Vinci: Deep Thinking in Text-to-Image Generation using Unified Model with Reinforcement Learning

Wang Lin[1,*], **Wentao Hu**[2,*], **Liyu Jia**[2], **Kaihang Pan**[1], **Majun Zhang**[1], **Zhou Zhao**[1], **Fei Wu**[1], **Jingyuan Chen**[1,†], **Hanwang Zhang**[2]
[1]Zhejiang University, [2]Nanyang Technological University
linwanglw@zju.edu.cn, jingyuanchen@zju.edu.cn

Abstract

With the continuous development of large language models and reasoning chain technologies, the potential of deep reasoning based on reinforcement learning has shown remarkable promise in multi-task scenarios. However, existing unified models have yet to achieve end-to-end integration in image generation and understanding tasks, limiting the model's self-reflection ability and the realization of cross-modal reasoning chains. To address this, we propose Vinci, a novel framework designed to enable interleaved image generation and understanding through deep reasoning capabilities. We leverage a small amount of multimodal chain-of-thought (MCoT) data for cold-start and employ reinforcement learning to guide the integration of image generation and understanding tasks. Additionally, we introduce a momentum-based reward function, which dynamically adjusts the reward distribution by considering historical improvements, ensuring the stability of the model across multiple generations. Experimental results demonstrate that integrating MCoT can achieve a +22% improvement over the base model on Geneval, effectively enhancing both image generation quality and instruction alignment capabilities.

1 Introduction

With the rapid development of large language models [1, 9, 82, 61] and Chain of Thought [49, 37, 75, 32] (CoT) techniques, deep thinking through reinforcement learning has become a key research focus in artificial intelligence. It has shown exceptional reasoning capabilities in tasks like mathematical problem-solving [62, 18] and code generation [29, 28, 61], significantly improving both model performance and interpretability by providing clear, step-by-step deduction and decision-making.

This step-by-step reasoning ability is being applied to cross-modal tasks, particularly in image generation and understanding. In image understanding, many structured reasoning mechanisms [85, 79] have been proposed to enhance controllability and interpretability. In image generation, [10, 14, 57] focuses on refining the input prompts or employing text-based layout construction prior to synthesis. Recently, breakthroughs in unified models have enabled a single model to simultaneously perform image generation and understanding. However, the potential of Chain of Thought techniques in existing unified models has yet to be fully explored.

The main challenge lies in the lack of end-to-end integration between image generation and understanding. For example, in models like Janus-Pro [5] and Show-o [78], images are represented in two ways within the model (*i.e.*, the model cannot directly understand the representations of the images it generates), thus limiting the model's ability to reflect on and optimize the generated results. Furthermore, while there have been attempts [89, 31, 22] to introduce CoT into these models, they are mostly limited to unimodal reasoning chains. For instance, in the T2I-R1 [31] model, semantic-level CoT is applied only to enhance text prompts, while token-level CoT is optimized only on image tokens.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

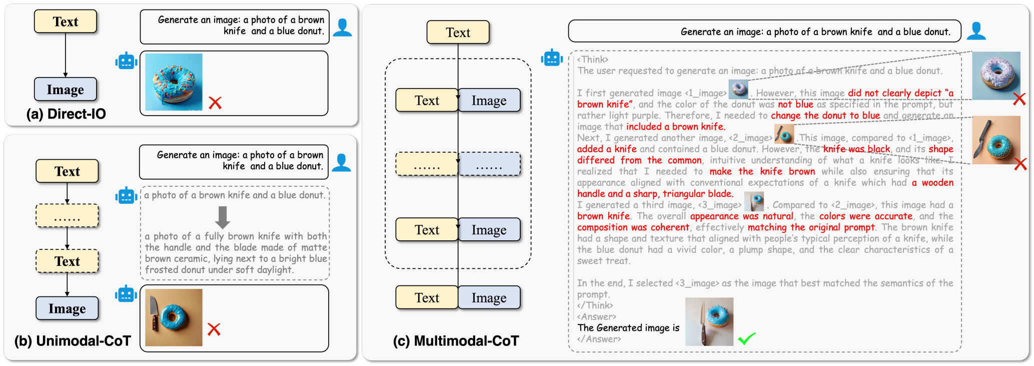

Figure 1: The illustration of CoT in text-to-image generation, showing three different approaches: (a) Direct-IO, where the image is generated directly from the prompt, (b) Unimodal-CoT, which involves generating an extended prompt before producing the image, and (c) Multimodal-CoT, where both text and image are processed iteratively to refine the image generation process based on the evolving understanding of the prompt.

This unimodal approach to reasoning chains restricts the model's ability to perform cross-modal deep thinking, further limiting its performance improvements.

Our goal is to integrate interleaved Multimodal Chain-of-Thought (MCoT) capabilities into the model, allowing it to observe and reflect on its generated results in real-time, thus achieving deep integration of image generation and understanding. As shown in Figure 1, initially, the model generates an image from the prompt *"a brown knife and a blue donut"*. Upon reviewing the first image, the model identifies that the donut is not blue and the knife is missing. It reflects on these issues and adjusts the generation strategy to correct the donut's color to blue and adds a knife, but the knife's color and shape still do not match typical expectations. Realizing the mistake, the model gives the knife's appearance description. This iterative process reflects the deep thinking of image generation and understanding, refining the generated images based on real-time evaluation and adjustments.

To achieve this, this paper introduces **Vinci**, which is the first image generation model capable of deep thinking. Specifically, we employ a progressive training strategy and fine-tune the model using a small amount of MCoT data to provide initial CoT guidance. Subsequently, we use reinforcement learning to guide the model to simultaneously focus on both image generation and image understanding tasks during the generation process. In addition to applying the reward model to evaluate the generated images, we introduce a momentum reward function to further enhance the model's stability and generation quality. The core of this method lies in dynamically distributing process rewards by considering the improvements made by the model in historical generation processes. By employing this approach, the Vinci can better balance the model's immediate performance with its long-term stability, preventing performance fluctuations or degradation over multiple iterations. Our contributions are summarized as follows:

- We propose **Vinci**, the first image generation model capable of deep reasoning, which integrates interleaved MCoT capabilities. This enables the model to not only generate images but also observe and reflect on its outputs in real-time.
- We propose a momentum-based reward function that dynamically distributes process rewards based on historical improvements. This approach enhances the model's stability while balancing immediate performance with long-term stability.
- Experimental results demonstrate that integrating MCoT can achieve a +22% improvement over the base model on Geneval [17] benchmarks.

2 Related Work

2.1 Unified Generation and Understanding LMM

Recent advancements in multimodal understanding and image generation have led to efforts to integrate these tasks into a unified framework. The main approaches can be categorized into two types. (1) Fused autoregressive and diffusion models such as Transfusion[88], Show-o[78], and MonoFormer [86], combine the strengths of autoregressive models for text generation with the high-quality image generation of diffusion models. Text is generated autoregressively, while images are produced through a multi-step denoising process. It balances symbolic control with visual fidelity, but fails to perform generation and understanding simultaneously. (2) Autoregressive models unify image and text generation within a sequential framework, using powerful autoregressive language models (e.g., LLaMA [1], Vicuna [9]) at their core. These models encode images into discrete visual tokens and process them alongside text tokens. Various image tokenization strategies, including pixel-based like LWM [50], Chameleon [65], and ANOLE [8], semantic-based like Emu [64], LaVIT [35], and DreamLLM [12], and learnable query encoding like SEED [15] and MetaQueries [55], allow the integration of visual information within the autoregressive framework. These autoregressive models demonstrate significant improvements in both image generation quality and efficiency. Note, we aim to perform generation and understanding simultaneously. Thus, the autoregressive models such as Chameleon [65] and Emu3 [71], which have a unified image representation, provide the foundation for MCoT in image generation.

