Plan, Edit, Refine: Joint RL Training of LLM Agents and Diffusion Editors for Compositional Image Generation

Xuan Qi [1]

Abstract

Compositional text-to-image generation requires satisfying multiple attribute constraints simultaneously—a task that single-pass generation routinely fails on. Two recent lines of work address this from opposite ends: agentic iterative refinement trains only the language planner while freezing the image generator (Jiang et al., 2026), and unified policy optimization trains language and visual generation components but in a single generation round (Liu et al., 2026). We present **NEXUS**, a work-in-progress framework combining both properties: iterative, multi-round refinement with *joint* RL training of both the LLM planner (Qwen2.5-7B) and the reference-conditioned diffusion editor (FLUX.1-Kontext-dev) via shared GRPO reward. A **dual-channel Bridge** routes continuous LLM hidden states (differentiable pathway) and discrete text instructions (interpretable pathway) to the editor's conditioning mechanism. Empirically, using three-seed means on the full 553-prompt GenEval prompt set scored with our Qwen-VQA compositional score rather than the official GenEval evaluator: (1) zero-shot Gemini with the same iterative refinement loop reaches 0.740 and serves as the matched baseline; (2) NEXUS full reaches 0.734 at step 50 and 0.780 at step 100, surpassing this matched Gemini baseline by +4.0 pp at step 100; and (3) planner-only and no-Bridge variants do not exceed the matched Gemini baseline at step 100, highlighting the importance of the full co-adaptive system.

1. Introduction

Compositional image generation—producing images that precisely satisfy multi-attribute prompts such as *"a red apple next to a blue vase on a wooden table"*—remains an open challenge. State-of-the-art flow-matching models like FLUX (Black Forest Labs, 2024) excel at photorealism but systematically fail on attribute binding and spatial relations (Ghosh et al., 2023; Huang et al., 2023). A single diffusion forward pass must simultaneously satisfy competing constraints with no mechanism for iterative correction.

Two complementary strategies address this gap. *Agentic approaches* (Jiang et al., 2026; Feng et al., 2026) wrap the generator in an LLM-driven iterative loop, using VLM feedback to issue successive corrective editing instructions. They demonstrate strong multi-round refinement but treat the image generator as a *frozen* tool: the editor never adapts to the instruction style the planner develops. *Unified policy optimization approaches* (Liu et al., 2026) optimize language reasoning and visual generation under shared reward, but in a *single generation round* with no mechanism to observe failures and issue targeted corrections.

Neither paradigm is sufficient alone. A frozen tool creates a performance ceiling regardless of planner quality. A single round cannot observe and correct its own output. The natural combination—iterative co-adaptation of both components—has not been explored.

The iterative refinement challenge runs deeper than it appears. When a planner issues T sequential editing instructions, credit assignment becomes non-trivial: the terminal VQA reward must be distributed across T independent decisions, each of which may be responsible for satisfying (or violating) a different attribute constraint. Score-function estimators suffer $O(T)$ variance growth, making credit assignment progressively noisier for longer trajectories. This is particularly acute in our setting where the planner must learn *which* attribute failed in round t and *how* to correct it precisely enough that the diffusion editor can execute the fix faithfully. Without a differentiable pathway connecting the planner's internal reasoning to the editor's denoising process, gradient signal must cross two discrete bottlenecks: the sampling step in the LLM *and* the implicit VQA evaluation signal.

Modular vs. fused architectures present a second fundamental tension. Fused backbones (Liu et al., 2026) unify the language and diffusion parameters into a single model,

[1]IIIS Tsinghua University. Correspondence to: Xuan Qi <kx80777801@gmail.com>.

Preprint. June 7, 2026.

