WHEN REFLECTION FAILS: IMPROVING SELF-REFLECTIVE UNIFIED MULTIMODAL MODELS WITH CORRECTION AMPLIFICATION AND FALLBACK

Anonymous authors
Paper under double-blind review

ABSTRACT

Self-reflective unified multimodal models (UMMs) offer a promising paradigm for complex image generation by enabling a single model to inspect its own outputs and iteratively refine them. However, they often struggle with generation tasks involving multiple objects and spatial relationships. To mitigate this problem, existing approaches typically rely on utilizing agents to perform reflection and generation separately, rather than exploiting the unified model's intrinsic capability. We revisit the source of these failures within the unified model and identify two distinct bottlenecks: *reflection failure*, where the model fails to diagnose errors in its reflection, and *correction failure*, where the model cannot effectively translate correction feedback into visual outputs. Based on this decomposition, we propose *RECAF*, a training-free framework that enhances both stages while retaining the unified model as the central component for reflection and generation. Experiments on *GenEval*, *GenEval++*, and *GenEval2* demonstrate that *RECAF* consistently improves self-reflective image generation, outperforming baselines by 5.0, 5.7, and 3.5 points, respectively.

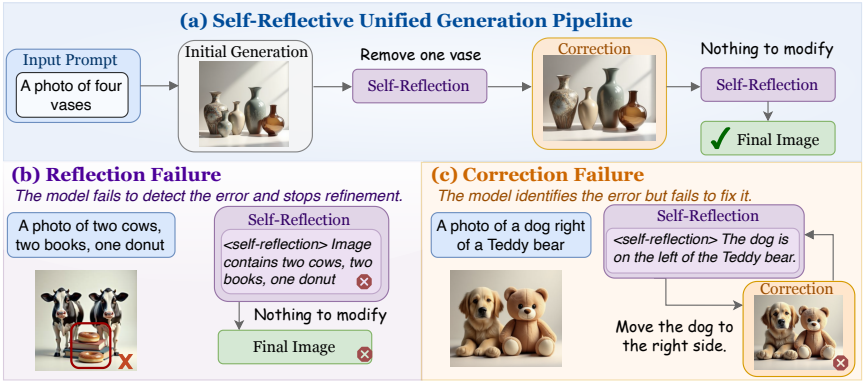

Figure 1: **Self-reflective unified multimodal models (UMMs) show great potential in correcting their own mistakes in image generation, but struggle with compositional image generation.** We show the overall workflow of self-reflective UMMs (a). We further identify two bottlenecks of self-reflective UMMs in compositional image generation: reflection (b) and correction (c).

1 INTRODUCTION

As one of the fundamental human capabilities, self-reflection enables individuals to monitor their own cognitive processes (Nelson, 1990) and adapt subsequent behavior accordingly (Rouault et al., 2018). Inspired by this, recent efforts (Liu et al., 2026; Qin et al., 2026; Huang et al., 2026a) have begun equipping Unified Multimodal Models (UMMs) with self-reflective capability. They offer an appealing approach for complex image generation by enabling a single model to inspect its own generations and iteratively refine them. Despite this progress, existing self-reflective UMMs still struggle to generate scenes involving multiple objects and spatial relationships, as shown in Fig. 1.

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.

To address these failures, a common strategy is to fully externalize different stages of the reflection process. Methods such as GenAgent (Jiang et al., 2026) and GenArtist (Wang et al., 2024) employ specialized agents to inspect generated outputs, provide corrective feedback, and invoke a separate diffusion model for image re-generation. Although such pipelines can substantially improve compositional image generation, this design departs from the central premise of a self-reflective UMM, in which a single model can autonomously inspect and refine its own generations. We therefore ask: *Why do self-reflective UMMs struggle on compositional image generation, and how can these failures be mitigated?*

To answer these questions, we first revisit potential reasons why self-reflective compositional image generation fails. Rather than treating unsuccessful refinement as a single holistic failure, our analysis reveals that it stems from two distinct sources: *reflection failure (Fig. 1 b)*, where the model fails to recognize errors in its own generation, and *correction failure (Fig. 1 c)*, where the model correctly diagnoses the error but fails to realize the intended suggestion during correction. Based on this decomposition, we propose *RE*flection *C*orrection *A*mplification with *F*allback (*RECAF*), a training-free framework that improves self-reflective image generation by explicitly targeting these two failure modes. For reflection failure, *RECAF* adds a fallback that rechecks the prompt's constraints whenever the model proposes to stop, catching inconsistencies that the model's own reflection has missed; if the model catches some errors, we allow the self-reflection to proceed naturally. For correction failure, *RECAF* amplifies correction-relevant information in the model's KV cache, strengthening its influence on subsequent image generation. By separating reflection from correction while retaining the unified model as the central component for understanding and generation, *RECAF* preserves the core self-reflective paradigm while improving iterative image refinement without additional training. To summarize, our main contributions are:

- We identify two bottlenecks in self-reflective image generation: *reflection failure*, where the model fails to diagnose errors in its reflection, and *correction failure*, where the model identifies the error but fails to transfer the intended correction in the regenerated output.
- Building on this decomposition, we propose *RECAF*, a training-free framework that strengthens compositional image generation, while retaining the unified model as the core component for reflection and generation.
- We validate the effectiveness of *RECAF* on *GenEval*, *GenEval++* and *GenEval2* benchmarks. *RECAF* consistently outperforms strong baselines, improving the overall score by 5.0, 5.7, and 3.5 points on the three benchmarks, respectively.

2 Related Work

RECAF builds on unified multimodal models with self-reflective capabilities and is related to agentic image generation. We now introduce these two lines of work in detail.

Unified multimodal models (UMMs) and self-reflection in UMMs. UMMs (Ge et al., 2024; Xie et al., 2025a;b; Wu et al., 2025; Chen et al., 2025; Fang et al., 2025) aim to integrate visual understanding and generation within a shared architecture, allowing a single model to perform tasks such as visual question answering and image generation. Building upon UMMs, recent work has further explored how the understanding capability of UMMs can be leveraged to improve their own generations. For instance, Uni-CoT (Qin et al., 2026) equips BAGEL (Deng et al., 2025) with self-reflective refinement by constructing correction-oriented training data and further optimizing the model for iterative diagnosis and correction. AlphaGRPO (Huang et al., 2026a) instead enhances the underlying UMM via reinforcement learning with decomposed verifiable rewards together with an external reflection module. Related approaches have also explored reasoning-guided generation (Fang et al., 2025; Huang et al., 2026b) and refining the intermediate image generation via regeneration (Guo et al., 2026), however, such methods require collecting extra data and extra training. Building upon these works, we propose *RECAF*, a training-free framework that enhances self-reflective generation by introducing a fallback to validate the model's reflection before termination, while strengthening subsequent correction through KV cache amplification.