2.2 Multimodal-CoT Reasoning

Multimodal Chain-of-Thought (MCoT) has gained significant attention in tasks such as Visual Question Answering (VQA), where early works like IPVR [6] and Multimodal-CoT [85] laid the foundation by generating intermediate rationales before final predictions. Subsequent research improved MCoT by introducing self-consistency [72] with word-level majority voting and dynamic reasoning selection based on human cognitive strategies. Frameworks like CoCoT [84] and RelationLMM [77] enhanced multi-image comprehension and object relationship modeling. Methods such as DDCoT [87] and Socratic Questioning [25] employed staged reasoning for better interpretability, while Chain-of-Spot [51] and DCoT [30] focused on region-of-interest analysis to improve contextual understanding. Recent advancements expanded MCoT beyond VQA. Techniques like G-CoT [52] and STIC [11] addressed data annotation limitations through automated augmentation, and DPMM-CoT [24] regenerated image features from the latent space to handle complex reasoning. Additionally, multimodal rationales, as seen in Visual-CoT [59] and MVoT [39], have enhanced reasoning across modalities, improving overall model performance. In image generation tasks, models like Prompt-CoT [83] and LayoutLLM-T2I [57] optimize input prompts and layout construction, respectively, to improve image synthesis quality. These developments highlight the growing importance of MCoT in multimodal reasoning, enabling models to tackle complex image-related tasks with greater efficiency and interpretability.

3 Method

3.1 Preliminary

Recently, reinforcement learning has become the primary method for unlocking the reasoning capabilities of large language models (LLMs). [62] introduces the Group Relative Policy Optimization (GRPO) framework. Unlike reinforcement learning algorithms such as PPO [60] that require a critic model to evaluate policy performance, GRPO compares groups of candidate responses directly, eliminating the need for an additional critic model. Given the input instruction q, GRPO first generates G distinct predictions $\{o_1, o_2, ..., o_g\}$ from the old policy $\pi_{\theta_{old}}$. Then, GRPO takes actions based on these predictions and denotes the obtained rewards as $\{r_1, r_2, ..., r_g\}$. By computing their mean and standard deviation for normalization, GRPO determines the relative quality of these responses:

$$A_i = \frac{r_i - \text{mean}(\{r_1, \ldots, r_g\})}{\text{std}(\{r_1, \ldots, r_g\})}, \tag{1}$$

where A_i represents the relative quality of the i-th answer. The GRPO method employs a clipped objective function, similar to PPO, and introduces a KL penalty term that compares the current policy π_θ with the reference model $\pi_{\theta_{\text{ref}}}$ into the loss, as follows:

$$\mathcal{L}_{\text{GRPO}}(\theta) = -\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \tag{2}$$

$$\left[\min\left(r_{i,t}(\theta)\hat{A}_{i,t},\ \text{clip}\left(r_{i,t}(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_{i,t}\right) - \beta\mathbb{D}_{KL}\left[\pi_\theta || \pi_{\text{ref}}\right]\right] \tag{3}$$

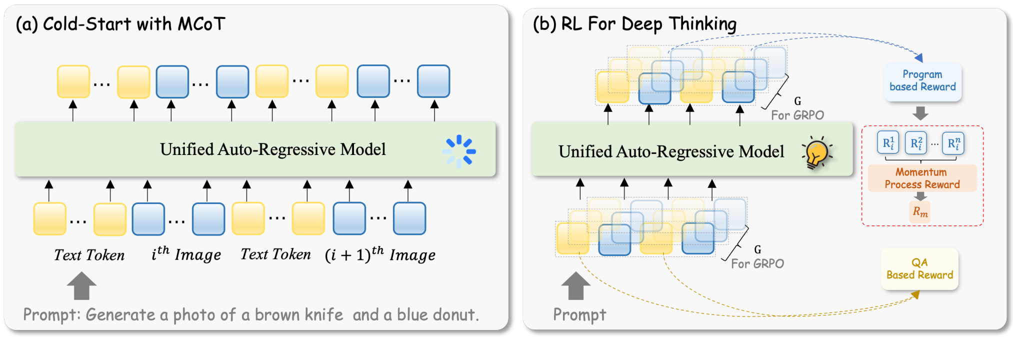

Figure 2: Overview of the Vinci model architecture. (a) Cold-start for long Multimodal Chain-of-Thought (MCoT), where the unified autoregressive model sequentially processes text tokens and image tokens. (b) Reinforcement Learning (RL) for deep thinking, where the model incorporates multiple reward signals (program-based, momentum process, and QA-based rewards) to iteratively refine the generated content during the generation process.

where $r_{i,t}(\theta)$ represents the ratio between the probabilities of π_θ and $\pi_{\theta_{\text{old}}}$ for generating the current token:

$$r_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t}|q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t}|q, o_{i,<t})}. \tag{4}$$

In text-to-image generation, the model is guided to follow a predefined template to produce the reasoning process and final images. The reward functions evaluate the generated image in visual RL and provide feedback for optimization.

3.2 Cold Start with Multimodal-CoT

Inspired by recent advances in reasoning models [19, 61, 53], we aim to incorporate a long-chain-of-thought reasoning into the unified models generation process as shown in Figure 2. However, existing unified models are typically designed to generate images in a single forward pass, lacking the ability to perform multi-round understanding and reflection in the image generation process. We first introduce a cold-start stage to adjust the model's output format and prevent instability of RL training during early-stage.

Multimodal-CoT Data Collection. Unlike Unimodal-CoT, which only refines the prompt without considering the generated image, the Multimodal-CoT we introduce refers to a process in which the model engages in understanding and reflecting on intermediate images during text-to-image generation. In each generation attempt, the model analyzes the generated image, identifies inconsistencies or areas for improvement, and proposes adjustments to better align with the prompt and enhance visual quality. This iterative, interleaved process of textual reasoning and visual feedback ultimately leads to the generation of the final, high-quality image.

To construct Multimodal-CoT data for the cold-start stage, we design a three-stage data pipeline.

- **Generate the Image for MCoT.** For each text query q, we generate a set of n candidate images, denoted as $\{p_1, p_2, \ldots, p_n\}$. We then apply an object detection model to each image to identify objects and obtain detection outputs, resulting in a set of tuples $\{(p_1, o_1), (p_2, o_2), \ldots, (p_n, o_n)\}$, where o_i represents the detected objects in image p_i.
- **Generate the Text for MCoT.** To prevent confusion in multi-image understanding, we first require the multimodal large model (MLLM) to independently evaluate each generated tuple (p_i, o_i) for its semantic alignment with the original query and overall quality. Note that we also provide the output of the object detection model o_i, which helps reduce hallucinations in the multimodal large model. In this process, MLLM generates a caption c_i and a score g_i for each image p_i. These outputs are then combined into annotated triples (p_i, c_i, g_i), which are sorted by their scores in ascending order, resulting in the final set $\{(p_1, c_1, g_1), (p_2, c_2, g_2), \ldots, (p_n, c_n, g_n)\}$, where $g_1 \leq g_2 \leq \ldots \leq g_n$.
- **MCoT Construction:** Given a predefined reasoning step count k, we randomly select k triples from this sorted set for combination. It is important to maintain the relative order

of the triples according to their scores and ensure that each combination contains at least one high-scoring (correct) triple, which serves as the endpoint for the MCoT. This results in combinations of the form $\{(p_i^1, c_i^1, g_i^1), \ldots, (p_i^k, c_i^k, g_i^k)\}$, where $k \leq n$. Each combination is then input into the MLLM to construct the final MCoT data.

For the final constructed MCoT, we use GPT-4o [53] to evaluate and filter out data that contains hallucinations or failures at the MCoT endpoints. Through these three stages, we construct an interleaved multimodal sequence that represents an iterative process of generation and reasoning.

3.3 Reinforcement Learning For Deep Thinking

Reward Functions Based on Comprehension Models The overall design philosophy of our reward model is to utilize comprehension models to evaluate both the generated image and text in RL. In this paper, we categorize the comprehension-based reward into two major types:

Program-based Reward Function. For the generated image, the comprehension model should understand and evaluate the consistency between the generated image and the textual prompt. Given the prompt like "*3 clocks and 1 dog*" and generated image, we can use visual detectors [4] to evaluate the generation quality. For example, we count the clocks based on the detector's confidence, returning 1 if the count is correct and 0 otherwise. Each prompt has its own item sets to be tested, and the average of the scores for each test is used as the reward score R_i.

