified model's own understanding role U specifies the requested output from text u; its generation role G synthesizes $\widehat{Y}$ from the resulting condition $\widehat{C}$. The interface represents generation requirements rather than source-relative edits and preservation constraints. Both roles are trained with local/global feedback and bidirectional gates, following the DuplexBridge design.

The responsibilities become $u \to \widehat{C}$ and $\widehat{C} \to \widehat{Y}$. Prompt-fidelity checks compare the condition with u; execution checks assess whether $\widehat{Y}$ follows $\widehat{C}$; the global evaluator compares the image with u. Prompt trust and execution trust retain their respective roles in admitting global feedback to G and U. This extension uses a generation-specific interface rather than the editing CE-v1 schema.

Results and metric interpretation. Table 30 reports three adapted configurations across GenEval2, R2I, WISE-V, and DPG. The adapted models score above the corresponding initial configurations on all five displayed metrics. BAGEL's baseline uses native chain-of-thought. The gains assess the complete T2I adaptation. The extension evaluates generation after retraining; it does not evaluate zero-shot transfer from the editing checkpoints.

GenEval2 uses Soft-TIFA: AM takes the arithmetic mean and GM the geometric mean of soft VQA scores within a prompt, followed by averaging over prompts (Kamath et al., 2025). GM therefore measures compositional fulfillment on a continuous scale, rather than binary complete compliance. R2I evaluates reasoning-driven generation (Chen et al., 2025); WISE evaluates world-knowledge consistency (Niu et al., 2026); DPG evaluates dense prompt following (Hu et al., 2024). R2I and WISE-V use the displayed $\times 100$ scale. Each metric is interpreted separately, with no cross-benchmark average.

Table 30: **Text-to-image adaptation after task-specific retraining.** BAGEL uses native chain-of-thought (CoT) in its baseline configuration. R2I and WISE-V use the displayed $\times 100$ scale. Dark/light pink mark the best/second-best value across all six systems in each column, including ties. Metric definitions and the adaptation setting are given in section L.

Model	GenEval2 AM ↑	GenEval2 GM ↑	R2I $\times 100$ ↑	WISE-V $\times 100$ ↑	DPG ↑
BAGEL + native CoT	70.9	23.1	48.0	62.8	85.1
BAGEL + DuplexBridge-T2I	77.8	31.5	56.0	**70.5**	86.8
Qwen-Image-Edit-2511	80.8	33.8	52.0	51.0	88.3
Qwen-Image-Edit-2511 + DuplexBridge-T2I	**85.0**	**42.0**	**61.0**	66.0	**89.7**
Wan-Image-7B	79.4	32.2	51.0	54.0	87.7
Wan-Image-7B + DuplexBridge-T2I	84.2	40.6	59.5	64.5	89.3

M Public Requirement Load and Joint Fulfillment

Population and definitions. We analyze the ungated and full Wan-Image outputs on all 1,200 CE-Bench tasks, paired by task. Public requirement load $K_i = K_i^{\text{change}} + K_i^{\text{preserve}}$ counts change and preservation groups from the task requirements, independently of the predicted bridge. The observed loads are 3, 5, 7, and 9; every task has both types of requirements. The number of change groups is 1/2/3/4 on 780/200/125/95 tasks, so total load is not simply the number of requested edits. Let f_i^{change} and f_i^{preserve} be failed-group counts. Then

$$C_i = \mathbf{1}[f_i^{\text{change}} = 0], \qquad P_i = \mathbf{1}[f_i^{\text{preserve}} = 0], \tag{35}$$

$$\text{CCR}_i = C_i P_i, \qquad \text{AC}_i = 1 - \frac{f_i^{\text{change}} + f_i^{\text{preserve}}}{K_i}, \qquad \text{Pres}_i = 1 - \frac{f_i^{\text{preserve}}}{K_i^{\text{preserve}}}. \tag{36}$$

Each metric is averaged over tasks. In particular, all-preservation-pass P and average preservation Pres. measure different outcomes.

Performance across observed loads. Full has higher CCR point estimates at each observed K (table 31). The paired overall gain is 6.17 points (95% interval [4.50, 7.83]); the $K = 9$ gain is 3.33 points $[-0.56, 7.22]$, leaving its direction uncertain at this sample size. Gains are not monotonic in K. Average failed groups per task fall from 0.659 to 0.508 overall, but improvement is not uniform across components (table 32). At $K = 9$, change all-pass falls from 73.33% to 60.56%, a -12.78-point contrast $[-17.78, -7.78]$, while preservation all-pass rises from 49.44% to 52.78%. Higher joint success therefore does not imply uniformly better change execution.

