RICO: Improving Accuracy and Completeness in Image Recaptioning via Visual Reconstruction

Yuchi Wang[1], **Yishuo Cai**[1], **Shuhuai Ren**[1], **Sihan Yang**[2], **Linli Yao**[1], **Yuanxin Liu**[1], **Yuanxing Zhang**[3], **Pengfei Wan**[3], **Xu Sun**[1]

[1] National Key Laboratory for Multimedia Information Processing, Peking University
[2]Xi'an JiaoTong University [3]Kuaishou Technology
wangyuchi369@gmail.com gaohuan05@kuaishou.com xusun@pku.edu.cn

Figure 1: Analysis of image captions generated by Qwen2-VL and its recaptioned variants. Despite the advanced capabilities of Qwen2-VL, the generated captions still contain incorrect or ambiguous information—for example, misidentifying the number of buses—a mistake that remains uncorrected even by GPT-4o. Furthermore, both GPT-4o and human-generated recaptions often overlook fine-grained details, such as attributes and spatial relationships, which are accurately captured by our model. By reconstructing images from captions, it becomes evident that our model better preserves such details, resulting in reconstructions that more closely resemble the original image.

Abstract

Image recaptioning is widely used to generate training datasets with enhanced quality for various multimodal tasks. Existing recaptioning methods typically rely on powerful multimodal large language models (MLLMs) to enhance textual descriptions, but often suffer from inaccuracies due to hallucinations and incompleteness caused by missing fine-grained details. To address these limitations, we propose RICO, a novel framework that refines captions through visual reconstruction. Specifically, we leverage a text-to-image model to reconstruct a caption into a reference image, and prompt an MLLM to identify discrepancies between the original and reconstructed images to refine the caption. This process is performed iteratively, further progressively promoting the generation of more faithful and comprehensive descriptions. To mitigate the additional computational cost induced by the iterative process, we introduce RICO-Flash, which learns to generate captions like RICO using DPO. Extensive experiments demonstrate that our approach significantly improves caption accuracy and completeness, outperforms most baselines by approximately 10% on both CapsBench and CompreCap. Code released at https://github.com/wangyuchi369/RICO.

1 Introduction

The availability of hundreds of millions of image-text pairs collected from the internet has played a pivotal role in advancing modern multimodal learning (Chen et al., 2023; Liu et al., 2023; Bai et al., 2023b). However, the alt text associated with web

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 21785–21804
November 4-9, 2025

images is frequently of low quality, offering uninformative descriptions or even text unrelated to the image content. Consequently, recaptioning methods have been widely employed to generate enhanced captions for downstream multimodal tasks, such as training multimodal large language models (MLLMs) (Chen et al., 2023), text-to-image models (Betker et al.; Yang et al., 2024), and CLIP-like models (Fan et al., 2023; Lai et al., 2024).

Typically, recaptioning methods primarily depend on powerful MLLMs (Lai et al., 2024; Chen et al., 2023). While MLLMs significantly enhance captions over the alt-text by leveraging their strong perceptual capabilities, the generated descriptions still face two key challenges: (1) **Inaccuracy**, where some descriptions are incorrect, often exacerbated by the notorious hallucination problem of MLLMs (Bai et al., 2025); and (2) **Incompleteness**, where important details are frequently omitted. These issues cannot be fully resolved even with the integration of additional models or human editing. For example, as illustrated in Fig. 1, the caption generated by Qwen2-VL (Wang et al., 2024b) contains ambiguous or incorrect information that cannot be fully corrected even with GPT-4o (OpenAI et al., 2024). Moreover, several visual details remain undetected by either GPT-4o or human annotators, whereas our method successfully captures them. This appears to stem from the natural tendency of both humans and models to focus on salient objects in an image, often neglecting attributes and subtle details. We further validate this observation through experiments in § 4.2.

From a semantic space perspective, the challenges above suggest that the semantic space constructed through recaptioning is often biased and lossy compared to that of the original image. As illustrated in Fig. 2, conventional captioners typically follow a one-way mapping from image to text, without enforcing explicit semantic alignment between the two modalities, resulting in the omission of critical semantic elements in the generated captions. We argue that an ideal cross-modal semantic alignment should involve a bi-directional mapping: when text is generated from an image, the reconstructed image from that text should remain consistent with the original. In cases of misalignment, the discrepancy between the original and reconstructed images can be used to adjust the semantic space of the caption. Based on this intuition, we propose RICO (**R**econstruction-guided **I**mage **C**aption **O**ptimization), a novel recaptioning framework. As shown in Fig. 2, our method incorporates a visual reconstruction step that makes semantic discrepancies more observable in the visual domain compared to simply contrasting image and text, thereby facilitating the recovery of omitted details and producing descriptions that are both more semantically aligned and comprehensive.

Technically, we use powerful text-to-image models to reconstruct each caption into a reference image. Next, we input the original image, the generated reference image, and the candidate caption into a reviser—an MLLM and prompt it to refine the caption based on the discrepancies between the original and reference images. Through experiments, we find that a single-step refinement is insufficient, so we design the refinement process to iterate multiple times to progressively improve the caption. Given the significant time and computational resources required for iterative refinement, we propose an end-to-end variant as a more efficient alternative to RICO. This model is constructed by learning the naturally induced preference relationships during the iterative refinement process using Direct Preference Optimization (DPO) (Rafailov et al., 2024). Specifically, we employ RICO to generate a batch of training data, which is then used to fine-tune a base model via DPO, resulting in the compact RICO-Flash model.

Through experiments, we demonstrate that our pipeline effectively constructs well-aligned image-text information spaces. From the captioning perspective, we evaluate both the RICO framework and the compact RICO-Flash model on some benchmarks. Results show that RICO significantly enhances caption quality in terms of both accuracy and comprehensiveness. For instance, it consistently achieves improvements of over 10 points on CapsBench (Liu et al., 2024a). Moreover, RICO-Flash outperforms all recaptioning baselines. From the reverse perspective of text-to-image generation, we find that models trained on captions refined by RICO-Flash exhibit a stronger understanding of fine-grained prompts, particularly with regard to attributes and relationships. Further analysis also reveals that our method demonstrates strong robustness and generalization across diverse settings.

2 Related Works

2.1 Multimodal Large Language Models

Inspired by the success of large language models (LLMs) (Sun et al., 2025; Ouyang et al., 2022;

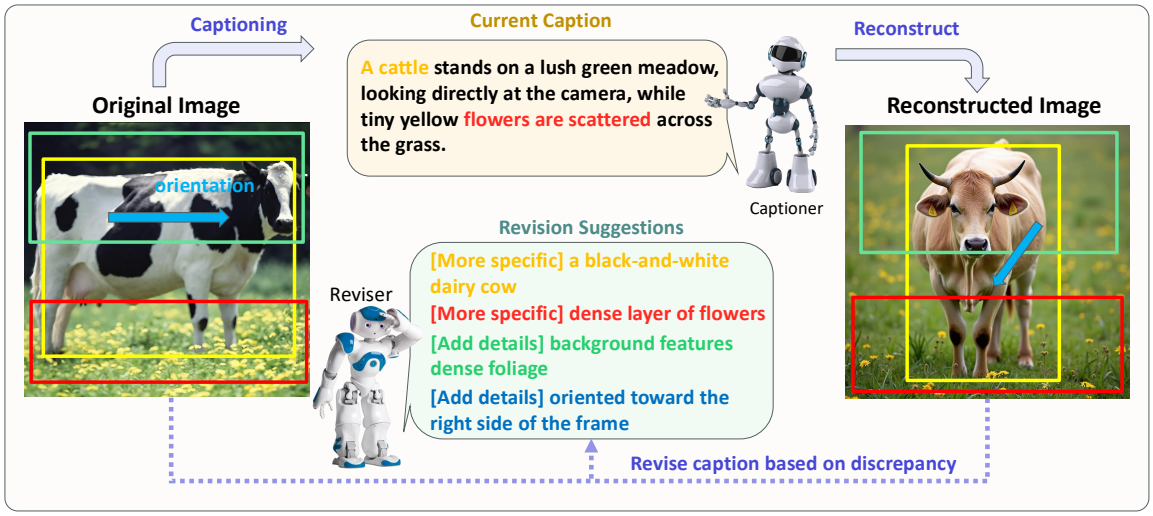

Figure 2: Illustration of the motivation for introducing the visual reconstruction mechanism. Conventional recaptioning methods typically map images directly to text without explicitly aligning the semantic spaces of the two modalities, often leading to information loss in the generated captions. In contrast, our approach incorporates visual reconstruction to make this loss more observable. By identifying discrepancies between the original and reconstructed images through the reviser, we refine the caption to produce a more semantically aligned and comprehensive description.

DeepSeek-AI et al., 2025; Bai et al., 2023a) in natural language processing, several works have extended them to multimodal settings by incorporating visual encoders (OpenAI, 2023; Liu et al., 2023; Team et al., 2024a; Bai et al., 2023b; Ren et al., 2024), contributing to multimodal large language models (MLLMs). Flamingo (Alayrac et al., 2022) is an early effort that inserts gated attention layers into a pretrained language model to enable vision-language understanding. Subsequent works explore various strategies for connecting vision encoders to language models. For example, BLIP-2 (Li et al., 2023a) introduces the Q-Former to bridge the modalities, LLaVA (Liu et al., 2023) employs a simple MLP projection layer, and Gemini (Team et al., 2024a) feeds image and text tokens jointly into a unified Transformer. In addition to architectural design, recent research has also focused on improving the quality of pretraining and fine-tuning data (Bai et al., 2023b; Wang et al., 2024c; Zhu et al., 2023). While modern MLLMs demonstrate impressive visual perception capabilities, they still suffer from hallucination issues (Bai et al., 2025)—occasionally generating inaccurate or fabricated content—which undermines the faithfulness of the generated captions.