QA-based Reward Function for Generated Text. For the generated text, the comprehension model should evaluate whether the text accurately describes the generated image. Given both the generated image and text, like in Figure 1 (c), we leverage the Multi-modal Large Language Models (MLLMs) to judge the quality of the evaluation text for the generated image. We require the model to be evaluated on three aspects: the completeness of the image description, whether it identifies issues in the image, and whether it includes strategies for improvement. Each aspect is rated on a scale of 0, 1, or 2, with the final reward score R_t being the average of the three scales.

Momentum Process Reward Function Given a sequence of n generated images $\{I_t\}_{t=1}^n$, let $s_t \in [0, 1]$ denote the quality score of the t-th image produced by the reward function. We aim to design a process reward function R_m that simultaneously evaluates both instantaneous quality and improvement dynamics.

Inspired by the Adam optimizer [36], we propose a dynamic process reward that is used to encourage consistent improvements as follows:

$$R_m = \frac{1}{n}\sum_{t=1}^{n} R_i(s_t) + \alpha \sum_{k=1}^{n-1} \frac{\Delta_k}{\sqrt{V_k} + \epsilon}\lambda^{k-1} \tag{5}$$

where the hyperparameter α balances between absolute quality and improvement momentum, the time decay factor $\lambda \in (0, 1)$ imposes attenuation on later improvements, and $\epsilon = 10^{-8}$ ensures numerical stability.

The current improvement $\Delta_k = \max(0, R_i(s_k) - R_i(s_{k-1}))$, quantifies non-negative increments between generated images. With the momentum normalization term $\Delta_k/\sqrt{V_k}$, which establishes dynamic scaling based on historical improvement patterns. This reward function achieves three objectives. First, it normalizes current improvements against historical volatility, preventing disproportionate rewards from isolated quality spikes. Second, the exponentially weighted variance calculation prioritizes recent trends while maintaining memory of long-term patterns. Third, the non-linear response curve generates superlinear rewards when sustained improvements exceed historical baselines ($\Delta_k \gg \sqrt{V_k}$), while penalizing inconsistent progress through variance accumulation.

The historical volatility $V_k = \gamma V_{k-1} + (1 - \gamma)\Delta_k^2$, evolves through exponential moving averaging. The momentum decay rate $\gamma \in [0, 1)$ controls historical variance adaptation. This formulation ensures older improvements contribute diminishing weights to the variance estimate, creating adaptive resistance to quality fluctuations.

The λ^{k-1} term implements exponential temporal decay to prioritize early-stage improvements, modeling the empirical observation that establishing baseline quality early enables more effective subsequent refinements. The decay schedule follows $\lambda^{k-1} = e^{(k-1)\ln\lambda}$, which creates implicit temporal milestones where early improvements receive more weighting, while late optimizations

Type	Method	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attr.	Overall
Gen. Only	PixArt-α [3]	0.98	0.50	0.44	0.80	0.8	0.70	0.48
	SDXL [56]	0.98	0.74	0.39	0.85	0.15	0.23	0.55
	FLUX.1-dev [38]	0.98	0.79	0.73	0.77	0.22	0.45	0.66
	DALL-E 3 [63]	0.96	0.87	0.47	0.83	0.43	0.45	0.67
	CogView4-6B [2]	0.99	0.86	0.66	0.79	0.48	0.58	0.73
	SD3-Medium [13]	0.99	0.94	0.72	0.89	0.33	0.60	0.74
Unified.	SEED-X [16]	0.97	0.58	0.26	0.80	0.19	0.14	0.49
	Emu3-Gen [71]	0.98	0.71	0.34	0.81	0.17	0.21	0.54
	TokenFlow-XL [58]	0.95	0.60	0.41	0.81	0.16	0.24	0.55
	Transfusion [88]	-	-	-	-	-	-	0.63
	D-DiT [41]	0.97	0.80	0.54	0.76	0.32	0.50	0.65
	Show-o [78]	0.98	0.80	0.66	0.84	0.31	0.50	0.68
	ILLUME+ [26]	0.99	0.88	0.62	0.84	0.42	0.53	0.72
	Infinity [23]	-	0.85	-	-	0.49	0.57	0.73
	Janus-Pro-7B [5]	0.99	0.89	0.59	0.90	0.79	0.66	0.80
	GPT-4o [53]	0.99	0.92	0.85	0.91	0.75	0.66	0.85
	Vinci(Ours)	0.99	0.86	0.48	0.83	0.86	0.54	0.76

Table 1: Evaluation of text-to-image generation ability on GenEval benchmark. Emu3-Gen is our base model. *Gen. Only* indicate models that can only do generation tasks, *Unified.* indicate models that can do both generation and understanding tasks

contribute minimally to the momentum score. The decay rate $\ln \lambda$ controls the transition steepness between these regimes.

To effectively guide the model's learning process and ensure that it can generate high-quality outputs, we combine the three reward functions as:

$$R = R_i + R_t + R_m \tag{6}$$

By integrating these three components, the overall reward function leads to more accurate, coherent, and consistent multimodal generation.

4 Experiments

4.1 Experimental Setup

Training Data. Our training data consists of two distinct phases. In the cold-start phase, we randomly generated 20,000 MCoT; the images are generated by Flux [38] and the text is generated by Qwen-VL [68]. Following this, the reinforcement learning (RL) phase began with the utilization of 30,000 prompts, all without corresponding images. Notably, to ensure the integrity of the training and evaluation processes, we performed careful deduplication of the training prompts, eliminating any overlap with the test set. This measure was critical in preventing data leakage and ensuring a robust evaluation of the model's performance on unseen prompts.

Implementation details. For our training, we adopted Emu-Gen as the base model, which has a unified image representation. The model's context length was extended to 15,360 tokens, which allowed for processing longer sequences of text and image pairs and supported approximately three iteration within each context window. The learning rate was set to 1e-5 and a beta of 0.01.

4.2 Main Results

The experimental results presented in Table 1 highlight the performance of various models on the GenEval [17] benchmark, which evaluates the text-to-image generation ability across different tasks. The results highlight the superior performance of Vinci over the *Gen. Only* models, demonstrating the necessity of integrating comprehension abilities into generative models for enhanced instruction following capabilities.

In comparison with the *Unified* models, Vinci only lags behind Janus-Pro-7B [5] and GPT-4o [53], indicating its strong competitiveness among models capable of both generation and understanding tasks. The enhancements in Vinci over the baseline model Emu3-Gen [71] are particularly significant across several dimensions. Specifically, Vinci elevates the Position generation score from 0.17 to 0.86