Agentic generation. Unlike self-reflective UMMs, agentic generation methods (Zhao et al., 2026; Jiang et al., 2026; Wang et al., 2024; Zeng et al., 2026) often decouple the reflection and generation processes and employ external evaluators or multiple cooperating agents in the generation loop. For example, MCCD (Li et al., 2025) employs multiple MLLM-based agents to parse complex scenes

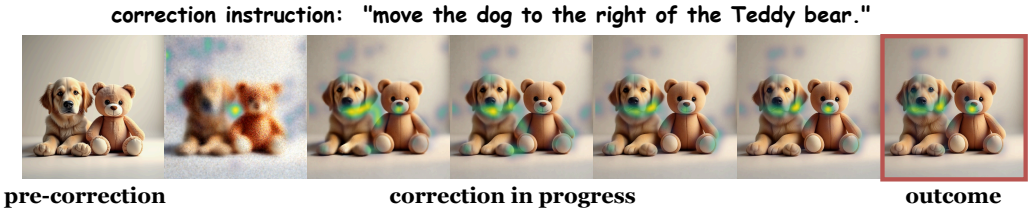

Figure 2: **The correction instruction attends to the target object but fails to realize the edit.** We visualize the correction process during denoising. Although the correction-token attention progressively concentrates on the target object, the final image still fails to implement the desired instruction.

and combines their outputs for more accurate object and relation generation. GenAgent (Jiang et al., 2026) uses a multimodal reasoning agent to perform reasoning and reflection, while using diffusion models as a tool for iterative refinement. Other approaches, such as PARM (Zhang et al., 2025), scale inference-time generation and select better generations. Unlike agentic pipelines that distribute reflection and generation across multiple specialized agents, *RECAF* retains the self-reflective UMM as the core engine for understanding and generation.

3 RECAF: Reflection Correction Amplification with Fallback

Before introducing *RECAF*, let us first describe the generation process of self-reflective unified multimodal models, using Uni-CoT as a representative example. We then analyze the bottlenecks that limit this process, which motivates the design of our method.

Preliminary on Self-Reflective UMM. As illustrated in Fig. 1 a, Uni-CoT follows an iterative reflection-correction process. Let $\mathcal{M}$ denote a unified multimodal model capable of both textual reasoning and image generation. During textual reasoning, $\mathcal{M}$ autoregressively generates tokens conditioned on the preceding context. For image generation, it synthesizes an image through an iterative denoising process. The generated image is then encoded by the vision encoder (e.g., ViT (Dosovitskiy et al., 2021)) into visual tokens and fed back into the model $\mathcal{M}$, enabling $\mathcal{M}$ to inspect its own output and produce a textual reflection. Based on this reflection, the model generates a correction instruction and revises the image accordingly. The revised image is then fed back into $\mathcal{M}$ for the next round of reflection, forming an iterative reflection–correction loop.

Formally, given an input prompt p, the model first generates a planning trace $t \sim \mathcal{M}_{\text{txt}}(\cdot \mid p)$, followed by an initial image $x^{(0)} \sim \mathcal{M}_{\text{img}}(\cdot \mid p, t)$. Starting from $x^{(0)}$, the model iteratively evaluates and refines the current image. At refinement round r, the model observes the current image $x^{(r-1)}$ and produces an evaluation $e^{(r)} \sim \mathcal{M}_{\text{txt}}\left(\cdot \mid p, t, x^{(r-1)}\right)$, which assesses whether the current generation satisfies the input prompt. If further refinement is needed, the model generates a correction suggestion $s^{(r)} \sim \mathcal{M}_{\text{txt}}\left(\cdot \mid p, t, x^{(r-1)}, e^{(r)}\right)$. The correction suggestion is then incorporated into the correction context $c^{(r)} = \left[p,\, t,\, \phi\left(x^{(r-1)}\right),\, s^{(r)}\right]$, where $\phi(x^{(r-1)})$ denotes the visual-token representation of the current image. Conditioned on this context, the model generates a revised image $x^{(r)} \sim \mathcal{M}_{\text{img}}\left(\cdot \mid c^{(r)}\right)$. The revised image $x^{(r)}$ is then fed into the next refinement round, and the process continues until the model predicts that no further modification is required.

Our goal is to improve the compositionality of image generation within this self-reflective generation process in a *training-free* manner, without collecting additional data or training.

3.1 Diagnosing Self-Reflection Process Failures

As shown in Fig. 1, self-reflective UMMs struggle with compositional image generation tasks. To understand the source of these failures, we analyze its two constituent stages, *reflection* and *correction*, separately, rather than treating refinement as a single holistic process.

Reflection failure. We first examine whether the model can reliably identify errors in its own generation. This is critical because once the model evaluates the current image as satisfactory, the refinement loop terminates, and the image is returned as the final output. An incorrect "pass", therefore, prevents subsequent correction. We refer to these unsuccessful cases as *reflection failures*.

To quantify the reflection failure, we evaluate Uni-CoT on GenEval (Ghosh et al., 2023) and GenEval++ (Ye et al., 2025). Among the final incorrect generations, Uni-CoT accepts a large fraction of incorrect initial reflections. Specifically, 78.3% of incorrect generations on GenEval and 74.3% on GenEval++ are judged as satisfactory (Fig. 3), thus bypassing the correction step. For a sanity check, we test whether the reflection decision is actually conditioned on the visual input by replacing the generated image with a mismatched image while keeping the remaining context unchanged. Under this sanity check, the pass rate drops to 0% on both benchmarks (Fig. 3). This change indicates that the model does attend to visual evidence during reflection. However, it often fails to detect the errors in its own generations, while readily rejecting images that exhibit strong visual inconsistency. **This suggests that the primary issue is not the absence of visual grounding, but insufficiently reliable reflection of the generated image, which motivates our design of reflection with fallback.**

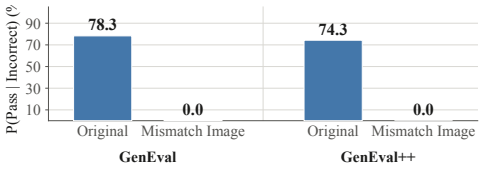

Figure 3: **Among the final incorrect generations, more than 70% were already accepted at the initial reflection stage by Uni-CoT, causing them to bypass subsequent refinement.** We report the pass rate among incorrect generations on two benchmarks under the original setting and a mismatched-image sanity check, where the generated image is replaced with a completely unrelated image before reflection. Under this sanity check, $P(\text{Pass} \mid \text{Incorrect})$ drops to 0% on both benchmarks. This indicates that the model does attend to visual evidence during reflection, while its high pass rate in the original setting suggests that it often fails to detect more subtle compositional errors in its own generations.

