UNIRECT-COT: ENHANCING GENERATION IN UNIFIED MULTIMODAL MODELS VIA REFLECTIVE RECTIFICATION WITH INHERENT UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Unified Multimodal Models (UMMs) aim to integrate visual understanding and generation within a single structure. However, these models exhibit a notable capability mismatch, where their understanding capability often outperforms their generation capability. This mismatch suggests that the model's rich internal knowledge, while effective for understanding tasks, is not fully utilized during generation. To address this, we draw inspiration from the human "Thinking-While-Drawing" paradigm, where humans continuously reflect on intermediate results and revise them according to their understanding of the intended target. In this paper, we propose UniRect-CoT, a training-free unified rectification chain-of-thought framework. We regard the multi-step denoising process in UMMs as an intrinsic visual reasoning process, whose intermediate states provide opportunities for continuous reflection and rectification. By leveraging the UMM's inherent understanding to reflect on intermediate results in light of the target instruction and rectify them accordingly, our framework forms a reflective chain of thought within a single denoising trajectory, enabling the model's internal knowledge to guide generation throughout the process. Specifically, the UMM generates a state-conditioned reflection, whose autoregressive negative log-likelihood under the same model defines a model-native differentiable semantic loss that translates textual reflection into gradients for latent rectification. Extensive experiments demonstrate that UniRect-CoT generalizes across existing flow-based UMMs, consistently enhancing overall text-to-image generation performance.

1 INTRODUCTION

Building upon the success of Large Language Models (LLMs) Guo et al. (2025); Team et al. (2025), Multimodal Large Language Models (MLLMs) have been developed to incorporate visual representations into language-centric frameworks. Recently, Unified Multimodal Models (UMMs) have emerged to integrate "understanding" and "generation" within a single structure. Such a unified structure creates the potential to leverage the model's understanding capability during generation.

Although UMMs achieve structural unification of understanding and generation through shared parameters, they exhibit a notable capability mismatch, where the understanding capability of UMMs often outperforms their generation capability Xie et al. (2025a). As shown in Fig. 1(a), given the instruction "A photo of four vases", the UMM incorrectly generates five vases, while its understanding branch correctly identifies the extra object and describes the desired correction. This example suggests that, while UMMs effectively leverage their internal knowledge for understanding, this same knowledge is not fully used during generation. Therefore, a natural question arises: Can we use the UMM's inherent understanding capability to guide generation?

To achieve this, we draw inspiration from the human creation process of "Thinking-While-Drawing." Specifically, given an instruction, humans continuously reflect on intermediate results by comparing the current result with the intended target and revise them accordingly. This reflective process allows existing knowledge to be used throughout creation. In contrast, current UMMs lack an explicit reflection mechanism for generation. Fortunately, the multi-step denoising process in UMMs can be regarded as an intrinsic visual reasoning process, naturally producing intermediate results that

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.

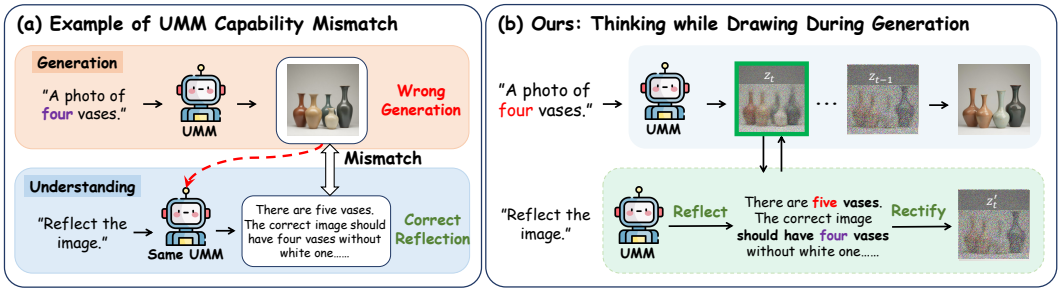


Figure 1: **Capability mismatch in UMMs and overview of UniRect-CoT.** (a) A UMM may fail to follow the generation instruction while its understanding branch can identify the error and describe the desired correction. (b) UniRect-CoT realizes "Thinking-While-Drawing" by interleaving state-conditioned reflection and rectification within the same denoising trajectory.

resemble the human creation process. This offers a valuable opportunity to leverage the UMM's understanding capability to iteratively reflect on and rectify intermediate generation results during denoising, forming a chain of thought that guides generation.

Based on these insights, we propose UniRect-CoT (Fig. 1(b)), a training-free unified rectification chain-of-thought framework that interleaves reflection and rectification within the same denoising trajectory. At each step of this chain, the UMM first inspects the current intermediate result, reflects on its discrepancies with the user instruction, and then uses this reflection to rectify the intermediate state. Specifically, at selected timesteps, we map the noisy intermediate state to an estimated clean image that can be inspected by the UMM's understanding branch. Based on the estimated image and the user instruction, the UMM produces a state-conditioned reflective description of what the generated image should depict to satisfy the instruction. Using the estimated image as visual context, we teacher-force the reflective description through the same UMM and use its autoregressive negative log-likelihood as a model-native differentiable semantic loss. Backpropagating this loss to the intermediate state provides gradients for rectification without external semantic supervision. To improve the reliability of rectification, we further introduce Greedy Iterative Trajectory Optimization (GITO), which explores successive rectified candidates and selects among them according to the user instruction. By repeatedly applying this process, UniRect-CoT enables the UMM to leverage its understanding capability for continuous reflection and rectification during generation, realizing "Thinking-While-Drawing."

The key contributions of this work are summarized as follows:

- We propose UniRect-CoT, a training-free unified rectification chain- of-thought framework that realizes "Thinking-While-Drawing" by interleaving state-conditioned reflection and rectification within a single denoising trajectory.
- We develop a model-native differentiable semantic loss from the UMM's autoregressive likelihood, converting discrete textual reflection into gradients for latent rectification without additional training or external semantic supervision.
- Extensive experiments across four flow-based UMMs and two text-to-image benchmarks demonstrate the effectiveness and generality of UniRect-CoT.

2 Related Work

2.1 Unified Multimodal Models

Unified Multimodal Models (UMMs) integrate multimodal understanding and generation within a single framework Zhang et al. (2025). Existing UMMs follow two paradigms: autoregressive (AR) and AR+Diffusion. *The AR paradigm* includes next-token prediction architectures Team (2024); Chen et al. (2025); Liu et al. (2024); Qu et al. (2024), which unify the prediction of textual and visual tokens, as well as Masked Autoregressive (MAR) models Li et al. (2024) and their variants Wu et al. (2025c), which support direct generation without vector quantization. *The AR+Diffusion paradigm* combines Transformers with diffusion or flow-matching objectives through architectural designs.

