Decomposed Graded Verifier for Rule-Governed Generative World Modeling

Bowei Liu[1,2*] **Xinchen Zhang**[1*] **Xuhuan Li**[1*] **Tianlin Zhang**[2] **Kaian Jiang**[1]
Jingwei Liu[2,4] **Congyang Zhao**[2] **Yifei Qian**[2] **Junhong Liu**[3] **Zhiheng Li**[1]
Yujiu Yang[1†] **Yang Cai**[2] **Ling Yang**[4†]

[1]Tsinghua University [2]Amap, Alibaba Group [3]Microcyto [4]Peking University
https://github.com/Cominclip/OmniVerifier

Abstract

Generative visual verifiers play a crucial role in advancing multimodal intelligence. To enhance the robustness of visual verifiers in complex multimodal scenarios involving compositional requirements, we move beyond conventional binary verifiers that produce only True/False discrete judgments and introduce a verifier that automatically decomposes complex prompts or environmental rules into fine-grained components and assigns pointwise scores to provide a continuous and more informative critique signal. Our experiments further demonstrate that these structured and interpretable outputs improve the verifier's generalization to out-of-domain multimodal critic tasks. To train this verifier, we introduce a Decoupled RL training strategy that jointly optimizes sub-question-level binary judgment and global alignment scoring, using precise local feedback to develop fine-grained verification capabilities that provide a foundation for reliable continuous scoring at the global level. Under a general reinforcement learning framework, we first empirically demonstrate that the proposed pointwise graded verifier significantly outperforms binary verifiers in traditional text-to-image scenarios. We then extend our investigation to rule-governed generative world modeling where generative models are leveraged for world simulation and world reasoning under explicit rules. From a theoretical standpoint, we further show that a pairwise verification paradigm yields more accurate critiques than the pointwise formulation when ground-truth images are available, with empirical results on diverse structured world-modeling tasks such as Maze and Sudoku further validating the effectiveness of our approach. Overall, our findings suggest a key insight: the transition from generative models to structured world-modeling agents critically hinges on the availability of accurate pairwise graded visual verifiers.

1 Introduction

Multimodal generative models have made remarkable progress in recent years [2, 35, 29], producing visually compelling images conditioned on textual descriptions. Beyond visual fidelity [40, 50, 46], a central challenge lies in whether generated images faithfully follow complex instructions and reflect coherent world knowledge [12, 6, 45]. Motivated by this challenge, visual verifiers have emerged as a crucial component for assessing the correctness of generated results. By providing feedback on the alignment between images and textual descriptions, verifiers not only serve as evaluation tools, but also act as general critic models for inference-time scaling and reinforcement learning-based training. As generative models are increasingly applied to open-ended environments and reasoning-intensive

*Equal contribution.
†Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

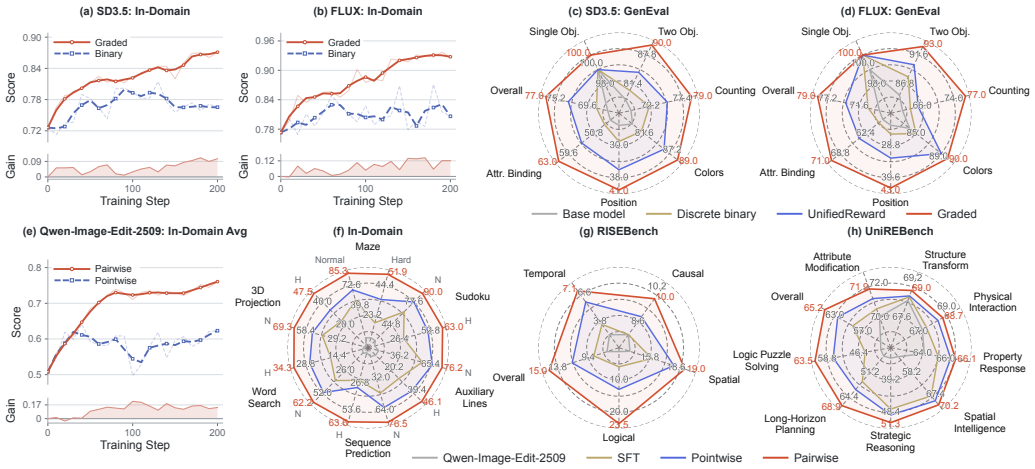


Figure 1: Panels (a)–(d) highlight the advantages of the graded verifier over the binary verifier as a generative reward model, while panels (e)–(h) show that pairwise verification provides more reliable and accurate feedback signals than pointwise verification.

tasks, the reliability and expressiveness of visual verifiers become a key bottleneck for advancing multimodal intelligence.

Despite their importance, most existing visual verifiers [52, 37, 43] adopt a binary judgment paradigm that produces only True/False decisions about whether an image satisfies a given prompt. Although simple and efficient, this design has several inherent limitations. First, binary supervision provides coarse feedback that fails to capture the partial correctness of generated samples. By assigning the same negative label to imperfect samples of varying quality, binary rewards obscure meaningful quality differences and weaken the within-group learning signals required by group-relative reinforcement learning. Moreover, the limited information conveyed by binary labels may encourage shortcut learning and increase susceptibility to reward hacking [30, 39], as models may exploit superficial patterns to predict the expected labels without adequately reasoning about the e subtl compositional relationships among objects, attributes, and spatial constraints.

To address these limitations, we propose the **Decomposed Graded Verifier (OmniVerifier-C1)**, which automatically decomposes complex prompts or environmental rules into fine-grained sub-questions and performs structured reasoning over each component. It assigns continuous scores based on the proportion of satisfied sub-questions, capturing partial correctness and providing informative critique signals. Our experiments demonstrate that decomposition-based structured reasoning significantly improves the verifier's out-of-domain multimodal critic capabilities. To train this verifier, we introduce a **Decoupled RL training strategy** that jointly optimizes sub-question-level binary judgment and global alignment scoring, using precise 0/1 feedback on individual sub-questions to develop strong fine-grained verification capabilities that support reliable global scoring. Moreover, our theoretical analysis shows that the decomposition-and-scoring paradigm better preserves the underlying true quality ordering among generated samples, providing finer-grained learning signals for group-relative reinforcement learning.

Building upon this verifier, we further explore its role as a generative reward model for reinforcement learning across two types of tasks. First, we apply the verifier to traditional text-to-image alignment, where it provides fine-grained feedback that encourages generative models to satisfy detailed compositional relationship. Second, we extend our study to the more reasoning-intensive and challenging setting of **structured generative world modeling**, where generative models are used to simulate environments and reason over sequential states. In these rule-governed environments, we find that pairwise verification, which compares generated states against ground-truth states, provides more accurate and reliable critiques than pointwise verification. This observation suggests that the effectiveness of visual verification in rule-governed world-modeling scenarios critically depends on the availability of reference states for comparison. Extensive experiments across diverse tasks, such as Maze and Sudoku, demonstrate that the proposed verifier provides more reliable reward signals and substantially improves reinforcement learning performance.

Our contributions can be summarized as follows:

- We propose the **Decomposed Graded Verifier**, which automatically decomposes complex prompts or environmental rules into fine-grained sub-questions, performs structured reasoning over each component, and assigns continuous scores based on the proportion of satisfied sub-questions. Our experiments demonstrate that its structured outputs significantly improve out-of-domain multimodal critic capabilities.
- We introduce a **Decoupled RL training strategy** that jointly optimizes sub-question-level binary judgment and global alignment scoring, using precise 0/1 feedback on individual sub-questions to develop fine-grained verification capabilities that provide a foundation for reliable continuous scoring.
- We demonstrate that the proposed verifier serves as an effective generative reward model for reinforcement learning, improving performance in both traditional text-to-image generation and rule-governed generative world modeling tasks.
- We provide empirical validation across diverse structured reasoning environments, showing that accurate **pairwise decomposed verifiers** are a key enabler for transforming generative models into rule-governed world-modeling agents.