Correction failure. We next examine cases in which the model successfully identifies an error and proposes a correction instruction, but the regenerated image still fails to implement the requested change. We quantify this behavior using an MLLM-based evaluator (OpenAI, 2025) to assess whether the regenerated image follows the model's editing instruction (Fig. 4; the evaluation prompt and model details are provided in App. B.2.1). Despite being given an explicit correction instruction, the regenerated image fully follows it in only 32% of cases on GenEval and 55% on GenEval++. We refer to these unsuccessful cases as *correction failures*. To better understand their source, we further inspect the model's attention during regeneration. As shown in Fig. 2, when the model suggests "move the dog to the right of the Teddy bear," the correction-conditioned attention is already concentrated on the dog and its surrounding region, yet the regenerated image still preserves the original incorrect spatial relationship. **This observation suggests that correction failure does not necessarily arise from an inability to localize the relevant visual content. Rather, the corrective information may already be represented in the appropriate region but remain too weak to exert sufficient influence on the subsequent generation process.**

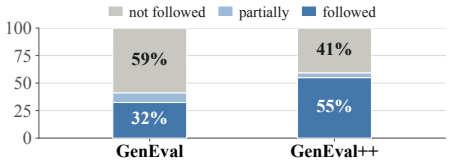

Figure 4: **Instructions are not faithfully realized during correction.** We use a MLLM-based evaluator to assess whether the regenerated image follows the model's editing instruction. A substantial fraction of edits are not followed, with only 32% and 55% of instructions realized on GenEval and GenEval++, respectively.

Together, reflection and correction failure reveal two bottlenecks in self-reflective generation: the model may fail to *recognize* an error during reflection, or, despite correctly identifying the error, fail to *realize* the intended correction into regenerated image. This decomposition motivates *RECAF* to target the two bottlenecks separately. We next introduce the corresponding components for reflection with fallback and correction amplification.

3.2 Reflection with Fallback

We first address the reflection failure. To strengthen the model's reflection capability, we introduce a targeted fallback step at termination. When the model proposes to stop, we use SAM 3 (Carion et al., 2026) to localize the relevant objects, while a BLIP-based VQA model (Li et al., 2022) checks

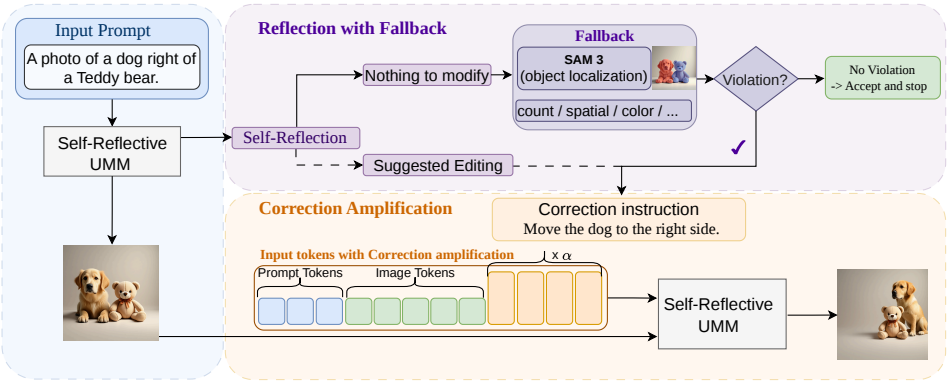

Figure 5: **Overview of *RECAF*. *RECAF* strengthens self-reflective UMMs in compositional image generation via reflection with fallback and correction amplification.** Details see Sec. 3.

their attributes. Together, these signals provide a final assessment of measurable compositional constraints before the image is accepted.

Given the current image x and prompt requirements $\mathcal{R}(p)$, the fallback first localizes the required object categories using an open-vocabulary segmentation model: $\mathcal{I}_c = \text{Segment}(x, c)$, where $\mathcal{I}_c$ contains the detected instances of category c. The same object-level representation is then reused to verify the compositional constraints $\mathcal{V}$:

$$\mathcal{V} = \{V_1, \ldots, V_j\},$$

where each verifier V_j evaluates the corresponding requirement and returns either a violation or no violation:

$$f_j = V_j(x, \mathcal{R}(p), \{\mathcal{I}_c\}). \tag{1}$$

We aggregate all measurable violations into $\mathcal{F}(x)$ and use a binary signal $D(x)$ to indicate whether any violation exists in the current image, where

$$\mathcal{F}(x) = \{f_j \mid f_j \neq \varnothing\}, \text{ and } D(x) = \mathbb{I}[|\mathcal{F}(x)| > 0]\,. \tag{2}$$

We apply the fallback only when the model attempts to stop the refinement process. If no measurable violation is found, we accept the model's decision and return the current image. Otherwise, the stopping decision is overridden and another correction round is triggered based on the violation information. The stopping criterion is defined as:

$$\text{Stop}(r) = A(s^{(r)}) \wedge \left[D(x^{(r)}) = 0 \vee k^{(r)} \geq M\right], \tag{3}$$

where $A(s^{(r)})$ indicates that the model proposes no further edit, $D(x^{(r)})$ indicates whether a measurable violation is detected, $k^{(r)}$ is the number of previous correction rounds, and M is the maximum number of correction rounds.

3.3 Correction Amplification

We next address correction failure, observing that it can occur even when the suggestion tokens already attend to the image region corresponding to the request edit. For example, as shown in Fig. 2, when the model suggests moving the dog to the right of the Teddy bear, the conditioned attention concentrates on the dog and its surroundings, while the regenerated image still preserves the incorrect spatial arrangement. This suggests that correction failure may not caused by missing spatial focus, rather, the corrective signal may be insufficiently strong to steer generation trajectory. Prior work has shown that intervening on key-value representations can effectively steer model behavior and control generation without modifying model parameters (Zhu et al., 2025; Belitsky et al., 2025). Motivated by this, we introduce the second training-free component, *Correction Amplification*, which strengthens the influence of the correction instruction during regeneration via a targeted key-value (KV) cache intervention. The conditioning context at reflection round r is:

$$c^{(r)} = \left[p,\, t,\, \phi(x^{(r-1)}),\, s^{(r)}\right], \tag{4}$$

where $s^{(r)}$ denotes the editing instruction and $\phi(x^{(r-1)})$ represents the visual tokens of the current image. We amplify only the correction-related tokens in $c^{(r)}$, while leaving all other context tokens unchanged. Let $\mathcal{S}^{(r)}$ denote the token positions of the correction instruction $s^{(r)}$, and let $\mathcal{L}$ denote the transformer layers where amplification is applied. For each $\ell \in \mathcal{L}$ and $j \in \mathcal{S}^{(r)}$, we rescale the cached Key and Value states as:

$$K_\ell[j] \leftarrow \alpha K_\ell[j], \qquad V_\ell[j] \leftarrow \alpha V_\ell[j], \qquad \alpha \geq 1. \tag{5}$$

All other cached states remain unchanged. For an image-token query q_i, attention is then computed using the modified keys and values:

$$a'_{ij} = \mathrm{softmax}_j \left(\frac{q_i^\top k'_j}{\sqrt{d}} \right), \qquad k'_j = \begin{cases} \alpha k_j, & j \in \mathcal{S}^{(r)}, \\ k_j, & \text{otherwise.} \end{cases} \tag{6}$$

The corresponding attention output is:

$$h'_i = \sum_j a'_{ij} v'_j, \quad with \quad v'_j = \begin{cases} \alpha v_j, & j \in \mathcal{S}^{(r)}, \\ v_j, & \text{otherwise.} \end{cases} \tag{7}$$

Scaling the Key states increases the relative attention assigned to the editing instruction, while scaling the Value states strengthens its contribution after attention aggregation. Importantly, the intervention does not introduce a new correction signal, it increases the influence of the existing one. The intervention is applied only to the conditional generation branch, while the unconditional branches remain unchanged.