Serial configurations Lin et al. (2025); Wu et al. (2025a;b) use AR representations to condition the diffusion generation process; parallel mechanisms Shi et al. (2024); Deng et al. (2025); Wang et al. (2025) connect understanding and generation through shared intermediate features; and unified structures Zhou et al. (2025); Xie et al. (2025b) directly incorporate diffusion or flow-matching heads into a shared Transformer backbone. In this paper, we focus on AR+Diffusion UMMs, as our method leverages the UMM's understanding capability to rectify its denoising trajectory.

2.2 Test-Time Scaling for Multimodal Generation

Test-time scaling improves multimodal generation by allocating additional computation at inference time. Recent methods introduce chain-of-thought (CoT) reasoning through textual or interleaved multimodal representations Gu et al. (2025); Qin et al. (2025), while others perform multi-round generation and refinement by revising textual conditions based on previously generated outputs Khan et al. (2025); Lyu et al. (2025). These methods mainly reason before generation or refine across multiple generation rounds. In contrast, UniRect-CoT performs state-conditioned reflection and rectification within the same denoising trajectory, allowing the reflection to adapt to intermediate generation results. Gradient-based guidance steers diffusion or flow-based sampling through differentiable objectives Dhariwal & Nichol (2021); Liu et al. (2023); Song et al. (2023); Bansal et al. (2023); Yu et al. (2023). These methods establish a general mechanism for modifying the generation trajectory at inference time. UniRect-CoT builds on this mechanism but differs in how the semantic loss is constructed: we derive a model-native differentiable semantic loss from the UMM's own reflection and autoregressive likelihood, enabling textual reflection to guide trajectory rectification without an external semantic oracle.

3 Preliminaries

Flow Matching. Flow Matching Lipman et al. (2022) has emerged as the successor to DDPMs in text-to-image synthesis, distinguished by its superior sampling efficiency. Standard implementations typically utilize a pre-trained autoencoder (comprising encoder $\mathcal{E}$ and decoder $\mathcal{D}$) to map input images x into a latent representation $z_0 = \mathcal{E}(x)$. Let z_0 denote a clean latent, $\epsilon \sim \mathcal{N}(0, I)$ Gaussian noise, and c a text prompt. The generative process is modeled as a deterministic Ordinary Differential Equation (ODE), $\frac{dz_t}{dt} = v(z_t, t)$ for $t \in [0, 1]$, which defines a continuous trajectory from Gaussian noise ϵ to data z_0. This trajectory is typically parameterized via linear interpolation: $z_t = (1-t)z_0 + t\epsilon$. To enable generation, a velocity network v_θ is trained to regress the vector field by minimizing the matching objective:

$$\mathcal{L}_{diff} = \mathbb{E}_{t,z_0,\epsilon}\|(\epsilon - z_0) - v_\theta(z_t, t, c)\|^2. \tag{1}$$

To bridge intermediate noisy states and the clean data, Look-Ahead Estimation leverages a core property of linear flow parameterization that z_0 can be estimated from any z_t. Similar to Tweedie's formula in the context of diffusion models, the estimated clean latent $\hat{z}_{0|t}$ can be derived directly from the current state and the predicted velocity:

$$\hat{z}_{0|t} = z_t - t \cdot v_\theta(z_t, t, c). \tag{2}$$

This estimation serves as a bridge for applying operations defined on clean data to noisy states.

Gradient-Based Guidance for Flow Models. Gradient-based guidance steers diffusion or flow-based generation using gradients from differentiable objectives Dhariwal & Nichol (2021); Bansal et al. (2023); Chung et al. (2022); Song et al. (2023); Yu et al. (2023). Given a target condition y and a differentiable objective $\mathcal{L}(z_0, y)$, the conditional distribution can be expressed as $p_0(z_0|y) \propto p_0(z_0)\exp(-\mathcal{L}(z_0, y))$. For flow-based generation, the corresponding conditional gradient modifies the velocity as

$$\hat{v}(z_t, t, c) = v_\theta(z_t, t, c) - \gamma \nabla_{z_t} \log p_t(y|z_t),$$

where γ controls the guidance strength. Since the exact conditional gradient is generally intractable, training-free guidance commonly approximates it by evaluating the differentiable objective on a predicted clean sample. Using the look-ahead estimate in Eq. 2, the guidance gradient can be approximated as

$$\nabla_{z_t} \log p_t(y|z_t) \approx -\nabla_{z_t} \mathcal{L}(\hat{z}_{0|t}, y). \tag{3}$$

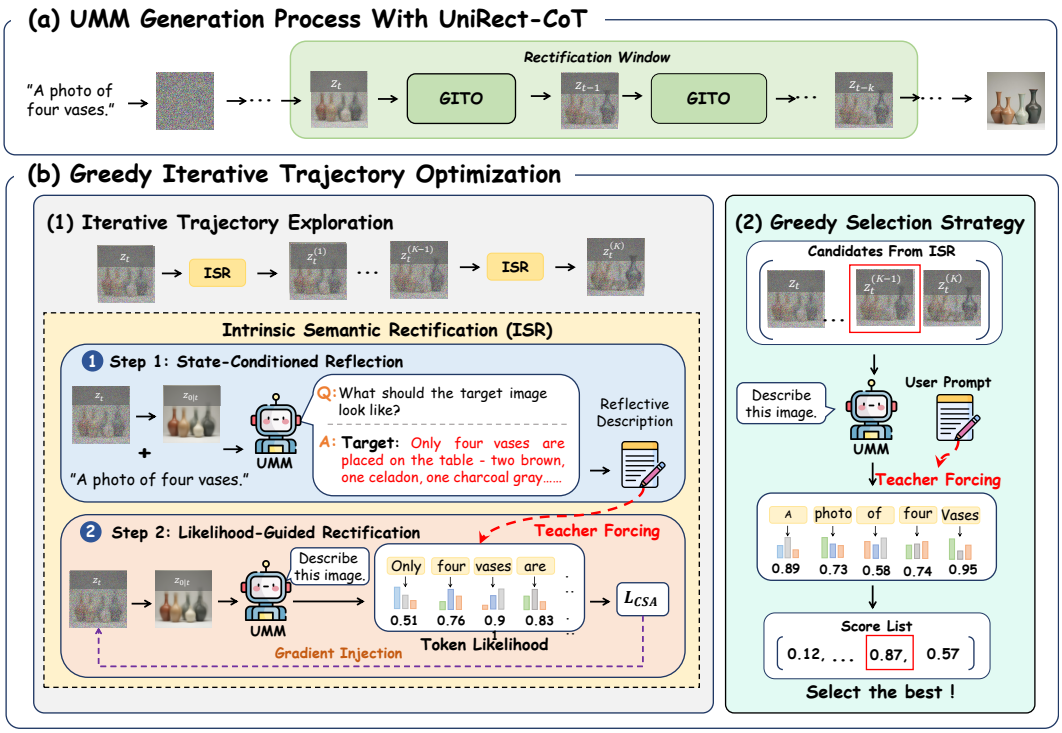


Figure 2: **Overall pipeline of UniRect-CoT.** (a) Reflection and rectification are interleaved with denoising within a selected rectification window. (b) At each rectification timestep, GITO performs iterative trajectory exploration and greedy selection. ISR combines state-conditioned reflection with likelihood-guided rectification. The reflective target is generated in the first ISR update and reused by subsequent updates at the same timestep. The resulting candidates, including the original state, are evaluated using the user instruction to determine the state retained for subsequent denoising.