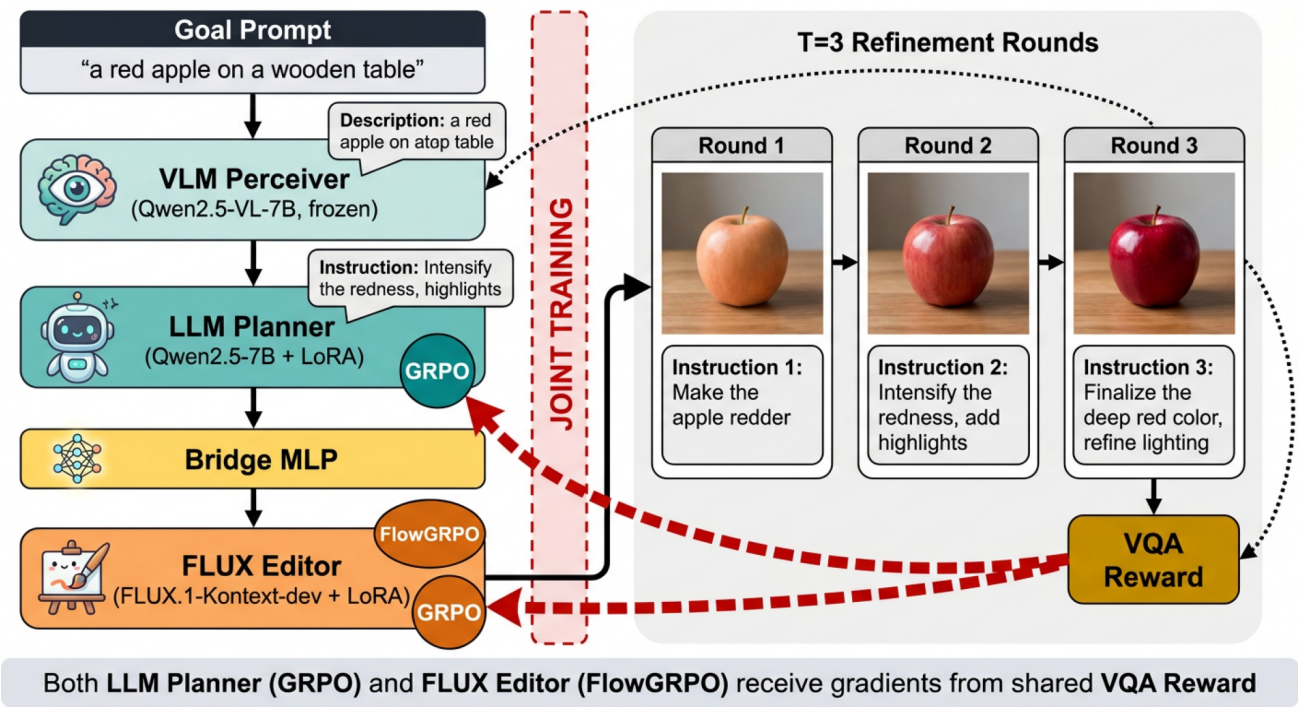

Figure 1. **NEXUS framework.** The iterative agent loop jointly trains the LLM planner and FLUX editor via shared VQA reward. The dual-channel Bridge routes both discrete text and continuous LLM hidden states to the editor's conditioning pathway, providing a differentiable gradient channel alongside the interpretable ReAct trace.

which simplifies gradient flow but sacrifices modularity: the LLM and diffusion components cannot be swapped, scaled, or fine-tuned independently. At the 7B + 12B parameter scale relevant to this work, maintaining independent checkpoints is essential for practical deployment—each component can be updated, audited, or replaced without retraining the system from scratch. The challenge is preserving gradient communication across independently checkpointed components without fusion.

We present NEXUS (**N**eural **E**ditor-e**X**ecutive **U**nified **S**ystem), a work-in-progress step toward filling this gap with three contributions:

- **Iterative joint training**: jointly RL-train both an LLM planner and a reference-conditioned diffusion editor across T=3 refinement rounds, enabling image-conditioned error correction with co-adapting components.
- **Dual-channel Bridge**: a lightweight MLP that simultaneously routes continuous LLM hidden states (differentiable gradient pathway) and discrete text (ReAct-style interpretable trace) to the diffusion editor's conditioning, bypassing the discrete sampling barrier without sacrificing interpretability.
- **Multi-actor GRPO**: credit assignment across independently checkpointed LLM and diffusion model parameters without backbone fusion, enabling modular, swappable agent components.

The rest of this paper is organized as follows. Section 2 presents the NEXUS architecture, including the iterative agent loop (Section 2.1), dual-channel Bridge (Section 2.2), multi-actor GRPO (Section 2.3), two-stage training protocol (Section 2.4), and credit assignment analysis (Section 2.5). Section 3 presents experimental results including the matched Gemini baseline, full pipeline training, and qualitative analysis. Section 4 discusses implications. Section 5 reviews related work. Appendices provide full hyperparameter tables, extended results, evaluation protocol, and intuition for the Bridge variance-reduction hypothesis.

2. Method

2.1. Iterative Agent Loop

NEXUS formulates compositional image generation as a finite-horizon MDP over T editing rounds. The state $s_t = (g, I_t, c_t)$ consists of the goal prompt g, current image I_t, and VLM-generated description c_t. Three components form the pipeline:

1. **VLM perceiver** ψ (Qwen2.5-VL-7B (Qwen Team, 2025), frozen): translates I_t to text description c_t.

2. **LLM planner** π_θ (Qwen2.5-7B (Qwen Team, 2024), LoRA rank 64): generates editing instruction a_t in ReAct format.
3. **Diffusion editor** f_ϕ (FLUX.1-Kontext-dev (Black Forest Labs et al., 2025), LoRA rank 64): applies the edit $I_{t+1} = f_\phi(I_t, a_t, \mathbf{z}_t)$.

A terminal VQA reward $R(\tau) = \frac{1}{n}\sum_k \mathbf{1}[\text{VQA}(I_T, q_k) = \text{True}]$ evaluates attribute satisfaction in I_T, where q_k are auto-generated yes/no questions for each attribute constraint in g (Ghosh et al., 2023).

FLUX.1-Kontext-dev is suited for iterative refinement because its reference-conditioning via sequence concatenation enables *consistency-preserving* edits: unmentioned attributes are preserved across rounds (Black Forest Labs et al., 2025), making sequential single-attribute correction tractable.

2.2. Dual-Channel Bridge

Standard agentic RL creates a *gradient barrier*: the discrete sampling $w_l \sim \text{Categorical}(\text{softmax}(\mathbf{W}^{lm}\mathbf{h}_l))$ makes $\partial w_l / \partial \mathbf{h}_l$ undefined, blocking reward gradients from reaching the diffusion editor parameters.

The **Bridge** $\mathcal{B}_\xi$ is a two-layer MLP that maps the LLM's final-token hidden state to the CLIP pooled projection dimension of FLUX.1-Kontext-dev:

$$\mathbf{z}_t = \mathcal{B}_\xi\big(\mathbf{h}^{(K)}_{|a_t|}\big) \;\in\; \mathbb{R}^{768} \tag{1}$$

where $\mathbf{h}^{(K)}_{|a_t|} \in \mathbb{R}^{4096}$ is the last-token hidden state at layer K. The editor receives *both*:

$$\tilde{\mathbf{c}}_t = \text{MLP}_{\text{proj}}\big([\mathbf{c}_t^{\text{CLIP}};\, \mathbf{z}_t]\big) \tag{2}$$

The text channel ($\mathbf{c}_t^{\text{CLIP}}$) preserves interpretability and backward compatibility with any string-accepting tool. The hidden channel ($\mathbf{z}_t$) provides a differentiable pathway: gradients flow through $f_\phi \to \mathbf{z}_t = \mathcal{B}_\xi(\mathbf{h}^{(K)}) \to \theta$, bypassing the sampling barrier. Both channels derive from the same forward pass, incurring no additional inference latency.

An alignment loss prevents semantic drift between channels: $\mathcal{L}_{\text{align}} = \lambda(1 - \cos(\mathbf{z}_t, \text{CLIP}(a_t)))$, $\lambda = 0.1$.