2.2 Image Recaptioning

Describing an image using text has been a fundamental task in multimodal learning (Li et al., 2022; Ghandi et al., 2023; Wang et al., 2024d; Yao et al., 2023). Among these efforts, image recaptioning aims to generate enhanced captions for original, noisy alt text associated with image-text pairs. It has become increasingly important for producing high-quality synthetic data to support various downstream applications. This trend was popularized by DALL-E 3 (Betker et al.), which introduced the idea of replacing low-quality or overly simplistic captions with synthetic alternatives. Since then, numerous approaches have leveraged image recaptioning to improve multimodal large language models (MLLMs) (Chen et al., 2023), text-to-image generation models (Betker et al.), and CLIP-style vision-language models (Lai et al., 2024; Fan et al., 2023). Among these efforts, LaCLIP (Fan et al., 2023) utilizes LLMs to rewrite alt-text, while VeCLIP (Lai et al., 2024) incorporates additional visual information. CapsFusion (Yu et al., 2024) trains a LLaMA-based model to fuse alt-text and synthetic captions, and ShareGPT4V (Chen et al., 2023) directly generates new captions using GPT-4V (OpenAI, 2023). More sophisticated approaches include Altogether (Xu et al., 2024), which employs iterative human annotation, and Ye et al. propose automated fine-grained feedback mechanisms to improve captioning capabilities. Additionally, methods based on local perception have also been explored (Peng et al., 2025; Sun et al., 2025). However, despite their ad-

vancements, these methods fundamentally follow a paradigm of directly generating captions without explicitly enforcing semantic alignment between visual and textual modalities, inevitably resulting in considerable information loss.

3 Methodology

In this section, we introduce our RICO framework and RICO-Flash model. § 3.1 provides an overview of the pipeline of RICO. Subsequently, § 3.2 describes how we generate the reference reconstruction image. § 3.3 presents the method designed to refine the caption. Finally, § 3.4 illustrates the process of training a compact model RICO-Flash to learn the iterative process using DPO.

3.1 Overall Pipeline of RICO

As illustrated in Fig. 3, in our RICO framework, the initial caption c_0 for the original image v_0 is generated by the initial captioning model. A reconstruction model $\mathbf{T}$ and a refinement model $\mathbf{R}$ are then alternately applied to iteratively improve the caption. In each iteration $i \geq 1$, the reconstruction procedure converts the previous candidate caption c_{i-1} into a reconstructed image v_i, and the refinement model generates a refined caption based on the previous caption c_{i-1}, the original image v_0, and the reconstructed reference image v_i. Formally, the refinement step is defined as:

$$c_i = \mathbf{R}(v_i, v_0, c_{i-1}) = \mathbf{R}(\mathbf{T}(c_{i-1}), v_0, c_{i-1}).$$

3.2 Reconstruct Candidate Caption into Reference Image

As discussed in § 1, the semantic information space of captions generated by typical captioning processes tends to be biased and lossy compared to the information contained in the original image. Specifically, we denote the semantic space of the original image as $\mathcal{V}$ and that of the generated caption as $\mathcal{C}$. A biased caption implies that for some information $i \in \mathcal{C}$, $f(i) \notin \mathcal{V}$, and a lossy caption implies that for some information $j \in \mathcal{V}$, $g(j) \notin \mathcal{C}$, where f represents the mapping from textual to visual information, and g denotes the reverse. A key insight of this work is that directly comparing the information spaces $\mathcal{V}$ and $\mathcal{C}$ is challenging due to the cross-modal nature of f and g. To address this, we leverage a powerful text-to-image model to reconstruct the caption into an image. This enables a more direct comparison between the original image

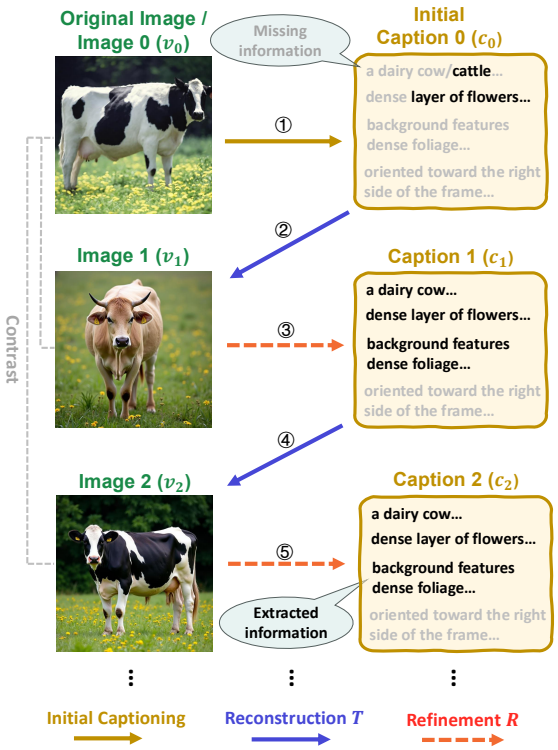

Figure 3: Illustration of the iterative process of RICO. After the **initial captioning** step, a **reconstruction** procedure is applied to generate an image from the candidate caption. The caption is then **refined** by comparing the original image with the reconstructed image.

$\mathcal{V}$ and the reconstructed image $\hat{\mathcal{V}}$, as both reside in the visual modality.

In particular, we use the FLUX.1-dev model (Labs, 2024) as our text-to-image generator, given its strong performance and open-source availability. A notable advantage of FLUX.1-dev is its use of a T5 text encoder (Raffel et al., 2023), which supports longer prompts, surpassing the 77-token limit imposed by CLIP-based models. This allows us to process more detailed captions and faithfully reconstruct their visual content. Formally, for a given generated caption c_{i-1}, we use the text-to-image model to produce a reference image v_i via $v_i = \mathbf{T}(c_{i-1})$, effectively translating the information space of the candidate caption into visual form and facilitating the identification of discrepancies from the original image.

3.3 Refine Caption with Reference Image Feedback

Having obtained the reconstructed reference image v_i, we proceed to refine the previous candidate caption c_{i-1} based on the discrepancy between the reconstructed image v_i and the original image v_0,

thereby generating an updated caption c_i, defined as $c_i = \mathbf{R}(c_{i-1}, v_i, v_0)$. Given the complexity of this task, we utilize one of the most advanced multimodal large language models, GPT-4o (OpenAI et al., 2024), to perform the refinement process. We observed that directly feeding all relevant information into the model yields suboptimal results, highlighting the importance of prompt engineering. To address this, we carefully design prompts with attention to several key aspects outlined below. The complete prompt is provided in § C.1.

Task Description We explicitly inform the model of the task objective, with a particular emphasis on how the reference image is generated. Additionally, the model is instructed to focus on the discrepancies between the reference image and the original image as the basis for refining the caption.

Aspects the Model Should Focus On It is not intuitive for the refinement model to determine what aspects of the discrepancy between the original image and the generated reference image it should focus on, and ranking the importance of different aspects is challenging. Therefore, we provide the model with some guidance. We define eight aspects for the model to prioritize, including: *'Visual Details, Composition & Layout, Human Attributes (if applicable), Perspective & Style, Text in the Image, Image Quality, World Knowledge, and Color Aesthetics.'*

Guidance for Improvement Method To guide the model in refining the candidate caption, we categorize improvements into two types: addressing *inaccuracy* and *incompleteness*. For inaccuracy, the model is instructed to identify and correct errors based on discrepancies between the original and reconstructed images, and to revise any ambiguous descriptions in the previous caption that may have caused inaccurate reconstruction. For incompleteness, the model is encouraged to incorporate missing details and to elaborate on key attributes of the main objects, such as color, shape, and other fine-grained characteristics.

Force Model to Output Analysis Process Inspired by the success of Chain of Thought (CoT) (Wei et al., 2023), we prompt the model to output not only the revised caption but also the corresponding analysis process. This technique serves two purposes: it allows us to examine the reasoning steps of the black-box multimodal large language model, and, as shown in our experiments in § 4.5, it improves the quality of the generated captions by encouraging the model to deliberate more deeply. For practical implementation, we instruct the model to enclose the analysis within special markers `<analysis>...</analysis>` to facilitate automated post-processing.

3.4 RICO-Flash: Leverage DPO to Mitigate Computational Cost

Preliminaries of DPO Direct Preference Optimization (DPO) (Rafailov et al., 2024) is a recently proposed algorithm for aligning language models with human preferences without relying on reinforcement learning. Unlike traditional Reinforcement Learning from Human Feedback (RLHF) methods, which involve separate reward modeling and policy optimization steps, DPO formulates preference learning as a binary classification problem between preferred and dispreferred responses. Formally, given a prompt x and a pair of responses (y^+, y^-), where y^+ is preferred over y^-, DPO optimizes the likelihood ratio between the two responses under a learned policy π_θ and a fixed reference policy π_{ref}, using the following objective:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{\pi_\theta(y^+\mid x)}{\pi_{\text{ref}}(y^+\mid x)} - \beta\log\frac{\pi_\theta(y^-\mid x)}{\pi_{\text{ref}}(y^-\mid x)}\right)\right],$$

Here, β is a temperature-like hyperparameter that controls the sharpness of the preference modeling. The objective encourages the model to assign higher relative likelihoods to preferred responses compared to dispreferred ones, with respect to the reference policy.