This formulation shows that a differentiable loss defined on the predicted clean sample can provide gradients to rectify the sampling trajectory. In UniRect-CoT, we use this mechanism as the optimization step for applying reflection to the denoising trajectory. Our key design in this step lies in the construction of the loss: we derive a model-native differentiable semantic loss from the UMM's state-conditioned reflection and autoregressive likelihood.

4 METHODS

In this section, we present UniRect-CoT(Fig. 2), a training-free framework that realizes "Thinking-While-Drawing" by leveraging the UMM's inherent understanding to reflect on and rectify intermediate results within a single denoising trajectory. We first evaluate the UMMs' ability to judge whether intermediate generation results satisfy the user instruction, and then describe how successive reflections form a chain along the denoising trajectory in Sec. 4.1. Then, in Sec. 4.2, we introduce Intrinsic Semantic Rectification (ISR), which first generates state-conditioned reflection and then converts its corrective target into a model-native differentiable semantic loss for latent rectification. Finally, Sec. 4.3 details our Greedy Iterative Trajectory Optimization (GITO), which explores candidate states through iterative ISR updates and selects the state for subsequent denoising according to the original user instruction.

4.1 REFLECTIVE REASONING ALONG THE DENOISING TRAJECTORY

To enable reflection during generation, we first make intermediate denoising states accessible to the UMM's understanding branch. For a latent state z at flow time t, Look-Ahead Estimation gives

$$\hat{z}_{0|t}(z) = z - t\, v_\theta(z, t, c_{\text{user}}), \qquad \hat{x}_{0|t}(z) = \mathcal{D}\big(\hat{z}_{0|t}(z)\big)\,. \tag{4}$$

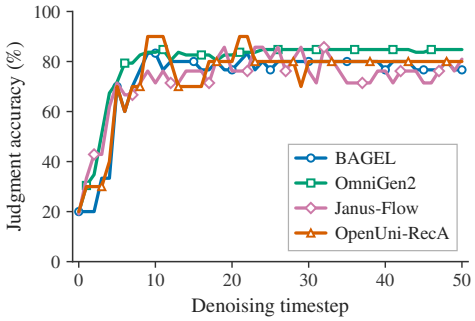


Figure 3: **Judgment accuracy across denoising timesteps.** We measure how often each UMM's judgment of whether a look-ahead image satisfies the user instruction agrees with the GenEval correctness label for the same image.

GenEval Ghosh et al. (2023) is a widely used text-to-image benchmark with explicit correctness criteria for each test case. To assess whether intermediate denoising states contain sufficient semantic information for evaluation, we apply look-ahead estimation at different timesteps along each trajectory and ask the UMM's understanding branch whether the estimated images satisfy the user instruction. We measure judgment accuracy at each timestep by comparing the UMM's judgments with the GenEval correctness labels computed on the same images. As shown in Fig. 3, all four UMMs achieve high judgment accuracy well before generation is complete. This observation motivates using the UMM's understanding branch to assess intermediate results with respect to the original user instruction during denoising.

We propose to interleave reflection and rectification with denoising to iteratively refine intermediate results. At selected denoising steps, the UMM reflects on and rectifies the current intermediate result, allowing its inherent understanding to guide generation. As denoising progresses, subsequent reflections are conditioned on intermediate results shaped by preceding rectifications. This alternation links successive reflections through the evolving visual state, naturally forming a reflective chain of thought within a single denoising trajectory and realizing "Thinking-While-Drawing."

4.2 Intrinsic Semantic Rectification

To operationalize this reflective process, we introduce Intrinsic Semantic Rectification (ISR), which aligns the intermediate result with a reflective target derived from the UMM's own understanding. The ISR consists of two steps: state-conditioned reflection and likelihood-guided rectification.

Step 1: State-Conditioned Reflection. The user instruction specifies the desired content, while state-conditioned reflection identifies discrepancies in the current intermediate result that require rectification. For example, when the instruction specifies "four vases" but the estimated image contains five, the UMM can identify the extra vase and describe the desired four-vase image. To obtain such feedback, we provide the estimated image $\hat{x}_{0|t}(z_t)$, a reflection prompt c_{ref}, and the user instruction c_{user} to the UMM's understanding branch:

$$r_t = \mathcal{M}_{\text{und}}\big(\hat{x}_{0|t}(z_t), [c_{\text{ref}}; c_{\text{user}}]\big)\,. \tag{5}$$

From the resulting reflection r_t, we extract the reflective target c^t_{target} that describes the desired visual content after rectification.

Step 2: Likelihood-Guided Rectification. The reflective target describes the desired visual content, but its discrete textual form cannot directly rectify the latent state. To construct a differentiable objective, we teacher-force $c^t_{\text{target}} = (y_1, \ldots, y_L)$ through the same UMM, using the estimated image as visual context. We define the *Cyclic Semantic Alignment* (CSA) loss as the average autoregressive negative log-likelihood of the reflective target:

$$\mathcal{L}_{\text{CSA}}\left(z; c^t_{\text{target}}\right) = -\frac{1}{L}\sum_{i=1}^{L} \log P_\theta\big(y_i \mid y_{<i}, \hat{x}_{0|t}(z)\big)\,. \tag{6}$$

This model-native loss encourages the estimated image to align with the reflective target under the UMM's own understanding, without external supervision. To minimize the alignment objective, we employ a gradient injection mechanism. At each selected denoising step, we treat the latent state z_t as the optimization variable while keeping all model parameters and the reflective target fixed. Differentiating the CSA loss with respect to z_t gives the rectification gradient:

$$g_t = \nabla_{z_t}\mathcal{L}_{\text{CSA}}\left(z_t; c^t_{\text{target}}\right) \tag{7}$$

The gradient is computed through the UMM's understanding branch and propagated through the decoder and generation branch to the intermediate latent state. To limit excessive perturbations to

the denoising trajectory, we apply ℓ_2-norm gradient clipping:

$$\hat{g}_t = \begin{cases} \delta \dfrac{g_t}{\|g_t\|_2}, & \text{if } \|g_t\|_2 > \delta, \\ g_t, & \text{otherwise}, \end{cases} \tag{8}$$

where δ denotes the clipping threshold. Finally, we update the latent state using the clipped gradient:

$$z_t \leftarrow z_t - \eta\, \hat{g}_t, \tag{9}$$

where η controls the rectification strength.

4.3 Greedy Iterative Trajectory Optimization

A single rectification update may be insufficient to realize the desired correction. Repeated updates allow further exploration, but excessive intervention may disrupt the generation trajectory and reduce consistency with the original user instruction. To address this trade-off, we introduce Greedy Iterative Trajectory Optimization (GITO), which combines iterative exploration with greedy selection to determine the state retained for subsequent denoising.