Taken together, ***RECAF*** introduces a training-free reflection with fallback module and a correction amplification module to address the two bottlenecks of self-reflective UMMs in compositional image generation. We next evaluate ***RECAF***'s capability.

4 Experimental Evaluation

With ***RECAF*** at hand, we now evaluate its capability on challenging compositional text-to-image generation benchmarks that require models to satisfy multiple semantic and spatial constraints.

Benchmarks. We use GenEval (Ghosh et al., 2023), GenEval++ (Ye et al., 2025) and GenEval2 (Kamath et al., 2025) as our benchmark datasets. GenEval evaluates image generation across categories, including single-object generation, multiple objects, counting, color, spatial position, and attribute binding. GenEval++ further increases the difficulty by introducing more compositional combinations, such as color-count, color-position, position-count, and position-size. GenEval2 is evaluated with Soft-TIFA, which decomposes each prompt into atomic visual requirements and assigns a soft VQA score to each one. These benchmarks are particularly suitable for our setting because many of their failure cases require both accurate reflection over the generated image and effective correction of the identified error.

Baselines. We consider baselines including Generative-only models: SD3-Medium (Esser et al., 2024), FLUX.1-dev (Labs, 2024), Unified models: Show-o (Xie et al., 2025a), Janus-Pro (Chen et al., 2025), BAGEL (Deng et al., 2025) and Reflective models: Uni-CoT (Qin et al., 2026) (w/ and w/o CoT), IRG (Huang et al., 2026b) and GenAgent (base) (Jiang et al., 2026).

Implementation details. ***RECAF*** consists of two components targeting the two failure modes identified in Sec. 3: reflection with fallback and correction amplification. For reflection with fallback, we augment the model's reflection with object-level visual cues from SAM 3 and attribute information from BLIP. These cues do not replace the unified model's reflection process. For correction amplification, we strengthen correction-relevant information in the model's KV cache before the subsequent image generation step. No additional training data, fine-tuning, or parameter updates are introduced. We use the same generation configuration as the base Uni-CoT model unless otherwise specified. We follow the official benchmark evaluation protocol. Details see App. B

4.1 Quantitative Results

Results on GenEval. The results show that *RECAF* is competitive, achieving a best overall accuracy of 90.0%. As shown in Tab. 1, ***RECAF*** achieves the best overall accuracy of 90.0% on

Table 1: ***RECAF* achieves the best overall score on GenEval dataset.** Accuracy comparison on GenEval dataset (best performing models bolded). Higher values indicate better performance. The results of GenAgent and Uni-CoT are reproduced.

Model	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attri.	Overall↑
Gen.-Only Models							
SD3-Medium (Esser et al., 2024)	**99.0**	94.0	72.0	89.0	33.0	60.0	74.0
FLUX.1-dev (Labs, 2024)	**99.0**	81.0	79.0	74.0	20.0	47.0	67.0
Unified Models							
Show-o (Xie et al., 2025a)	98.0	80.0	66.0	84.0	31.0	50.0	68.0
Janus-Pro (Chen et al., 2025)	**99.0**	89.0	59.0	90.0	79.0	66.0	80.0
BAGEL (Deng et al., 2025)	**99.0**	92.0	78.0	87.0	53.0	64.0	79.0
Reflective Models							
IRG (Huang et al., 2026b)	98.0	94.0	**84.0**	86.0	74.0	73.0	85.0
GenAgent (Jiang et al., 2026)	**99.0**	**96.0**	79.0	90.0	61.0	70.0	83.0
Uni-CoT w/o CoT (Qin et al., 2026)	**99.0**	91.0	80.0	88.0	63.0	65.0	81.0
Uni-CoT (Qin et al., 2026)	**99.0**	92.0	83.0	88.0	64.0	67.0	82.0
RECAF (Ours)	**99.0**	93.0	**84.0**	**95.0**	**90.0**	**80.0**	**90.0**

Table 2: ***RECAF* outperforms the baselines and achieves the best overall performance on GenEval++ dataset.** Accuracy comparison on GenEval++ dataset (best performing models bolded). Higher values indicate better performance. The results of Show-o and Uni-CoT are reproduced.

Method	Col.	Cnt.	Col./Cnt.	Col./Pos.	Pos./Cnt.	Pos./Size	Multi-Cnt.	Overall↑
Gen.-Only Models								
SD3-Medium (Esser et al., 2024)	55.0	50.0	12.5	35.0	17.5	15.0	22.5	29.6
FLUX.1-dev (Labs, 2024)	35.0	62.5	15.0	27.5	20.0	37.5	22.5	31.4
Unified Models								
Show-o (Xie et al., 2025a)	25.0	30.0	7.5	17.5	5.0	22.5	20.0	18.2
Janus-Pro (Chen et al., 2025)	45.0	30.0	12.5	30.0	7.5	35.0	12.5	24.6
BAGEL (Deng et al., 2025)	32.5	60.0	25.0	32.5	25.0	47.5	37.5	37.1
Reflective Models								
GenAgent (Jiang et al., 2026)	30.0	60.0	32.5	30.0	32.5	30.0	35.0	35.7
Uni-CoT w/o CoT	42.5	60.0	52.5	50.0	47.5	40.0	60.0	50.4
Uni-CoT (Qin et al., 2026)	57.5	55.0	55.0	57.5	50.0	**50.0**	62.5	55.4
RECAF (Ours)	**60.0**	**65.0**	**65.0**	**65.0**	**57.5**	47.5	**67.5**	**61.1**

GenEval, outperforming the strongest baseline, IRG (85.0%), by 5 percentage points. The gains are particularly substantial on compositional categories, including colors (95.0%), position (90.0%), and color attribution (80.0%), corresponding to improvements of 5, 11, and 7 points over the strongest respective baselines. These results indicate that *RECAF* is particularly effective at correcting errors involving attributes and spatial relationships.

Results on GenEval++. *RECAF* achieves the highest overall score, with substantially stronger performance in most categories. On the more challenging GenEval++ benchmark (Tab. 2), *RECAF* achieves an overall accuracy of 61.1%, outperforming the strongest baseline, Uni-CoT (55.4%), by 5.7 points. *RECAF* also shows gains on joint-constraint categories, reaching 65.0% on color/count, 65.0% on color/position, 57.5% on position/count, and 67.5% on multi-object counting. These improvements demonstrate the effectiveness of *RECAF* on prompts requiring multiple compositional constraints to be satisfied simultaneously.