2 Related Work

2.1 Generative Verifier or Reward Models

Generative reward models [21, 22, 33, 20, 39] differ fundamentally from traditional reward models [44, 48, 26, 51] in that they produce explicit rationales rather than merely assigning scalar scores. OmniVerifier [52] first introduces the concept of Generative Universal Verifier and presents ViVerBench, a comprehensive and challenging benchmark for evaluating multimodal reasoning and critique capabilities. DeepSeekMath-V2 [30] further proposes meta-verification and integrates it into verifier training to enhance faithfulness and reliability. OmniVerifier-M1 [49] extends meta-verification to multimodal scenarios by introducing symbolic spatial annotations, such as bounding boxes, points or lines as verifiable rationales. These rule-based supervision signals substantially alleviate reward hacking and enhance verifier generalization. Despite their improved interpretability, existing generative verifiers still rely on coarse binary decisions as final judgments, and thus struggle to provide dense and optimization-friendly reward signals for reinforcement learning. In this paper, we introduce a Decoupled RL training strategy to train the Decomposed Graded Verifier, which provides reliable continuous scores through explicit fine-grained decomposition.

2.2 Generative Model as World Reasoner

Multimodal generative models have made remarkable progress [19, 7, 27] in semantic alignment and aesthetic quality. Beyond visual fidelity, they are increasingly expected to serve as reasoning-capable world simulators [25, 10, 9]. To quantify the world reasoning capabilities of current generative models, several benchmarks [18, 31, 14] have been proposed. WISE [23] focuses on evaluating generative models' understanding of cultural commonsense and scientific knowledge. MMGR [1] is designed to probe generative models' spatial, temporal, physical, and logical reasoning capabilities. Motivated by these challenges, recent studies [32, 55, 4, 42] have explored new training paradigms and input modalities to learn the underlying dynamics of world evolution. Emu3.5 [5] advances world modeling by learning native next-state prediction over long-horizon interleaved vision-language sequences. UnityVideo [16] improves world-aware video generation through unified multi-modal and multi-task learning across diverse visual representations. In contrast to these efforts, we focus on improving the reliability and accuracy of reinforcement learning rewards for visual generation in rule-governed world-modeling scenarios, thereby encouraging generative models to develop stronger and more generalizable world reasoning capabilities.

3 OmniVerifier-C1: Decomposed Graded Verifier

In this section, we first analyze the theoretical advantages of the graded verifier over the binary verifier as a generative reward model. We then empirically demonstrate that our verifier develops fine-grained and generalizable multimodal verification capabilities through the Decoupled RL strategy, which jointly optimizes sub-question-level binary judgment and global alignment scoring.

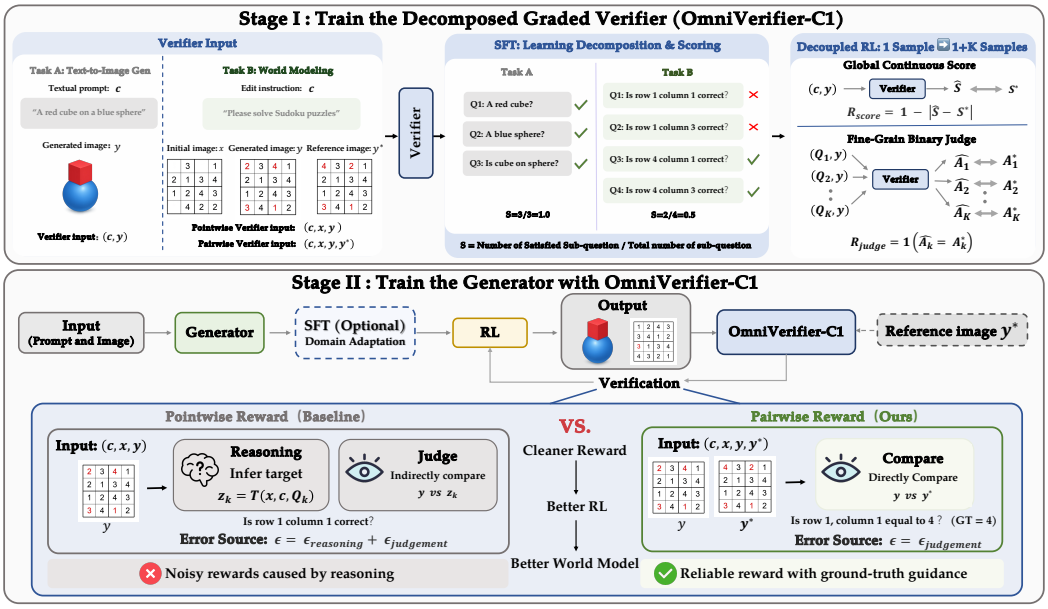


Figure 2: **Overview of our training pipeline.** Stage 1 trains the Decomposed Graded Verifier with decoupled reinforcement learning to produce fine-grained continuous rewards. Stage 2 introduces a pairwise judgment paradigm that provides cleaner and more reliable reward signals for generative model reinforcement learning.

3.1 Theoretical Advantage of Decomposed Graded Rewards for Generative RL

Drawbacks of Binary Reward. Binary verifiers have emerged as an effective tool for improving text-to-image generation quality through inference-time scaling [52]. When they detect semantic inconsistencies between a generated image and its textual description, they produce editing instructions that guide the generative model to iteratively refine the visual output. For this purpose, coarse binary feedback is often sufficient, as the verifier primarily serves as a gating mechanism that determines whether further refinement is needed. However, binary verifiers face inherent limitations when used as reward models for reinforcement learning, as their feedback is insufficiently dense and discriminative to distinguish among imperfect samples of varying visual quality and thereby guide effective generative policy optimization.

Revisiting Decomposition-Based Continuous Rewards. To address this limitation, we introduce the Decomposed Graded Verifier, a structured verification paradigm that elicits continuous scores from visual verifiers. Instead of reducing assessment to a single binary judgment, it decomposes a complex prompt into fine-grained sub-questions, evaluates each sub-question individually with binary judgments, and aggregates these local decisions into a continuous score that reflects the degree of prompt satisfaction. By preserving partial correctness in generated samples, this fine-grained reward signal provides denser and more optimization-friendly supervision for reinforcement learning.

Assume that the textual prompt c can be losslessly decomposed into K sub-questions $Q = \{q_1, q_2, \ldots, q_K\}$. Let $z(x) \in \{0, 1, \ldots, K\}$ denote the number of sub-questions satisfied by a visual output x. We define the continuous reward and the discrete reward as $r_c(x) = \frac{z(x)}{K}$ and $r_d(x) = \mathbf{1}[z(x) = K]$ respectively. Let $q_k := \mathbb{P}(z(x) = k)$ denote the probability that a sample satisfies exactly k sub-questions, where $k = 0, \ldots, K$. To quantify how well a reward recovers the true quality ordering, we define the ranking accuracy as

$$\mathrm{Acc}_{\mathrm{rank}}(r) := \mathbb{P}\big((r(x) - r(x'))(z(x) - z(x')) > 0 \mid z(x) \neq z(x')\big), \tag{1}$$

where $x, x' \sim$ i.i.d. $\pi_\theta(\cdot \mid c)$. This measures the probability that the reward assigns the correct ordering to a pair of samples with different true quality levels.

For the continuous reward, $r_c(x) = \frac{z(x)}{K}$ is strictly monotonic with respect to $z(x)$. Therefore, whenever $z(x) \neq z(x')$, we have $(r_c(x) - r_c(x'))(z(x) - z(x')) > 0$, which gives $\mathrm{Acc}_{\mathrm{rank}}(r_c) = 1$.

Table 1: Performance on ViVerBench of the Binary Verifier and Graded Verifier.