Proposition 1 (Gradient variance intuition, informal)**.** *As trajectory length T grows, score-function credit assignment becomes noisier because the same terminal reward must supervise more stochastic editing decisions. The Bridge adds a pathwise conditioning signal through a shared MLP, which we hypothesize can improve gradient signal-to-noise by reducing reliance on score-function-only updates.*

We defer diagnostic training-dynamics evidence for this hypothesis to Appendix B.

2.3. Multi-Actor GRPO

We extend GRPO (Shao et al., 2024) to the two-actor (LLM + diffusion) setting. For a group of G=4 trajectories per prompt, the group-relative advantage is $\hat{A}_i = (R(\tau_i) - \bar{R})/(\sigma_R + \epsilon)$. The joint log-likelihood decomposes as:

$$\log \pi_{\theta,\Delta\phi}(\tau) = \sum_t \log \pi_\theta(a_t \mid s_t) + \sum_t \log f_\phi(I_{t+1} \mid I_t, a_t, \mathbf{z}_t) \tag{3}$$

The diffusion term follows FlowGRPO (Liu et al., 2025) via ODE-to-SDE conversion. Gradients $\nabla_{\Delta\phi}\mathcal{L}$ update the editor LoRA directly; $\nabla_\theta \mathcal{L}$ updates the LLM via policy gradient. The Bridge additionally receives pathwise gradients through Eq. (1). Total: $\mathcal{L} = \mathcal{L}_{\text{GRPO}} + \mathcal{L}_{\text{align}}$.

2.4. Training Protocol

Training proceeds in two stages.

Stage 0: Bridge warm-up (50 SFT steps). Before joint GRPO, the Bridge MLP is pre-aligned to the CLIP embedding space using a supervised objective. For each prompt g in the warm-up set, we run one LLM forward pass, extract $\mathbf{h}^{(K)}_{|a_0|}$, and minimize $\mathcal{L}_{\text{align}}$ with frozen LLM and diffusion parameters. This ensures the Bridge produces CLIP-compatible vectors at the start of Stage 1 rather than random noise that could destabilize early GRPO updates. Warm-up converges in ≈50 steps with learning rate 10^{-4}.

Stage 1: Full GRPO (up to 100 steps). All three components (LLM LoRA, FLUX LoRA, Bridge) are updated jointly. Each training step processes one batch of prompts. For each prompt, G=4 full trajectories of T=3 rounds are collected, producing $4T = 12$ total image-text pairs per batch element. Group-relative advantages are computed across the G trajectories. The LLM receives score-function gradients; the diffusion editor receives FlowGRPO gradients (Liu et al., 2025); the Bridge receives pathwise gradients via the differentiable path in Eq. (1). The combined objective is:

$$\mathcal{L} = \mathcal{L}_{\text{GRPO}}(\theta, \Delta\phi) + \lambda \mathcal{L}_{\text{align}}(\xi), \quad \lambda = 0.1.$$

2.5. Credit Assignment Across Rounds

A key challenge in multi-round iterative generation is *temporal credit assignment*: the terminal reward $R(\tau)$ at round T must be attributed to each of the T editing decisions, each potentially responsible for a different attribute. Under standard GRPO, the score-function gradient

$$\nabla_\theta \mathcal{L} = \hat{A}_i \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t \mid s_t)$$

accumulates T gradient terms per trajectory. Under standard simplifying assumptions, the variance of this sum grows

Algorithm 1 NEXUS Training

Require: Prompts $\mathcal{P}$, rounds T, group size G
Stage 0 (Bridge warm-up): Train ξ on $\mathcal{L}_{\text{align}}$ for 50 steps; freeze θ, ϕ
Stage 1 (Joint GRPO):
for each batch of prompts $\{g_j\}$ **do**
  for $g = 1$ **to** G **do**
    Initialize $I_0 \leftarrow$ blank / previous image
    for $t = 0$ **to** $T - 1$ **do**
      $c_t \leftarrow \psi(I_t)$ {VLM percept}
      $a_t \sim \pi_\theta(\cdot \mid s_t)$ {LLM action}
      $\mathbf{z}_t \leftarrow \mathcal{B}_\xi(\mathbf{h}^{(K)}_{|a_t|})$ {Bridge hidden vector}
      $I_{t+1} \leftarrow f_\phi(I_t, a_t, \mathbf{z}_t)$ {FLUX edit}
    end for
    $R_g \leftarrow \frac{1}{n}\sum_k \mathbf{1}[\text{VQA}(I_T,\ q_k)]$
  end for
  $\hat{A}_g \leftarrow (R_g - \bar{R})/(\sigma_R + \epsilon)$
  Update θ, $\Delta\phi$, ξ via $\mathcal{L}_{\text{GRPO}} + \lambda\mathcal{L}_{\text{align}}$
end for

with horizon length, making longer refinement trajectories harder to optimize.

The Bridge mitigates this bottleneck by providing an additional pathwise gradient pathway. Because $\mathbf{z}_t = \mathcal{B}_\xi(\mathbf{h}_t)$ depends on the same hidden representation that generates a_t, the pathwise update can carry information about useful editing directions without passing solely through sampled text. We treat the resulting variance-reduction interpretation as a hypothesis rather than a formal guarantee; diagnostic evidence is reported in Appendix B, and Appendix D gives the intuition.

3. Experiments

3.1. Setup

Models and training. LLM planner: Qwen2.5-7B (Qwen Team, 2024) with LoRA rank 64. Diffusion editor: FLUX.1-Kontext-dev (Black Forest Labs et al., 2025) with LoRA rank 64 on all cross-attention layers. VLM perceiver: Qwen2.5-VL-7B (Qwen Team, 2025) (frozen). Bridge MLP: hidden dim 2048, <10M parameters. Configuration: T=3 rounds, G=4 rollouts, VQA reward. Learning rates: LLM 5×10^{-5}, editor 1×10^{-5}, Bridge 1×10^{-4}. Hardware: 8×NVIDIA L20Z 80 GB GPUs. Full hyperparameter table is in Appendix A.