Given that our iterative refinement process incurs substantial inference time and computational overhead, we explore the development of an end-to-end variant. Noting that the iterative procedure implicitly induces a preference relationship between captions, we adopt Direct Preference Optimization (DPO) to learn these preferences. Specifically, we collect a high-quality image dataset and apply RICO to generate refined captions. For each image $v^{(i)}$, we extract the initial caption $c_0^{(i)}$ and the final caption after N refinement steps, $c_N^{(i)}$, forming a preference tuple $(v^{(i)}, c_0^{(i)}, c_N^{(i)})$. Based on our empirical observation that $c_N^{(i)}$ consistently outperforms $c_0^{(i)}$ in most cases, we treat this pairwise preference as supervision for DPO training. We adopt Qwen2-VL (Wang et al., 2024b) as the base model and fine-tune it using the DPO objective, yielding an end-to-end variant we denote as RICO-Flash.

Table 1: Performance of RICO and RICO-Flash under different initial MLLM recaptioning models. For RICO-Flash, we use the corresponding MLLM as the base model. In CapsBench, *Acc.* denotes overall accuracy, and *Rel.Pos.* indicates relative position accuracy. In CompreCap, *Obj.*, *Pix.*, *Attr.*, and *Rel.* represent object coverage, pixel coverage, attribute score, and relation score, respectively. *Over.* in Amber refers to overall performance (see § B.2 for details). Green text indicates improvements. RICO demonstrates significant gains over the original captions, while RICO-Flash achieves performance close to that of RICO.

Method	CapsBench			Amber	CompreCap			
	Acc. ↑	Color ↑	Rel. Pos. ↑	Over. ↑	Obj. ↑	Pix. ↑	Attr. ↑	Rel. ↑
Qwen2-VL Init.	42.0	48.1	32.4	59.7	69.82	60.02	2.66	2.81
+ RICO-Flash	55.3 (+13.3)	66.7 (+18.6)	55.1 (+22.7)	60.6 (+0.9)	74.80 (+4.98)	**63.35** (+3.33)	2.84 (+0.18)	**2.84** (+0.03)
+ RICO ($N = 2$)	**59.0** (+17.0)	**67.1** (+19.0)	**59.5** (+27.1)	**62.2** (+2.5)	**75.04** (+5.22)	63.04 (+3.02)	**2.85** (+0.19)	2.82 (+0.01)
LLaVA-1.5 Init.	29.5	27.8	18.1	44.7	57.14	44.48	2.02	2.38
+ RICO-Flash	46.2 (+16.7)	49.6 (+21.8)	38.7 (+20.6)	53.1 (+8.4)	66.68 (+9.54)	56.52 (+12.04)	2.53 (+0.51)	2.43 (+0.05)
+ RICO ($N = 2$)	**53.1** (+23.6)	**61.1** (+33.3)	**48.1** (+30.0)	**59.7** (+15.0)	**76.38** (+19.24)	**61.49** (+17.01)	**2.82** (+0.80)	**2.82** (+0.44)

Table 2: Recaptioning results by humans and models based on the initial caption. In our RICO method, a single iteration of refinement is performed.

Model	CapsBench (Subset)			
	Acc.	Color	Rel. Pos.	Shape
Original	43.55	44.30	39.45	20.41
+ GPT-4o Edit	49.50	53.02	44.04	24.49
+ Human Edit	50.96	51.30	47.82	27.01
+ RICO Edit	**54.08**	**65.47**	**34.04**	**49.51**

This model directly generates improved captions without requiring iterative alternation between a text-to-image model and a caption refinement module, thereby significantly reducing inference cost while maintaining competitive performance.

4 Experiments

4.1 Setup

4.1.1 Implementation Details

For the implementation details, the text-to-image generation is performed using the FLUX.1-dev model (Labs, 2024), while the caption refinement process is conducted with GPT-4o (24-08-06) (OpenAI et al., 2024). We set the number of interaction steps $N = 2$, based on empirical observations that this configuration achieves a good balance between performance and computational efficiency. For the DPO experiments, we initialize with the Qwen2-VL model and set the preference scaling parameter $\beta = 0.1$. The model is fine-tuned for 3 epochs with a learning rate of $\eta = 1.0 \times 10^{-5}$. More implementation details can be found in § C.

4.1.2 Evaluation Benchmarks

In the era of MLLMs, traditional captioning metrics (Papineni et al., 2002; Vedantam et al., 2015) often fail to capture fine-grained details and inadequately penalize hallucinations. To address these limitations, in addition to the recently proposed reference-based metric CAPTURE (Dong et al., 2024a), we adopt more advanced benchmarks to more faithfully evaluate the quality of our method. Specifically, we employ CapsBench (Liu et al., 2024a), which uses QA pairs to assess the accuracy and comprehensiveness of generated captions. We also utilize CompreCap (Lu et al., 2025), which leverages a Directed Scene Graph to evaluate the correctness of object mentions and their relationships. Furthermore, we adopt Amber (Wang et al., 2024a) to assess hallucinations in the generated descriptions. More details can be found in § B.2.

4.2 Effectiveness of RICO and RICO-Flash

We verify that our RICO pipeline effectively addresses both inaccuracy and incompleteness in recaptioning. Firstly, we use two popular open-source models Qwen2-VL (Wang et al., 2024b) and LLaVA-1.5 (Liu et al., 2024b) as the initial captioning models to produce baseline captions, which are then refined by RICO. As shown in Tab. 1, even with just two refinement iterations, the captions generated by RICO exhibit substantial improvements across all benchmarks and metrics. Notably, the improvement in the overall score on the Amber indicates that RICO mitigates hallucination. Furthermore, on CapsBench, we emphasize two critical aspects—color and relative position—and show that the reconstruction step helps the model more accurately identify and correct fine-grained discrep-

Table 3: Comparison with baseline methods across various evaluation metrics. Our method achieves the best performance on most metrics, while RICO-Flash demonstrates performance comparable to RICO. Bold text indicates the best results, and underlined text denotes the second-best.

Method	CapsBench				CompreCap				Amber	Capture
	Acc.	Color	Shape	Rel. Pos.	Obj.	Pix.	Rel.	Attr.	Over.	
LaCLIP (Fan et al.)	22.65	21.65	9.09	11.11	48.02	42.59	1.73	2.29	43.8	39.56
CapsFusion (Yu et al.)	35.04	38.14	12.12	25.46	61.67	52.63	2.32	2.59	44.5	56.03
Self-Loop (Dong et al.)	29.63	29.55	9.09	17.13	65.77	51.54	2.30	2.53	49.5	56.61
VeCLIP (Lai et al.)	25.19	27.84	11.11	13.43	49.60	42.25	2.50	1.77	41.0	38.13
ShareGPT4V (Chen et al.)	50.46	62.13	38.78	49.34	67.47	62.00	2.83	2.81	56.2	59.80
RICO-Flash (Ours)	55.32	66.67	50.29	55.09	74.80	**63.35**	2.84	**2.84**	60.6	65.52
RICO (Ours)	**59.02**	**67.14**	**53.68**	**59.51**	**75.04**	63.04	**2.85**	2.82	**62.2**	**65.98**

ancies. In addition, we can see that RICO-Flash achieves performance that closely matches RICO while still demonstrating substantial improvements over the initial captions, validating its effectiveness as a non-iterative alternative.

Secondly, we assess recaptioning quality by comparing RICO against GPT-4o and human annotators. We randomly select 100 images from CapsBench, generate initial captions using Qwen2-VL, and perform one round of editing using GPT-4o, RICO, and human annotators. The results, shown in Tab. 2, demonstrate that RICO achieves strong recaptioning performance, even surpassing humans, who tend to overlook fine-grained details. Some experiment details can be found in § B.3.

Finally, we conduct a qualitative analysis of the refinement process and present examples showcasing the step-by-step improvement of captions through RICO in § A.1.

4.3 Comparison with Other Recaptioning Methods

We compare our approach with other recaptioning methods, and the results are presented in Tab. 3. RICO demonstrates strong performance across all evaluation metrics, particularly in fine-grained aspects such as color, entity shape, and relative position. This highlights the importance of reconstruction for achieving better alignment between textual descriptions and visual content. Details on how the baseline methods perform recaptioning are provided in § B.1.

4.4 Further Analysis

We conduct more experiments to help better understand our RICO pipeline.

Table 4: Evaluation of a text-to-image generation model trained with original captions versus captions refined by our RICO-Flash model. *Rel.* and *Attr.* represent relation and attribute respectively.