Iterative Trajectory Exploration. Instead of relying on a single update, we explore the local neighborhood of the current latent state through iterative rectification. For each sampling step within the rectification window $\mathcal{W} = [s_{\text{start}}, s_{\text{end}}]$, we perform K ISR updates from $z_t^{(0)} = z_t$ to construct the candidate set $\mathcal{Z}_{\text{cand}} = \{z_t^{(0)}, \ldots, z_t^{(K)}\}$. The first update generates c_{target}^t from $z_t^{(0)}$; subsequent updates at the same timestep reuse this target. At each iteration, we recompute the look-ahead image and CSA gradient from the current candidate and update the latent as

$$z_t^{(k+1)} = z_t^{(k)} - \eta\, \hat{g}_t\Big(z_t^{(k)}\Big), \qquad k = 0, \ldots, K-1. \tag{10}$$

Greedy Selection Strategy. To avoid over-rectification, we do not assume the final iteration $z_t^{(K)}$ is optimal. Instead, we employ a Greedy Selection Strategy (GSS) to evaluate all candidates against the original user instruction c_{user}. The reflective target guides rectification of the current result, while the original instruction provides a common criterion for selection. This reduces selection's dependence on potentially inaccurate reflective targets and encourages alignment with the user's original instruction. Specifically, for $c_{\text{user}} = (u_1, \ldots, u_J)$, we teacher-force the instruction through the same UMM with each candidate's estimated image as visual context and compute

$$S_t(z) = \frac{1}{J} \sum_{i=1}^{J} \log P_\theta\big(u_i \mid u_{<i}, \hat{x}_{0|t}(z)\big)\,. \tag{11}$$

The retained state is selected as

$$z_t^* = \arg\max_{z \in \mathcal{Z}_{\text{cand}}} S_t(z). \tag{12}$$

We resume denoising from the selected state z_t^* and repeat reflection, iterative rectification, and greedy selection at subsequent rectification timesteps.

5 Experiments

5.1 Experimental Setup

Benchmarks and Baselines. To comprehensively assess text-to-image generation capabilities, we utilize GenEval Ghosh et al. (2023) to evaluate compositional generation skills, specifically focusing on object co-occurrence and attribute binding, while employing DPG-Bench Hu et al. (2024) to test robustness in handling complex and dense prompts. We apply UniRect-CoT to four representative flow-based UMMs: Janus-Flow Ma et al. (2025), OpenUni-RecA Xie et al. (2025a), OmniGen2 Wu et al. (2025b), and BAGEL Deng et al. (2025).

Implementation Details. Unless otherwise specified, we use 50 flow-matching sampling steps with the default classifier-free guidance settings from each baseline's official implementation. The rectification window is $\mathcal{W} = [5, 10]$ in sampling-step indices, with $K = 3$ ISR updates per rectification timestep. We set the gradient clipping threshold to $\delta = 0.001$ and the latent update step size to $\eta = 200$. Evaluations are conducted on NVIDIA H100 GPUs.

Table 1: **Quantitative comparison on GenEval and DPG-Bench.** We compare our method with state-of-the-art unified multimodal models (top) and verify the effectiveness of our method on standard baselines (bottom). Results marked with † are reproduced in this work by generating 4 consecutive samples per prompt, starting with a fixed initial seed. ↑ denotes that higher is better. We highlight **improvements** over the baseline and the best overall performance.

	DPG ↑	GenEval ↑						
Model	Score	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attri.	Overall
TokenFlow	73.4	0.97	0.66	0.40	0.84	0.17	0.26	0.55
OpenUni	79.0	0.99	0.71	0.55	0.82	0.25	0.42	0.62
Emu3	80.6	0.98	0.71	0.34	0.81	0.17	0.21	0.54
Show-o2	85.0	0.99	0.86	0.55	0.86	0.46	0.63	0.73
Janus	79.6	0.97	0.68	0.30	0.84	0.46	0.42	0.61
Janus-Pro-7B	84.1	0.99	0.89	0.59	0.90	0.79	0.66	0.80
Ours & Baselines								
Janus-Flow †	73.0	0.950	0.482	0.346	0.771	0.352	0.277	0.530
+ Ours	**75.2**	**0.959**	**0.609**	**0.419**	**0.811**	**0.455**	**0.350**	**0.601**
OpenUni-RecA †	80.1	0.987	0.906	0.478	0.877	0.415	0.680	0.724
+ Ours	**82.5**	**0.994**	**0.934**	**0.538**	**0.891**	**0.423**	**0.698**	**0.746**
OmniGen2 †	83.0	0.993	0.929	0.728	0.869	0.535	0.665	0.786
+ Ours	**84.2**	**0.994**	**0.957**	0.728	**0.875**	**0.553**	**0.670**	**0.796**
BAGEL †	85.0	0.991	0.957	0.743	0.867	0.490	0.610	0.776
+ Ours	**85.9**	**0.997**	**0.970**	**0.819**	**0.872**	0.485	**0.680**	**0.804**

Table 2: **Comparison of test-time strategies on BAGEL.** "Pre": pre-generation reasoning or prompt enhancement; "Multi": multiple full generation/refinement rounds; "Traj.": intervention during denoising; "State": feedback conditioned on the current visual result. † marks results reproduced under our unified inference and evaluation protocol. Boldface highlights results with UniRect-CoT.

Configuration					**GenEval ↑**						
Method	**Pre**	**Multi**	**Traj.**	**State**	**Single Obj.**	**Two Obj.**	**Counting**	**Colors**	**Position**	**Color Attri.**	**Overall**
Training-free test-time scaling on BAGEL											
BAGEL	✗	✗	✗	✗	99.1	95.7	74.3	86.7	49.0	61.0	77.6
Qwen Rewrite †	✓	✗	✗	✗	100.0	95.7	73.4	89.9	47.3	64.0	78.4
UiG †	✓	✓	✗	✓	99.1	95.7	75.3	87.8	48.8	61.5	78.1
TiR †	✓	✓	✗	✓	98.4	95.7	80.0	83.5	54.0	67.2	79.8
UniRect-CoT	✗	✗	✓	✓	**99.7**	**97.0**	**81.9**	**87.2**	**48.5**	**68.0**	**80.4**
Test-time scaling on reasoning-tuned BAGEL											
BAGEL Think †	✓	✗	✗	✗	99.1	93.6	77.8	87.2	54.0	66.7	79.7
Think + UniRect-CoT	✓	✗	✓	✓	**99.4**	**93.9**	**79.4**	**89.4**	**54.5**	**70.8**	**81.3**

5.2 Quantitative Results

Comparison across Unified Multimodal Models. As shown in Tab. 1, UniRect-CoT improves the overall scores of all four baselines on both benchmarks. Janus-Flow exhibits the largest GenEval gain, from 0.530 to 0.601, while its DPG-Bench score increases from 73.0 to 75.2. On the stronger BAGEL baseline, UniRect-CoT improves GenEval Overall from 0.776 to 0.804, with notable gains in counting ($0.743 \rightarrow 0.819$) and attribute binding ($0.610 \rightarrow 0.680$). These results support the applicability of trajectory rectification across UMMs with different architectures and baseline strengths.