Conditions. Three training conditions are compared: **(1) Joint** (LLM + FLUX LoRA, no Bridge); **(2) Frozen-tool** (LLM LoRA only, FLUX frozen); **(3) NEXUS full** (Joint + Bridge). All conditions use the same iterative T=3 loop and VQA reward. The **zero-shot Gemini** planner (no training) serves as the strong off-the-shelf baseline.

Table 1. Matched zero-shot Gemini baseline (no RL training) under the same iterative refinement loop and Qwen-VQA compositional score used for the trained systems. This score, rather than the single-round score, is used as the baseline for Table 2.

Protocol	Score	Δ vs. single round
Gemini, single round	0.640	—
Gemini, iterative refinement	**0.740**	**+10.0 pp**

Reporting convention. This workshop paper reports work in progress. Unless otherwise stated, trained-condition scores are means over three random seeds on the full 553-prompt GenEval prompt set, scored with Qwen-VQA compositional score rather than the official object-detection-based GenEval evaluator. The present version focuses on mean trends; standard deviations and confidence intervals are left to a longer version.

Training data. We train on 130 compositional prompts sampled from CC12M captions (Changpinyo et al., 2021), not from GenEval. Captions are filtered to contain at least two distinct attribute constraints (*e.g.*, object+color, object+spatial relation), then grouped to approximately cover the six GenEval skill categories: single-object, two-objects, counting, colors, position, and color-attribution (Ghosh et al., 2023). These prompts are used only for RL rollouts; no GenEval evaluation prompt is used for training. Appendix C provides details on scoring.

Evaluation. We use the full 553-prompt GenEval prompt set as the prompt source, but do not use the official GenEval evaluation code. Instead, we score images with Qwen2.5-VL-7B as a VQA judge over auto-generated binary questions per attribute constraint (Appendix C); we call this metric Qwen-VQA compositional score. The only external baseline in the main experiments is a zero-shot Gemini planner run with the same iterative refinement loop and the same Qwen-VQA scoring protocol.

3.2. Iterative Refinement Without Training

Before any gradient updates, we establish whether iterative refinement helps with a zero-shot Gemini planner. Table 1 shows that using the same iterative loop improves Gemini from 0.640 to 0.740. We therefore use 0.740 as the matched Gemini baseline in the main results, avoiding comparison between trained T=3 systems and a weaker single-round baseline. This pre-training baseline motivates the RL experiments: if zero-shot iteration already produces substantial improvements, can joint RL training further amplify these gains with a task-specialized planner and co-adapted editor?

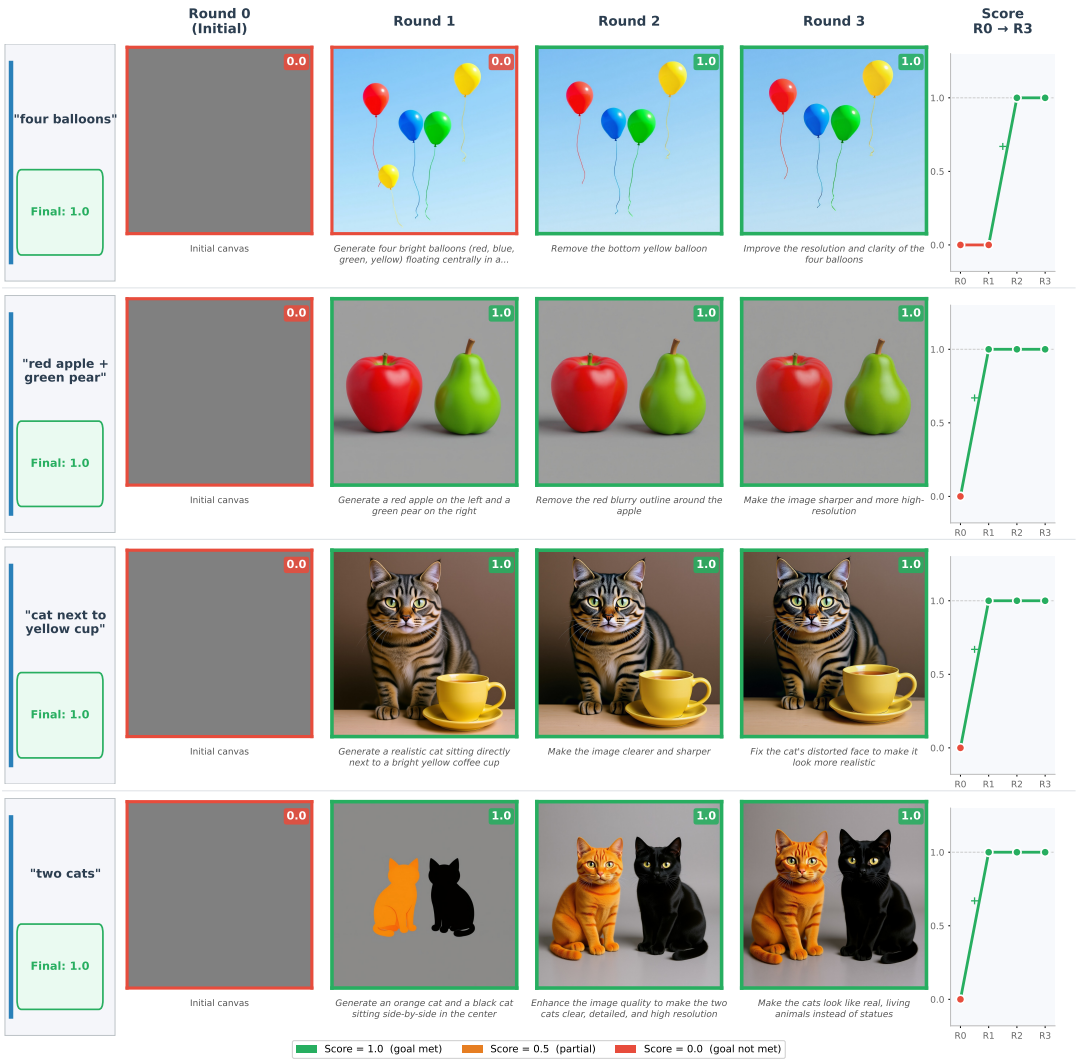

Figure 2. **Qualitative examples of iterative refinement.** Three rounds of the NEXUS pipeline on a multi-attribute compositional prompt. In some examples, Round 1 already satisfies the main object/attribute constraints; later rounds preserve those constraints while improving clarity or realism. In harder examples, later rounds correct remaining failures such as count or object placement. The VQA score is non-decreasing in the shown trajectories.