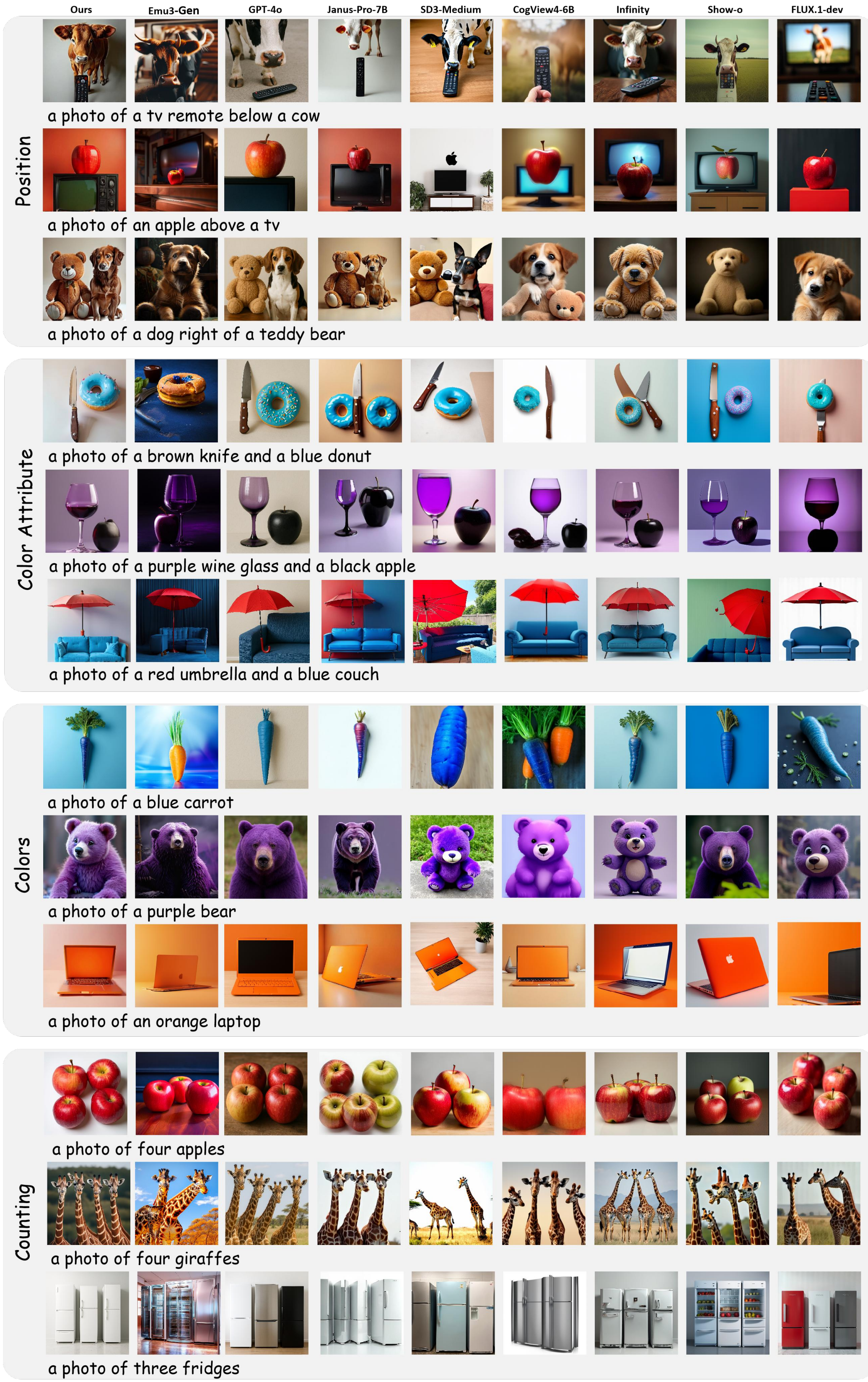

Figure 3: Qualitative experimental results of Vinci. Compared to existing text-to-image generation models, the images generated by Vinci demonstrate better alignment with the given prompts.

Method	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attr.	Overall
Emu3-Gen [71]	0.98	0.71	0.34	0.81	0.17	0.21	0.54
$w.R_t$	0.98	0.72	0.31	0.77	0.51	0.30	0.59
$w.R_i$	0.96	0.73	0.36	0.75	0.72	0.44	0.66
$w.R_i\&R_t$	0.99	0.84	0.44	0.79	0.75	0.46	0.71
$w.R_i\&R_t\&R_m$	0.99	0.86	0.48	0.83	0.86	0.54	0.76

Table 2: Ablation study results on the GenEval benchmark. We compare the performance of the base model Emu3-Gen and its variants with modified reward functions.

(+0.69) and Color attribution score from 0.21 to 0.54 (+0.33). These improvements underscore the efficacy of the Multimodal-CoT in achieving a deep integration of image generation and understanding. By enabling the model to observe and reflect on its generated results in real-time, Vinci not only generates images but also enhances the interpretability and controllability of the generated content.

Furthermore, the results in Figure 3 show that Vinci demonstrates a robust performance across a variety of tasks. For instance, when tasked with generating an image of a "tv remote below a cow" or an "apple above a tv" Vinci successfully positions the objects as per the instructions, showcasing its ability to understand and execute complex instructions involving multiple objects.

4.3 Ablation Study

The ablation study results presented in Table 2 provide valuable insights into the impact of different reward functions on the performance of the Vinci model.

Image understanding ability can feed back into image generation. Initially, when the model is trained using only the text quality evaluation through R_t, we observe improvements in image generation, particularly in areas such as position and color attribution. This result suggests that improving the model's ability to understand the generated image can have a positive effect on the image generation process itself.

Explicit image quality evaluation is important. By providing explicit feedback on the image quality, R_i acts as a direct optimization signal, guiding the model to produce higher-quality images. The explicit evaluation of image fidelity ensures that the generated images are not only contextually accurate but also visually compelling. The model is therefore able to focus more on producing realistic and visually appealing images. The combined use of both image and text reward functions (*w.* $R_i\&R_t$) results in additional performance gains.

Stable iteration process leads to further improvement. The final enhancement comes from the incorporation of a momentum-based process reward function R_m, which takes into account historical improvements over time. This mechanism encourages the model to maintain a steady and consistent progression in its learning process. It draws parallels with human cognitive processes, where sustained, incremental progress is highly valued. By emphasizing gradual improvement, the momentum-based reward ensures that the model continuously refines its understanding and generation abilities over time, rather than oscillating between high-performance peaks and low-performance troughs. This method not only leads to enhanced model performance but also stabilizes the training trajectory, making the reinforcement learning process more reliable and robust in the long term.

4.4 Visualization of Multimodal-COT

The visualizations in Figure 4 demonstrate the adaptive nature of Vinci, which allows the model to autonomously control the length of the reasoning process. This capability enables the model to directly produce the final image after a single round of thought for simpler cases, while more complex scenarios may require two to three iterations of contemplation to yield the desired output. When tasked with creating an image of a blue vase and a black banana, the model's initial attempt missed the banana and had an unappealing vase shape. Recognizing these shortcomings, the model engaged in further reasoning to introduce the banana and enhance the vase's aesthetics. It continued this process until it achieved an image that not only met the user's specifications but also exhibited high visual quality.

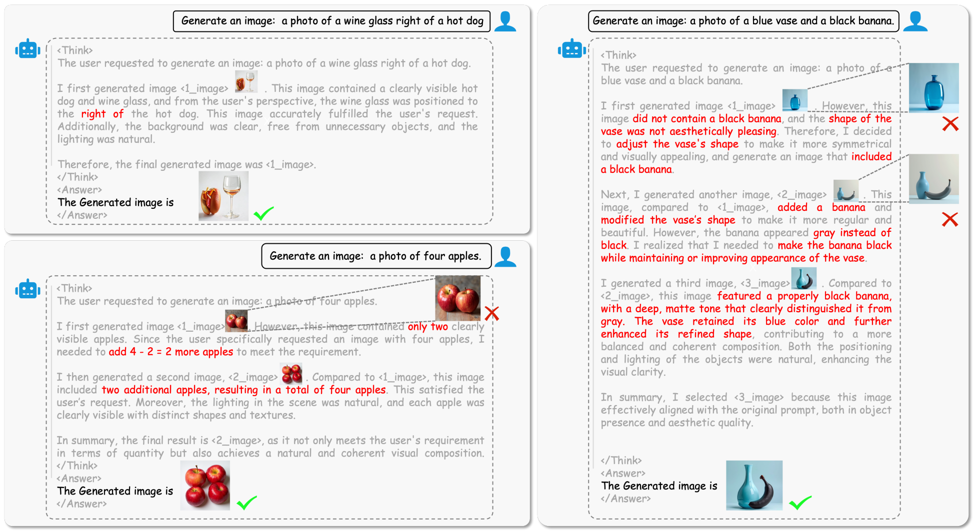

Figure 4: Case study demonstrating the variable-length MCoT generation process. For each query, the model iteratively generates and refines images based on feedback from the reasoning process.