Methods	Concept Existence			Object Relation		World Dynamics		Image Annotation			State Value Evaluation				STEM		Overall
	Obj.	Attr.	Abs.P.	Spat.	N-Spat.	S-Phy	D-Phy	BBox	Point	Count	Maze	F.Lake	Robot.	GUI	Chart	LaTeX	
Qwen3-VL-8B	0.710	0.690	0.562	0.642	0.716	0.604	0.593	0.831	0.741	0.489	0.420	0.693	0.671	0.787	0.540	0.773	0.654
+ RL(Binary)	0.732	0.727	0.579	0.713	0.754	0.600	0.596	0.843	0.770	0.566	0.434	0.674	0.645	0.782	0.526	0.795	0.671
+ SFT(Binary)	0.689	0.695	0.620	0.654	0.703	0.546	0.627	0.802	0.735	0.504	0.414	0.714	0.661	0.792	0.538	0.802	0.656
+SFT(Binary) + RL(Binary)	0.735	0.721	0.554	0.719	0.748	0.521	0.636	0.840	0.765	0.532	0.446	0.687	0.682	0.799	0.553	0.814	0.672
+SFT(Graded)	0.724	0.709	0.535	0.679	0.657	0.613	0.596	0.817	0.752	0.543	0.427	0.661	0.576	0.769	0.656	0.830	0.659
+SFT(Graded)+RL(Graded)	0.758	0.735	0.541	0.736	0.766	0.573	0.582	0.859	0.781	0.559	0.466	0.679	0.693	0.810	0.567	0.839	0.684

For the discrete reward, let $A := \{(r_d(x) - r_d(x'))(z(x) - z(x')) > 0\}$ and $B := \{z(x) \neq z(x')\}$. Then $\mathrm{Acc}_{\mathrm{rank}}(r_d) = \mathbb{P}(A \mid B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}$. Since x and x' are independent, we have $\mathbb{P}(B) = 1 - \sum_{k=0}^{K} q_k^2$. The event $A \cap B$ occurs only when one sample satisfies all sub-questions while the other does not. Therefore, $\mathbb{P}(A \cap B) = 2q_K(1 - q_K)$, and thus $\mathrm{Acc}_{\mathrm{rank}}(r_d) = \frac{2q_K(1-q_K)}{1-\sum_{k=0}^{K} q_k^2} \leq 1$. A complete proof is provided in Appendix A. Moreover, as long as there exist two imperfect samples with different true quality levels, i.e., there exist $0 \leq k_1 \neq k_2 < K$, we have $\mathrm{Acc}_{\mathrm{rank}}(r_d) < 1$.Therefore, $\mathrm{Acc}_{\mathrm{rank}}(r_c) = 1 \geq \mathrm{Acc}_{\mathrm{rank}}(r_d)$.

Therefore, graded rewards preserve the true quality ordering more faithfully by assigning larger advantage weights to samples that satisfy more sub-questions, thereby better aligning optimization with improvements in true sample quality. By contrast, discrete rewards collapse partially correct samples into the same value, failing to distinguish imperfect outputs of differing true visual quality and potentially destabilizing generative policy optimization, especially under complex compositional requirements.

3.2 Graded Verification Enhances Multimodal Reasoning and Critique Capabilities

Drawbacks of Binary Outcome Supervision. Binary verifiers supervise model training solely through the final True/False outcome, implicitly assuming that a correct verdict reflects a correct reasoning process. This assumption is often reasonable in domains such as mathematical problem solving, where the solution space is large and arriving at the correct answer typically requires coherent multi-step reasoning. However, it becomes much more fragile in binary multimodal critique tasks. Without explicit supervision over the judgment process, a verifier may exploit shallow visual patterns or shortcut cues to produce seemingly correct verdicts without genuinely understanding the underlying multimodal reasoning process [39, 30], thereby leading to reward hacking

Experiment Setup. We collect textual prompts from ShareGPT-4o-Image [3] and control prompt length to construct samples with varying levels of complexity. To enhance visual diversity, we randomly sample from multiple generative models, including FLUX.1-dev [17], Qwen-Image [41], and Z-Image [36], to generate the corresponding images. We then perform supervised fine-tuning (SFT) to learn the sub-question decomposition paradigm distilled from Seed1.8 [28]. Using the SFT-trained model, we filter out samples whose predictions exactly match the ground truth, and further apply a GPT-5 [24] Bo10 filtering strategy to remove samples with accuracy below 60% while controlling the proportion of fully correct samples at 30%. The remaining high-quality samples are used for *Decoupled Reinforcement Learning* with DAPO [47] on Qwen3-VL-8B [34]. The model is trained for 80 steps on 16 NVIDIA A800-80G GPUs and evaluated on ViVerBench [52].

Decoupled Reinforcement Learning. As shown in Fig. 2, given a textual prompt c and its corresponding generated image y, we convert each pair into $K + 1$ reinforcement learning samples, where K denotes the number of decomposed sub-questions. Specifically, one sample is used for decomposition-based continuous scoring policy optimization, while the remaining K samples are used for binary fine-grained judgment policy optimization. For the decomposition-based scoring sample, the verifier takes the textual prompt c and the generated image y as input, and is trained to predict the global alignment score induced by the decomposition. Given the predicted score $\hat{S} \in [0, 1]$ and the ground-truth score $S^* \in [0, 1]$, the reward is defined as $\mathcal{R}_{\mathrm{score}} = 1 - |\hat{S} - S^*|$. For each fine-grained judgment sample, the verifier takes the k-th sub-question Q_k and the generated image y as input, and is trained to predict the binary answer associated with this sub-question. Given the predicted answer $\hat{A}_k \in \{0, 1\}$ and the ground-truth answer $A_k^* \in \{0, 1\}$, the reward is defined as $\mathcal{R}_{\mathrm{judgment}} = \mathbb{1}[\hat{A}_k = A_k^*]$. During reinforcement learning, we mix these two types of samples and



Figure 3: Qualitative visualization of traditional text-to-image generation.

optimize them with their respective rewards, forming a decoupled RL training scheme that jointly optimizes sub-question-level binary judgment and global alignment scoring. Precise 0/1 feedback on individual sub-questions enables the verifier to develop fine-grained verification capabilities that provide a foundation for reliable continuous scoring.

Experiment Analysis. Table 1 shows that the Decomposed Graded Verifier achieves stronger generalization than the Discrete Binary Verifier on ViVerBench, a comprehensive and challenging benchmark for evaluating multimodal reasoning and critique. Notably, the Decomposed Graded Verifier requires supervised fine-tuning (SFT) to learn the structured decomposition paradigm. By contrast, the Discrete Binary Verifier only needs to produce a binary True/False verdict with a corresponding textual explanation, and can therefore be directly optimized through reinforcement learning. For a fair comparison, we evaluate the Discrete Binary Verifier in both RL-only and SFT+RL settings. The results show that the Discrete Binary Verifier consistently underperforms the Decomposed Graded Verifier, regardless of whether it is first adapted with SFT. This indicates that the key factor is not the training pipeline itself, but the nature of the supervision signal. Binary supervision is highly compressed and sparse, which encourages the model to rely on shallow patterns rather than capturing complex semantic structures. In contrast, the Decomposed Graded Verifier decomposes complex tasks into multiple sub-questions and performs fine-grained judgments, thereby explicitly enforcing a structured reasoning process. This mechanism provides richer and more informative feedback signals, effectively mitigating reward hacking and substantially improving the model's multimodal reasoning and critique capabilities. Moreover, our **Decoupled RL training strategy** jointly optimizes sub-question-level binary judgment and global alignment scoring, using precise 0/1 feedback on individual sub-questions to strengthen fine-grained verification capabilities and thereby provide a foundation for robust generalization to out-of-distribution (OOD) scenarios.

4 Verifier-Guided Reinforcement Learning for Generative Models

In this section, we investigate how the Decomposed Graded Verifier guides generative model optimization through reinforcement learning. We begin with traditional text-to-image generation task, and then focus on studying how to adapt the verifier to provide more reliable reward signals for rule-governed world modeling scenarios.

Table 2: Performance on GenEval and CLIPScore under different reward models.