Taken together, results on GenEval and GenEval++ demonstrate that *RECAF* effectively strengthens the compositional image generation capability of the underlying self-reflective UMM by jointly addressing failures in both reflection and correction. Furthermore, we report additional results on GenEval2 (Kamath et al., 2025) in the ablation study to support the effectiveness of *RECAF*.

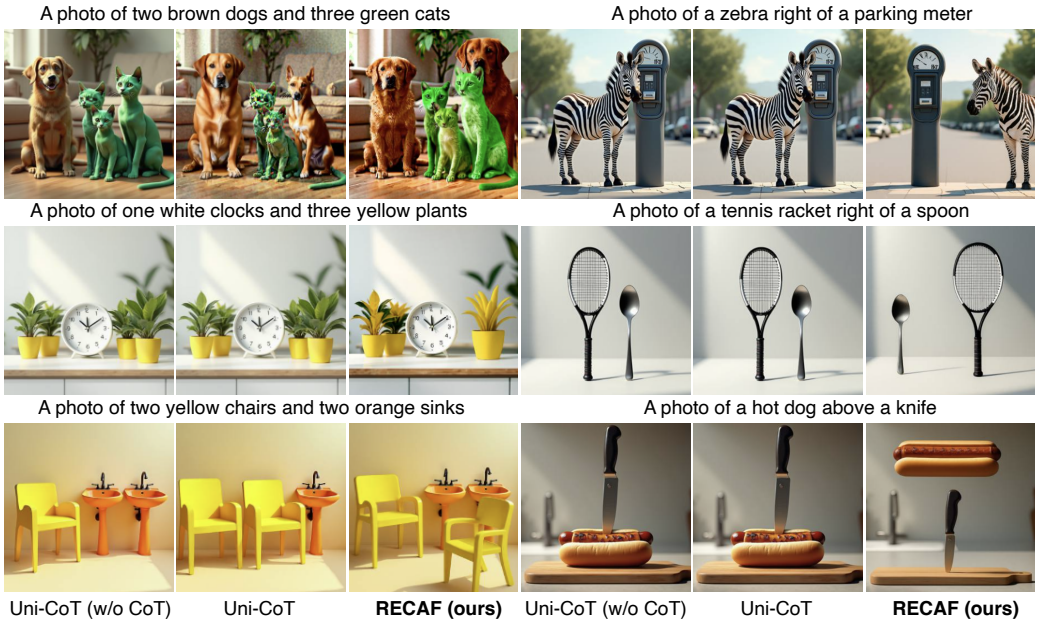

Figure 6: **Qualitative results.** We compare *RECAF* with Uni-CoT (w/ and w/o CoT). *RECAF* consistently produces images that better satisfy compositional prompts, while Uni-CoT and its variants often fail to correct errors in object counts or spatial relations, resulting in incorrect final generations. Details see Sec. 4.2.

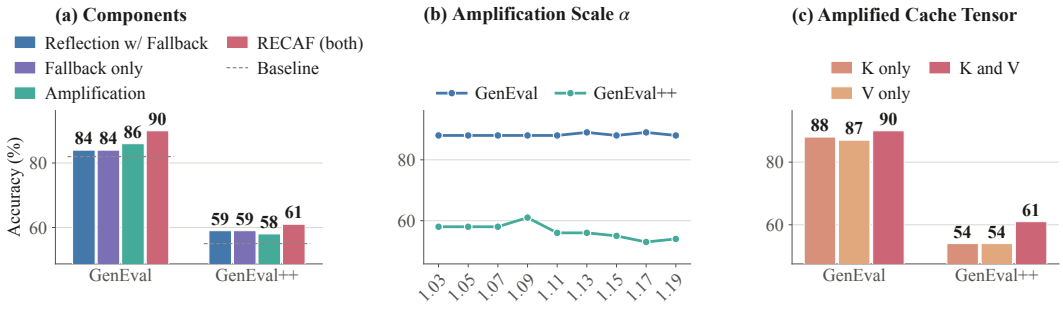

Figure 7: **Ablation studies of *RECAF*.** We analyze the contribution of reflection with fallback and correction amplification, the effect of amplification scale, and the choice of amplified cache tensor. Combining both components and amplifying both K and V yields the strongest performance.

4.2 Qualitative Results

To complement the quantitative evaluation, we next present qualitative examples that illustrate how *RECAF* improves compositional image generation. We qualitatively compare *RECAF* against baseline Uni-CoT and Uni-CoT w/o CoT in Fig. 6. Across the examples, all methods generally produce realistic images, but baseline methods frequently miss individual constraints, for instance, violating a requested spatial relation. *RECAF*, by contrast, more consistently satisfies the prompt. The benefit is especially visible for prompts containing multiple interacting constraints. For example, they may correctly generate the requested objects and colors but fail to preserve their relative positions, or satisfy the spatial arrangement while assigning an attribute to the wrong object. *RECAF* revisits these mismatches when reflection terminates and refines the image toward a composition that better meets all constraints simultaneously.

4.3 Ablation Studies

Finally, we conduct ablation studies to examine the contribution of each component in *RECAF* and the design choices behind correction amplification. We further evaluate the full framework on GenEval2 (Kamath et al., 2025) as an additional generalization study. Results are summarized in Fig. 7. We provide detailed results in App. C and App. D.

Effect of each component. Both reflection with fallback and correction amplification contribute to performance, with their combination achieving the best results on GenEval and GenEval++. We first ablate the two key components of *RECAF*: *reflection with fallback* and *correction amplification*. As shown in Fig. 7 a, using reflection with fallback alone achieves 84% on GenEval and 59% on GenEval++, amplification alone improves the performance to 86% and 58% compared to baseline, respectively. We further compare against an *Fallback only* variant, which replaces every reflection round with fallback. Fallback only achieves 84% on GenEval and 59% on GenEval++, providing no additional gain over fallback alone. Combining both components yields the best performance, reaching 90% on GenEval and 61% on GenEval++. These results indicate that the two components are complementary: reflection with fallback helps prevent erroneous termination, whereas correction amplification improves the model's ability to realize identified corrections.

Effect of the amplification scale. *RECAF* is relatively robust to the amplification scale on GenEval, with moderate amplification performing best on GenEval++. We next study the sensitivity of *RECAF* to the amplification scale. As shown in Fig. 7 b, performance on GenEval remains relatively stable over a broad range of amplification strengths, whereas GenEval++ is more sensitive to this hyperparameter. In particular, an amplification scale at 1.09 gives the strongest performance on the more challenging GenEval++ benchmark, larger scales degrade performance. This suggests that effective correction requires strengthening the relevant internal signal without excessively perturbing the original generation dynamics.