This variable-length reasoning approach enhances both the output's fidelity and the process's overall efficiency. By allowing the model to determine the appropriate depth of thought required for each task, it ensures that while complex images receive the detailed consideration needed to meet high standards, simpler images are generated swiftly without unnecessary computational overhead.

5 Limitation and Future Work

There remains considerable room for exploration in the unified model for multimodal understanding and generation. Currently, the community lacks a truly general-purpose foundational model that can seamlessly integrate cross-modal understanding and generation capabilities. Although state-of-the-art models like Janus-Pro strive to balance these capabilities, they sacrifice a unified representation of understanding and generation at the foundational level. Additionally, the length of image tokens (which often requires hundreds or more tokens to represent a single image), combined with the limitations of the context window in autoregressive models, restricts the depth and efficiency of the model's reasoning capabilities. Future work can focus on improving both the understanding and generation capabilities of unified models, as well as developing better reasoning strategies to optimize efficiency and performance.

6 Conclusion

In this paper, we introduced Vinci, a novel image generation model that integrates deep reasoning capabilities through the use of interleaved multimodal Chain-of-Thought (MCoT). Unlike previous models that primarily focus on either image generation or understanding, Vinci achieves a unified approach that combines these two tasks in a way that allows the model to reflect on and iteratively improve its generated outputs in real time. Our method leverages reinforcement learning, where the model learns to refine its output based on the feedback from reward model. During the iterative generation process, we introduce a momentum-based process reward function that effectively balances exploration and exploitation. Experimental results demonstrate that Vinci significantly improves both image generation and instruction-following abilities, achieving a remarkable +22% performance improvement over the base model. This establishes Vinci as a new paradigm for reasoning-centric generative systems.

7 Acknowledgements

This work was supported by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang (2025C02022) and the National Natural Science Foundation of China (No.62037001, No.62307032).

References

[1] Meta AI. Llama: Open and efficient foundation language models. `https://www.llama.com/`. Accessed: 2025-04-07.

[2] Zhipu AI. Cogview4: Next-generation image creation. `https://cogview4.net/`, 2025.

[3] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023.

[4] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

[5] 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.

[6] Zhenfang Chen, Qinhong Zhou, Yikang Shen, Yining Hong, Hao Zhang, and Chuang Gan. See, think, confirm: Interactive prompting between vision and language models for knowledge-based visual reasoning. *arXiv preprint arXiv:2301.05226*, 2023.

[7] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1290–1299, 2022.

[8] Ethan Chern, Jiadi Su, Yan Ma, and Pengfei Liu. Anole: An open, autoregressive, native large multimodal models for interleaved image-text generation. *arXiv preprint arXiv:2407.06135*, 2024.

[9] 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. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[10] Jaemin Cho, Abhay Zala, and Mohit Bansal. Visual programming for step-by-step text-to-image generation and evaluation. *Advances in Neural Information Processing Systems*, 36:6048–6069, 2023.

[11] Yihe Deng, Pan Lu, Fan Yin, Ziniu Hu, Sheng Shen, Quanquan Gu, James Y Zou, Kai-Wei Chang, and Wei Wang. Enhancing large vision language models with self-training on image comprehension. *Advances in Neural Information Processing Systems*, 37:131369–131397, 2024.

[12] Runpei Dong, Chunrui Han, Yuang Peng, Zekun Qi, Zheng Ge, Jinrong Yang, Liang Zhao, Jianjian Sun, Hongyu Zhou, Haoran Wei, et al. Dreamllm: Synergistic multimodal comprehension and creation. *arXiv preprint arXiv:2309.11499*, 2023.

[13] 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*, 2024.

[14] Weixi Feng, Wanrong Zhu, Tsu-jui Fu, Varun Jampani, Arjun Akula, Xuehai He, Sugato Basu, Xin Eric Wang, and William Yang Wang. Layoutgpt: Compositional visual planning and generation with large language models. *Advances in Neural Information Processing Systems*, 36:18225–18250, 2023.

[15] Yuying Ge, Yixiao Ge, Ziyun Zeng, Xintao Wang, and Ying Shan. Planting a seed of vision in large language model. *arXiv preprint arXiv:2307.08041*, 2023.

[16] 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.

[17] Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework for evaluating text-to-image alignment. *Advances in Neural Information Processing Systems*, 36:52132–52152, 2023.

[18] Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. *arXiv preprint arXiv:2501.04519*, 2025.

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

[20] Zirun Guo, Tao Jin, Wenlong Xu, Wang Lin, and Yangyang Wu. Bridging the gap for test-time multimodal sentiment analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 16987–16995, 2025.

[21] Zirun Guo, Shulei Wang, Wang Lin, Weicai Yan, Yangyang Wu, and Tao Jin. Efficient prompting for continual adaptation to missing modalities. *arXiv preprint arXiv:2503.00528*, 2025.

[22] Ziyu Guo, Renrui Zhang, Chengzhuo Tong, Zhizheng Zhao, Peng Gao, Hongsheng Li, and Pheng-Ann Heng. Can we generate images with cot? let's verify and reinforce image generation step by step. *arXiv preprint arXiv:2501.13926*, 2025.

[23] Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis, 2024.

[24] Liqi He, Zuchao Li, Xiantao Cai, and Ping Wang. Multi-modal latent space learning for chain-of-thought reasoning in language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18180–18187, 2024.

[25] Wanpeng Hu, Haodi Liu, Lin Chen, Feng Zhou, Changming Xiao, Qi Yang, and Changshui Zhang. Socratic questioning: Learn to self-guide multimodal reasoning in the wild. *arXiv preprint arXiv:2501.02964*, 2025.

[26] Runhui Huang, Chunwei Wang, Junwei Yang, Guansong Lu, Yunlong Yuan, Jianhua Han, Lu Hou, Wei Zhang, Lanqing Hong, Hengshuang Zhao, and Hang Xu. Illume+: Illuminating unified mllm with dual visual tokenization and diffusion refinement. *arXiv preprint arXiv:2504.01934*, 2025.

[27] Zihan Huang, Tao Wu, Wang Lin, Shengyu Zhang, Jingyuan Chen, and Fei Wu. Autogeo: Automating geometric image dataset creation for enhanced geometry understanding. *arXiv preprint arXiv:2409.09039*, 2024.

[28] Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

[29] Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024.

[30] Zixi Jia, Jiqiang Liu, Hexiao Li, Qinghua Liu, and Hongbin Gao. Dcot: Dual chain-of-thought prompting for large multimodal models. In *The 16th Asian Conference on Machine Learning (Conference Track)*, 2024.

[31] Dongzhi Jiang, Ziyu Guo, Renrui Zhang, Zhuofan Zong, Hao Li, Le Zhuo, Shilin Yan, Pheng-Ann Heng, and Hongsheng Li. T2i-r1: Reinforcing image generation with collaborative semantic-level and token-level cot. *arXiv preprint arXiv:2505.00703*, 2025.

[32] Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanwei Li, Yu Qi, Xinyan Chen, Liuhui Wang, Jianhan Jin, Claire Guo, Shen Yan, et al. Mme-cot: Benchmarking chain-of-thought in large multimodal models for reasoning quality, robustness, and efficiency. *arXiv preprint arXiv:2502.09621*, 2025.

[33] Tao Jin, Wang Lin, Hao Jiang, Jian Wang, Xiao Jin, Zhimeng Zhang, Jingyuan Chen, Zhou Zhao, and Zhongfei Zhang. Recognize-and-tell: Generating video captions with textual cue in scene. *Expert Systems with Applications*, page 127831, 2025.

[34] Tao Jin, Wang Lin, Ye Wang, Linjun Li, Xize Cheng, and Zhou Zhao. Rethinking the multimodal correlation of multimodal sequential learning via generalizable attentional results alignment. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5247–5265, 2024.