Methods	GenEval							CLIPScore
	Single Obj.	Two Obj.	Counting	Colors	Position	Attri. Bind.	Overall	
Stable-Diffusion-3.5-medium	0.99	0.78	0.69	0.79	0.25	0.45	0.66	32.8
+RL(UnifiedReward)	0.99	0.84	0.75	0.86	0.36	0.55	0.73	33.7
+RL(Discrete Binary Verifier)	0.99	0.80	0.72	0.81	0.29	0.47	0.68	33.1
+RL(Decomposed Graded Verifier)	**1.00**	**0.90**	**0.79**	**0.89**	**0.41**	**0.63**	**0.77**	**34.4**
FLUX.1-dev	0.99	0.85	0.61	0.84	0.22	0.58	0.68	33.2
+RL(UnifiedReward)	1.00	0.90	0.65	0.89	0.33	0.64	0.74	34.0
+RL(Discrete Binary Verifier)	1.00	0.88	0.63	0.85	0.25	0.59	0.70	33.3
+RL(Decomposed Graded Verifier)	**1.00**	**0.93**	**0.77**	**0.90**	**0.43**	**0.71**	**0.79**	**34.5**

4.1 Verifier-Guided RL for Traditional Text-to-Image Generation

Given a textual prompt c and its generated image y, the Decomposed Graded Verifier first decomposes c into a set of fine-grained sub-questions $Q = \{q_1, q_2, \ldots, q_K\}$ and then produces a continuous score by aggregating their verification results: $S = \frac{1}{K}\sum_{k=1}^{K} \mathcal{J}(q_k, y)$, where $\mathcal{J}(q_k, y) \in \{0, 1\}$ denotes a binary judgment function that evaluates whether the generated image y satisfies sub-question q_k. This score is used as the reward signal for reinforcement learning to guide policy optimization.

4.2 Verifier-Guided RL for Rule-Governed World Modeling

Traditional text-to-image generation primarily targets semantic alignment, where a prompt may admit multiple diverse and visually plausible outputs, leaving no well-defined unique ground-truth image. By contrast, world simulation and reasoning tasks are rule-governed and highly structured, where the desired final state is often uniquely determined by the initial state and the underlying rules, thereby making a ground-truth image available in many scenarios. Motivated by this property, we explicitly introduce the ground-truth image as an additional visual input to the verifier, which simplifies the verification process and enables the verifier to produce more accurate and reliable reward signals for rule-governed generative world modeling.

Formally, given an initial visual state x and a rule-based instruction c, the generative model π_θ is required to **understand the underlying rules and perform rule-governed reasoning** to produce a resulting image $y \sim \pi_\theta(y \mid x, c)$, which can be viewed as a visual prediction of the target final state. Similar to the traditional text-to-image setting, the instruction c can be decomposed into K fine-grained sub-questions $Q = \{q_1, q_2, \ldots, q_K\}$, each assessing whether the resulting image y satisfies a specific sub-rule or achieves a corresponding sub-goal.

Definition1: Pointwise Verification. As shown in Fig. 2, the pointwise verifier takes (x, c, y) as input. For each sub-question q_k, it must first infer the corresponding target state from the initial visual state x and the underlying rules c, denoted as $z_k = \mathcal{T}(q_k, x, c)$. The verifier then assesses whether the resulting image y is consistent with this internally inferred target state z_k. Accordingly, the pointwise verification score is defined as $S_{\text{point}} = \frac{1}{K}\sum_{k=1}^{K} \mathcal{J}(q_k, y, z_k)$, and its deviation from the true score consists of both reasoning error and judgment error: $\epsilon_{\text{point}} = \epsilon_{\text{reasoning}} + \epsilon_{\text{judge}}$.

Definition2: Pairwise Verification. In contrast, pairwise verification explicitly incorporates the ground-truth image y^* as an additional visual input, with the verifier taking (x, c, y, y^*) as input. For each sub-question q_k, the verifier compares the generated image y against the ground-truth image y^* and assigns a corresponding score, hence the name pairwise verification. By avoiding internal target-state inference, the pairwise verification score is defined as $S_{\text{pair}} = \frac{1}{K}\sum_{k=1}^{K} \mathcal{J}(q_k, y, y^*)$, and its deviation from the true score is reduced to the judgment error: $\epsilon_{\text{pair}} = \epsilon_{\text{judge}}$.

By simplifying the reason-then-judge process into direct reference-based comparison and thereby reducing the reasoning errors associated with target-state inference, pairwise verification provides a substantially more reliable reward mechanism than pointwise verification in rule-governed world-modeling scenarios. This property is crucial for post-training generative models, where the effectiveness of reinforcement learning hinges on the accuracy and stability of the reward signal.

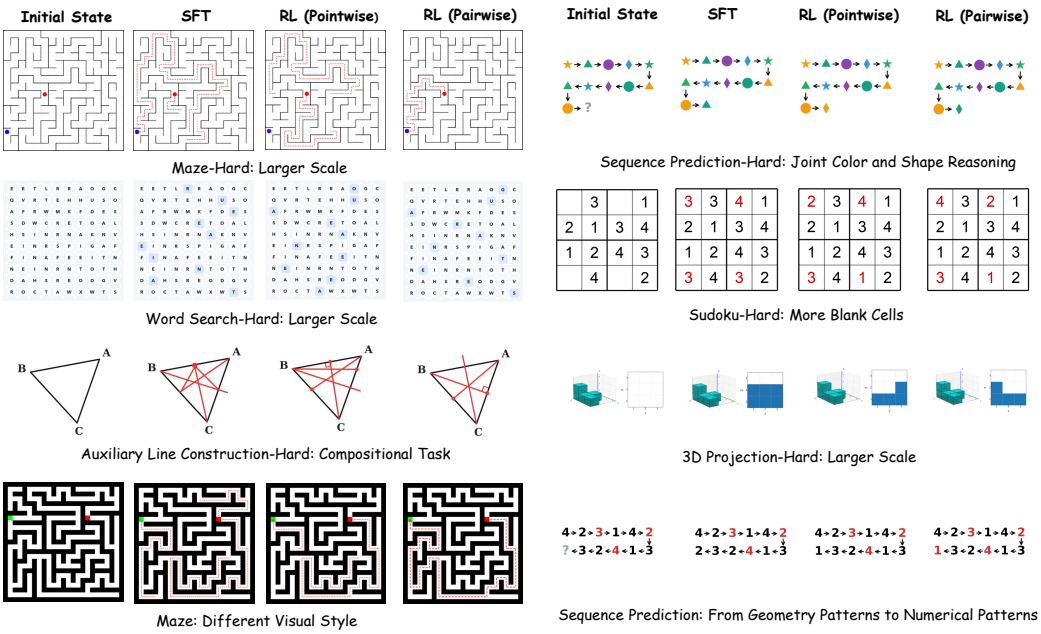


Figure 4: Qualitative visualization of rule-governed world-modeling tasks.

5 Experiments

In this section, we conduct comprehensive experiments to demonstrate the advantages of the Decomposed Graded Verifier as a generative reward model and the superiority of the pairwise verification mechanism in rule-governed world modeling scenarios.

5.1 Experiment Setup

Training Datasets. For traditional text-to-image tasks, we use textual prompts from ShareGPT-4o-Image [3], the same data source used for verifier training. For rule-governed world modeling scenarios, we construct six game-based tasks to comprehensively evaluate generative models' abilities in **path planning, logical reasoning, geometric construction, pattern induction, character recognition, and spatial reasoning** .

- **Maze & Maze-Hard.** Find a valid path from the red start circle to the blue end cross without touching or crossing any black walls. Maze-Hard contains larger-scale maze instances.
- **Sudoku & Sudoku-Hard.** Fill the blanks in a 4×4 Sudoku grid so that each row, column, and 2×2 subgrid contains $\{1, 2, 3, 4\}$ exactly once. Sudoku-Hard contains more blanks.
- **Auxiliary Line Construction & Auxiliary Line Construction-Hard.** Draw auxiliary lines in a geometric figure. The hard setting requires multiple compositional constructions.
- **Sequence Prediction & Sequence Prediction-Hard.** Predict the next geometric figure from shape or color patterns. The hard setting requires simultaneous color-shape changes.
- **Word Search & Word Search-Hard.** Locate the target word in an $N \times N$ grid, where each row contains one character in the correct order. The hard setting corresponds to a larger N.
- **3D Projection & 3D Projection-Hard.** Generate the specified 2D orthographic view of a 3D structure, such as the top, front, or side view. The hard setting uses larger 3D structures.