Table 31: **Complete and average compliance by public requirement load.** U/F denote ungated/full Wan-Image; CCR and AC are percentages. Δ is full minus ungated CCR in points, calculated before rounding. Intervals use 50,000 paired task bootstrap resamples within observed categories; they are pointwise, not simultaneous. The $K = 9$ interval includes zero.

K	N	U CCR	F CCR	Δ	95% interval	U AC	F AC
3	300	85.00	90.33	+5.33	[+2.33, +8.33]	95.00	96.78
5	360	74.44	80.83	+6.39	[+3.33, +9.44]	89.78	92.33
7	360	66.39	74.44	+8.06	[+4.72, +11.39]	88.53	92.58
9	180	49.44	52.78	+3.33	[-0.56, +7.22]	83.15	84.26
All	1200	70.92	77.08	+6.17	[+4.50, +7.83]	89.72	92.31

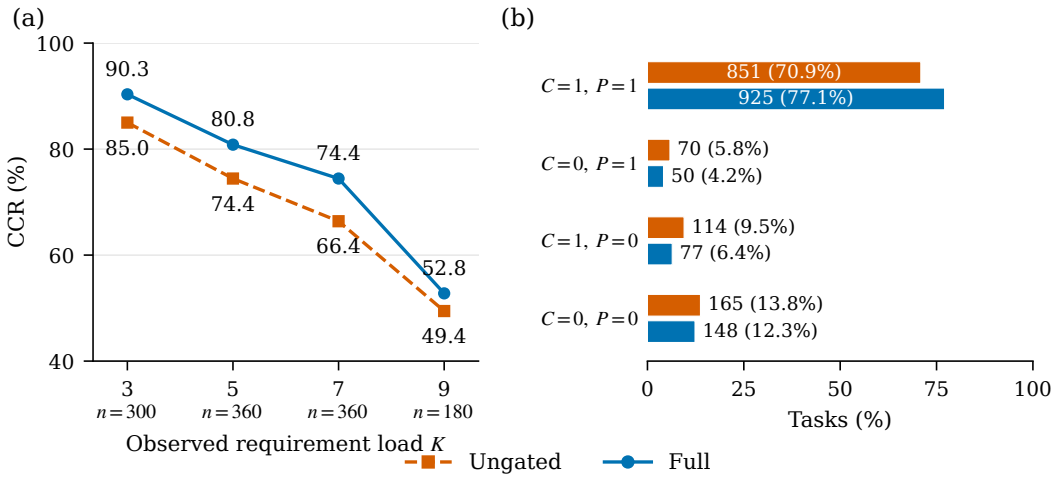


Figure 8: **Requirement load and joint change–preservation outcomes.** (a) CCR for full and ungated on the same 1,200 public tasks, grouped by their observed total number K of change and preservation requirements; each stratum's task count is shown below its tick. Task-category composition varies across strata, so these observations do not isolate a causal effect of increasing K. (b) Counts and percentages of the same tasks in the four joint states (ungated above, full below), derived from task-level change and invariant failure counts. $C = 1$ means every requested change passes, and $P = 1$ means every preservation requirement passes; P is an all-pass indicator, distinct from the mean preservation fraction reported as Pres. The fraction of tasks in the $C = P = 1$ state equals CCR.

Table 32: **Change and preservation fulfillment.** C and P require all change or all preservation groups to pass, respectively, and are reported as percentages. P differs from mean Pres. Fail is the average number of failed public groups per task. At $K = 9$, higher joint success coexists with lower change all-pass.