Model	DPG-Bench			VQAScore
	Rel.	Attr.	Overall	
FLUX w/ Init. Cap.	89.95	80.08	78.50	0.84
FLUX w/ RICO-Flash	**90.55**	**82.83**	**80.34**	**0.85**

4.4.1 Verify Alignment via Text-to-Image Generation

To verify that RICO effectively builds a well-aligned image-text semantic space, we evaluate it on a classical downstream task: text-to-image generation. We collect an image dataset from Huggingface[1] and use RICO to perform recaptioning. Specifically, for each image v, we obtain both the initial caption c_0 and the refined caption c_N, forming two datasets: $\mathcal{D}_{\text{initial}} = \{(v^{(i)}, c_0^{(i)})\}$ and $\mathcal{D}_{\text{refined}} = \{(v^{(i)}, c_N^{(i)})\}$. We then use these datasets to train two separate text-to-image generation models based on FLUX.1-dev. For evaluation, considering that the prompts in our dataset are typically long and thus incompatible with many existing benchmarks (Ghosh et al., 2023; Huang et al., 2025), we adopt the recently proposed DPG-Bench (Hu et al., 2024), which is designed to evaluate detailed prompts. Moreover, we also employ VQAScore (Lin et al., 2024)—a reference-free metric that serves as a robust alternative to CLIPScore (Hessel et al., 2022; Imagen-Team-Google et al., 2024). Following the official implementation in the original paper, we use CLIP-FlanT5-XXL (Lin et al., 2024) as the VQA model As shown in Tab. 4, the model fine-tuned on the refined dataset consis-

[1]Mainly from https://huggingface.co/datasets/jackyhate/text-to-image-2M

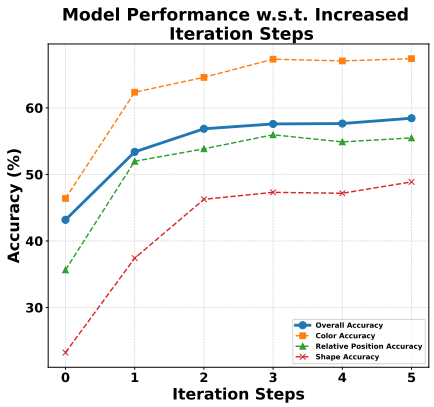

Figure 4: Performance of the RICO pipeline under different numbers of refinement iterations.

tently outperforms the baseline across all metrics. Notably, it achieves improvements in entity, relation, and attribute dimensions, demonstrating that our reconstruction-refinement pipeline enhances the alignment between image and caption in fine-grained semantic aspects. Detailed training configurations are provided in § B.4.

4.4.2 Saturation with Increased Iteration Steps

In RICO, the caption is progressively refined as the number of iteration steps increases. As shown in Fig. 4, performance consistently improves with each additional iteration. However, the gains begin to plateau after approximately the second step, with only marginal improvements observed thereafter. This suggests that the generated caption reaches a satisfactory quality level, at least given the capabilities of the reconstruction and refinement modules. Detailed results can be found in § A.5.

4.4.3 Generalization to Different Initial Captions

To evaluate the generalization capability of RICO, we examine whether it can consistently enhance captions through the reconstruction-refinement pipeline. As shown in the upper part of Tab. 5, we generate initial captions using different captioning models. The results indicate that RICO significantly improves captions from all initial models, demonstrating its robustness. Notably, although our refinement module is based on GPT-4o, captions generated by GPT-4o alone do not outperform the final outputs, suggesting that RICO does more than simply distill the captioning ability of GPT-4o. In the lower part of Tab. 5, we assess performance using various initial prompts. The results show that

Table 5: Generality of RICO across different initial recaptioning models and prompts. For CapsBench, we report overall accuracy, and for CompreCap, we use the unified metric for evaluation.

Model	CapsBench	CompreCap
GPT-4o	49.6 / 57.7 (+8.1)	58.6 / 60.4 (+1.8)
Gemini 1.5 Pro	49.7 / 57.7 (+8.0)	60.1 / 61.5 (+1.4)
BLIP-3	37.0 / 56.2 (+19.2)	55.4 / 60.2 (+4.8)
CogVLM 2	45.1 / 57.5 (+12.4)	56.0 / 60.3 (+4.3)
Qwen2-VL (Prompt 1)	42.0 / 59.0 (+17.0)	55.9 / 61.4 (+5.5)
Qwen2-VL (Prompt 2)	46.0 / 57.6 (+11.6)	57.2 / 60.6 (+3.4)
Qwen2-VL (Prompt 3)	41.9 / 54.9 (+13.0)	56.9 / 60.9 (+4.0)

Table 6: Performance of different refinement models. For CapsBench, we report overall accuracy.

Model	CapsBench
Qwen2-VL as reviser	42.0 / 45.8 (+3.8)
GPT-4o as reviser	42.0 / 59.0 (+17.0)

our pipeline yields substantial improvements across different prompts. While modifying the prompt within the same MLLM can lead to some gains, these are relatively minor compared to the improvements achieved by RICO. Importantly, our method is also orthogonal to prompt-based strategies and can be combined with more effective prompts for further enhancement.

4.4.4 Discussion on the Reliance on GPT-4o

We acknowledge the computational cost and the closed-source nature introduced by the GPT-4o of the original RICO model. Our RICO-Flash model was specifically designed to address this issue. By generating captions in a single step and using an open-source base model, RICO-Flash is well-suited for large-scale dataset re-captioning tasks.

Actually, the rationale for using GPT-4o lies in the complexity of the caption refinement process guided by the reference image. This process requires (1) accurately perceiving the discrepancies between the original and reference images, and (2) strong textual organization capabilities to effectively revise and integrate feedback into the original caption. These requirements demand a powerful multimodal language model (MLLM), and GPT-4o was chosen for its demonstrated strength in both visual understanding and coherent text generation.

We also evaluated other models, such as Qwen2-VL, using the same prompt. However, we found that Qwen2-VL struggled to fully understand the refinement setting, and the resulting revised cap-

Table 7: Performance of different comparison methods.

Methods	Capsbench Acc.
Original Qwen2-VL	43.55
+ Image-text comparison	49.50
+ Image-image comparison	54.08

Table 8: Ablation studies.

Method	CapsBench			
	Acc.	Color	Rel. Pos.	Shape
RICO	**59.02**	**67.14**	**59.51**	**53.68**
RICO-Flash	55.32	66.67	55.09	50.29
(a) wo/ tips	54.33	62.23	50.95	42.11
(b) wo/ output analy.	50.40	62.54	53.24	36.36
(c) finetune w/ pos.	51.16	59.79	51.85	32.32
(d) infer with ICL	45.26	49.83	42.13	26.26

tions were less satisfactory. We further evaluated its performance on CapsBench, with the results shown in Tab. 6. We can see that compared to GPT-4o, Qwen2-VL yields only limited improvements over the baseline. We also plan to evaluate more powerful open-sourced models in future work.

4.4.5 Influence of Comparison Mechanism

In our RICO model, we adopt an image-to-image comparison strategy. Compared to image-to-caption comparison, image-to-image comparison offers several advantages. First, it is inherently a unimodal task and thus avoids the cross-modal translation required by image-to-caption comparison. This allows both humans and models to directly align visual content at corresponding spatial locations to identify discrepancies. In contrast, image-to-caption comparison is more challenging since captions provide high-level, abstract summaries that require semantic grounding to specific visual elements. Moreover, the sequential and linguistic structure of captions often does not align with the spatial structure of images, further increasing the difficulty of accurate comparison.

To validate this, we conducted experiments using GPT-4o to refine captions based on either image-to-caption comparison or image-to-image comparison. The results are summarized in Tab. 7. As shown, image-to-image comparison leads to a greater improvement, suggesting that it provides more effective feedback for caption refinement.

4.5 Ablation Studies

We conduct ablation studies to validate our design choices, with results presented in Tab. 8. The findings are: **(a)** When the refinement model is not guided on which aspects to focus, it struggles to identify key elements, resulting in a performance drop. **(b)** Omitting the requirement for the model to output an analysis process, which is intended to promote deliberate reasoning, also leads to degraded performance. Regarding the DPO method, we evaluate two alternative strategies: **(c)** directly fine-tuning the base model using positive samples, and **(d)** incorporating a positive sample into the prompt for in-context learning (Dong et al., 2024b). Both approaches yield inferior results compared to the DPO method, underscoring the effectiveness of DPO in our setting.

5 Conclusion

In this paper, we propose the RICO pipeline, which leverages visual reconstruction to improve the accuracy and completeness of image recaptioning. We also introduce an efficient variant, RICO-Flash, which learns the iterative refinement process of RICO by DPO. Experimental results show that our method achieves well-aligned semantic representations between images and their captions, and delivers strong recaptioning performance compared to prior baselines. Further evaluations also confirm the generalizability of our approach. We hope RICO will inspire new techniques in image recaptioning and may contribute to advancements in broader multimodal research.

Acknowledgement

This research was partially supported by the National Natural Science Foundation of China under Grant No. 92470205 and No. 62176002. Xu Sun and Yuanxing Zhang are corresponding authors.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, and 8 others. 2022. Flamingo: a visual language model for few-shot learning. *Preprint*, arXiv:2204.14198.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic

propositional image caption evaluation. *Preprint*, arXiv:1607.08822.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, and 29 others. 2023a. Qwen technical report. *Preprint*, arXiv:2309.16609.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023b. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *Preprint*, arXiv:2308.12966.

Zechen Bai, Pichao Wang, Tianjun Xiao, Tong He, Zongbo Han, Zheng Zhang, and Mike Zheng Shou. 2025. Hallucination of multimodal large language models: A survey. *Preprint*, arXiv:2404.18930.