Training Details. For traditional text-to-image tasks, we adopt Stable-Diffusion-3.5-medium and FLUX.1-dev [17] as the backbone generative models. For rule-governed world-modeling scenarios, we use Qwen-Image-Edit-2509 [41] as the backbone. All models are trained for 200 steps using the DiffusionNFT [54] algorithm on 16 NVIDIA A800-80G GPUs.

Table 3: Performance comparison on game-based image-to-image tasks for world modeling.

Methods	Maze		Sudoku		Aux. Line Construction		Sequence Prediction		Word Search		3D Projection	
	Normal	Hard	Normal	Hard	Normal	Hard	Normal	Hard	Normal	Hard	Normal	Hard
Qwen-Image-Edit-2509	17.31	8.73	22.21	5.62	16.79	7.32	9.18	4.33	6.28	2.46	3.86	1.71
+ SFT	56.77	18.96	63.62	29.99	57.62	26.84	46.53	27.96	37.36	17.29	42.94	23.37
+SFT+RL(Pointwise)	67.81	34.78	77.74	50.71	66.98	37.49	61.27	34.18	50.92	26.94	52.19	33.16
+SFT+RL(Pairwise)	**85.31**	**51.92**	**89.97**	**62.99**	**76.19**	**46.12**	**73.48**	**63.64**	**62.18**	**34.27**	**69.28**	**47.51**

Table 4: Performance comparison on RISEBench and UniREditBench.

Methods	RISEBench					UniREdit-Real					UniREdit-Game					Overall
	T	C	S	L	Overall	AM	ST	PI	PR	Avg.	SI	SR	LHP	LPS	Avg.	
Qwen-Image-Edit-2509	2.4	7.8	14.0	1.2	6.7	70.04	68.49	67.26	64.11	67.48	52.31	32.96	42.46	37.89	41.41	53.16
+ SFT	3.5	7.8	16.0	5.9	8.6	70.56	68.68	67.55	64.98	67.94	66.92	47.48	56.06	43.98	53.61	59.14
+ SFT + RL(Pointwise)	5.9	8.9	18.0	12.9	11.7	71.24	68.73	68.38	65.77	68.53	68.73	49.16	60.32	56.41	58.66	62.37
+ SFT + RL(Pairwise)	**7.1**	**10.0**	**19.0**	**23.5**	**15.0**	**71.87**	**69.02**	**68.72**	**66.08**	**68.92**	**70.23**	**51.33**	**68.93**	**63.53**	**63.51**	**65.15**

5.2 Text-to-Image Generation

Graded Rewards Are Better than Binary Rewards. Table 2 compares the Decomposed Graded Verifier with the Discrete Binary Verifier as reinforcement learning reward models across different generative backbones, including Stable-Diffusion-3.5-Medium and FLUX.1-dev. The Decomposed Graded Verifier consistently achieves superior performance, demonstrating that fine-grained graded reward signals are more effective for improving the semantic alignment capability of generative models. Fig. 3 further supports this conclusion through qualitative visualizations. Fig. 1 (a)(b) shows the training dynamics under the two reward models. We evaluate intermediate checkpoints every 10 steps on a held-out set of 500 textual prompts under the same decomposition-based scoring protocol, **using Gemini3.1-Pro [11] as an evaluator not involved in data processing**. Compared with the Discrete Binary Verifier, training with the Decomposed Graded Verifier yields steadier and more pronounced performance improvements, suggesting that denser and more discriminative rewards support more stable and effective optimization.

Comparison with Other Continuous Reward Models. We compare the Decomposed Graded Verifier with UnifiedReward [38], which also focuses on semantic alignment and produces continuous reward scores. As shown in Table 2, our method achieves superior performance on both GenEval [8] and CLIPScore [15], supporting the ability of the Decomposed Graded Verifier trained with our Decoupled RL strategy to provide more accurate and fine-grained reward signals and highlighting its advantages as a generative reward model.

5.3 Rule-Governed Generative World Modeling

Advantages of Pairwise Verification over Pointwise Verification. As shown in Tables 3 and 4, pairwise verification consistently outperforms pointwise verification on RISEBench [53], UniREdit-Bench [13], and our constructed game-based benchmark. These results indicate that incorporating the ground-truth image as an explicit reference significantly enhances the accuracy and reliability of reward signals. The training dynamics in Fig. 1(e) summarize average performance across our six constructed game-based tasks and further highlight the optimization advantages of pairwise verification over pointwise verification. We evaluate intermediate checkpoints every 10 training steps on the held-out validation sets. **To improve evaluation reliability, we use Gemini3.1-Pro [11] as an evaluator not involved in data processing and explicitly provide the ground-truth image following the pairwise verification protocol.**

The pairwise-trained model exhibits steadier and more pronounced performance improvements, whereas the pointwise-trained model shows clear performance fluctuations. Pointwise rewards require the verifier to infer an implicit target state, which can introduce substantial noise into the feedback in complex reasoning scenarios. In contrast, pairwise verification directly compares the generated result with the ground-truth image, reducing reliance on target-state inference and enabling reward signals to better reflect the true visual quality. This provides more reliable supervision for policy optimization and highlights the value of pairwise verification for developing generative models into rule-governed world-modeling agents.

From Easy-to-Hard and Domain Generalization. To more comprehensively evaluate the generalization potential of the model trained with pairwise verification, we construct a more challenging Hard Dataset that is excluded from training. This dataset contains larger-scale instances and compositional tasks designed to test reasoning in more complex structured environments. As shown in Table 3 and Fig. 4, the model exhibits strong performance across out-of-distribution scenarios, including shifts in the visual domain. These results suggest that the model acquires transferable capabilities for structured reasoning and rule-governed world modeling beyond task-specific training patterns, highlighting the advantages of pairwise verification in supporting robust generalization.

SFT Provides Domain-Specific Execution Priors. Tables 3 and 4 show that SFT yields substantial performance gains, which can be primarily attributed to task-domain adaptation. Since Qwen-Image-Edit-2509 has not been specifically trained on game-based image-to-image generation tasks, it often fails to interpret the required transformation. In practice, the model tends to either the model tends to either leave the input image unchanged or distort its original structure, indicating that it lacks the task-specific action priors required for this domain. After supervised fine-tuning on our constructed game-based dataset, the generative model learns the visual style of these tasks and the corresponding rule-based transformation patterns. Although the model still often fails to produce fully correct solutions, it begins to perform meaningful operations toward the desired final state rather than generating irrelevant or destructive visual outputs, leading to clear performance improvements. These results indicate that **SFT primarily equips the model with domain-specific execution ability**, while subsequent reinforcement learning further endows the generative model with reasoning intelligence.

Generative Intelligence Beyond the Text Encoder. In both multimodal understanding and generation models, the text encoder is often regarded as the primary source of reasoning intelligence. For generative models, this view implies that the text encoder interprets instructions and provides semantic guidance, while the DiT primarily executes the visual generation process. However, in our training pipeline, the text encoder remains frozen throughout both SFT and RL, and only the DiT parameters are updated. The reasoning capabilities observed after training suggest that reasoning intelligence is not encoded exclusively in the text encoder, as the DiT can also learn, internalize, and express capabilities for structured reasoning and rule-governed world modeling.

6 Conclusion

This work introduces the Decomposed Graded Verifier as an effective reward model for improving visual generation in both traditional text-to-image tasks and structured world-modeling scenarios. Our contributions are twofold. First, we propose a Decoupled RL training strategy that jointly optimizes sub-question-level binary judgment and global alignment scoring, using precise local feedback to develop fine-grained verification capabilities that support accurate continuous scoring. Empirical results further demonstrate improved out-of-domain multimodal critic capabilities. Second, focusing on rule-governed world modeling, we introduce a pairwise verification paradigm that explicitly incorporates ground-truth images as visual references. This simplifies verification by shifting from a reason-then-judge process to direct reference-based comparison, thereby producing more accurate and reliable reward signals. Extensive experiments validate the effectiveness of both the Decomposed Graded Verifier trained with our Decoupled RL strategy and the pairwise verification paradigm.