[35] Yang Jin, Kun Xu, Kun Xu, Liwei Chen, Chao Liao, Jianchao Tan, Yadong Mu, et al. Unified language-vision pretraining in llm with dynamic discrete visual tokenization. In *International Conference on Learning Representations*, 2024.

[36] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[37] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

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

[39] Chengzu Li, Wenshan Wu, Huanyu Zhang, Yan Xia, Shaoguang Mao, Li Dong, Ivan Vulić, and Furu Wei. Imagine while reasoning in space: Multimodal visualization-of-thought. *arXiv preprint arXiv:2501.07542*, 2025.

[40] Linjun Li, Tao Jin, Wang Lin, Hao Jiang, Wenwen Pan, Jian Wang, Shuwen Xiao, Yan Xia, Weihao Jiang, and Zhou Zhao. Multi-granularity relational attention network for audio-visual question answering. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[41] Zijie Li, Henry Li, Yichun Shi, Amir Barati Farimani, Yuval Kluger, Linjie Yang, and Peng Wang. Dual diffusion for unified image generation and understanding. *arXiv preprint arXiv:2501.00289*, 2024.

[42] Wang Lin, Jingyuan Chen, Jiaxin Shi, Zirun Guo, Yichen Zhu, Zehan Wang, Tao Jin, Zhou Zhao, Fei Wu, YAN Shuicheng, et al. Action imitation in common action space for customized action image synthesis. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

[43] Wang Lin, Jingyuan Chen, Jiaxin Shi, Yichen Zhu, Chen Liang, Junzhong Miao, Tao Jin, Zhou Zhao, Fei Wu, Shuicheng Yan, et al. Non-confusing generation of customized concepts in diffusion models. *arXiv preprint arXiv:2405.06914*, 2024.

[44] Wang Lin, Yueying Feng, WenKang Han, Tao Jin, Zhou Zhao, Fei Wu, Chang Yao, and Jingyuan Chen. Exploring embodied emotion through a large-scale egocentric video dataset. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

[45] Wang Lin, Liyu Jia, Wentao Hu, Kaihang Pan, Zhongqi Yue, Wei Zhao, Jingyuan Chen, Fei Wu, and Hanwang Zhang. Reasoning physical video generation with diffusion timestep tokens via reinforcement learning. *arXiv preprint arXiv:2504.15932*, 2025.

[46] Wang Lin, Tao Jin, Wenwen Pan, Linjun Li, Xize Cheng, Ye Wang, and Zhou Zhao. Tavt: Towards transferable audio-visual text generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14983–14999, 2023.

[47] Wang Lin, Tao Jin, Ye Wang, Wenwen Pan, Linjun Li, Xize Cheng, and Zhou Zhao. Exploring group video captioning with efficient relational approximation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15281–15290, 2023.

[48] Wang Lin, QingSong Wang, Yueying Feng, Shulei Wang, Tao Jin, Zhou Zhao, Fei Wu, Chang Yao, and Jingyuan Chen. Non-natural image understanding with advancing frequency-based vision encoders. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29756–29766, 2025.

[49] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[50] Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with blockwise ringattention. *arXiv preprint arXiv:2402.08268*, 2024.

[51] Zuyan Liu, Yuhao Dong, Yongming Rao, Jie Zhou, and Jiwen Lu. Chain-of-spot: Interactive reasoning improves large vision-language models. *arXiv preprint arXiv:2403.12966*, 2024.

[52] Yingzi Ma, Yulong Cao, Jiachen Sun, Marco Pavone, and Chaowei Xiao. Dolphins: Multimodal language model for driving. In *European Conference on Computer Vision*, pages 403–420. Springer, 2024.

[53] OpenAI. Introducing 4o image generation. `https://openai.com/index/introducing-4o-image-generation/`, 2025. Accessed: 2025-04-22.

[54] Kaihang Pan, Wang Lin, Zhongqi Yue, Tenglong Ao, Liyu Jia, Wei Zhao, Juncheng Li, Siliang Tang, and Hanwang Zhang. Generative multimodal pretraining with discrete diffusion timestep tokens. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26136–26146, 2025.

[55] Xichen Pan, Satya Narayan Shukla, Aashu Singh, Zhuokai Zhao, Shlok Kumar Mishra, Jialiang Wang, Zhiyang Xu, Jiuhai Chen, Kunpeng Li, Felix Juefei-Xu, et al. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025.

[56] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[57] Leigang Qu, Shengqiong Wu, Hao Fei, Liqiang Nie, and Tat-Seng Chua. Layoutllm-t2i: Eliciting layout guidance from llm for text-to-image generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 643–654, 2023.

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

[59] Daniel Rose, Vaishnavi Himakunthala, Andy Ouyang, Ryan He, Alex Mei, Yujie Lu, Michael Saxon, Chinmay Sonar, Diba Mirza, and William Yang Wang. Visual chain of thought: bridging logical gaps with multimodal infillings. *arXiv preprint arXiv:2305.02317*, 2023.

[60] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[61] ByteDance Seed, Yufeng Yuan, Yu Yue, Mingxuan Wang, Xiaochen Zuo, Jiaze Chen, Lin Yan, Wenyuan Xu, Chi Zhang, Xin Liu, et al. Seed-thinking-v1. 5: Advancing superb reasoning models with reinforcement learning. *arXiv preprint arXiv:2504.13914*, 2025.

[62] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[63] Zhan Shi, Xu Zhou, Xipeng Qiu, and Xiaodan Zhu. Improving image captioning with better use of captions. *arXiv preprint arXiv:2006.11807*, 2020.

[64] Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. Emu: Generative pretraining in multimodality. *arXiv preprint arXiv:2307.05222*, 2023.

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

[66] Bohan Wang, Zhongqi Yue, Fengda Zhang, Shuo Chen, Li'an Bi, Junzhe Zhang, Xue Song, Kennard Yanting Chan, Jiachun Pan, Weijia Wu, et al. Selftok: Discrete visual tokens of autoregression, by diffusion, and for reasoning. *arXiv preprint arXiv:2505.07538*, 2025.

[67] Dongsheng Wang, Jiequan Cui, Miaoge Li, Wang Lin, Bo Chen, and Hanwang Zhang. Instruction tuning-free visual token complement for multimodal llms. In *European Conference on Computer Vision*, pages 446–462. Springer, 2024.

[68] 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. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[69] Shulei Wang, Wang Lin, Hai Huang, Hanting Wang, Sihang Cai, WenKang Han, Tao Jin, Jingyuan Chen, Jiacheng Sun, Jieming Zhu, et al. Towards transformer-based aligned generation with self-coherence guidance. *arXiv preprint arXiv:2503.17675*, 2025.

[70] Shulei Wang, Shuai Yang, Wang Lin, Zirun Guo, Sihang Cai, Hai Huang, Ye Wang, Jingyuan Chen, and Tao Jin. Omni-chart-600k: A comprehensive dataset of chart types for chart understanding. *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4051–4069, 2025.

[71] Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

[72] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[73] Ye Wang, Tao Jin, Wang Lin, Xize Cheng, Linjun Li, and Zhou Zhao. Semantic-conditioned dual adaptation for cross-domain query-based visual segmentation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9797–9815, 2023.

[74] Ye Wang, Wang Lin, Shengyu Zhang, Tao Jin, Linjun Li, Xize Cheng, and Zhou Zhao. Weakly-supervised spoken video grounding via semantic interaction learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10914–10932, 2023.

[75] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[76] Tao Wu, Mengze Li, Jingyuan Chen, Wei Ji, Wang Lin, Jinyang Gao, Kun Kuang, Zhou Zhao, and Fei Wu. Semantic alignment for multimodal large language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3489–3498, 2024.

[77] Chi Xie, Shuang Liang, Jie Li, Zhao Zhang, Feng Zhu, Rui Zhao, and Yichen Wei. Relationlmm: Large multimodal model as open and versatile visual relationship generalist. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[78] 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. *arXiv preprint arXiv:2408.12528*, 2024.