Which cache tensor should be amplified? Jointly amplifying both key and value caches outperforms amplifying either alone. Finally, we investigate whether correction amplification should be applied to the key cache, the value cache, or both. As shown in Fig. 7 c, amplifying only the key cache achieves 88.0% on GenEval and 53.9% on GenEval++, while amplifying only the value cache obtains 87.0% and 54.3%, respectively. Jointly amplifying both key and value caches performs best, reaching 90.0% and 61.1%. This result suggests that effective correction benefits from jointly strengthening both where the model attends, and the information propagated through attention, supporting our default design of amplifying both K and V.

Generalization to GenEval2. We additionally evaluate the full *RECAF* framework on GenEval2 (Kamath et al., 2025). As shown in Tab. 3, *RECAF* improves over the underlying self-reflective baseline by 3.5 points, demonstrating that its effectiveness extends beyond GenEval and GenEval++ to a more challenging compositional benchmark.

Table 3: ***RECAF* achieves the best Soft TIFA$_{\mathrm{GM}}$ score among unified models on GenEval2.** Higher is better.

Model	Soft TIFA$_{\mathrm{GM}}$ ↑
BAGEL (Deng et al., 2025)	23.1
Uni-CoT w/o CoT	27.5
Uni-CoT (Qin et al., 2026)	36.6
RECAF (Ours)	**40.1**

5 Conclusion

We revisit compositional image generation in self-reflective UMMs and show that refinement failures arise from two distinct bottlenecks: *reflection failure*, where the model fails to recognize errors in its own generation, and *correction failure*, where the model identifies the error but fails to realize the intended correction during regeneration. Based on this decomposition, we introduce *RECAF*, a training-free framework that targets the two bottlenecks separately through reflection with fallback and correction amplification. *RECAF* strengthens the compositional image generation capability of a self-reflective UMM without collecting additional data or updating model parameters. Experiments on GenEval, GenEval++ and GenEval2 demonstrate consistent improvements on overall scores over the underlying backbone, with particularly strong gains on compositional and spatial generation tasks. Our results suggest that explicitly diagnosing and addressing the internal failure modes of self-reflective generation provides a promising alternative to further training or fully externalized agentic pipelines. Due to space limits, we put the limitations and future work in App. A.

AI USE STATEMENT

In this work, we used generative AI tools for supporting data analysis (we use GPT for the evaluation of correction failure, the evaluation of GenEval++ benchmark are conducted via GPT), editing our code, edit a research paper to improve readability, identify relevant literature, propose a title or keywords for a research paper. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for creating and editing code, evaluating the proposed title by authors for the paper, analyse existing literature, edit and polish the text to improve readability. We have reviewed all AI-assisted work. The LLM-generated code was verified and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our work focuses on improving compositional image generation in self-reflective UMMs through a training-free method. We aim to improve the model's ability to produce images that more closely follow user instructions. However, as with other image-generation systems, this capability may be misused to create misleading, harmful, or deceptive visual content. Our method does not introduce new modalities of misuse beyond those already associated with the underlying generative model, but stronger instruction following and iterative refinement may increase the quality of generated outputs. We strongly discourage any efforts to adopt our work to create inappropriate or unsafe content.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide detailed descriptions of our framework in Sec. 3, implementation details, the evaluation protocols in Sec. 4, and in App. B.

REFERENCES

Max Belitsky, Dawid J Kopiczko, Michael Dorkenwald, M Jehanzeb Mirza, James R Glass, Cees GM Snoek, and Yuki M Asano. Kv cache steering for controlling frozen llms. *arXiv preprint arXiv:2507.08799*, 2025.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International conference on learning representations (ICLR)*, 2026.

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.

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.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning (ICML)*, 2024.

Rongyao Fang, Chengqi Duan, Kun Wang, Linjiang Huang, Hao Li, Hao Tian, Shilin Yan, Weihao Yu, Xingyu Zeng, Jifeng Dai, Xihui Liu, and Hongsheng Li. Got: Unleashing reasoning capability of MLLM for visual generation and editing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025.

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

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Jiayi Guo, Linqing Wang, Jiangshan Wang, Yang Yue, Zeyu Liu, Zhiyuan Zhao, Qinglin Lu, Gao Huang, and Chunyu Wang. Refinement via regeneration: Enlarging modification space boosts image refinement in unified multimodal models. In *2nd Workshop on MUCG: Multimodal Large Language Models for Unified Comprehension and Generation*, 2026.

Runhui Huang, Jie Wu, Rui Yang, Zhe Liu, and Hengshuang Zhao. AlphaGRPO: Unlocking self-reflective multimodal generation in unified multimodal models via decompositional verifiable reward. In *Forty-third International Conference on Machine Learning (ICML)*, 2026a.

Wenxuan Huang, Shuang Chen, Zheyong Xie, Shaosheng Cao, Shixiang Tang, Yufan Shen, Qingyu Yin, Wenbo Hu, Xiaoman Wang, Yuntian Tang, et al. Interleaving reasoning for better text-to-image generation. In *International Conference on Learning Representations (ICLR)*, 2026b.

Kaixun Jiang, Yuzheng Wang, Junjie Zhou, Pandeng Li, Zhihang Liu, Chen-Wei Xie, Zhaoyu Chen, Yun Zheng, and Wenqiang Zhang. Genagent: Scaling text-to-image generation via agentic multimodal reasoning. In *European Conference on Computer Vision (ECCV)*, 2026.

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

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014.

Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2024.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning (ICML)*, 2022.

Mingcheng Li, Xiaolu Hou, Ziyang Liu, Dingkang Yang, Ziyun Qian, Jiawei Chen, Jinjie Wei, Yue Jiang, Qingyao Xu, and Lihua Zhang. Mccd: Multi-agent collaboration-based compositional diffusion for complex text-to-image generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Qingyang Liu, Bingjie Gao, Canmiao Fu, Zhipeng Huang, Chen Li, Feng Wang, Shuochen Chang, Shaobo Wang, Yali Wang, Keming Ye, Jiangtong Li, and Li Niu. Breaking dual bottlenecks: Evolving unified multimodal models into self-adaptive interleaved visual reasoners. In *Forty-third International Conference on Machine Learning (ICML)*, 2026.

Thomas O Nelson. Metamemory: A theoretical framework and new findings. In *Psychology of learning and motivation*. Elsevier, 1990.

OpenAI. Gpt-4-1. `https://openai.com/index/gpt-4-1/`, 2025.

Luozheng Qin, GONG JIA, Yuqing Sun, Tianjiao Li, Haoyu Pan, Mengping Yang, Xiaomeng Yang, Chao Qu, Zhiyu Tan, and Hao Li. Uni-cot: Towards unified chain-of-thought reasoning across text and vision. In *The Fourteenth International Conference on Learning Representations (ICLR)*, 2026.

Marion Rouault, Andrew McWilliams, Micah G Allen, and Stephen M Fleming. Human metacognition across domains: insights from individual differences and neuroimaging. *Personality neuroscience*, 2018.