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh. Improving image generation with better captions.

David Chan, Suzanne Petryk, Joseph E. Gonzalez, Trevor Darrell, and John Canny. 2023. Clair: Evaluating image captions with large language models. *Preprint*, arXiv:2310.12971.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. 2023. Sharegpt4v: Improving large multimodal models with better captions. *Preprint*, arXiv:2311.12793.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Hongyuan Dong, Jiawen Li, Bohong Wu, Jiacong Wang, Yuan Zhang, and Haoyuan Guo. 2024a. Benchmarking and improving detail image caption. *Preprint*, arXiv:2405.19092.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024b. A survey on in-context learning. *Preprint*, arXiv:2301.00234.

Lijie Fan, Dilip Krishnan, Phillip Isola, Dina Katabi, and Yonglong Tian. 2023. Improving clip training with language rewrites. *Preprint*, arXiv:2305.20088.

Taraneh Ghandi, Hamidreza Pourreza, and Hamidreza Mahyar. 2023. Deep learning approaches on image captioning: A review. *ACM Computing Surveys*, 56(3):1–39.

Dhruba Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. 2023. Geneval: An object-focused framework for evaluating text-to-image alignment. *Preprint*, arXiv:2310.11513.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2022. Clipscore: A reference-free evaluation metric for image captioning. *Preprint*, arXiv:2104.08718.

Wenyi Hong, Weihan Wang, Ming Ding, Wenmeng Yu, Qingsong Lv, Yan Wang, Yean Cheng, Shiyu Huang, Junhui Ji, Zhao Xue, Lei Zhao, Zhuoyi Yang, Xiaotao Gu, Xiaohan Zhang, Guanyu Feng, Da Yin, Zihan Wang, Ji Qi, Xixuan Song, and 6 others. 2024. Cogvlm2: Visual language models for image and video understanding. *Preprint*, arXiv:2408.16500.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Xiwei Hu, Rui Wang, Yixiao Fang, Bin Fu, Pei Cheng, and Gang Yu. 2024. Ella: Equip diffusion models with llm for enhanced semantic alignment. *Preprint*, arXiv:2403.05135.

Kaiyi Huang, Chengqi Duan, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. 2025. T2i-compbench++: An enhanced and comprehensive benchmark for compositional text-to-image generation. *Preprint*, arXiv:2307.06350.

Imagen-Team-Google, :, Jason Baldridge, Jakob Bauer, Mukul Bhutani, Nicole Brichtova, Andrew Bunner, Lluis Castrejon, Kelvin Chan, Yichang Chen, Sander Dieleman, Yuqing Du, Zach Eaton-Rosen, Hongliang Fei, Nando de Freitas, Yilin Gao, Evgeny Gladchenko, Sergio Gómez Colmenarejo, Mandy Guo, and 243 others. 2024. Imagen 3. *Preprint*, arXiv:2408.07009.

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

Zhengfeng Lai, Haotian Zhang, Bowen Zhang, Wentao Wu, Haoping Bai, Aleksei Timofeev, Xianzhi Du, Zhe Gan, Jiulong Shan, Chen-Nee Chuah, Yinfei Yang, and Meng Cao. 2024. Veclip: Improving clip training via visual-enriched captions. *Preprint*, arXiv:2310.07699.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *Preprint*, arXiv:2301.12597.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *Preprint*, arXiv:2201.12086.

Zhuang Li, Yuyang Chai, Terry Yue Zhuo, Lizhen Qu, Gholamreza Haffari, Fei Li, Donghong Ji, and Quan Hung Tran. 2023b. Factual: A benchmark for faithful and consistent textual scene graph parsing. *Preprint*, arXiv:2305.17497.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. 2024. Evaluating text-to-visual generation with image-to-text generation. *Preprint*, arXiv:2404.01291.

Bingchen Liu, Ehsan Akhgari, Alexander Visheratin, Aleks Kamko, Linmiao Xu, Shivam Shrirao, Chase Lambert, Joao Souza, Suhail Doshi, and Daiqing Li. 2024a. Playground v3: Improving text-to-image alignment with deep-fusion large language models. *Preprint*, arXiv:2409.10695.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024b. Improved baselines with visual instruction tuning. *Preprint*, arXiv:2310.03744.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Preprint*, arXiv:2304.08485.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Fan Lu, Wei Wu, Kecheng Zheng, Shuailei Ma, Biao Gong, Jiawei Liu, Wei Zhai, Yang Cao, Yujun Shen, and Zheng-Jun Zha. 2025. Benchmarking large vision-language models via directed scene graph for comprehensive image captioning. *Preprint*, arXiv:2412.08614.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

OpenAI. 2023. Gpt-4v(ision) system card.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ruotian Peng, Haiying He, Yake Wei, Yandong Wen, and Di Hu. 2025. Patch matters: Training-free fine-grained image caption enhancement via local perception. *Preprint*, arXiv:2504.06666.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

Shuhuai Ren, Linli Yao, Shicheng Li, Xu Sun, and Lu Hou. 2024. Timechat: A time-sensitive multimodal large language model for long video understanding. *Preprint*, arXiv:2312.02051.

Yanpeng Sun, Jing Hao, Ke Zhu, Jiang-Jiang Liu, Yuxiang Zhao, Xiaofan Li, Gang Zhang, Zechao Li, and Jingdong Wang. 2025. Descriptive caption enhancement with visual specialists for multimodal perception. *Preprint*, arXiv:2412.14233.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, and 1331 others. 2024a. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, Soroosh Mariooryad, Yifan Ding, Xinyang Geng, Fred Alcober, Roy Frostig, Mark Omernick, Lexi Walker, Cosmin Paduraru, Christina Sorokin, and 1118 others. 2024b. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. *Preprint*, arXiv:1411.5726.

Junyang Wang, Yuhang Wang, Guohai Xu, Jing Zhang, Yukai Gu, Haitao Jia, Jiaqi Wang, Haiyang Xu, Ming Yan, Ji Zhang, and Jitao Sang. 2024a. Amber: An llm-free multi-dimensional benchmark for mllms hallucination evaluation. *Preprint*, arXiv:2311.07397.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024b. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *Preprint*, arXiv:2409.12191.

Weihan Wang, Qingsong Lv, Wenmeng Yu, Wenyi Hong, Ji Qi, Yan Wang, Junhui Ji, Zhuoyi Yang, Lei Zhao, Xixuan Song, Jiazheng Xu, Bin Xu, Juanzi Li, Yuxiao Dong, Ming Ding, and Jie Tang. 2024c. Cogvlm: Visual expert for pretrained language models. *Preprint*, arXiv:2311.03079.

Yuchi Wang, Shuhuai Ren, Rundong Gao, Linli Yao, Qingyan Guo, Kaikai An, Jianhong Bai, and Xu Sun. 2024d. Ladic: Are diffusion models really inferior to autoregressive counterparts for image-to-text generation? *Preprint*, arXiv:2404.10763.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Hu Xu, Po-Yao Huang, Xiaoqing Ellen Tan, Ching-Feng Yeh, Jacob Kahn, Christine Jou, Gargi Ghosh, Omer Levy, Luke Zettlemoyer, Wen tau Yih, Shang-Wen Li, Saining Xie, and Christoph Feichtenhofer. 2024. Altogether: Image captioning via re-aligning alt-text. *Preprint*, arXiv:2410.17251.

Le Xue, Manli Shu, Anas Awadalla, Jun Wang, An Yan, Senthil Purushwalkam, Honglu Zhou, Viraj Prabhu, Yutong Dai, Michael S Ryoo, Shrikant Kendre, Jieyu Zhang, Can Qin, Shu Zhang, Chia-Chih Chen, Ning Yu, Juntao Tan, Tulika Manoj Awalgaonkar, Shelby Heinecke, and 8 others. 2024. xgen-mm (blip-3): A family of open large multimodal models. *Preprint*, arXiv:2408.08872.

Ling Yang, Zhaochen Yu, Chenlin Meng, Minkai Xu, Stefano Ermon, and Bin Cui. 2024. Mastering text-to-image diffusion: Recaptioning, planning, and generating with multimodal llms. *Preprint*, arXiv:2401.11708.

Linli Yao, Weijing Chen, and Qin Jin. 2023. Capenrich: Enriching caption semantics for web images via cross-modal pre-trained knowledge. *Preprint*, arXiv:2211.09371.

Qinghao Ye, Xianhan Zeng, Fu Li, Chunyuan Li, and Haoqi Fan. 2025. Painting with words: Elevating detailed image captioning with benchmark and alignment learning. *Preprint*, arXiv:2503.07906.

Qiying Yu, Quan Sun, Xiaosong Zhang, Yufeng Cui, Fan Zhang, Yue Cao, Xinlong Wang, and Jingjing Liu. 2024. Capsfusion: Rethinking image-text data at scale. *Preprint*, arXiv:2310.20550.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *Preprint*, arXiv:2304.10592.

A Additional Experimental Results

A.1 Qualitative Analysis of RICO

We present an example of the RICO refinement process in Fig. 5. We can see that as the refinement progresses, the caption is progressively revised to incorporate important missing details. Additionally, Fig. 6 provides a case accompanied by an in-depth analysis. The analysis illustrates that our refinement model effectively identifies discrepancies and generates reasonable revision suggestions, resulting in more accurate and comprehensive captions.