[79] Guowei Xu, Peng Jin, Li Hao, Yibing Song, Lichao Sun, and Li Yuan. Llava-o1: Let vision language models reason step-by-step. *arXiv preprint arXiv:2411.10440*, 2024.

[80] Weicai Yan, Wang Lin, Zirun Guo, Ye Wang, Fangming Feng, Xiaoda Yang, Zehan Wang, and Tao Jin. Diff-prompt: Diffusion-driven prompt generator with mask supervision. In *The Thirteenth International Conference on Learning Representations*, 2025.

[81] Weicai Yan, Ye Wang, Wang Lin, Zirun Guo, Zhou Zhao, and Tao Jin. Low-rank prompt interaction for continual vision-language retrieval. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 8257–8266, 2024.

[82] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[83] Junyi Yao, Yijiang Liu, Zhen Dong, Mingfei Guo, Helan Hu, Kurt Keutzer, Li Du, Daquan Zhou, and Shanghang Zhang. Promptcot: Align prompt distribution via adapted chain-of-thought. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7027–7037, 2024.

[84] Daoan Zhang, Junming Yang, Hanjia Lyu, Zijian Jin, Yuan Yao, Mingkai Chen, and Jiebo Luo. Cocot: Contrastive chain-of-thought prompting for large multimodal models with multiple image inputs. *arXiv preprint arXiv:2401.02582*, 2024.

[85] Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2023.

[86] Chuyang Zhao, Yuxing Song, Wenhao Wang, Haocheng Feng, Errui Ding, Yifan Sun, Xinyan Xiao, and Jingdong Wang. Monoformer: One transformer for both diffusion and autoregression. *arXiv preprint arXiv:2409.16280*, 2024.

[87] Ge Zheng, Bin Yang, Jiajin Tang, Hong-Yu Zhou, and Sibei Yang. Ddcot: Duty-distinct chain-of-thought prompting for multimodal reasoning in language models. *Advances in Neural Information Processing Systems*, 36:5168–5191, 2023.

[88] Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024.

[89] Le Zhuo, Liangbing Zhao, Sayak Paul, Yue Liao, Renrui Zhang, Yi Xin, Peng Gao, Mohamed Elhoseiny, and Hongsheng Li. From reflection to perfection: Scaling inference-time optimization for text-to-image diffusion models via reflection tuning. *arXiv preprint arXiv:2504.16080*, 2025.

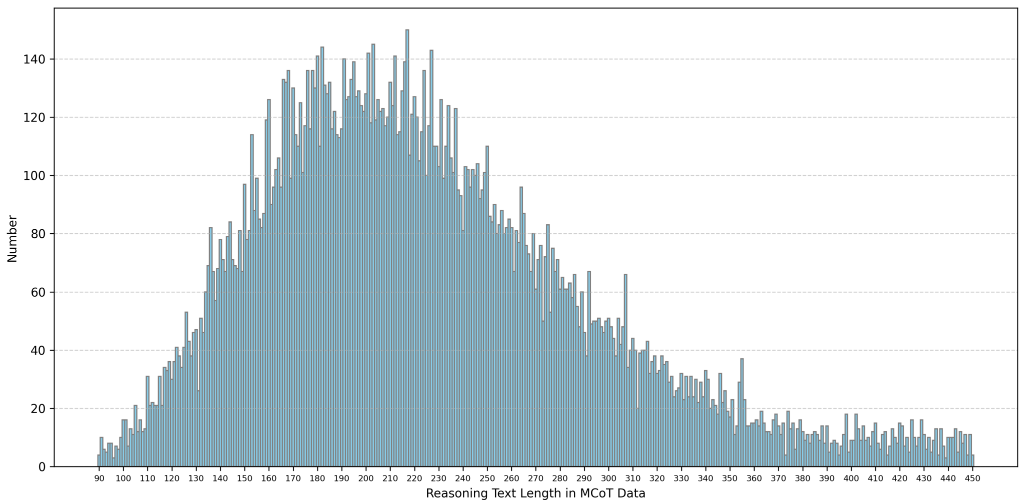

Figure 5: Distribution of reasoning text lengths in our MCoT dataset

A Dataset Construction

A.1 MCoT Construction Pipeline

To construct our Multimodal-CoT(MCoT) dataset, we designed a multi-stage pipeline comprising Image Generation for MCoT, Text Generation for MCoT, and MCoT Construction. Given a text query q, we first generated a set of n candidate images $\{p_1, p_2, ..., p_n\}$ using FLUX.1-dev [38]. To identify objects, we applied Mask2Former [7] to each image, yielding detection outputs o_i and forming the set of tuples $\{(p_1, o_1), (p_2, o_2), ..., (p_n, o_n)\}$, where o_i denoted the objects detected in image p_i. Then, to mitigate ambiguity in multi-image interpretation, we tasked Qwen-VL [68] with independently evaluating each tuple (p_i, o_i) for its semantic alignment with the original query and overall quality. The detection result o_i was explicitly provided to the model to reduce hallucinations. During this evaluation, the model produced a caption c_i and a quality score g_i, resulting in annotated triples (p_i, c_i, g_i). These were sorted in ascending order by g_i to yield a ranked sequence of annotated samples. Finally, to construct the MCoT data, we randomly selected k triples from the sorted list while preserving their relative score order. Each selected set must contain at least one high-quality (i.e., high-scoring) triple, which served as the final reasoning step. The resulting combination $\{(p_i^1, c_i^1, g_i^1), ..., (p_i^k, c_i^k, g_i^k)\}$ was input into Qwen-VL to construct the final MCoT data. The prompts used by Qwen-VL during the construction process are shown in Table 4.

The final MCoT dataset, consisting of 20,000 samples, comprises three types: 1-step MCoT, 2-step MCoT, and 3-step MCoT, which correspond to reasoning processes involving 1, 2, and 3 steps before generating the final image, respectively. Examples of these three types are shown in Table 3.

A.2 Distribution of Reasoning Text Lengths

A key characteristic of MCoT data is the ability to clearly understand and articulate the content of images generated throughout the reasoning process. Therefore, the length of the accompanying text is an important aspect to consider. The distribution of reasoning text lengths in our constructed MCoT data is shown in Figure 5.

B QA-based Reward for Generated Text

In our research, we designed a QA-based reward function to evaluate the quality of the generated text in the context of text-to-image generation. This reward function assesses whether the generated text accurately describes the generated image, identifies any issues in the image, and provides strategies for improvement. In Table 5, we provide examples of the prompts used to guide the evaluation process.

MCoT Data Type	Example
1-step MCoT	<think> The user asked to generate an image: a photo of two tennis rackets. One possible image was <1_image>. The image clearly reflected the prompt, showing two rackets placed side by side with distinct yet harmonious designs. The rackets were rendered in sharp detail, with red and black colors creating a vivid contrast. A soft, neutral background kept focus on the objects. Lighting and composition gave the image a polished, professional look, making it a faithful and visually appealing realization of the prompt. Overall, <1_image> was an excellent realization of the prompt, and I selected it as answer because it not only met the basic requirements but also elevated the concept through its high-quality execution and attention to detail. </think> <answer><1_image></answer>
2-step MCoT	<think> The user asked to generate an image: a photo of a sandwich below a sports ball. One possible image was <1_image>. While <1_image> did present a sandwich with a sports ball above it, the sandwich appeared somewhat simplistic and lacked detailed texturing. I needed to generate an image that clearly showed a sandwich below a sports ball, with the sandwich rendered in a more realistic and aesthetically pleasing way. Then, I generated <2_image>, which offered a more refined and visually compelling representation of the prompt. The image clearly showed a sports ball above a sandwich, and their spatial relationship was accurate. The sandwich in <2_image> had a more realistic texture, with a visibly toasted bun that added depth and detail. The layers, including lettuce and other fillings, were vibrant and well-defined. In summary, <2_image> correctly depicted a sandwich below a sports ball with a clear spatial arrangement. Both the sandwich and the sports ball featured improved textures and fine-grained details, contributing to the overall quality. These factors made <2_image> the preferred choice over <1_image>. </think> <answer><2_image></answer>
3-step MCoT	<think> The user asked to generate an image: a photo of an orange backpack and a green spoon. One possible image was <1_image>. However, it did not include an orange backpack as expected, which was a critical element of the prompt. I needed to generate an image that included an orange backpack. I then generated <2_image>, which included the required orange backpack and aligned better with the prompt. While <2_image> was an improvement, its color scheme was relatively dull, and the background did not effectively complement the orange backpack. I needed to make the colors more vibrant. Finally, I generated <3_image>, which featured a more vibrant and consistent color palette. The orange backpack and green spoon stood out clearly against a light green background. The lighting was evenly distributed, enhancing the textures and colors of the objects, which appeared more realistic and detailed. In summary, I selected <3_image> as the final result because it accurately captured both the orange backpack and the green spoon with a clear and visually pleasing composition. </think> <answer><3_image></answer>

Table 3: Introduction of the three types of MCoT data used in our cold-start stage. <1_image>, <2_image>, and <3_image> represent images generated after 1, 2, and 3 rounds of reasoning, respectively. The corresponding visual tokens are omitted for brevity.