K	U C	F C	U P	F P	U Fail	F Fail
3	88.33	90.33	96.67	100.00	0.150	0.097
5	75.28	80.83	74.44	80.83	0.511	0.383
7	82.50	91.94	76.11	80.28	0.803	0.519
9	73.33	60.56	49.44	52.78	1.517	1.417
All	80.42	83.50	76.75	81.25	0.659	0.508

Joint outcomes and paired transitions. Across all tasks, the four (C, P) states $(1, 1)$, $(0, 1)$, $(1, 0)$, and $(0, 0)$ contain 851/70/114/165 ungated outputs and 925/50/77/148 full outputs (figure 8). Thus, change all-pass rises from 80.42% to 83.50%, and preservation all-pass from 76.75% to 81.25%. These aggregate gains coexist with task-level regressions: 92 tasks switch from joint failure to joint success, while 18 switch in the opposite direction; 73 gain preservation all-pass, while 19 lose it. Marginal state differences are not counts of individually repaired tasks. At $K = 7$, preservation-only failures increase from 16.11% to 17.50%; at $K = 9$, outputs failing both components increase from 26.67% to 39.44%. The overall reduction in failure states should therefore not be interpreted as improvement in every load-specific outcome.

Category composition and interpretation. The load strata have different category mixtures (table 33): $K = 3$ has no D tasks, while $K = 9$ has no A tasks and contains 95/180 D tasks. The within-category contrasts include a negative cell for C at $K = 9$ (-6.67 points on 30 tasks). Requirement grouping and category both affect this observed-load analysis; it is not a controlled intervention on difficulty. Moreover, requiring every group to pass can itself reduce CCR as the number of groups grows. The breakdown tests where joint fulfillment improves on the observed tasks, rather than identifying a causal complexity effect.

Table 33: **Category composition and within-category contrasts.** A–E follow table 8. Counts total 300/240/240/240/180 across categories. A dash denotes an empty cell. Contrasts are descriptive; some cells are small and gains are not uniformly positive.

K	A	B	C	D	E
Task counts					
3	160	65	45	0	30
5	100	85	95	20	60
7	40	65	70	125	60
9	0	25	30	95	30
Full minus ungated CCR (points)					
3	+5.63	+3.08	+8.89	–	+3.33
5	+8.00	+5.88	+5.26	+15.00	+3.33
7	+12.50	+7.69	+4.29	+11.20	+3.33
9	–	+4.00	-6.67	+2.11	+16.67

Uncertainty and scope. Intervals use 50,000 paired task bootstrap resamples with seed 20260925, preserving category counts overall and within each reported K stratum. Both methods and all outcomes of a task remain paired. The 2.5th and 97.5th percentiles give pointwise intervals conditional on the observed strata, evaluated outputs, and group labels. Task-level contrasts are computed before rounding. The human evaluation in section N reports its own matched task population and is not pooled with these automatic outcomes.

N Human Evaluation of Final Outputs

Evaluation population and unresolved labels. The human evaluation contains 200 CE-Bench tasks: 50 source-grounded, 40 text/structured-content, 40 reasoning-grounded, 40 compositional, and 30 reference-conditioned tasks. Each task has complete-success labels for direct, ungated, and full outputs. Human labels are unresolved for 5, 4, and 3 outputs, respectively. Automated labels are available for all outputs. Per-method agreement and CCR use the same human-decidable subset within each method (table 5). This analysis evaluates final task success; it is distinct from the execution-check construction study in table 11.

Table 34: **Human–automatic agreement and sensitivity bounds.** TP/FN/FP/TN treat human task success as the positive reference. FA is FP/(FP+TN); FR is FN/(TP+FN), both in percent. Bounds assign unresolved human labels to failure or success over all 200 tasks; they are identification bounds, not confidence intervals.

Method	TP	FN	FP	TN	FA	FR	CCR bounds
Direct	94	10	12	79	13.19	9.62	[52.0, 54.5]
Ungated	129	7	10	50	16.67	5.15	[68.0, 70.0]
Full	144	6	7	40	14.89	4.00	[75.0, 76.5]

We retain unresolved cases in the original sample count and report their effect through identification bounds over all 200 tasks. For full, 150 tasks are human-confirmed successes, 47 are failures, and three are unresolved, giving bounds of 75.0–76.5%. These identification bounds vary only the unresolved labels, while agreement is evaluated on the decidable subset.

Matched gains and scoring bias. Let $H_i^m, A_i^m \in \{0, 1\}$ denote human and automatic complete success for method m. Per-method bias is the mean of $A_i^m - H_i^m$ on that method's decidable subset. For comparing full and ungated, we instead use the common set S of 194 tasks with both human

labels resolved:

$$\Delta_H = \frac{100}{|S|} \sum_{i \in S} (H_i^{\text{full}} - H_i^{\text{ungated}}), \tag{37}$$

$$\Delta_A = \frac{100}{|S|} \sum_{i \in S} (A_i^{\text{full}} - A_i^{\text{ungated}}), \qquad B_\Delta = \Delta_A - \Delta_H. \tag{38}$$

All three quantities are in percentage points. On S, the human paired counts are 123 both successful, 24 successful only for full, 11 successful only for ungated, and 36 both unsuccessful. The corresponding automatic counts are 131, 18, 6, and 39. Automatic and human gains therefore compare identical tasks, rather than subtracting rates with different denominators.