3.3. Full Pipeline RL Training

Table 2 reports the main results. Four observations stand out.

NEXUS full improves by step 100. NEXUS full achieves three-seed mean Qwen-VQA compositional score of 0.734 at step 50 and 0.780 at step 100, moving from slightly below the matched zero-shot Gemini baseline (0.740) to +4.0 pp above it. This provides preliminary evidence that co-adapting the diffusion editor, LLM planner, and Bridge under shared reward can improve compositional accuracy over adapting the planner alone. A frozen editor creates a one-sided mismatch: the planner learns instructions optimized for the reward, but the editor cannot adapt to the planner's evolving instruction style.

Matched baselines matter. Under the stronger Gemini iterative baseline, the planner-only and no-Bridge diagnostic variants do not surpass Gemini at step 100: joint reaches 0.700 (-4.0 pp) and frozen-tool reaches 0.728 (-1.2 pp). The improvement over Gemini is therefore attributable to the full NEXUS configuration rather than to the iterative loop alone.

GRPO oscillation remains a limitation. The joint condition's step-100 score (0.700) is below its step-50 score (0.718)—a pattern consistent with high-variance GRPO updates. When all trajectories in a batch receive uniform reward, the group-relative advantage $\hat{A}_i = 0$ and no gradi-

Table 2. Qwen-VQA compositional score on the full 553-prompt GenEval prompt set at steps 50 and 100 (T=3, VQA reward). Trained-condition scores are three-seed means. This is not the official GenEval metric. The matched baseline is zero-shot Gemini with the same iterative refinement loop (0.740). **Bold**: best at each step. †: best mean checkpoint among shown.

Condition	@50	@100	Δ@100 vs. Gemini
Joint (LLM+FLUX)	0.718	0.700	−4.0 pp
Frozen-tool	0.675	0.728	−1.2 pp
NEXUS full (+Bridge)	0.734	**0.780**†	+4.0 pp
Zero-shot Gemini (matched)	**0.740**	0.740	—
Untrained (base)	0.200	0.200	−54.0 pp

ent update is produced. We therefore interpret the step-50 advantage as an early-training signal rather than as evidence of monotonic dominance across all training budgets.

Bridge provides the main gain but remains preliminary. The NEXUS full (Bridge) condition improves by +4.6 pp from step 50 to step 100 and is the only trained condition that exceeds the matched Gemini baseline at step 100. We interpret this as evidence for the value of the full Bridge configuration at fixed checkpoints, while longer training still requires better stabilization and uncertainty estimates.

3.4. Iterative Refinement: Qualitative Analysis

Figure 2 illustrates the round-by-round refinement behavior that underpins the quantitative gains. The examples show two patterns rather than a single universal correction story. For simpler prompts such as *"red apple + green pear"*, *"cat next to yellow cup"*, and *"two cats"*, Round 1 already satisfies the main VQA constraints and later rounds mainly preserve the satisfied attributes while sharpening the image or improving realism. For the harder *"four balloons"* example, Round 1 still violates the target count; Round 2 removes the extra balloon and raises the score to 1.0. Round 3 then performs residual visual refinement while maintaining the achieved constraints.

This qualitative evidence supports a more modest interpretation: multi-round editing is useful both for correcting remaining compositional errors and for preserving satisfied constraints while improving visual quality. It should not be read as showing that every prompt requires a Round 2 semantic correction.

4. Discussion

Why modularity matters. Fused backbone architectures (Liu et al., 2026) achieve strong performance by eliminating the inter-component interface altogether. However, fused models sacrifice an important practical property: *component replaceability*. In a deployed multimodal pipeline, the image generator, language planner, and VLM perceiver will be updated at different rates as better models become available. NEXUS's independently checkpointed LoRA adapters allow, for instance, swapping FLUX.1-Kontext-dev for a successor model without retraining the LLM planner, or fine-tuning the planner on a new task domain without touching the diffusion parameters. The Bridge's alignment loss $\mathcal{L}_{\text{align}}$ further ensures that a newly swapped component can be re-warmed in ≈50 SFT steps rather than full retraining.

The oscillation pattern and its remedies. GRPO oscillation (the joint condition declining from step 50 to step 100) is a known challenge in small-group policy gradient training. When the group of G=4 trajectories collapses to uniform rewards, the advantage estimate degenerates to zero and training stalls. Two complementary remedies are future work: (1) KL regularization, which penalizes deviation from the reference policy and prevents reward-collapse episodes; and (2) group size increase to G=8 or G=16, which reduces the probability of reward collapse in any given batch. These modifications should make it possible to test whether the observed trends persist beyond step 100.

Future directions. Several extensions are natural: (a) *T5-pathway Bridge injection*, bypassing the CLIP bottleneck by injecting Bridge vectors directly into FLUX's T5-based text conditioning, which operates at higher semantic resolution; (b) *larger-scale co-training* with 70B LLM + a full-resolution diffusion model, where modular checkpointing is especially critical; (c) *video generation*, where temporal consistency across frames creates an analogous iterative refinement problem; (d) *reward model specialization*, replacing the frozen VQA judge with a jointly trained reward model that can assess compositional accuracy with higher precision than binary VQA questions.