Comparison with Test-Time Reasoning and Refinement. As shown in Tab. 2, UniRect-CoT achieves the highest GenEval Overall score among the evaluated training-free methods on BAGEL. It improves the baseline from 77.6 to 80.4, outperforming Qwen Rewrite Bai et al. (2025) (78.4), UiG Lyu et al. (2025) (78.1), and TiR Khan et al. (2025) (79.8). As shown in Fig. 5(c), UniRect-CoT achieves this score at $1.74\times$ the latency of standard 50-step BAGEL, with lower latency than both UiG and TiR. Qwen Rewrite is faster but achieves a lower score of 78.4, highlighting the performance–latency trade-off. We also evaluate BAGEL's Think mode, whose pre-generation reasoning capability is trained from reasoning aligned data Deng et al. (2025). Think improves the baseline score to 79.7, and applying UniRect-CoT raises it to 81.3 without additional training. This gain highlights the complementarity between learned pre-generation reasoning and our training-free trajectory rectification.

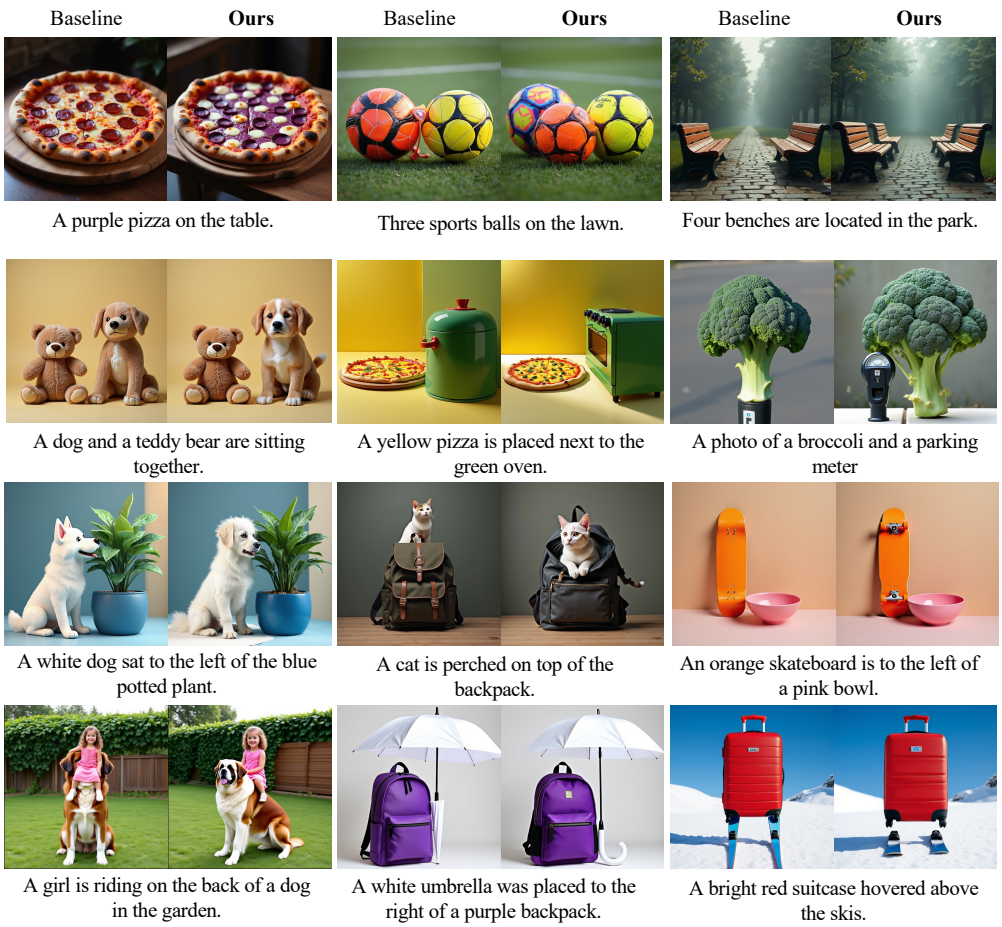


Figure 4: **Qualitative comparison.** Each pair shows BAGEL (left) and UniRect-CoT (right).

5.3 Qualitative Comparison

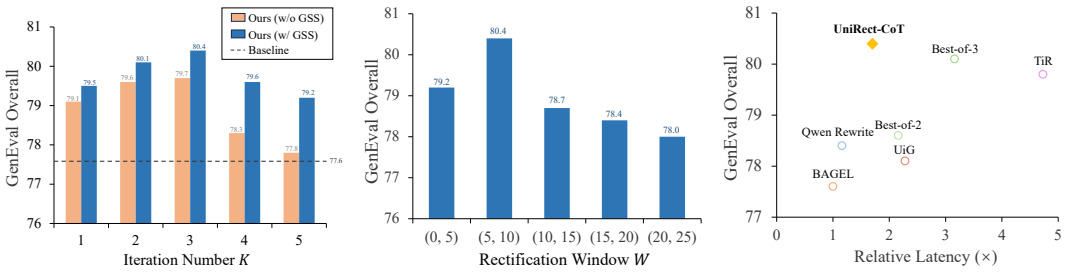
Fig. 4 presents representative comparisons with BAGEL on challenging compositional prompts. In these examples, UniRect-CoT more accurately follows the requested object counts, colors, attribute assignments, and spatial relations, illustrating its ability to correct compositional errors.

5.4 Ablation Studies and Efficiency Analysis

We conduct ablation studies on BAGEL using GenEval to evaluate rectification strategy, efficiency, and semantic target design. Detailed numerical results for the ablation and efficiency analyses are provided in Sec. A.1.

Rectification Strategy and Efficiency Analysis. Fig. 5(a) shows that performance peaks at $K = 3$, reaching 80.4 with GSS and 79.7 without GSS, and declines with further updates. GSS consistently improves performance and mitigates over-rectification: at $K = 5$, it achieves 79.2, compared with 77.8 without selection, although both remain below their scores at $K = 3$. We therefore use $K = 3$ to balance quality and cost. The best window is $\mathcal{W} = [5, 10]$ (Fig. 5(b)); earlier and later windows yield lower scores, suggesting a balance between reflection reliability and the flexibility of the intermediate state. Fig. 5(c) further compares UniRect-CoT with alternative test-time strategies. UniRect-CoT achieves higher GenEval Overall scores with lower latency than the evaluated Best-of-N configurations. Qwen Rewrite incurs less latency but achieves a lower score, highlighting a different trade-off.