Table 35: **Full versus ungated on 194 common tasks.** Point estimates and 95% percentile intervals are in percentage points. Intervals use 50,000 paired task bootstrap resamples within categories, preserving the observed category counts of this common decidable set.

Quantity	Estimate	95% interval
Human gain Δ_H	+6.70	[+1.03, +12.37]
Automatic gain Δ_A	+6.19	[+1.55, +10.82]
Gain bias $\Delta_A - \Delta_H$	-0.52	[-3.61, +2.58]

The interval for B_Δ includes zero, making the direction of gain bias inconclusive. Per-output errors remain (table 34), and the comparison is conditional on the observed decidable outputs.

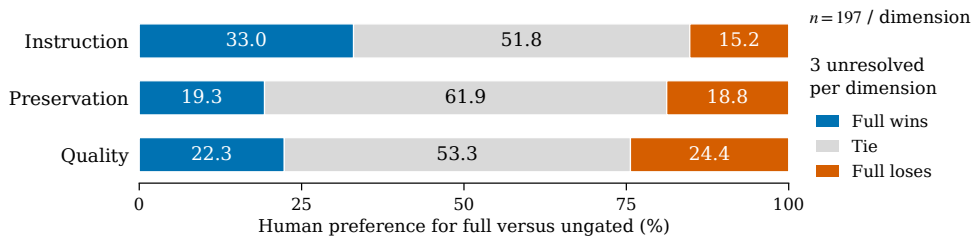


Figure 9: **Human preferences for full versus ungated.** Each dimension has 197 decidable task pairs and three unresolved pairs out of 200. Wins, ties, and losses retain their separate categories; interval estimates for net preference appear in table 36.

Dimension-specific preference. Each of the 200 tasks also has comparisons of full against ungated and direct for instruction fulfillment, preservation, and visual quality. Preferences take three resolved values: win, tie, and loss. Unresolved judgments are counted separately. Net preference is $100(N_{\text{win}} - N_{\text{loss}})/N_{\text{decidable}}$; ties remain in $N_{\text{decidable}}$.

Table 36: **Human preference for full.** W/T/L are wins/ties/losses; Unres. counts unresolved judgments out of 200 task pairs per comparison. Net preference is 100(W-L)/(W+T+L), in points. The 95% intervals use category-stratified task bootstrap; ties remain in the denominator.

Full comparison	Dimension	W/T/L	Unres.	Net	95% interval
vs. ungated	Instruction	65/102/30	3	+17.77	[+8.63, +26.90]
vs. ungated	Preservation	38/122/37	3	+0.51	[-8.12, +9.14]
vs. ungated	Quality	44/105/48	3	-2.03	[-11.68, +7.11]
vs. direct	Instruction	89/83/25	3	+32.49	[+22.84, +42.13]
vs. direct	Preservation	70/94/33	3	+18.78	[+9.14, +28.43]
vs. direct	Quality	61/87/48	4	+6.63	[-3.57, +16.84]

Instruction preference favors full in both comparisons. Against ungated, preservation and quality intervals span zero; against direct, preservation also favors full, while the quality interval includes

zero. Figure 9 visualizes wins, ties, and losses against ungated. These are complementary evaluations of the same task pool, not independent replications. Failure to detect a quality difference does not establish non-inferiority or an absence of degradation.

Uncertainty. We use 50,000 category-stratified task bootstrap resamples with seed 20260925 and report the 2.5th and 97.5th percentiles. Each analysis retains the observed number of decidable tasks in each category; the full–ungated common set has 49/39/39/39/28 tasks in A–E. A task's paired method and evaluator labels are resampled together. Preference intervals use the relevant dimension's decidable task pairs. These intervals describe task-sampling uncertainty conditional on the evaluated outputs and their observed labels.