A.2 Detailed Results on the Generalization of RICO

As discussed in § 4.4.3, our method consistently performs well across various initial captioning models and prompt configurations. Extended results for different prompt variants are presented in Tab. 17, with the corresponding prompt templates listed in Tab. 18. Detailed results using different initial captioning models are provided in Tab. 16. These findings further validate the robustness and effectiveness of RICO under diverse settings.

A.3 Detailed Results of the Text-to-Image Generation Experiment

We present the expanded results of Tab. 4 in Tab. 19. The text-to-image model trained with captions generated by our method consistently outperforms the model trained with initial captions across nearly all metrics, demonstrating improved alignment between image and text semantic spaces in RICO.

A.4 Computational Efficiency of RICO-Flash

We report the total inference time per image in the Tab. 9 (note that the time may vary depending on the specific GPT-4o API used). As shown, RICO-Flash significantly reduces inference time by removing the need for iterative refinement. Additionally, when the analysis process is omitted, the impact on inference time is relatively small.

A.5 Results of More Iterations

We show the results of more refinement steps in Tab. 10. We observe that after Step 4, there is no significant improvement, and the variations are largely attributable to random fluctuations introduced by the evaluation process. In fact, the captions undergo only minimal changes in the later steps.

Table 9: Inference time comparison of different models. "RICO wo/ ana." denotes the RICO model without the necessity to generate an analysis process.

Model	Time (s)
RICO	40.1
RICO wo/ ana.	37.8
RICO-Flash	4.2

A.6 Performance on CLAIR Metric

We evaluate our model using the CLAIR metric (Chan et al., 2023), as reported in Tab. 11. CLAIR is a novel method that leverages the zero-shot language modeling capabilities of large language models (LLMs) to evaluate candidate captions. Our model consistently outperforms the baselines, highlighting the effectiveness of the proposed reconstruction–refinement pipeline.

A.7 Potential Bias from Using GPT-4o for Both Recaptioning and Evaluation

In our CapsBench evaluation, we employ GPT-4o for both caption refinement and evaluation. We acknowledge that reusing the same model in both stages may introduce evaluation bias. To address this concern, we conduct additional evaluations using two alternative evaluators—GPT-4V and Gemini 2.5 Pro. The results, summarized in Tab. 12, exhibit consistent trends across evaluators, supporting the reliability of our findings and indicating that the observed improvements are not artifacts of model-specific bias.

Furthermore, we perform an additional experiment where Gemini 2.5 Pro served as the evaluation model, while GPT-4o remained the refinement model in our RICO pipeline. As reported in Tab. 13, our method continues to substantially outperform all baselines, thereby demonstrating both robustness and general effectiveness across diverse evaluation settings.

A.8 Discussion on the Rel. Pos. in Capsbench

In CapsBench, the Rel. Pos. (Relative Position) metric evaluates whether the caption accurately captures the relative spatial relationships between objects in the image. Our model performs well in this aspect, which we believe is due to the advantage of image reconstruction in our framework. Specifically, in traditional image-caption comparison, relative position information is often underemphasized or lost due to the abstract nature of

Table 10: Performance of different refinement steps.

Steps	1	2	3	4	5	6	7	8	9	10
Capsbench Acc.	43.18	53.39	56.86	57.59	57.65	58.45	58.23	58.73	57.99	58.62

Table 11: Performance on CLAIR metric.

Methods	CLAIR
LaCLIP	0.749
CapsFusion	0.795
RICO-Flash	0.839
RICO	0.864

Table 12: Evaluations using different models.

Model Used for Evaluation	CapsBench Acc.
GPT-4o	59.0
GPT-4v	58.7
Gemini 2.5pro	59.2

Table 13: Results on CapsBench, evaluated by Gemini 2.5 Pro.

Models	CapsBench
LaCLIP	23.4
CapsFusion	34.7
Self-Loop	30.4
VeCLIP	23.5
ShareGPT4V	50.6
RICO-Flash	54.3
RICO	59.2

captions. However, when a reference image is reconstructed and directly compared with the original, discrepancies in object arrangements become more salient, allowing the model to more easily refine captions to include accurate positional information.

Table. 14 is an example of an original caption and its refined version, where the newly added relative position cues are highlighted with <**>.

B Additional Information on Experimental Settings

B.1 Details of Baselines and Our Implementations

We compare our method with several recaptioning baselines. The details of each are provided below:

LaCLIP (Fan et al., 2023) LaCLIP identifies that in CLIP training, text inputs tend to be underutilized due to a lack of augmentation. To address this, the authors propose leveraging large language models (LLMs) to rewrite the given text. Specifically, ChatGPT is used to generate meta input-output pairs, which are then used as in-context examples to prompt LLaMA (Touvron et al., 2023) for generating refined captions. In our implementation, we follow the same procedure to obtain enhanced captions. Specifically, we first employ Qwen2-VL-7B-Instruct to simulate the generation of alt text using the prompt: "Describe the image using a few essential keywords. Keep it concise, within 10 words." The meta input-output pairs generated by ChatGPT are then used as in-context examples to prompt Qwen2-VL-7B-Instruct, which generates the final refined captions.

VeCLIP (Lai et al., 2024) While previous methods like LaCLIP focus solely on textual rewriting, VeCLIP emphasizes the incorporation of visual concepts into the caption. It first employs a multimodal LLM (LLaVA) to generate captions independently of the original alt text, and then fuses these captions with the original using another LLM, such as Vicuna (Chiang et al., 2023). In our implementation, we follow the official pipeline. We adopt the same approach as in LaCLIP to generate the initial alt texts. We then utilize LLaVA-1.5-7B-Chat to generate supplementary captions. Finally, Qwen2-VL-7B-Instruct is prompted to fuse these two captions.

CapsFusion (Yu et al., 2024) CapsFusion highlights the importance of combining web-based alt texts and synthetic captions. The authors construct a dataset of 1 million examples by prompting ChatGPT to fuse these two types of captions, which is then used to fine-tune LLaMA, resulting in the CapsFusion-LLaMA model. Technically, we adopt the official implementation: we use the same approach as LaCLIP to generate alt texts, utilize Qwen2-VL-7B-Instruct to produce synthetic captions, and apply the official CapsFusion-LLaMA model weights for fusion.

Self-Loop (Dong et al., 2024a) In the CAPTURE (Dong et al., 2024a) paper, the authors introduce a new metric to evaluate image captioning and design a self-looping caption improvement pipeline

Table 14: An example of an original caption and its refined version.

Original:
The image depicts a scene from an animated series, featuring three characters seated on a green, ornate couch. The man has red hair and is dressed in a brown suit with a white shirt and tie. The woman has blonde hair styled in a high ponytail and is wearing a blue dress with a white collar. The child has pink hair and is dressed in a red outfit with a green bow. The background is a simple, muted color, focusing attention on the characters. The overall style is vibrant and detailed, typical of anime art.

Revised:
The image depicts a scene from an animated series featuring three characters seated on a green, ornate couch with an elaborate, dark wooden backrest. The man <*__on the left__*> has light pink hair styled in a straight cut and is dressed in a brown suit with a white shirt and a black tie. The woman <*__on the right__*> has long blonde hair styled in a sleek, high ponytail and is wearing a light blue dress with a white collar and a sash tied at the waist. The child <*__in the center__*> has light pink hair in a cute bob cut and is dressed in a red outfit with a bright green bow. The background is plain with subtle tones, highlighting the characters. The overall style is soft and detailed, typical of high-quality anime art, with pastel colors. <*__The composition is a medium shot, allowing clear visibility of the characters' attire and expressions.__*>

Table 15: Instructions provided to human annotators in the caption editing experiment.

== INSTRUCTION TO ANNOTATORS ==

We are working on an image captioning task. The following caption was generated by an AI model. Please help refine this caption by correcting any errors or ambiguities based on the image, and feel free to add any important details that are missing from the original caption.

guided by this metric. In detail, the method detects objects in the image, generates local captions, filters out hallucinated objects, and merges local descriptions with the overall caption. We use the official repository to run this baseline.

ShareGPT4V (Chen et al., 2023) ShareGPT4V underscores the critical role of captions in MLLM training. It uses carefully crafted prompts to guide GPT-4V in generating high-quality descriptions, and then trains a Share-Captioner model to replicate this behavior. In our experiments, we use Share-Captioner to generate captions as part of the baseline comparison.

B.2 Details of Evaluation Benchmarks

Traditional caption evaluation metrics (Anderson et al., 2016; Lin, 2004) are not well-suited for evaluating captions generated by modern MLLMs. In our work, we adopt the following evaluation metrics:

CapsBench. Proposed in Playground v3 (Liu et al., 2024a), CapsBench introduces a benchmark designed to evaluate the comprehensiveness and accuracy of image captions. For each image, a set of "yes-no" question-answer pairs is generated across 17 semantic categories. During evaluation, an LLM is tasked with answering these questions based solely on the candidate caption. The possible answers are "yes", "no", and "n/a" (for unanswerable questions). The predicted answers are compared with the ground-truth to compute the overall accuracy. This benchmark effectively assesses whether a model can capture accurate and comprehensive information from the image. In our implementation, we use GPT-4o (2024-08-06) as the judge model.