Acknowledgments and Disclosure of Funding

This work was partly supported by the National Key Research and Development Program of China (No. 2024YFB2808903), the Shenzhen Science and Technology Program (ZDCY20250901103533010).

We thank the anonymous reviewers for their valuable comments and constructive suggestions. The authors declare no competing interests.

References

[1] Zefan Cai, Haoyi Qiu, Tianyi Ma, Haozhe Zhao, Gengze Zhou, Kung-Hsiang Huang, Parisa Kordjamshidi, Minjia Zhang, Wen Xiao, Jiuxiang Gu, et al. Mmgr: Multi-modal generative reasoning. *arXiv preprint arXiv:2512.14691*, 2025.

[2] Siyu Cao, Hangting Chen, Peng Chen, Yiji Cheng, Yutao Cui, Xinchi Deng, Ying Dong, Kipper Gong, Tianpeng Gu, Xiusen Gu, et al. Hunyuanimage 3.0 technical report. *arXiv preprint arXiv:2509.23951*, 2025.

[3] Junying Chen, Zhenyang Cai, Pengcheng Chen, Shunian Chen, Ke Ji, Xidong Wang, Yunjin Yang, and Benyou Wang. Sharegpt-4o-image: Aligning multimodal models with gpt-4o-level image generation, 2025.

[4] Kaijin Chen, Dingkang Liang, Xin Zhou, Yikang Ding, Xiaoqiang Liu, Pengfei Wan, and Xiang Bai. Out of sight but not out of mind: Hybrid memory for dynamic video world models. *arXiv preprint arXiv:2603.25716*, 2026.

[5] Yufeng Cui, Honghao Chen, Haoge Deng, Xu Huang, Xinghang Li, Jirong Liu, Yang Liu, Zhuoyan Luo, Jinsheng Wang, Wenxuan Wang, et al. Emu3. 5: Native multimodal models are world learners. *arXiv preprint arXiv:2510.26583*, 2025.

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

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

[8] 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.

[9] Google. Veo 3 model card. `https://storage.googleapis.com/deepmind-media/Model-Cards/Veo-3-Model-Card.pdf`, 2024.

[10] Google. Introducing nano banana pro. `https://blog.google/technology/ai/nano-banana-pro/`, 2025.

[11] Google DeepMind. Gemini 3.1 Pro model card. `https://deepmind.google/models/model-cards/gemini-3-1-pro/`, 2026.

[12] Ziyu Guo, Xinyan Chen, Renrui Zhang, Ruichuan An, Yu Qi, Dongzhi Jiang, Xiangtai Li, Manyuan Zhang, Hongsheng Li, and Pheng-Ann Heng. Are video models ready as zero-shot reasoners? an empirical study with the mme-cof benchmark. *arXiv preprint arXiv:2510.26802*, 2025.

[13] Feng Han, Yibin Wang, Chenglin Li, Zheming Liang, Dianyi Wang, Yang Jiao, Zhipeng Wei, Chao Gong, Cheng Jin, Jingjing Chen, et al. Unireditbench: A unified reasoning-based image editing benchmark. *arXiv preprint arXiv:2511.01295*, 2025.

[14] Haonan Han, Jiancheng Huang, Xiaopeng Sun, Junyan He, Rui Yang, Jie Hu, Xiaojiang Peng, Lin Ma, Xiaoming Wei, and Xiu Li. Vigor-bench: How far are visual generative models from zero-shot visual reasoners? *arXiv preprint arXiv:2603.25823*, 2026.

[15] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 7514–7528, 2021.

[16] Jiehui Huang, Yuechen Zhang, Xu He, Yuan Gao, Zhi Cen, Bin Xia, Yan Zhou, Xin Tao, Pengfei Wan, and Jiaya Jia. Unityvideo: Unified multi-modal multi-task learning for enhancing world-aware video generation. *arXiv preprint arXiv:2512.07831*, 2025.

[17] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

[18] Ouxiang Li, Yuan Wang, Xinting Hu, Huijuan Huang, Rui Chen, Jiarong Ou, Xin Tao, Pengfei Wan, Xiaojuan Qi, and Fuli Feng. Easier painting than thinking: Can text-to-image models set the stage, but not direct the play? *arXiv preprint arXiv:2509.03516*, 2025.

[19] Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. Uniworld-v1: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025.

[20] Bowei Liu, Zheng Lu, Yuhan Bian, Xinchen Zhang, Xingming Shui, Yuesheng Huang, Xuhuan Li, Zihao Liu, Yifan Yang, Jun Zhou, et al. Vidforensics-m1: Meta-detection reinforcement learning with verifiable temporal grounding for ai-generated video forensics. *arXiv preprint arXiv:2608.11201*, 2026.

[21] Dakota Mahan, Duy Van Phung, Rafael Rafailov, Chase Blagden, Nathan Lile, Louis Castricato, Jan-Philipp Fränken, Chelsea Finn, and Alon Albalak. Generative reward models. *arXiv preprint arXiv:2410.12832*, 2024.

[22] Saumya Malik, Valentina Pyatkin, Sander Land, Jacob Morrison, Noah A Smith, Hannaneh Hajishirzi, and Nathan Lambert. Rewardbench 2: Advancing reward model evaluation. *arXiv preprint arXiv:2506.01937*, 2025.

[23] Yuwei Niu, Munan Ning, Mengren Zheng, Weiyang Jin, Bin Lin, Peng Jin, Jiaqi Liao, Chaoran Feng, Kunpeng Ning, Bin Zhu, et al. Wise: A world knowledge-informed semantic evaluation for text-to-image generation. *arXiv preprint arXiv:2503.07265*, 2025.

[24] OpenAI. Introducing gpt-5. `https://openai.com/index/introducing-gpt-5/`, 2025.

[25] OpenAI. Sora 2 system card. `https://openai.com/index/sora-2-system-card/`, 2025.

[26] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[27] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[28] Bytedance Seed. Seed1. 8 model card: Towards generalized real-world agency. *arXiv preprint arXiv:2603.20633*, 2026.

[29] Team Seedream, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, et al. Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427*, 2025.

[30] Zhihong Shao, Yuxiang Luo, Chengda Lu, ZZ Ren, Jiewen Hu, Tian Ye, Zhibin Gou, Shirong Ma, and Xiaokang Zhang. Deepseekmath-v2: Towards self-verifiable mathematical reasoning. *arXiv preprint arXiv:2511.22570*, 2025.

[31] Kaiyue Sun, Rongyao Fang, Chengqi Duan, Xian Liu, and Xihui Liu. T2i-reasonbench: Benchmarking reasoning-informed text-to-image generation. *arXiv preprint arXiv:2508.17472*, 2025.

[32] Boming Tan, Xiangdong Zhang, Ning Liao, Yuqing Zhang, Shaofeng Zhang, Xue Yang, Qi Fan, and Yanyong Zhang. Dreamworld: Unified world modeling in video generation. *arXiv preprint arXiv:2603.00466*, 2026.

[33] Sijun Tan, Siyuan Zhuang, Kyle Montgomery, William Y Tang, Alejandro Cuadron, Chenguang Wang, Raluca Ada Popa, and Ion Stoica. Judgebench: A benchmark for evaluating llm-based judges. *arXiv preprint arXiv:2410.12784*, 2024.

[34] Qwen Team. Qwen3 technical report, 2025.

[35] Wan Team. Wan: Open and advanced large-scale video generative models. 2025.

[36] Z-Image Team. Z-image: An efficient image generation foundation model with single-stream diffusion transformer. *arXiv preprint arXiv:2511.22699*, 2025.

[37] Xiyao Wang, Chunyuan Li, Jianwei Yang, Kai Zhang, Bo Liu, Tianyi Xiong, and Furong Huang. Llava-critic-r1: Your critic model is secretly a strong policy model. *arXiv preprint arXiv:2509.00676*, 2025.

[38] Yibin Wang, Yuhang Zang, Hao Li, Cheng Jin, and Jiaqi Wang. Unified reward model for multimodal understanding and generation. *arXiv preprint arXiv:2503.05236*, 2025.