Zhenyu Wang, Aoxue Li, Zhenguo Li, and Xihui Liu. Genartist: Multimodal llm as an agent for unified image generation and editing. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

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

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. Show-o2: Improved native unified multimodal models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025b.

Junyan Ye, Dongzhi Jiang, Zihao Wang, Leqi Zhu, Zhenghao Hu, Zilong Huang, Jun He, Zhiyuan Yan, Jinghua Yu, Hongsheng Li, et al. Echo-4o: Harnessing the power of gpt-4o synthetic images for improved image generation. *arXiv preprint arXiv:2508.09987*, 2025.

Ziyun Zeng, Hang Hua, and Jiebo Luo. Mira: Multimodal iterative reasoning agent for image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, June 2026.

Renrui Zhang, Chengzhuo Tong, Zhizheng Zhao, Ziyu Guo, Haoquan Zhang, Manyuan Zhang, Jiaming Liu, Peng Gao, and Hongsheng Li. Let's verify and reinforce image generation step by step. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Jiahao Zhao, Xiaomin Yu, Zhongxiang Sun, Fengwei Teng, Chengwei Qin, Xiaobin Hu, Jun Xu, and Shuicheng Yan. Toolartist: Tool-using unified multimodal models for agentic image generation. *arXiv preprint arXiv:2608.04436*, 2026.

Tianrui Zhu, Shiyi Zhang, Jiawei Shao, and Yansong Tang. Kv-edit: Training-free image editing for precise background preservation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.



Figure 8: Illustration of *RECAF*'s limitation in counting editing. See App. A for details.

A Limitations and Future Work

Although *RECAF* achieves strong improvements on GenEval, GenEval++ and GenEval2, it still exhibits limitations in counting editing, particularly for exact object addition and removal. For example, in Fig. 8, the model is asked to remove exactly one apple (left) or add one more TV (right), but instead removes two apples and adds two TVs, respectively. We attribute this partly to the gap between language-level correction instructions and the precise visual region to be modified: the current correction signal specifies *what* should change, but not always *which instance* should be edited. A promising direction for future work is to incorporate more explicit spatial grounding, such as bounding boxes or instance-level localization, into the correction stage to enable more precise control over object addition or removal.

B Implementation Details

We provide the implementation details of our framework, including the base model, the verifier configuration, and the evaluation settings.

B.1 Base model and generation

Our frameworks use Uni-CoT-7B-MoT (a BAGEL-7B mixture-of-transformers backbone with the Uni-CoT self-reflection fine-tuning) as base model, loaded from the released checkpoint in official repository. The model is run with the system prompt written with Uni-CoT, which instructs the model to (1) plan in **<think>** tags before generating, (2) self-evaluate in **<sr>** tags, and (3) write a brief modification suggestion in **<sugg>** tags and regenerate. We do not alter any weights.

The initial image is sampled from the flow-matching decoder at 1024 resolution with 50 timesteps, timestep shift 3.0, text guidance scale $w_{\text{txt}} = 4.0$ applied on the interval $[0.4, 1.0]$ of the sampling trajectory, and global guidance renormalisation with $\text{renorm}_{\min} = 0$. The planning trace is decoded greedily (**do_sample=False**) with a maximum length of 2048 tokens.

Each regeneration conditions on the system prompt, the prompt, the planning trace, the VAE (Kingma & Welling, 2014) and ViT tokens of the current image, and the (possibly rewritten) **<sugg>** instruction. Sampling uses 50 timesteps, timestep shift 3.0, serial text-then-image classifier-free guidance with $w_{\text{txt}} = 4.0$ and $w_{\text{img}} = 1.5$ over the full interval $[0, 1]$, and per-text-channel renormalisation with $\text{renorm}_{\min} = 0$. The output size is the input size, clamped to the $[512, 1024]$ range on the shorter and longer side respectively. All text in the reflection rounds (the **<sr>** evaluation, the **<sugg>**, and the rewrite) is decoded greedily with a maximum context of 8192 tokens.

After the initial generation, the model reflects on its own image. The loop stops when the **<sugg>** text contains one of the literal approval strings "Everything is good", "Everything is well", "No editing" or "no editing". Otherwise, the suggestion is passed to a fixed rewrite prompt ("Rewrite the text within the **<sugg>...</sugg>** tags to make it more specific and detailed as an editing instruction. Ensure that only the revised instruction is enclosed within **<sugg>...</sugg>** tags.")

and the image is regenerated. At most 5 correction rounds are run per prompt, after which the last image is taken as the final result.

For correction amplification, we use $\alpha = 1.09$ for GenEval++, $\alpha = 1.03$ for GenEval2 and best performing α for GenEval.

B.2 Fallback Component

The Fallback receives the current image and a requirement set (which can be derived from the metadata of the benchmark), runs one open-vocabulary segmentation call per distinct class, and evaluates checks on the resulting instance masks. Tab. 4 lists all constants.

Table 4: Hyperparameters for the Fallback components.

Component	Setting	Value
SAM 3	detection score threshold	0.6
	mask threshold	0.6
BLIP-VQA	question	"is the {class} {color}?", one per vocabulary color
	decision	margin $\Delta = \max_\kappa P(\text{yes} \mid \kappa) - P(\text{yes} \mid \kappa_{\text{req}})$
	threshold τ	0.25

B.2.1 Experiment Details in evaluation of correction failure

GPT-based evaluation of correction fidelity. To evaluate whether each regeneration faithfully executes its corresponding correction instruction, we employ GPT-4.1 (OpenAI, 2025) as an independent multimodal judge. Importantly, this evaluation is designed to measure *instruction-following* of the correction step in the self-reflection process, rather than the correctness of the correction instruction itself. For each corrected refinement round, the evaluator is provided with three inputs: (i) the image before correction, (ii) the regenerated image after correction, and (iii) the correction instruction used to produce the regeneration. The instruction is extracted from the **<sugg>** span after removing formatting tags.

We make one GPT-4.1 call for each corrected round, with max tokens of 300. The user message contains the correction instruction in the form **Editing instruction: instruction**, followed by the two images explicitly identified as **First image = ORIGINAL** and **Second image = EDITED**. The evaluator is instructed to compare the edited image against the original and determine whether the requested modification is actually realized. This relative comparison is important because a target property that was already satisfied in the original image should not be counted as a successful edit unless the requested change is newly introduced by the regeneration.

The evaluator returns a structured JSON object with four fields: **follows instruction**, **degree**, **change detected**, and **reason**. The binary field **follows instruction** is set to 1 only when the edited image fully satisfies the requested correction. The categorical field **degree** takes one of three values **fully**, **partially**, or **not** which we subsequently map to **followed**, **partially**, and **not followed**, respectively. The field **change detected** indicates whether any instruction-relevant visual change occurred between the original and edited images. Finally, **reason** provides a concise textual justification for the decision.