In our implementation, we used a multi-modal large language model (MLLM) to evaluate the generated text based on the prompts provided above. The scores and explanations provided by the MLLM were then used as part of the overall reward function to guide the training of our text-to-image generation model. By incorporating this QA-based reward function, we aimed to enhance the model's ability to generate high-quality, contextually accurate, and self-reflective image descriptions, thereby improving the overall performance of the text-to-image generation process.

C Implementation Details

C.1 Training Details

For our training, we adopted Emu3-Gen [71] as the base model, which has a unified image representation. The model's context length was extended to 15,360 tokens, which allowed for processing longer sequences of text and image pairs and supported approximately three iterations within each context window. The learning rate was set to 1e-5 and a beta of 0.01.

During the training, we utilized 16 A800 GPUs. The training was divided into two stages. In the first stage, known as the cold start, we set the batch size to 64 and trained for approximately 20 hours, allowing the model to learn the fundamental features and patterns of the data. In the second stage,

Stage	Prompt Example
Text Generation	You are tasked with generating a caption for a given image and its detection results, and evaluating how well the image aligns with the original prompt in terms of semantic accuracy and generation quality. Your task involves the following steps: 1. Caption Generation - Generate a coherent and informative caption that accurately describes the given image, using the provided object detection results as a reference. - The caption should cover key visual elements, including object types, positions, colors, and spatial relationships, and reflect the intent of the original prompt. 2. Prompt-Image Alignment and Generation Quality Evaluation - Evaluate how well the image itself matches the original prompt in terms of semantic content. - In addition, assess the visual quality of the generated image, including realism, clarity, and overall aesthetic quality. 3. Scoring - Provide an overall score on a scale from 0 to 4 based on both prompt alignment and image quality: - 0 indicates the image does not align with the prompt and is of low visual quality. It fails both semantically and aesthetically. - 1 indicates the image is visually decent but fails to capture the core semantics of the prompt. It may contain hallucinated or unrelated content. - 2 indicates the image is partially aligned with the prompt and has moderate quality. Some key elements may be missing or inaccurately rendered. - 3 indicates the image correctly reflects the prompt but suffers from low visual quality (e.g., blurry, distorted, or unnatural rendering). - 4 indicates the image is fully aligned with the prompt and of high visual quality. It accurately presents all required elements in a realistic, clear, and aesthetically pleasing manner.
MCoT Construction	You are given a sequence of (image, caption, score) triples ranked in ascending order by their score and their original prompt, where each triple consists of: - an image generated based on a text prompt, - a caption describing the image, - a score indicating how well the image aligns with the original prompt in terms of semantic relevance and generation quality. Your task is to simulate a step-by-step reasoning process that leads to the final decision about which image best satisfies the original prompt. This process should reflect how a human might evaluate and revise image generations based on feedback and visual inspection. Please proceed as follows: 1. Analyze each image and caption in order, reflecting on what aspects are missing, incorrect, or can be improved. 2. Describe how the reasoning evolves across steps and why one image is better than the previous ones. 3. End the reasoning by selecting the best image and briefly summarizing why it is the final choice. Output: Your response must strictly follow the format below: <think> The user asked/requested to generate an image: [prompt]. One possible image is `<1_image>` [your generation and reasoning process] </think> <answer>[your choice]</answer>

Table 4: Introduction of prompts used in MCoT construction

which involved reinforcement learning, we configured the group size to 8 and trained for 60 hours, further optimizing the model's performance through external feedback mechanisms.

C.2 Data Preparation

All of our images are in the resolution of 512×512, and vision tokens were generated using vision tokenizer of Emu3. Following the Emu3 design, we incorporated five special tokens to merge textual and visual data, constructing interleaved vision-language MCoT data and document-like inputs for the training process.

Taking the 2-step MCoT data as an example, whenever `<1_image>` or `<2_image>` appears for the first time in the MCoT sequence, it is immediately followed by the corresponding image token block. The resulting training data can be structured as follows:

```
<1_image>[BOI]{meta text}[SOT]{vision tokens}[EOL][EOF][EOI]
```

Example
You are tasked with evaluating the quality of an image description generated by a text-to-image model. Please provide a detailed evaluation based on the following aspects: 1. Completeness of the Image Description: - Assess whether the description covers all the key elements and details present in the image. Consider whether it includes descriptions of objects, their positions, colors, and any other relevant visual attributes. 2. Identification of Issues in the Image: - Determine if the description identifies any discrepancies or issues in the generated image compared to the original prompt. This could include missing elements, incorrect colors, misplaced objects, or any other inconsistencies. 3. Strategies for Improvement: - Suggest specific strategies or adjustments that could improve the accuracy and quality of the generated image. This could involve changes to the prompt or other recommendations to enhance the alignment between the generated image and the original prompt. Image Description: [Generated Image Description] Please provide a score for each aspect on a scale of 0 to 2, where: - 0 indicates poor performance, - 1 indicates average performance, - 2 indicates excellent performance. Additionally, provide a brief explanation for each score to justify your evaluation.

Table 5: Introduction of the prompt used to evaluate the generated text's quality.

Here, the image token block begins with `[BOI]`, where `{meta text}` contains information about the image resolution. The token `[SOT]` marks the beginning of the vision token sequence. Additionally, `[EOL]` and `[EOF]` are inserted into the token stream to indicate line breaks. The image token block ends with `[EOI]`.

D Societal Impacts

Vinci, as a novel framework that integrates deep reasoning capabilities into text-to-image generation through a unified model with reinforcement learning, has the potential to significantly enhance various applications that rely on visual content creation [42, 43, 54, 66, 69, 80] and understanding [20, 21, 33, 34, 40, 44, 46, 47, 67, 73, 74, 76, 81]. One of the key positive impacts is the improvement in the quality and alignment of generated images with textual prompts. This capability can greatly benefit educational and creative industries. For instance, in education, Vinci can generate highly accurate and contextually relevant illustrations for textbooks [27, 45, 48, 70], making learning materials more engaging and accessible for students. In creative industries, such as graphic design and advertising, Vinci can assist designers in quickly generating high-quality visual concepts, thereby accelerating the creative process and potentially leading to more innovative and diverse visual content.

Despite its potential benefits, Vinci also poses several risks that need to be carefully considered. One of the primary concerns is the potential misuse of the technology for generating misleading or harmful visual content. Vinci's ability to generate high-quality images based on textual prompts increases the risk of creating deepfakes or manipulated images that could be used to spread disinformation, manipulate public opinion, or harm individuals' reputations. The ease with which these images can be generated and disseminated poses a significant threat to societal trust and information integrity.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Appendix

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: N/A.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: See Project Page

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See section 4.Experiments

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: N/A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: See section 4.1 Experimental Setup

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Appendix

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: N/A.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: N/A.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: N/A.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: N/A.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: N/A.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

Justification: See Section 3. Method

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.