5. Related Work

Text-to-image generation and compositional evaluation. Large diffusion and flow-matching models have made high-quality text-to-image generation practical (Rombach et al., 2022; Saharia et al., 2022; Black Forest Labs, 2024), but semantic faithfulness remains difficult for prompts with multiple objects, attributes, counts, or spatial relations. This gap has motivated a line of fine-grained evaluation benchmarks, including GenEval (Ghosh et al., 2023), T2I-CompBench (Huang et al., 2023), TIFA (Hu et al., 2023), and DSG (Cho et al., 2024). These works shift evaluation away from holistic image quality and toward decomposed checks of object presence, attribute binding, counting, and relational correctness. NEXUS follows this view by using the full GenEval prompt set with per-attribute Qwen-VQA questions; the resulting scores are not official GenEval scores.

Compositional control for diffusion models. Several methods improve compositional generation by modifying the generation process itself. Composable Diffusion (Liu et al., 2022) combines diffusion components to better satisfy multiple concepts, while Attend-and-Excite (Chefer et al., 2023) adjusts cross-attention during sampling to reduce object neglect and attribute-binding failures. These approaches show that compositionality can be improved without changing the user-facing task, but they are primarily single-shot inference-time control methods. In contrast, NEXUS frames compositional generation as a multi-round agent-editor interaction and optimizes the planner and editor jointly.

Instruction-guided image editing. Instruction-following editors such as InstructPix2Pix (Brooks et al., 2023) and datasets such as MagicBrush (Zhang et al., 2023) establish text-guided editing as a natural interface for modifying existing images. FLUX.1-Kontext-dev extends this direction with reference-conditioned editing suitable for consistency-preserving refinements (Black Forest Labs et al., 2025). NEXUS builds on this editing interface but asks a different question: rather than training an editor to follow isolated instructions, can an LLM planner and diffusion editor co-adapt over repeated feedback-driven edits?

Agentic image generation. General LLM-agent work uses interleaved reasoning and acting traces (Yao et al., 2023) and feedback-based self-improvement (Shinn et al., 2023) to solve sequential decision problems. Recent image-generation agents adapt this idea to visual synthesis: GenAgent (Jiang et al., 2026) uses an LLM/VLM loop to call an image generator iteratively, and Gen-Searcher (Feng et al., 2026) adds retrieval-augmented search. These systems demonstrate the value of closed-loop multimodal reasoning, but the generator is typically treated as a frozen tool. NEXUS keeps the interpretable agent loop while allowing the generator to adapt to the planner's evolving instruction distribution.

Preference learning and RL for visual generation. Human-preference resources and reward models such as ImageReward (Xu et al., 2023) and Pick-a-Pic/PickScore (Kirstain et al., 2023) enabled reward-based alignment of text-to-image models. DDPO (Black et al., 2024) formulates diffusion denoising as an RL problem; AlignProp (Prabhudesai et al., 2024) optimizes differentiable rewards through backpropagation; DiffusionDPO (Wallace et al., 2024) adapts direct preference optimization to diffusion; and FlowGRPO (Liu et al., 2025) extends group-relative policy optimization to flow-matching models. These methods mainly align the generator itself. NEXUS instead optimizes a trajectory containing both language actions and image edits, so credit assignment spans multiple actors and refinement rounds.

Unified and modular LM–diffusion training. UniGRPO (Liu et al., 2026) fuses LM and diffusion backbones into one model. Fused architectures simplify gradient flow because the language and visual generation modules share a single optimization interface. However, this also reduces modularity: replacing the language model, upgrading the image generator, or auditing one component independently becomes difficult. NEXUS explores a different point in the design space. It keeps Qwen2.5 as the planner (Qwen Team, 2024), Qwen2.5-VL as the frozen perceiver (Qwen Team, 2025), and FLUX.1-Kontext-dev as the reference-conditioned editor (Black Forest Labs et al., 2025), then connects them through a lightweight Bridge. This preserves component replaceability while still allowing shared reward optimization across the agent and editor.

6. Conclusion

We presented NEXUS, a work-in-progress framework for jointly RL-training an LLM planning agent and a reference-conditioned diffusion editor across multiple iterative refinement rounds via shared GRPO reward. The dual-channel Bridge provides differentiable gradient routing alongside interpretable text communication, enabling modular co-adaptation without backbone fusion or architecture modification.

Three preliminary findings summarize the contribution: (1) iterative refinement improves zero-shot Gemini from 0.640 to 0.740 before any training under Qwen-VQA compositional score; (2) NEXUS full reaches 0.734 at step 50 and 0.780 at step 100, surpassing the matched Gemini iterative baseline by +4.0 pp at step 100; and (3) planner-only and no-Bridge variants do not surpass the matched Gemini baseline at step 100, suggesting that the full co-adaptive system is important.

Limitations. This is a workshop work-in-progress paper. The main trained-condition scores are reported as three-seed means, but this version does not include standard deviations or confidence intervals. The VLM perceiver is frozen; T5-pathway Bridge injection and KL-regularization remain future directions. Evaluation uses the full 553-prompt GenEval prompt set, but the score is our Qwen-VQA metric rather than the official GenEval evaluator; this version also does not yet include the full uncertainty analysis needed for strong statistical claims.

Broader impact. NEXUS suggests that jointly training modular multimodal agent components through lightweight differentiable interfaces is practically feasible at the 7B+12B parameter scale, pointing toward a broader class of composable, jointly optimized multimodal agents.

LLM/Agent Usage Disclosure

LLMs were used to assist with writing polish and figure creation for this paper.

References

Black, K., Janner, M., Du, Y., Kostrikov, I., and Levine, S. Training diffusion models with reinforcement learning. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2305.13301.

Black Forest Labs. FLUX.1 [dev]. https://github.com/black-forest-labs/flux, 2024. Official inference repository. Accessed: 2026-04-29.

Black Forest Labs, Batifol, S., Blattmann, A., Boesel, F., Consul, S., Diagne, C., Dockhorn, T., English, J., English, Z., Esser, P., Kulal, S., Lacey, K., Levi, Y., Li, C., Lorenz, D., Müller, J., Podell, D., Rombach, R., Saini, H., Sauer, A., and Smith, L. FLUX.1 Kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025. URL https://arxiv.org/abs/2506.15742.