Semantic Target Design. Tab. 3 examines whether reflection provides better guidance than directly using the original instruction as the target for gradient-based latent updates. Our method improves Overall from 78.7 to 79.7 without GSS and from 79.1 to 80.4 with GSS compared with user-only



(a) Iterative Rectification (b) Rectification Window (c) Performance vs. Latency

Figure 5: **Ablation of rectification strategy and efficiency Analysis.** (a) Effect of the number of rectification updates K and the Greedy Selection Strategy (GSS). (b) Effect of the rectification window $\mathcal{W}$ at $K = 3$. (c) GenEval Overall versus relative latency for UniRect-CoT and competing test-time strategies. Latency is normalized to standard BAGEL inference ($1.0\times$, 24.7 s/image).

Table 3: **Ablation of semantic target design.** "User", "Exp.", and "Ref." denote the original instruction c_{user}, its text-only expansion, and the state-conditioned reflective target c^t_{target}, respectively. Static targets remain fixed across the rectification window; dynamic targets are regenerated at each rectification timestep and reused for its K updates. "–" indicates that the corresponding component is disabled.

Configuration				GenEval ↑						
Setting	**Update**	**ITE Target**	**GSS Target**	**Single Obj.**	**Two Obj.**	**Counting**	**Colors**	**Position**	**Color Attri.**	**Overall**
BASELINE	–	–	–	99.1	95.7	74.3	86.7	49.0	61.0	77.6
ITE ONLY	Static	User	–	99.7	93.9	76.3	86.7	48.5	67.0	78.7
	Static	Exp.	–	99.7	94.2	76.9	87.2	48.8	60.5	77.9
	Static	Ref.	–	100.0	94.7	77.8	86.2	46.5	69.3	79.1
	Dynamic	Ref.	–	100.0	95.5	80.3	87.8	46.0	69.0	79.7
ITE + GSS	Static	User	User	99.4	95.2	76.5	88.3	51.0	64.3	79.1
	Static	Exp.	User	99.4	94.4	77.8	87.5	48.8	60.5	78.1
	Static	Ref.	User	99.7	96.2	77.2	86.2	47.8	69.3	79.4
	Dynamic	Ref.	Ref.	99.1	95.7	79.4	85.4	47.0	66.5	78.8
	Dynamic	Ref.	User	99.7	97.0	81.9	87.2	48.5	68.0	80.4

guidance. In contrast, text-only expansion underperforms the original instruction in both settings, suggesting that the gains come from feedback on the current image rather than prompt expansion alone. Static reflection already improves upon the original instruction, and dynamic reflection provides further gains, supporting the value of updating the corrective target as the visual state evolves. For candidate selection, however, reflection-based GSS achieves 78.8, below both instruction-based GSS (80.4) and no selection (79.7). These results support distinct roles for guidance and selection: reflection adapts the proposed correction to the current image, while the original instruction provides a consistent criterion for evaluating candidates. Building on the general guidance mechanism of UG Bansal et al. (2023), our method introduces adaptive targets generated through the UMM's own understanding and a separate selection criterion based on the original instruction. The ablation results demonstrate the benefit of these choices over directly optimizing a fixed user instruction.

6 CONCLUSION

In this paper, we proposed UniRect-CoT, a training-free unified rectification chain- of-thought framework that uses the inherent understanding of UMMs to improve image generation. Inspired by "Thinking-While-Drawing," our framework interleaves reflection and rectification within a single denoising trajectory. Intrinsic Semantic Rectification converts state-conditioned corrective targets into gradients for latent updates, while Greedy Iterative Trajectory Optimization selects among the resulting candidates using the original user instruction. Extensive experiments on four flow-based UMMs show consistent improvements on GenEval and DPG-Bench, demonstrating the effectiveness and generality of UniRect-CoT.

AI USE STATEMENT

We used generative AI tools to assist with literature searches and translation, to provide feedback on the methodology and experimental evaluation, and to edit and rewrite selected manuscript passages for clarity and readability. The authors take full responsibility for the final manuscript, including all references, scientific claims, and experimental results.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *Proc. CVPR*, pp. 843–852, 2023.

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

Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

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

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Proc. NeurIPS*, 34:8780–8794, 2021.

Dhruba Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment, 2023. URL https://arxiv.org/abs/2310.11513.

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

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

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.

Mohammad Abdul Hafeez Khan, Yash Jain, Siddhartha Bhattacharyya, and Vibhav Vineet. Test-time prompt refinement for text-to-image models, 2025. URL https://arxiv.org/abs/2507.22076.

Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Proc. NeurIPS*, 37:56424–56445, 2024.

Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. Uniworld: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with ringattention. *arXiv preprint*, 2024.

Xihui Liu, Dong Huk Park, Samaneh Azadi, Gong Zhang, Arman Chopikyan, Yuxiao Hu, Humphrey Shi, Anna Rohrbach, and Trevor Darrell. More control for free! image synthesis with semantic diffusion guidance. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 289–299, 2023.

Yuanhuiyi Lyu, Chi Kit Wong, Chenfei Liao, Lutao Jiang, Xu Zheng, Zexin Lu, Linfeng Zhang, and Xuming Hu. Understanding-in-generation: Reinforcing generative capability of unified model via infusing understanding into generation, 2025. URL https://arxiv.org/abs/2509.18639.

Yiyang Ma, Xingchao Liu, Xiaokang Chen, Wen Liu, Chengyue Wu, Zhiyu Wu, Zizheng Pan, Zhenda Xie, Haowei Zhang, Xingkai yu, Liang Zhao, Yisong Wang, Jiaying Liu, and Chong Ruan. Janusflow: Harmonizing autoregression and rectified flow for unified multimodal understanding and generation, 2025. URL https://arxiv.org/abs/2411.07975.

Luozheng Qin, Jia Gong, Yuqing Sun, Tianjiao Li, Mengping Yang, Xiaomeng Yang, Chao Qu, Zhiyu Tan, and Hao Li. Uni-cot: Towards unified chain-of-thought reasoning across text and vision. *arXiv preprint arXiv:2508.05606*, 2025.

Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. *arXiv preprint arXiv:2412.03069*, 2024.

Weijia Shi, Xiaochuang Han, Chunting Zhou, Weixin Liang, Xi Victoria Lin, Luke Zettlemoyer, and Lili Yu. Lmfusion: Adapting pretrained language models for multimodal generation. *arXiv preprint arXiv:2412.15188*, 2024.

Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In *Proc. ICML*, pp. 32483–32498. PMLR, 2023.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.

Zeyu Wang, Zilong Chen, Chenhui Gou, Feng Li, Chaorui Deng, Deyao Zhu, Kunchang Li, Weihao Yu, Haoqin Tu, Haoqi Fan, and Cihang Xie. Lightfusion: A light-weighted, double fusion framework for unified multimodal understanding and generation, 2025. URL https://arxiv.org/abs/2510.22946.