CompreCap. CompreCap (Lu et al., 2025) is a benchmark that evaluates the compositional understanding of detailed visual scenes through a directed scene graph framework. Each image is segmented into semantically meaningful regions, and objects within these regions are annotated with at-

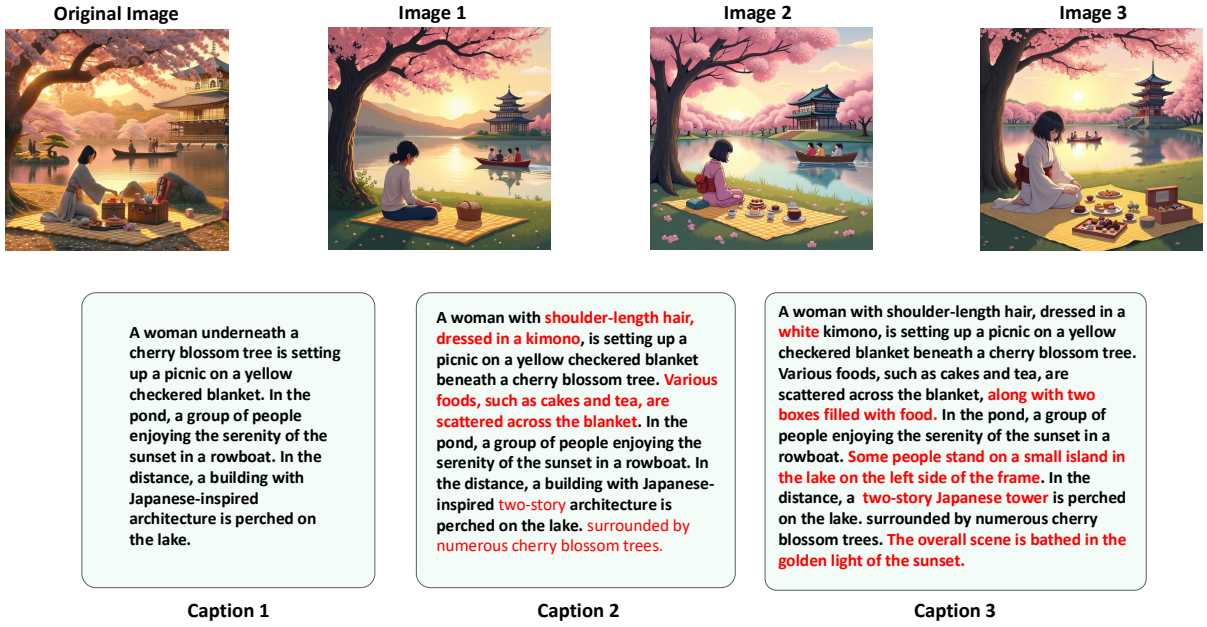

Figure 5: An example demonstrating the iterative refinement process performed by our model, where red text indicates added or corrected information.

Table 16: Detailed performance of RICO across different initial captioning models.

Model	CapsBench				CompreCap			
	Acc.	Color	Shape	Rel. Pos.	Obj.	Pix.	Rel.	Attr.
Qwen2-VL (Wang et al.)	42.02	48.11	27.27	32.41	69.82	60.02	2.66	2.81
Qwen2-VL + RICO	59.02 (+17.00)	67.14 (+19.03)	53.68 (+26.41)	59.51 (+27.10)	75.04 (+5.22)	63.04 (+3.02)	2.85 (+0.19)	2.82 (+0.01)
CogVLM 2 (Hong et al.)	45.10	47.77	28.23	39.81	68.54	59.21	2.57	2.61
CogVLM 2 + RICO	57.51 (+12.41)	63.67 (+15.90)	35.46 (+7.23)	48.76 (+8.95)	75.37 (+6.83)	61.65 (+2.44)	2.78 (+0.21)	2.75 (+0.14)
GPT-4o (OpenAI et al.)	49.63	54.64	28.28	48.15	70.93	60.09	2.67	2.77
GPT-4o + RICO	57.68 (+8.05)	63.24 (+8.60)	44.57 (+16.29)	59.47 (+11.32)	74.47 (+3.54)	62.11 (+2.02)	2.76 (+0.09)	2.81 (+0.04)
Gemini 1.5 Pro (Team et al.)	49.71	51.20	23.23	36.57	71.77	60.28	2.89	2.71
Gemini 1.5 Pro + RICO	57.72 (+8.01)	65.70 (+14.50)	37.50 (+14.27)	50.48 (+13.91)	75.77 (+4.00)	61.97 (+1.69)	2.85 (+-0.04)	2.83 (+0.12)
BLIP-3 (Xue et al.)	37.03	40.55	19.19	29.63	67.85	56.99	2.61	2.50
BLIP-3 + RICO	56.21 (+19.18)	66.20 (+25.65)	37.76 (+18.57)	55.61 (+25.98)	74.31 (+6.46)	61.47 (+4.48)	2.79 (+0.18)	2.75 (+0.25)
LLaVA 1.5 (Liu et al.)	29.51	27.84	9.09	18.06	57.14	44.48	2.02	2.38
LLaVA 1.5 + RICO	53.13 (+23.62)	61.07 (+33.23)	36.84 (+27.75)	48.10 (+30.04)	76.38 (+19.24)	61.49 (+17.01)	2.82 (+0.80)	2.82 (+0.44)

Table 17: Detailed performance of RICO across different initial prompts.

Model	CapsBench				CompreCap			
	Acc.	Color	Shape	Rel. Pos.	Obj.	Pix.	Rel.	Attr.
Prompt #1	42.02	48.11	27.27	32.41	69.82	60.02	2.66	2.81
+ RICO	59.02 (+17.00)	67.14 (+19.03)	53.68 (+26.41)	59.51 (+27.10)	75.04 (+5.22)	63.04 (+3.02)	2.85 (+0.19)	2.82 (+0.01)
Prompt #2	45.97	49.14	23.22	40.74	69.29	60.41	2.69	2.62
+ RICO	57.64 (+11.67)	65.45 (+16.31)	39.08 (+15.86)	56.45 (+15.71)	74.83 (+5.54)	62.65 (+2.24)	2.80 (+0.11)	2.79 (+0.17)
Prompt #3	41.85	43.30	23.23	36.11	68.46	58.89	2.72	2.59
+ RICO	54.85 (+13.00)	66.54 (+23.24)	47.25 (+24.02)	52.85 (+16.74)	75.15 (+6.69)	62.40 (+3.51)	2.80 (+0.08)	2.82 (+0.23)

Table 18: Different prompts used to generate initial captions.

========== DIFFERENT PROMPTS TO GENERATE INITIAL CAPTIONS ==========

Prompt #1: Describe this image in detail. Your answer should be concise and informative.

Prompt #2: Describe the image with rich and detailed observations. You may pay attention to the dimensions of overall, main subject, background, movement of main subject, style, camera movement and so on.

Prompt #3: Give this image a detailed caption.

Table 19: Extended version of the evaluation of the text-to-image model.

Model	DPG-Bench					VQAScore
	Entity	Relation	Attribute	Global	Overall	
FLUX w/ Init. Cap.	85.110	89.950	80.080	72.414	78.502	0.841
FLUX w/ RICO-DPO	**86.850**	**90.551**	**82.831**	**75.172**	**80.336**	**0.852**

tributes and directional relations to form a directed scene graph. The benchmark then assesses generated captions based on three levels: (1) object-level coverage, (2) accuracy of attribute descriptions, and (3) correctness of key relationships. This benchmark is particularly effective at evaluating the model's ability to capture relational and compositional details. We adopt the official implementation for our evaluation.

Amber. Amber (Wang et al., 2024a) is designed to evaluate hallucinations in MLLM-generated captions by comparing the set of objects mentioned in the caption with a pre-annotated object list for the image. It defines several metrics: *CHAIR*, which quantifies the frequency of hallucinated (i.e., nonexistent) objects, and *Cover*, which measures how well the caption covers the annotated objects. Following the original paper's claim that "an ideal response is considered to be one that minimizes hallucinatory content without significantly compromising the coverage of objects in the image," we adopt a unified metric, $Cover - CHAIR$, to reflect this trade-off. This provides a concise and interpretable measure of caption faithfulness.

CAPTURE CAPTURE (Dong et al., 2024a) introduce a benchmark designed to evaluate detailed image captioning performance by extracting and comparing core visual elements in generated captions. Unlike traditional metrics that rely on n-gram overlaps, CAPTURE focuses on the alignment of semantic content by parsing captions into structured scene graphs comprising objects, attributes, and relationships. Using the Factual T5-based parser (Li et al., 2023b), these elements are extracted and then matched across candidate and reference captions through a three-stage strategy involving exact, synonym, and soft matching. The final evaluation score is computed as a weighted sum of F1 scores for each element type. This benchmark is particularly effective for assessing fine-grained visual grounding in generated captions. We adopt the official implementation for our evaluation.

B.3 Details of Recaptioning Experiment with GPT-4o and Human Annotators

In § 4.2, we analyze the recaption behavior for captions generated by an MLLM. Specifically, we randomly select 100 images from CapsBench and prompt Qwen2-VL-7B-Instruct to generate initial captions. These captions, along with the corresponding images, are then input to GPT-4o, which is asked to refine the captions. For human recaptioning, we recruit two well-educated researchers proficient in English and instruct them to refine the captions. We verify that their edits are consistent and of high quality. The instruction provided to both GPT-4o and the human annotators is shown in Tab. 15. For our RICO, to ensure a fair comparison, we set the number of refinement iterations $N = 1$, i.e., each caption is refined only once.