Brooks, T., Holynski, A., and Efros, A. A. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2211.09800.

Changpinyo, S., Sharma, P., Ding, N., and Soricut, R. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

Chefer, H., Alaluf, Y., Vinker, Y., Wolf, L., and Cohen-Or, D. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics*, 42(4), 2023. URL https://arxiv.org/abs/2301.13826.

Cho, J., Hu, Y., Garg, R., Anderson, P., Krishna, R., Baldridge, J., Bansal, M., Pont-Tuset, J., and Wang, S. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.18235.

Feng, K., Zhang, M., Chen, S., Lin, Y., Fan, K., Jiang, Y., Li, H., Zheng, D., Wang, C., and Yue, X. Gensearcher: Reinforcing agentic search for image generation. *arXiv preprint arXiv:2603.28767*, 2026. URL https://arxiv.org/abs/2603.28767.

Ghosh, D., Hajishirzi, H., and Schmidt, L. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2310.11513.

Hu, Y., Liu, B., Kasai, J., Wang, Y., Ostendorf, M., Krishna, R., and Smith, N. A. 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://arxiv.org/abs/2303.11897.

Huang, K., Sun, K., Xie, E., Li, Z., and Liu, X. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2307.06350.

Jiang, K., Wang, Y., Zhou, J., Li, P., Liu, Z., Xie, C.-W., Chen, Z., Zheng, Y., and Zhang, W. GenAgent: Scaling text-to-image generation via agentic multimodal reasoning. *arXiv preprint arXiv:2601.18543*, 2026. URL https://arxiv.org/abs/2601.18543.

Kirstain, Y., Polyak, A., Singer, U., Matiana, S., Penna, J., and Levy, O. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2305.01569.

Liu, J., Liu, G., Liang, J., Li, Y., Liu, J., Wang, X., Wan, P., Zhang, D., and Ouyang, W. Flow-GRPO: Training flow matching models via online RL. *arXiv preprint arXiv:2505.05470*, 2025. URL https://arxiv.org/abs/2505.05470.

Liu, J., Ye, Z., Yuan, L., Zhu, S., Gao, Y., Wu, J., Li, K., Wang, X., Nie, X., Huang, W., and Ouyang, W. UniGRPO: Unified policy optimization for reasoning-driven visual generation. *arXiv preprint arXiv:2603.23500*, 2026. URL https://arxiv.org/abs/2603.23500.

Liu, N., Li, S., Du, Y., Torralba, A., and Tenenbaum, J. B. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pp. 423–439, 2022. URL https://arxiv.org/abs/2206.01714.

Prabhudesai, M., Goyal, A., Pathak, D., and Fragkiadaki, K. Aligning text-to-image diffusion models with reward backpropagation. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.03739.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025. URL https://arxiv.org/abs/2502.13923.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022. URL https://arxiv.org/abs/2112.10752.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, S. K. S., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., Ho, J., Fleet, D. J., and Norouzi, M. Photorealistic text-to-image diffusion models with deep language understanding. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2205.11487.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y. K., Wu, Y., and Guo, D. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2303.11366.

Wallace, B., Dang, M., Rafailov, R., Zhou, L., Lou, A., Purushwalkam, S., Ermon, S., Xiong, C., Joty, S., and Naik, N. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://arxiv.org/abs/2311.12908.

Xu, J., Liu, X., Wu, Y., Tong, Y., Li, Q., Ding, M., Tang, J., and Dong, Y. Imagereward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2304.05977.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.03629.

Zhang, K., Mo, L., Chen, W., Sun, H., and Su, Y. Magicbrush: A manually annotated dataset for instruction-guided image editing. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2306.10012.

A. Hyperparameters and Implementation Details

Table 3 provides a complete listing of all hyperparameters used in NEXUS experiments. All experiments were conducted on a node of 8×NVIDIA L20Z 80 GB GPUs. The LLM and VLM components were loaded in `bfloat16`; the diffusion model was loaded in `float32` for numerical stability in the SDE conversion step.

Table 3. Complete hyperparameter table for all NEXUS training runs.

Component	Hyperparameter	Value
LLM Planner	Model	Qwen2.5-7B
	LoRA rank	64
	LoRA target	all attention proj.
	Learning rate	5×10^{-5}
Diffusion Editor	Model	FLUX.1-Kontext-dev
	LoRA rank	64
	LoRA target	all cross-attention
	Learning rate	1×10^{-5}
Bridge MLP	Architecture	2-layer MLP
	Hidden dim	2048
	Input dim	4096 (LLM hidden)
	Output dim	768 (CLIP proj.)
	Learning rate	1×10^{-4}
Training	Rounds T	3
	Group size G	4
	Alignment weight λ	0.1
	Bridge warm-up steps	50
	Max GRPO steps	100
	Reward	VQA (binary, mean)
Hardware	GPUs	8× NVIDIA L20Z
	VRAM	80 GB per GPU

B. Diagnostic Training Dynamics

Table 4. Diagnostic training dynamics on a 5-prompt held-out set. Both selected runs ultimately achieve peak reward 0.600, but joint training reaches this earlier (step 100 vs. 150). ⋆: peak checkpoint. †: monotonically improving (no oscillation).

Config.	Stp.50	Stp.100	Stp.150	Best
Joint $T=1$, s1	0.180	**0.600***	0.200	0.600
Frozen $T=2$, s1	0.200	0.200	**0.600***	0.600
Frozen $T=1$, s2	0.200	0.040	**0.500***	0.500
Frozen $T=2$, s2	0.200	0.300	**0.400**†	0.400

Table 4 shows that joint training peaks at step 100 while frozen-tool peaks at step 150 (both reaching 0.600 on the 5-prompt set). This diagnostic trend is consistent with Proposition 1: the shared GRPO gradient simultaneously updates both planner and editor, potentially carrying information about both components' failure modes. We do not treat this small-set result as statistical proof of faster convergence.