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

Chenyuan Wu, Pengfei Zheng, Ruiran Yan, Shitao Xiao, Xin Luo, Yueze Wang, Wanli Li, Xiyan Jiang, Yexin Liu, Junjie Zhou, Ze Liu, Ziyi Xia, Chaofan Li, Haoge Deng, Jiahao Wang, Kun Luo, Bo Zhang, Defu Lian, Xinlong Wang, Zhongyuan Wang, Tiejun Huang, and Zheng Liu. Omnigen2: Exploration to advanced multimodal generation. *arXiv preprint arXiv:2506.18871*, 2025b.

Size Wu, Wenwei Zhang, Lumin Xu, Sheng Jin, Zhonghua Wu, Qingyi Tao, Wentao Liu, Wei Li, and Chen Change Loy. Harmonizing visual representations for unified multimodal understanding and generation. *arXiv preprint arXiv:2503.21979*, 2025c.

Ji Xie, Trevor Darrell, Luke Zettlemoyer, and XuDong Wang. Reconstruction alignment improves unified multimodal models. *arXiv preprint arXiv:2509.07295*, 2025a.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*, 2025b.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. *arXiv preprint arXiv:2303.09833*, 2023.

Xinjie Zhang, Jintao Guo, Shanshan Zhao, Minghao Fu, Lunhao Duan, Jiakui Hu, Yong Xien Chng, Guo-Hua Wang, Qing-Guo Chen, Zhao Xu, et al. Unified multimodal understanding and generation models: Advances, challenges, and opportunities. *arXiv preprint arXiv:2505.02567*, 2025.

Chunting Zhou, LILI YU, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *ICLR*, 2025.

A Appendix

A.1 Detailed Ablation and Efficiency Results

This section provides the numerical results underlying the three ablation and efficiency panels in the main paper.

A.1.1 Number of Rectification Updates and Greedy Selection

Tab. 4 reports the detailed effects of the number of ISR updates and GSS. Overall performance improves as K increases from 1 to 3, reaching 79.7 without GSS and 80.4 with GSS, before declining with further updates. GSS improves Overall at every evaluated value of K, with a larger benefit under more aggressive rectification. At $K = 5$, selection raises the score from 77.8 to 79.2, although performance remains below that of $K = 3$. These results indicate that greedy selection mitigates, but does not fully prevent, the degradation caused by excessive rectification. We therefore adopt $K = 3$ as the default setting.

Table 4: **Detailed ablation of ISR iterations and GSS on BAGEL.** The rectification window is fixed to $\mathcal{W} = [5, 10]$. We report category-level GenEval scores for different iteration numbers K, with and without GSS.

K	GSS	GenEval ↑						
		Single	Two	Count	Colors	Position	Attri.	Overall
1	✗	99.7	95.0	78.8	86.2	46.5	68.5	79.1
	✓	99.7	96.5	80.3	85.6	48.0	67.0	79.5
2	✗	100	95.5	80.6	86.7	48.0	67.0	79.6
	✓	100	96.5	82.2	85.6	48.8	67.5	80.1
3	✗	100	95.5	80.3	87.8	46.0	69.0	79.7
	✓	99.7	97.0	81.9	87.2	48.5	68.0	80.4
4	✗	99.7	93.9	75.6	85.6	46.5	68.3	78.3
	✓	99.4	95.2	80.0	85.1	48.5	69.5	79.6
5	✗	98.8	92.7	75	85.9	47.0	68.0	77.8
	✓	98.8	96.0	78.1	85.4	49.3	68.0	79.2

A.1.2 Rectification Window

Tab. 5 examines the effect of the rectification window with $K = 3$ and GSS enabled. The 5–10 window achieves the highest Overall score of 80.4, while shifting rectification to either the earlier window or later windows reduces performance. The category-level results also reveal a trade-off: the 0–5 window achieves a higher Position score (51.0 versus 48.5), but lower Counting and Attribute scores than the default window. Thus, the best intervention window for overall instruction following does not necessarily maximize every individual category. These observations support the default window as a balance across the evaluated compositional capabilities.

Table 5: **Detailed ablation of the rectification window on BAGEL.** Window labels denote sampling-step ranges. All configurations use $K = 3$ and GSS.

Window	GenEval ↑						
	Single	Two	Count	Colors	Position	Attri.	Overall
0–5	99.7	94.9	78.1	85.9	51.0	65.7	79.2
5–10	99.7	97.0	81.9	87.2	48.5	68.0	80.4
10–15	99.7	95.7	79.7	84.8	47.3	64.5	78.7
15–20	99.4	95.5	77.2	85.1	47.5	65.8	78.4
20–25	100	94.7	77.2	85.1	46.5	65	78.0

A.1.3 Performance and Absolute Latency

Tab. 6 complements the main efficiency figure with category-level GenEval scores and absolute inference latency. UniRect-CoT achieves an Overall score of 80.4 with a latency of 43.2 s/image. It attains a higher score with lower latency than UiG, TiR, and the evaluated Best-of-N configurations. For example, Best-of-3 achieves 80.1 at 76.9 s/image, while TiR achieves 79.8 at 116.9 s/image. Qwen Rewrite offers a faster alternative at 28.7 s/image, but achieves a lower score of 78.4. These results show that trajectory-level rectification provides a favorable performance–latency trade-off among the evaluated strategies, while still introducing additional computation over standard BAGEL inference.

Table 6: **Detailed performance–latency comparison on BAGEL.** Absolute latency is reported in seconds per output image, and relative latency is normalized to standard 50-step BAGEL.

Method	GenEval ↑							Latency ↓	
	Single	Two	Count	Colors	Position	Attri.	Overall	Absolute (s/image)	Relative
BAGEL	99.1	95.7	74.3	86.7	49.0	61.0	77.6	24.7	1.00×
Qwen Rewrite	100.0	95.7	73.4	89.9	47.3	64.0	78.4	28.7	1.16×
UiG	99.1	95.7	75.3	87.8	48.8	61.5	78.1	56.4	2.28×
TiR	98.4	95.7	80.0	83.5	54.0	67.2	79.8	116.9	4.73×
Best-of-2	99.3	93.4	79.3	87.5	49.2	63.0	78.6	51.8	2.09×
Best-of-3	99.7	95.6	82.5	86.4	48.3	68.3	80.1	76.9	3.11×
UniRect-CoT	99.7	97.0	81.9	87.2	48.5	68.0	80.4	43.2	1.74×

B Repair-Rate Comparison of Training-Free Methods

Tab. 7 evaluates both the correction of baseline failures and the preservation of previously correct outputs on BAGEL. Repair and regression rates are computed over the same 510 baseline failures and 1702 baseline successes, respectively. UniRect-CoT corrects 86 failures and degrades 28 previously correct outputs, corresponding to a repair rate of 16.86% and a regression rate of 1.65%. Its net gain of 58 correct cases is the largest among the evaluated methods, while its regression rate is the lowest. Although Qwen Rewrite and TiR achieve higher repair rates (39.02% and 32.35%), they also incur higher regression rates (10.81% and 6.99%), resulting in smaller net gains of 15 and 46 cases, respectively. Compared with UiG, UniRect-CoT combines a higher repair rate with a lower regression rate. These results suggest that the observed net benefit of UniRect-CoT comes from balancing error correction with the preservation of correct outputs, rather than maximizing repair rate alone. Net gain measures changes in sample-level correctness and is distinct from the macro-averaged GenEval Overall score.