[39] Zongqi Wang, Rui Wang, Yuchuan Wu, Yiyao Yu, Pinyi Zhang, Shaoning Sun, Yujiu Yang, and Yongbin Li. Reward modeling from natural language human feedback. *arXiv preprint arXiv:2601.07349*, 2026.

[40] Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. Video models are zero-shot learners and reasoners. *arXiv preprint arXiv:2509.20328*, 2025.

[41] Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, Yuxiang Chen, Zecheng Tang, Zekai Zhang, Zhengyi Wang, An Yang, Bowen Yu, Chen Cheng, Dayiheng Liu, Deqing Li, Hang Zhang, Hao Meng, Hu Wei, Jingyuan Ni, Kai Chen, Kuan Cao, Liang Peng, Lin Qu, Minggang Wu, Peng Wang, Shuting Yu, Tingkun Wen, Wensen Feng, Xiaoxiao Xu, Yi Wang, Yichang Zhang, Yongqiang Zhu, Yujia Wu, Yuxuan Cai, and Zenan Liu. Qwen-image technical report, 2025.

[42] Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video world models with long-term spatial memory. *arXiv preprint arXiv:2506.05284*, 2025.

[43] Tianyi Xiong, Xiyao Wang, Dong Guo, Qinghao Ye, Haoqi Fan, Quanquan Gu, Heng Huang, and Chunyuan Li. Llava-critic: Learning to evaluate multimodal models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13618–13628, 2025.

[44] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.

[45] Ling Yang, Ye Tian, Bowen Li, Xinchen Zhang, Ke Shen, Yunhai Tong, and Mengdi Wang. Mmada: Multimodal large diffusion language models. *arXiv preprint arXiv:2505.15809*, 2025.

[46] Ling Yang, Xinchen Zhang, Ye Tian, Chenming Shang, Minghao Xu, Wentao Zhang, and Bin Cui. Hermesflow: Seamlessly closing the gap in multimodal understanding and generation. *arXiv preprint arXiv:2502.12148*, 2025.

[47] Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

[48] Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. *arXiv preprint arXiv:2408.15240*, 2024.

[49] Xinchen Zhang, Bowei Liu, Jiale Liu, Chufan Shi, Yizhen Zhang, Junhong Liu, Youliang Zhang, Zhiheng Li, Yujiu Yang, and Ling Yang. Omniverifier-m1: Multimodal meta-verifier with explicit structured recalibration. *arXiv preprint arXiv:2605.28805*, 2026.

[50] Xinchen Zhang, Ling Yang, Yaqi Cai, Zhaochen Yu, Kai-Ni Wang, Jiake Xie, Ye Tian, Minkai Xu, Yong Tang, Yujiu Yang, et al. Realcompo: Balancing realism and compositionality improves text-to-image diffusion models. *Advances in Neural Information Processing Systems*, 37:96963–96992, 2024.

[51] Xinchen Zhang, Ling Yang, Guohao Li, Yaqi Cai, Yong Tang, Yujiu Yang, Mengdi Wang, Bin CUI, et al. Itercomp: Iterative composition-aware feedback learning from model gallery for text-to-image generation. In *The Thirteenth International Conference on Learning Representations*, 2024.

[52] Xinchen Zhang, Xiaoying Zhang, Youbin Wu, Yanbin Cao, Renrui Zhang, Ruihang Chu, Ling Yang, and Yujiu Yang. Generative universal verifier as multimodal meta-reasoner. *arXiv preprint arXiv:2510.13804*, 2025.

[53] Xiangyu Zhao, Peiyuan Zhang, Kexian Tang, Xiaorong Zhu, Hao Li, Wenhao Chai, Zicheng Zhang, Renqiu Xia, Guangtao Zhai, Junchi Yan, et al. Envisioning beyond the pixels: Benchmarking reasoning-informed visual editing. *arXiv preprint arXiv:2504.02826*, 2025.

[54] Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. *arXiv preprint arXiv:2509.16117*, 2025.

[55] Sixiao Zheng, Minghao Yin, Wenbo Hu, Xiaoyu Li, Ying Shan, and Yanwei Fu. Versecrafter: Dynamic realistic video world model with 4d geometric control. *arXiv preprint arXiv:2601.05138*, 2026.

A Proof of Ranking Preservation

In this section we prove that Decomposed Graded Verification preserves the true quality ordering of generated samples more faithfully than Discrete Binary Verification. The argument rests on a single observation: under GRPO, the relative update strength among samples is dictated by their group-normalized advantages, whose ordering is fully inherited from the underlying reward values. It therefore suffices to compare how the two reward designs rank pairs of samples with distinct latent quality.

Setup. Let c be a prompt and $x \sim \pi_\theta(\cdot \mid c)$ a sample drawn from the current policy. Suppose c admits a lossless decomposition into K sub-questions, and let

$$z(x) \in \{0, 1, \ldots, K\} \tag{2}$$

denote the number of sub-questions satisfied by x. We treat $z(x)$ as the latent fine-grained quality level of x and write

$$q_k := \mathbb{P}(z(x) = k), \qquad k = 0, \ldots, K, \tag{3}$$

where the probability is taken with respect to $x \sim \pi_\theta(\cdot \mid c)$.

We compare two reward designs:

$$r_c(x) = \frac{z(x)}{K}, \qquad r_d(x) = \mathbf{1}[z(x) = K], \tag{4}$$

namely the decomposed graded reward r_c and the discrete binary reward r_d.

For two independent samples $x, x' \sim \pi_\theta(\cdot \mid c)$, the pairwise ranking accuracy of a reward function r is defined as

$$\mathrm{Acc}_{\mathrm{rank}}(r) := \mathbb{P}\left((r(x) - r(x'))(z(x) - z(x')) > 0 \mid z(x) \neq z(x')\right). \tag{5}$$

In words, conditional on the two samples carrying distinct true quality levels, $\mathrm{Acc}_{\mathrm{rank}}(r)$ is the probability that r assigns them a strictly correct relative ordering.

Remark on non-degeneracy. Throughout the proof we assume

$$\mathbb{P}(z(x) \neq z(x')) > 0, \tag{6}$$

i.e., the distribution of $z(x)$ is not a point mass on a single quality level. When this fails, no pair of samples carries distinct quality levels and Eq. (5) is undefined; the comparison is vacuous in this case and we exclude it without loss of generality.

Lemma A.1 (Advantage ordering is inherited from reward ordering)**.** *Consider a GRPO rollout group $\{x_i\}_{i=1}^n$ with rewards $r_i = r(x_i)$, and let*

$$A_i(r) = \frac{r_i - \bar{r}}{\sigma_r}, \qquad \bar{r} = \frac{1}{n}\sum_{j=1}^n r_j, \qquad \sigma_r^2 = \frac{1}{n}\sum_{j=1}^n (r_j - \bar{r})^2 \tag{7}$$

denote the group-normalized advantage. If $\sigma_r > 0$, then for any pair of samples x_i, x_j,

$$\mathrm{sign}\left(A_i(r) - A_j(r)\right) = \mathrm{sign}\left(r_i - r_j\right). \tag{8}$$

Consequently, the relative ordering of GRPO update weights coincides exactly with the relative ordering of rewards.