Averaging frozen-tool $T=2$ across both seeds gives $(0.600+0.400)/2 = 0.500$, versus joint seed 1 best of 0.600—a **mean +10 pp advantage** in these diagnos-

tic runs. A full convergence claim requires more seeds and uncertainty estimates.

Early reward dynamics. The progression of training rewards across conditions provides further insight. At step 0, Bridge achieves 0.800, joint achieves 0.600, and frozen-tool achieves 0.200—reflecting the Bridge's better pre-training initialization from the Stage 0 warm-up. By step 10, all conditions are improving (Bridge 0.800, joint 0.700, frozen-tool 0.500). At step 30, the expected oscillation pattern emerges: Bridge oscillates around 0.500 while joint reaches its 0.800 peak and frozen-tool also reaches 0.800, consistent with GRPO's known reward saturation dynamics in small-batch settings.

C. Qwen-VQA Compositional Score Protocol

Evaluation prompt set. We use the full 553-prompt GenEval prompt set (Ghosh et al., 2023) as the source of evaluation prompts. The prompts cover single-object generation, two-object generation, counting, colors, position, and color-attribution. The 130-prompt RL training set is constructed separately from CC12M captions and has no overlap with these GenEval prompts.

VQA judge. We use Qwen2.5-VL-7B (Qwen Team, 2025) as the VQA judge. For each prompt g and generated image I_T, we auto-generate one binary yes/no question per attribute constraint in g. Example: for *"a red apple next to a blue vase"*, questions are: *"Is there a red apple in the image? (yes/no)"* and *"Is there a blue vase in the image? (yes/no)"*.

Score computation. The Qwen-VQA compositional score for a single image is the fraction of per-attribute questions answered "yes" by the VQA judge. The reported score is the mean over all 553 prompts. We use the same scoring protocol for zero-shot Gemini and all trained NEXUS variants; the matched Gemini iterative score (0.740) is therefore the primary baseline for the main experiments.

Relationship to official GenEval. This score should not be interpreted as the official GenEval metric. The official GenEval evaluation pipeline relies on object-focused discriminative vision models, including detection-based checks for object presence, count, position, and related attributes. Our current workshop version instead uses a Qwen2.5-VL VQA judge and automatically generated yes/no questions. The numbers reported in this paper are therefore internally comparable across our conditions, but not directly comparable to official GenEval scores reported by other papers.

D. Variance Reduction Intuition

We provide intuition for Proposition 1. This appendix is not a formal proof; a future version should directly measure gradient variance and signal-to-noise across training runs.

Setup. Consider a trajectory of T steps with terminal reward $R(\tau)$. Under standard GRPO with score-function estimation, the gradient estimator for the LLM is:

$$\hat{g}_{\text{SF}} = \hat{A}_i \sum_{t=0}^{T-1} \nabla_\theta \log \pi_\theta(a_t \mid s_t).$$

Variance of $\hat{g}_{\text{SF}}$. Under a simplified assumption that per-round score-function terms are weakly correlated, adding more refinement rounds increases the number of noisy terms attached to the same terminal reward:

$$\text{Var}(\hat{g}_{\text{SF}}) = \sum_{t=0}^{T-1} \text{Var}\Big(\hat{A}_i \nabla_\theta \log \pi_\theta(a_t \mid s_t)\Big) = O(T).$$

Real trajectories are not independent, so we use this expression only as a horizon-scaling intuition.

Bridge pathwise gradient. The Bridge provides an additional gradient pathway:

$$\hat{g}_{\text{Bridge}} = \sum_{t=0}^{T-1} \frac{\partial \mathcal{L}}{\partial \mathbf{z}_t} \cdot \frac{\partial \mathbf{z}_t}{\partial \mathbf{h}_t} \cdot \frac{\partial \mathbf{h}_t}{\partial \theta}.$$

Because all T Bridge vectors $\mathbf{z}_t = \mathcal{B}_\xi(\mathbf{h}_t)$ are computed by the *same* MLP $\mathcal{B}_\xi$, the gradients $\partial \mathbf{z}_t / \partial \xi$ share the same Jacobian structure. When the Bridge is trained (Stage 1), it can learn a smoother CLIP-aligned conditioning space than the sampled text channel alone. The intended benefit is therefore not that correlation mathematically guarantees lower variance for an unnormalized sum; rather, the Bridge adds a pathwise signal that may improve the useful gradient signal-to-noise ratio relative to score-function-only updates.

Combined estimator. The combined gradient $\hat{g} = \alpha \hat{g}_{\text{SF}} + (1-\alpha)\hat{g}_{\text{Bridge}}$ (for some $\alpha \in (0,1)$) can benefit from both discrete policy exploration and continuous conditioning gradients. The current experiments show an earlier peak in a small diagnostic setting (Section B), which is consistent with this intuition but not by itself a definitive variance-reduction proof.

E. Additional Qualitative Examples

Table 5 summarizes additional NEXUS pipeline multi-round refinement examples. The pattern is consistent with the qualitative examples in Figure 2: some prompts are solved at the first generated round, while harder prompts benefit from a later corrective edit. Later rounds often sharpen residual details or improve realism while preserving already satisfied constraints. The table contains only measured scores and removes the placeholder training curves used in earlier drafts.

Table 5. Prompt-level summary of additional NEXUS iterative refinement trajectories shown in the qualitative study. Scores are VQA fractions after each round. The table replaces the earlier appendix figure to avoid placeholder curves and keep only measured outcomes.

Prompt	R0	R1	R2	R3	Main observation
Four balloons	0.0	0.0	1.0	1.0	Initial generation misses the required count/color set; the second edit recovers the compositional constraint and the final round preserves it.
Red apple + green pear	0.0	1.0	1.0	1.0	The first generated image satisfies the object/color binding; later edits mainly remove artifacts and sharpen the image.
Cat next to yellow cup	0.0	1.0	1.0	1.0	The spatial relation and object identity are satisfied after the first edit; later rounds improve clarity and realism.
Two cats	0.0	1.0	1.0	1.0	The two-object constraint is satisfied early; subsequent edits focus on visual quality while preserving the satisfied constraint.