Table 7: **Repair and regression analysis on BAGEL.** Repair rate is computed over 510 baseline failures, while regression rate is computed over 1702 baseline successes. Net gain denotes the number of corrected cases minus the number of degraded cases. All methods are evaluated on the same paired sample set. Boldface highlights the best value in each column.

Method	Repair rate (%) ↑	Regression rate (%) ↓	Net gain (cases) ↑
Qwen Rewrite	**39.02**	10.81	+15
UiG	7.84	2.06	+5
TiR	32.35	6.99	+46
UniRect-CoT	16.86	**1.65**	**+58**

C Additional Visualizations

Fig. 6 presents additional qualitative comparisons with the baseline on prompts involving object counts, attribute binding, and spatial relations. In these selected examples, UniRect-CoT corrects errors such as missing or excess objects and mismatched attributes, improving alignment with the

requested content. These visual comparisons complement the quantitative results by illustrating how the method addresses compositional errors in individual outputs.

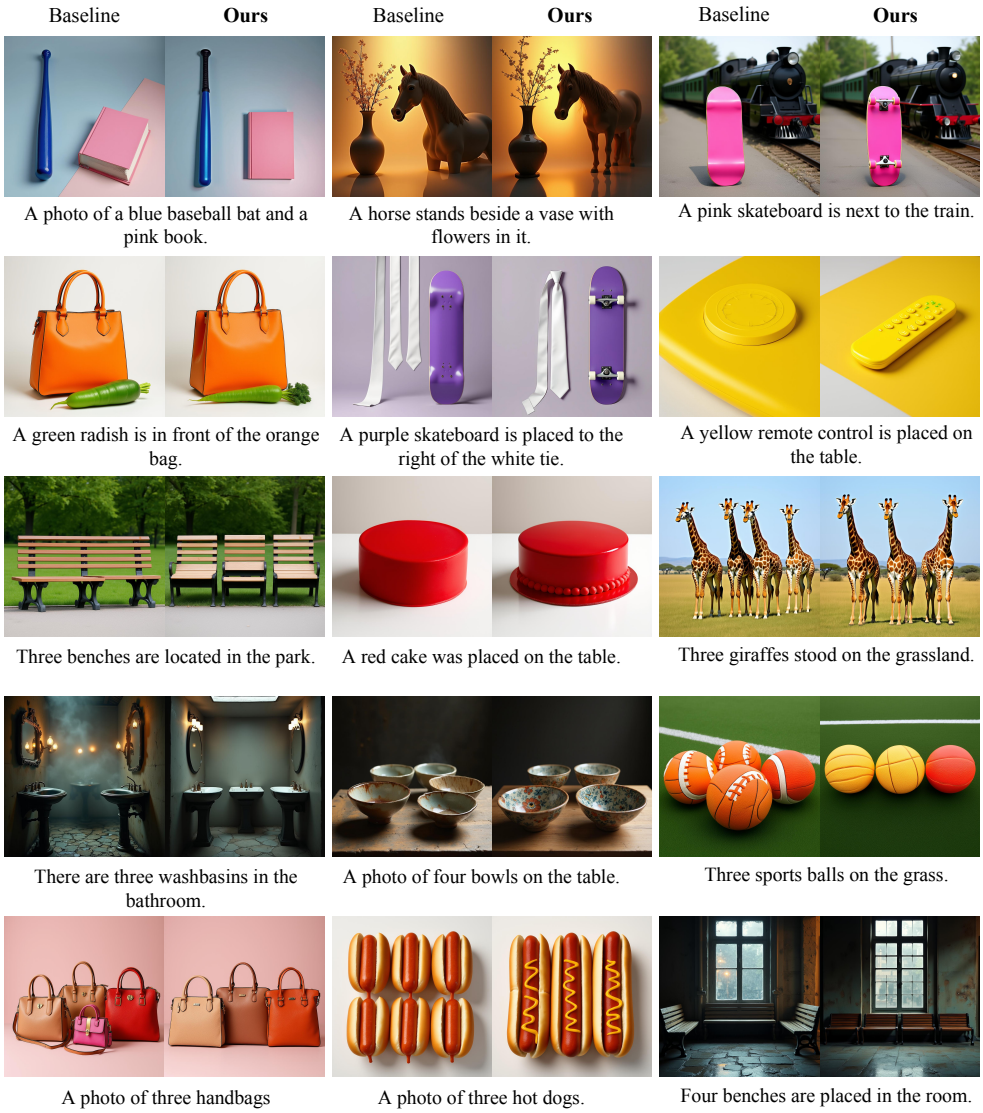


Figure 6: **More Qualitative Comparisons with Baseline.** We demonstrate further visual comparisons with baseline methods in various challenging scenarios. This further proves the shortcomings of baseline methods in terms of semantic misalignment and object omission, while our method faithfully follows user instructions and generates accurate visual details.

D Prompt Template

Fig. 7 presents the prompt template used to generate reflections based on the look-ahead image and the original user instruction. The `modifications` field describes the proposed changes, while the text in the `final_image` field serves as the reflective target c_{target} for latent rectification.

E Limitations

Although UniRect-CoT improves generation without additional training, it introduces extra inference cost through reflection generation, latent gradient computation, and candidate scoring. The number of rectification updates and the intervention window therefore require balancing generation quality against computational cost. Moreover, additional updates do not necessarily improve

Prompt Template

Inputs:

Given the text prompt "{prompt}" and the provided noisy intermediate image, identify the changes needed to make the final image fully match the prompt. Ignore noise and artifacts, and prioritize the text prompt whenever it conflicts with the image.

Outputs:

Return a JSON dictionary with exactly two fields:
{
"Modifications": "A concise, actionable description of what should be changed, including any corrections to objects, counts, attributes, spatial relationships, composition, or atmosphere. Preserve visible details that are consistent with the prompt.",
"Reflection_Target": "One concise paragraph describing the envisioned final image after those modifications. Describe its visual details and atmosphere directly, without mentioning the editing process, noise, artifacts, or the original image."
}

Figure 7: **Prompt Template.**

performance, and GSS mitigates but does not eliminate the degradation associated with excessive rectification. The framework also depends on the UMM's inherent understanding capability. When the model misinterprets an intermediate image or produces an inaccurate reflective target, the resulting semantic gradients may guide generation toward an incorrect correction. Using the original instruction for candidate selection reduces reliance on the reflective target, but the model-native likelihood remains a proxy for semantic correctness. Consequently, a higher selection score does not guarantee a more accurate final image, and intervention can still disrupt outputs that would otherwise be correct.