The evaluator is instructed to judge only the visual attributes explicitly specified by the correction instruction, including object position, count, color, size, and presence or absence. Unrelated variations in style, illumination, background, or fine-grained appearance are ignored unless they are directly relevant to the requested edit. For spatial edits, a modification is considered successful only if the requested spatial relation is actually established in the edited image; a small displacement that does not realize the intended relation is insufficient. Similarly, for addition or removal instructions, the object count must change accordingly. We use a strict definition of partial success: **partially** is assigned only when there is a clearly observable change in the intended direction that nevertheless fails to fully satisfy the instruction.

The exact system prompt used for GPT-4.1 is provided below for reproducibility:

Role. You are an expert evaluator of image-editing results.

Input. You will be given: (1) an **ORIGINAL** image, (2) an **EDITED** image, and (3) an editing instruction describing the intended modification.

Evaluation objective. Determine whether the EDITED image successfully implements the given instruction *relative to* the ORIGINAL image. A requested change is considered successful only if it is visibly realized in the EDITED image and was not already satisfied in the ORIGINAL image in the same manner.

Evaluation criteria. Evaluate only the visual properties explicitly specified by the instruction, including object position, count, color, size, and presence or absence. Ignore unrelated changes in style, lighting, background, or fine-grained appearance unless they are directly relevant to the instruction.

For spatial-editing instructions, the target object must actually satisfy the requested spatial relation in the EDITED image. A minor displacement that does not establish the intended relation should not be considered successful.

For object addition or removal instructions, the corresponding object count must change consistently with the instruction.

Assign **`partially`** only when the EDITED image exhibits a clear modification in the intended direction but does not fully satisfy the requested edit.

Output. Return only a valid JSON object following the predefined output schema.

To ensure consistent parsing and exclude malformed responses, we validate every returned object before accepting it. Specifically, **`follows instruction`** and **`change detected`** must belong to $\{0, 1\}$, and **`degree`** must be one of **`fully`**, **`partially`**, or **`not`**. Responses that do not satisfy this schema are rejected rather than included in the evaluation.

C Additional experimental results

To further evaluate the generalization of RECAF beyond the benchmarks used in the main paper, we conduct more additional experiments on GenEval2 (Kamath et al., 2025). GenEval2 is evaluated with Soft-TIFA, which decomposes each prompt into atomic visual requirements and assigns a soft VQA score to each one. Soft-TIFA$_{\mathrm{AM}}$ averages these scores arithmetically to measure atom-level correctness, while Soft-TIFA$_{\mathrm{GM}}$ uses the geometric mean, which gives a lower score when any individual requirement is poorly satisfied and therefore better reflects prompt-level correctness.

Table 5: **RECAF outperforms the baselines and achieves the best overall performance on GenEval2 dataset.** Accuracy comparison on GenEval2 dataset (best performing models bolded). Higher values indicate better performance. The results of Uni-CoT are reproduced.

Model	Soft TIFA$_{\mathrm{GM}}$↑	Soft TIFA$_{\mathrm{AM}}$↑
Gen.-Only Models		
SD3-Medium (Esser et al., 2024)	21.3	66.3
FLUX.1-dev (Labs, 2024)	21.1	67.1
Unified Models		
BAGEL (Deng et al., 2025)	23.1	70.9
Uni-CoT w/o CoT (Qin et al., 2026)	27.5	73.5
Uni-CoT (Qin et al., 2026)	36.6	75.9
`RECAF (Ours)`	**40.1**	**77.8**

As shown in Tab. 5, RECAF achieves the best performance on both Soft-TIFA$_{\mathrm{GM}}$ and Soft-TIFA$_{\mathrm{AM}}$. Compared with the Uni-CoT baseline, RECAF improves Soft-TIFA$_{\mathrm{GM}}$ from 36.6 to 40.1 and Soft-TIFA$_{\mathrm{AM}}$ from 75.96 to 77.8. This suggests that RECAF not only improves individual requirement satisfaction but also reduces cases in which one or more compositional constraints remain unaddressed.

Table 6: **Datailed results on GenEval.** Higher values indicate better performance.

Model	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attri.	Overall↑
Contribution of Each Component							
Reflection with Fallback	0.99	0.92	0.8	0.91	0.76	0.68	0.84
Amplification	0.99	0.92	0.8	0.93	0.76	0.77	0.86
RECAF	0.99	0.93	0.84	0.95	0.9	0.8	0.9
Effect of Amplification Scale							
α = 1.03	0.99	0.93	0.84	0.94	0.82	0.79	0.88
α = 1.05	0.99	0.92	0.8	0.94	0.85	0.8	0.88
α = 1.07	0.99	0.91	0.83	0.94	0.84	0.80	0.88
α = 1.09	0.99	0.91	0.83	0.94	0.84	0.8	0.88
α = 1.11	0.99	0.92	0.83	0.94	0.86	0.76	0.88
α = 1.13	0.99	0.92	0.8	0.94	0.89	0.78	0.89
α = 1.15	0.99	0.91	0.8	0.95	0.9	0.76	0.88
α = 1.17	0.99	0.92	0.83	0.95	0.89	0.78	0.89
α = 1.19	0.99	0.92	0.79	0.95	0.89	0.76	0.88
Effect of K / V Tensor							
K Only	0.99	0.93	0.81	0.94	0.81	0.79	0.88
V Only	0.99	0.93	0.82	0.94	0.76	0.77	0.87
KV	0.99	0.93	0.84	0.95	0.9	0.8	0.9

Table 7: **Datailed results on GenEval++.** Higher values indicate better performance.

Method	Col.	Cnt.	Col./Cnt.	Col./Pos.	Pos./Cnt.	Pos./Size	Multi-Cnt.	Overall↑
Contribution of Each Component								
Reflection with Fallback	55	57.5	62.5	57.5	67.5	47.5	70	59.64
Amplification	57.5	52.5	60	55	62.5	52.5	65	57.86
RECAF	60	65	65	65	57.5	47.5	67.5	61.07
Effect of Amplification Scale								
α = 1.03	52.5	60	65	67.5	57.5	45	60	58.21
α = 1.05	52.5	65	65	62.5	55	45	60	57.86
α = 1.07	62.5	62.5	62.5	62.5	55	47.5	55	58.21
α = 1.09	60	65	65	65	57.5	47.5	67.5	61.07
α = 1.11	52.5	60	57.5	65	60	45	55	56.43
α = 1.13	52.5	60	60	60	67.5	47.5	52.5	57.14
α = 1.15	57.5	50	55	57.5	60	50	57.5	55.36
α = 1.17	45	52.5	52.5	60	55	52.5	50	52.5
α = 1.19	47.5	50	57.5	62.5	57.5	50	52.5	53.93
Effect of K / V Tensor								
K Only	62.5	52.5	60	57.5	40	47.5	57.5	53.93
V Only	65	45	60	60	42.5	50	57.5	54.29
KV	60	65	65	65	57.5	47.5	67.5	61.07

D Detailed Ablation Results

We provide the detailed results of the ablation studies summarized in the main paper for GenEval in Tab. 6 and GenEval++ in Tab. 7.