O Extended Related Work and Method Comparison

O.1 Extended related work

Understanding–generation coordination. MGIE learns expressive instructions and jointly trains the multimodal language model and image editor (Fu et al., 2024). SmartEdit strengthens complex instruction editing with bidirectional interaction between image features and MLLM outputs (Huang et al., 2024). These works establish understanding-guided editing and coupled adaptation; SmartEdit's bidirectionality concerns feature interaction, whereas ours concerns training feedback between the understanding and generation roles. BAGEL unifies image understanding and generation; UniEdit studies editing-driven adaptation (Deng et al., 2025; Zheng et al., 2026). Unified Thinker uses successive understanding and generation phases, ThinkGen applies SepGRPO, and ThinkRL-Edit decouples the two roles' updates (Zhou et al., 2026; Jiao et al., 2026; Li et al., 2026). UniGRPO and BRAID further study joint reasoning and synthesis (Liu et al., 2026; Hu et al., 2026). DuplexBridge focuses on the supervision references and feedback eligibility within this co-adaptation: task fidelity, received-condition execution, and task outcome train distinct loss components. Here, *native* understanding means that the unified model uses its own understanding capability for prompt enhancement and passes the result to its own generation capability, without an external understanding module in that inference path. This definition permits external judges and teachers for training supervision.

Structured prompt enhancement and feedback. PromptEnhancer optimizes a rewriter with a frozen generator; its published alignment reward evaluates generated images against the rewritten prompt (Wang et al., 2026). Thus, received-condition image assessment is already an established component. Wan-Image and SenseNova-U1.5 use structured editing conditions (Mao et al., 2026; Diao et al., 2026). DDA-Thinker already assesses executable plans without observing generated images and maintains separate cognitive and visual reward streams, while keeping the editor frozen (Yang et al., 2026). DARS addresses the same ambiguity between planning and execution failures. It jointly trains a planner and renderer with an original-task checklist, post-render planner diagnostics, prefix gating across plan fields, and variance-based module weights (Cao et al., 2026). DuplexBridge combines different local references within co-adaptation: U-local judges task fidelity without the rendered image, and G-local judges execution against the condition actually received. Reciprocal partner trust selects only the additional global losses, preserving valid local supervision when global feedback is withheld. This combination makes local responsibility and global-feedback eligibility explicit within the same training procedure. The gates implement selective feedback, not counterfactual credit estimation (Foerster et al., 2018). Detailed comparisons appear in section O.2.

Reward modeling and editing evaluation. SLD detects and corrects generation errors through an inference-time loop (Wu et al., 2024); DuplexBridge uses execution checking to determine which feedback can train each role. RewardDance uses comparative token probabilities; UnifiedReward-Think and Z-Reward study reasoning-enhanced and distilled judgment (Wu et al., 2025a; Wang et al., 2025; Jin et al., 2026). Our distilled RM supports the shared outcome signal through direct pairwise scoring against fixed reference edits. TIFA and Davidsonian Scene Graph motivate decomposed visual checking (Hu et al., 2023; Cho et al., 2024). ImgEdit, GEdit, RISEBench, and KRIS-Bench assess complementary dimensions of editing and reasoning (Ye et al., 2025; Liu et al., 2025b; Zhao et al., 2025; Wu et al., 2025b). CE-Bench emphasizes *complete* compliance with

source-grounded changes and preservation constraints. Its requirements come from the original task, independently of a model's predicted bridge, and evaluation accepts final images.

O.2 Optimization and feedback mechanisms

Table 37 distinguishes optimization and supervision choices. DuplexBridge combines a received-condition execution target with reciprocal eligibility on global losses. The comparison separates these feedback choices from the broader use of structured PE and two-role optimization.

Table 37: Mechanism comparison. U and G refer to functional understanding and generation roles, which need not share an architecture across methods.