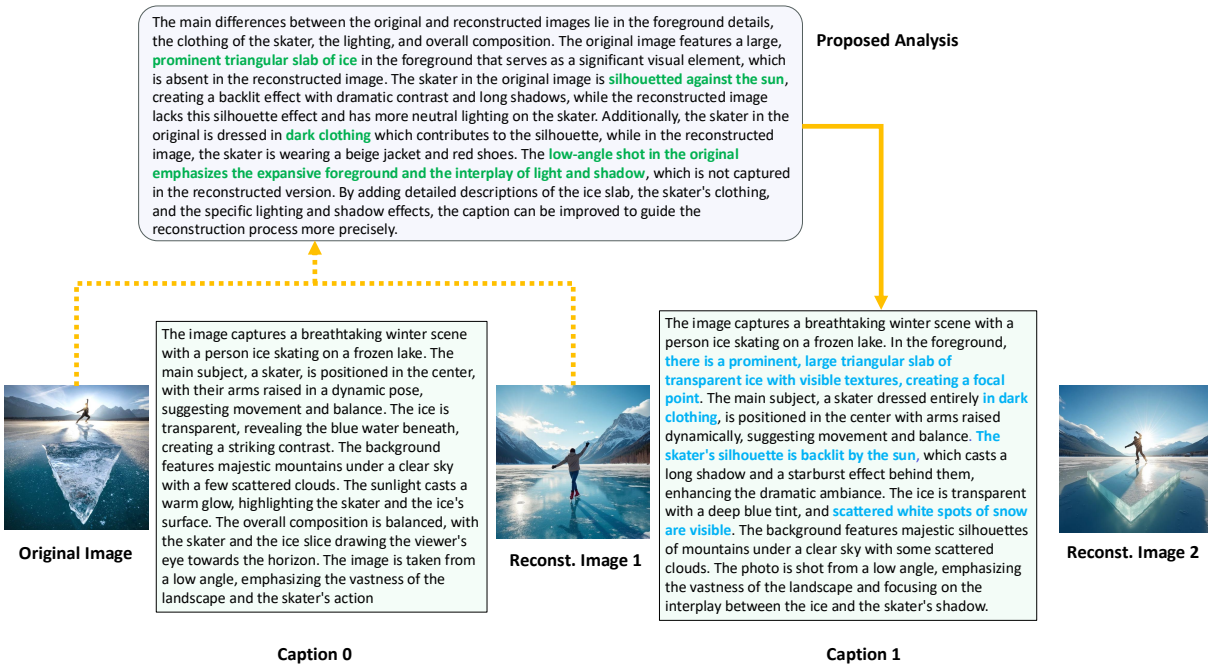

Figure 6: An example demonstrating the output analysis produced by our model, where green text highlights important aspects identified in the analysis, and blue text indicates information incorporated into the updated caption.

B.4 Details of Text-to-Image Generation

For the text-to-image generation experiment described in § 4.4.1, we adopt the FLUX.1-dev model (Labs, 2024). To accelerate training, we employ a LoRA-tuned (Hu et al., 2021) version of the model. The training dataset is primarily sourced from Hugging Face [2], and we collect a total of 30K images for our experiments. Training is conducted for 10,000 steps using 8 GPUs, each with a batch size of 10. The image resolution is set to 1024 × 1024. We use the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of 10^{-4}.

C More Implementation Details

C.1 Prompt in the Refinement Procedure

We provide the prompt used to query GPT-4o (OpenAI et al., 2024) for the refinement procedure described in § 3.3, as shown in Tab. 20.

C.2 Details of DPO Training

For training the DPO variant, we primarily use data from the DCE dataset (Sun et al., 2025), which spans a diverse range of image domains. From this dataset, we randomly sample 10K instances to construct preference pairs, as outlined in § 3.4.

For the DPO experiments, we use the `LLaMA-Factory` toolkit (Zheng et al., 2024). We initialize the model with Qwen2-VL and set the preference scaling parameter to $\beta = 0.1$. The model is fine-tuned for 3 epochs using 8 GPUs. The batch size is set to 64, and the learning rate is $\eta = 1.0 \times 10^{-5}$. We use a cutoff length of 2048 tokens and a warmup ratio of 0.1.

D Basics for DPO

Direct Preference Optimization (DPO) (Rafailov et al., 2024) formulates preference learning as a probabilistic binary classification task, without the need to train an explicit reward model. Given a dataset of preference tuples (x, y^+, y^-)—where x denotes a shared context (e.g., a prompt), and y^+ and y^- represent the preferred and dispreferred responses respectively–DPO aims to train a policy $\pi_\theta(y \mid x)$ such that:

$$\pi_\theta(y^+ \mid x) > \pi_\theta(y^- \mid x)$$

DPO defines an implicit reward function based on the log-likelihood ratio between the current policy π_θ and a fixed reference policy π_0 (e.g., the base model):

$$r(y \mid x) = \log \frac{\pi_\theta(y \mid x)}{\pi_0(y \mid x)}$$

[2] https://huggingface.co/datasets/jackyhate/text-to-image-2M

This leads to a binary classification objective that maximizes preference likelihood with KL regularization:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{(x,y^+,y^-)} \left[\log \frac{\exp(\beta r(y^+ \mid x))}{\exp(\beta r(y^+ \mid x)) + \exp(\beta r(y^- \mid x))} \right] - \mathrm{KL}(\pi \,\|\, \pi_0)$$

Substituting $r(y \mid x)$, the DPO training loss becomes:

$$\mathcal{L}_{\mathrm{DPO}} = -\mathbb{E}_{(x,y^+,y^-)} \left[\log \frac{\pi_\theta(y^+ \mid x)^\beta}{\pi_\theta(y^+ \mid x)^\beta + \pi_\theta(y^- \mid x)^\beta} \right]$$

This loss encourages the model to prefer y^+ over y^- while implicitly regularizing against the reference model π_0. Unlike traditional reinforcement learning, DPO requires no reward model sampling or rollouts, offering both stability and efficiency. More mathematical details can be found in the original paper of DPO.[3]

E Limitations

Our work still has several limitations. First, a key assumption of the proposed pipeline is that the text-to-image model must be sufficiently powerful to faithfully recover as many details as possible from the candidate caption. This places high demands on the capability of the text-to-image model. In this work, we adopt the FLUX model, which demonstrates strong performance, but still leaves room for significant improvement. Secondly, given the discrepancies between the original and reconstructed images, multiple plausible caption revisions may exist. Determining how to refine the caption in a concise yet effective manner remains a significant challenge for the refinement model. Lastly, the iterative version of our method is resource-intensive. Although we propose a DPO-based variant to mitigate this issue, reducing the coupling within the pipeline and improving inference efficiency remain important directions for future work.

F Ethical Consideration

Due to the multi-stage nature of our framework, any biases present in the underlying models—whether the T2I model or the MLLM—can propagate through and affect the final results. Additionally, as most state-of-the-art T2I models and MLLMs are optimized primarily for English, our system also tends to be limited in linguistic diversity. We are actively working on mitigating these issues in future iterations of our model.

[3] https://arxiv.org/pdf/2305.18290

========== PROMPT IN THE REFINEMENT PROCEDURE ==========

We are working on a project that involves generating captions for images and using these captions to reconstruct the images. The process follows these steps:

1. **Original Image (First Image):** A caption is generated based on this image.
2. **Reconstructed Image (Second Image):** The generated caption is used as input for a text-to-image model to create this image.

Your Task

Compare the **original** and **reconstructed** images, analyzing their differences to identify potential improvements for the original caption. Based on your observations, provide a **revised caption** that could enhance the reconstruction quality.

Guidelines for Comparison

- **Visual Details:** Color, shape, texture, and material of objects.
- **Composition & Layout:** Object positioning, spatial relationships, and overall scene structure.
- **Human Attributes (if applicable):** Pose, facial expression, skin tone, clothing, and hairstyle.
- **Perspective & Style:** Type of image, camera angle, depth of field, lighting, and artistic style.
- **Text in the Image:** Accuracy of any visible words, symbols, or signs.
- **Image Quality:** Blurriness, artifacts, or inconsistencies in object rendering.
- **World Knowledge:** Proper nouns or specific real-world references that should be preserved.
- **Color Aesthetics:** Color palette, grading, and overall mood consistency.

How to Improve the Caption

- **Add missing details** that were lost in reconstruction.
- **Clarify ambiguous descriptions** to provide more precise information.
- **Correct any inaccuracies** based on observed differences.
- **Specify key attributes** (e.g., "a red leather couch" instead of "a couch").

Your revised caption should aim to **reduce discrepancies** between the original and reconstructed images while maintaining a natural and informative description. You are encouraged to make the revised caption less than 512 tokens.

Now I provide the original image, reconstructed image, and the original caption: `{orig_caption}`.

Please give me the revised caption that you believe could enhance the reconstruction quality (i.e., make the new reconstructed image more like the original one at pixel level), enclosed with `<revised caption>`. And provide your analysis enclosed with `<analysis>` after.

Table 20: The prompt used to query GPT-4o in the refinement procedure.