Method	Published optimization/feedback	Relation to DuplexBridge
PromptEnhancer (Wang et al., 2026)	Train a rewriter from image alignment to rewritten prompts; freeze G.	Received-condition assessment is a precedent; here it trains G alongside task-based U supervision.
DDA-Thinker (Yang et al., 2026)	Train a thinker with image-independent plan rewards and separate visual rewards; freeze G.	Both roles adapt here, with partner trust selecting global feedback while valid local feedback remains available.
Unified Thinker; ThinkGen (Zhou et al., 2026; Jiao et al., 2026)	Successive U/G RL phases; executable conditions or modular rewards.	DuplexBridge specifies separate local reference targets and reciprocal global-feedback eligibility.
ThinkRL-Edit (Li et al., 2026)	Decoupled U/G updates with planning, reflection, checklist feedback, and multi-reward preference-chain filtering.	Filtering reward rankings differs from accepting global feedback according to partner trust.
UniGRPO (Liu et al., 2026)	Joint TextGRPO/FlowGRPO on native reasoning and image generation with terminal image feedback.	Terminal feedback provides a related joint objective; DuplexBridge separates role-local supervision from partner-dependent global feedback.
BRAID (Hu et al., 2026)	Joint interleaved reasoning; terminal and visual-process feedback with separately normalized advantages.	DuplexBridge normalizes four eligible components, with partner-trust gates affecting only global feedback.
ToolArtist (Zhao et al., 2026a)	Unified tool-using policy with caption-intent and image-quality reward components.	Joint intent/image assessment is established; here their supervision targets and eligibility remain separate.
DARS (Cao et al., 2026)	Joint training with task-derived checklists, post-render planner diagnostics, prefix gating, and variance-based residual module weights.	G-local here checks the received C, U-local excludes Y, and partner trust selects only global components.
DuplexBridge	Alternating U/G updates with separately eligible local/global losses.	Prompt fidelity uses (X, u, C), execution uses (X, C, Y), and global outcome uses the pairwise RM on $(X, u, Y; \mathcal{R}_{X,u})$.

The UT-style and DARS-style rows in table 23 transplant mechanisms into the common Wan-Image configuration. These Wan-Image controls compare the stated mechanisms; complete-system comparisons provide separate architectural and optimization context.

P Implementation and Evaluation Details

Initialization and controlled comparisons. BAGEL, Qwen-2511, and Wan-Image-7B are evaluated before adaptation, after supervised initialization (B0), and with full DuplexBridge. The Wan-Image mechanism studies share B0. U-only and G-only optimize one role; local-only updates both roles without global losses; ungated and gated variants retain role-local supervision and differ in the use of global feedback. These are separate training conditions, rather than successive phases of one training run. The complete accepted teacher response supervises U during SFT (section B.1). At inference, only the public condition is passed to G; private reasoning is excluded from the generation context.

Reward model and downstream budget. The main DuplexBridge runs use M5, the reasoning-distilled direct pairwise reward model. It is frozen during U/G RL. Local-only training omits the global reward, and the reward-replacement study explicitly substitutes M0 or M1. The RM evaluation includes a difficult-pair subset. The downstream D0–D4 comparisons use equal update-step and rollout budgets. Relative latency measures one pairwise RM call end to end, normalized to base direct scoring; it does not measure whole-training cost.

Evaluation populations. CE-Bench contains 1,200 tasks with category sizes 300/240/240/240/180. Its public requirements are derived from source images, task references, and the original request, independently of predicted conditions. AC and Pres. average within each task before averaging across tasks; CCR requires every requirement group to pass. Failed outputs remain in the denominator. The 87.92% human-agreement result concerns execution-check construction on a content-matched subset (section D); it is separate from the CE-Bench final-output metrics.

The gate diagnostics apply every selector to one fixed pool of 6,000 rollouts. External-VLM judgments and human annotations provide wrong-prompt and unexecuted-prompt labels. Coverage uses all 6,000 rollouts as its denominator; error proportions use only the accepted subset for the corresponding global path. These diagnostics characterize which feedback is selected, without assigning causal responsibility for individual failures.

Comparison scope. Cross-system tables assess complete editing configurations; Wan-Image controls assess training mechanisms. All benchmark metrics keep their native scales, and no cross-benchmark average is formed. Aggregate comparisons use the displayed values; task-level analyses use unrounded outcomes. Interaction contrasts summarize the corresponding endpoint differences.

Q Ethics and Intended Use

Image editing can enable deceptive manipulation. Source provenance, data licenses, and disclosure of uncertainty are important safeguards. Structured instructions do not eliminate cultural bias, ambiguous intent, or visual-judgment errors. The method concerns controllable editing and diagnostic feedback, not a guarantee of authenticity or compliance with unstated user intent.

R Limitations

Our experiments demonstrate substantial improvements in image editing and generation, but DuplexBridge has not yet been evaluated on more complex video generation and editing tasks. Extending the method to the video domain is a promising direction for future research.