Proof. Subtracting the two normalized advantages cancels the common mean,

$$A_i(r) - A_j(r) = \frac{(r_i - \bar{r}) - (r_j - \bar{r})}{\sigma_r} = \frac{r_i - r_j}{\sigma_r}, \tag{9}$$

and dividing by the positive scalar σ_r preserves the sign. Therefore

$$\mathrm{sign}\left(A_i(r) - A_j(r)\right) = \mathrm{sign}\left(r_i - r_j\right), \tag{10}$$

as claimed. □

Lemma A.2 (The graded reward perfectly preserves the quality ordering)**.** *For the decomposed graded reward*

$$r_c(x) = \frac{z(x)}{K}, \tag{11}$$

the ranking accuracy satisfies

$$\mathrm{Acc}_{\mathrm{rank}}(r_c) = 1. \tag{12}$$

Proof. Since r_c is a strictly increasing affine function of z, for any pair x, x' with $z(x) \neq z(x')$ we have

$$(r_c(x) - r_c(x'))(z(x) - z(x')) = \frac{(z(x) - z(x'))^2}{K} > 0, \tag{13}$$

where the strict inequality uses $K > 0$ and $z(x) \neq z(x')$. Hence the event $(r_c(x) - r_c(x'))(z(x) - z(x')) > 0$ holds with probability one conditional on $z(x) \neq z(x')$, and $\mathrm{Acc}_{\mathrm{rank}}(r_c) = 1$. □

Lemma A.3 (The binary reward only ranks perfect-vs-imperfect pairs)**.** *For the discrete binary reward*

$$r_d(x) = \mathbf{1}[z(x) = K], \tag{14}$$

the event

$$(r_d(x) - r_d(x'))(z(x) - z(x')) > 0 \tag{15}$$

holds if and only if exactly one of the two samples attains the perfect quality level K.

Proof. Conditioning on $z(x) \neq z(x')$, there are exactly three disjoint cases:

(i) Exactly one sample is perfect. Without loss of generality, suppose $z(x) = K$ and $z(x') < K$. Then $r_d(x) - r_d(x') = 1$ and $z(x) - z(x') = K - z(x') > 0$, so the product is strictly positive. The case $z(x') = K, z(x) < K$ is symmetric.

(ii) Both samples are imperfect, i.e. $z(x) < K$ and $z(x') < K$. Then $r_d(x) = r_d(x') = 0$, so $r_d(x) - r_d(x') = 0$ and the product vanishes, even though $z(x) \neq z(x')$.

(iii) Both samples are perfect. This case violates $z(x) \neq z(x')$ and is excluded.

The product is strictly positive only in case (i), which proves the claim. □

Theorem A.4 (Ranking advantage of decomposed graded verification)**.** *Assume the quality distribution $\{q_k\}_{k=0}^K$ is non-degenerate, that is,*

$$1 - \sum_{k=0}^{K} q_k^2 > 0. \tag{16}$$

Then

$$\mathrm{Acc}_{\mathrm{rank}}(r_c) = 1, \qquad \mathrm{Acc}_{\mathrm{rank}}(r_d) = \frac{2q_K(1 - q_K)}{1 - \sum_{k=0}^{K} q_k^2}. \tag{17}$$

In particular,

$$\mathrm{Acc}_{\mathrm{rank}}(r_c) \geq \mathrm{Acc}_{\mathrm{rank}}(r_d), \tag{18}$$

with strict inequality whenever there exist two distinct imperfect quality levels with non-zero probability, i.e., $q_a q_b > 0$ for some $0 \leq a < b < K$.

Proof. The identity $\mathrm{Acc}_{\mathrm{rank}}(r_c) = 1$ is Lemma A.2. It remains to evaluate $\mathrm{Acc}_{\mathrm{rank}}(r_d)$ and to compare the two accuracies.

Step 1: Closed form for $\mathrm{Acc}_{\mathrm{rank}}(r_d)$. Define the events

$$A := \{(r_d(x) - r_d(x'))(z(x) - z(x')) > 0\}, \qquad B := \{z(x) \neq z(x')\}, \tag{19}$$

so that $\mathrm{Acc}_{\mathrm{rank}}(r_d) = \mathbb{P}(A \mid B) = \mathbb{P}(A \cap B)/\mathbb{P}(B)$.

By independence of x and x',

$$\mathbb{P}(B) = 1 - \mathbb{P}(z(x) = z(x')) = 1 - \sum_{k=0}^{K} q_k^2, \tag{20}$$

which is strictly positive by the non-degeneracy assumption. By Lemma A.3, the event $A \cap B$ is exactly the event that exactly one of the two samples attains quality level K. Hence

$$\mathbb{P}(A \cap B) = \mathbb{P}(z(x) = K,\, z(x') < K) + \mathbb{P}(z(x) < K,\, z(x') = K) \tag{21}$$

$$= 2q_K(1 - q_K). \tag{22}$$

Dividing Eq. (22) by Eq. (20) gives the claimed formula for $\mathrm{Acc}_{\mathrm{rank}}(r_d)$.

Step 2: $\mathrm{Acc}_{\mathrm{rank}}(r_d) \leq 1$. Expanding $(\sum_k q_k)^2 = 1$ yields the standard identity

$$1 - \sum_{k=0}^{K} q_k^2 = 2 \sum_{0 \leq a < b \leq K} q_a q_b, \tag{23}$$

and the same expansion gives

$$2q_K(1 - q_K) = 2 \sum_{a=0}^{K-1} q_a q_K. \tag{24}$$

Splitting the unordered pairs (a, b) according to whether $b = K$, we obtain the decomposition

$$\underbrace{1 - \sum_{k=0}^{K} q_k^2}_{\text{denominator}} = \underbrace{2 \sum_{a=0}^{K-1} q_a q_K}_{\text{numerator}} + 2 \sum_{0 \leq a < b < K} q_a q_b. \tag{25}$$

The second term on the right is non-negative, so the denominator is at least as large as the numerator and $\mathrm{Acc}_{\mathrm{rank}}(r_d) \leq 1$. Combined with Lemma A.2, this gives

$$\mathrm{Acc}_{\mathrm{rank}}(r_c) = 1 \geq \mathrm{Acc}_{\mathrm{rank}}(r_d). \tag{26}$$

Step 3: Strictness. From Eq. (25), the inequality is strict if and only if

$$\sum_{0 \leq a < b < K} q_a q_b > 0, \tag{27}$$

which holds precisely when $q_a q_b > 0$ for some $0 \leq a < b < K$, i.e., when at least two distinct imperfect quality levels are visited with positive probability. In this case,

$$\mathrm{Acc}_{\mathrm{rank}}(r_d) < 1 = \mathrm{Acc}_{\mathrm{rank}}(r_c), \tag{28}$$

which completes the proof. □

Corollary A.5 (Collapse of imperfect samples under binary verification)**.** *The ranking error of the discrete binary reward is exactly*

$$1 - \mathrm{Acc}_{\mathrm{rank}}(r_d) = \frac{2 \sum_{0 \leq a < b < K} q_a q_b}{1 - \sum_{k=0}^{K} q_k^2}. \tag{29}$$

Equivalently, every ranking error incurred by the binary reward comes from a pair of imperfect samples with different true quality levels.

Proof. Using the decomposition in Eq. (25),

$$1 - \mathrm{Acc}_{\mathrm{rank}}(r_d) = \frac{\left(1 - \sum_{k=0}^{K} q_k^2\right) - 2 \sum_{a=0}^{K-1} q_a q_K}{1 - \sum_{k=0}^{K} q_k^2} = \frac{2 \sum_{0 \leq a < b < K} q_a q_b}{1 - \sum_{k=0}^{K} q_k^2}. \tag{30}$$

The numerator counts (twice) the probability mass of unordered pairs of distinct *imperfect* quality levels, which by Lemma A.3 are precisely the pairs r_d fails to rank. This proves the corollary. □

Implication for GRPO. By Lemma A.1, GRPO assigns larger update weights to samples with larger rewards. Therefore, a reward model that better preserves the true quality ordering also provides more reliable relative policy updates. The decomposed graded reward is strictly monotonic with respect to the number of satisfied sub-questions and thus preserves all pairwise quality comparisons. In contrast, the binary reward collapses all imperfect samples into the same reward, making GRPO unable to distinguish partially correct samples from severely incorrect ones. Consequently, Decomposed Graded Verification provides a denser and more order-preserving reward signal for GRPO.

B Other Results

B.1 Training Dynamics across Six Game-Based Tasks

Fig. 5 presents the training dynamics of pairwise and pointwise verification across our six constructed rule-governed tasks. Following the evaluation protocol in the main text, we evaluate intermediate checkpoints every 10 training steps on the corresponding validation sets, using Gemini3.1-Pro [11] as an evaluator not involved in data processing and providing ground-truth images as explicit references.

Across all six tasks, pairwise verification generally yields more sustained performance improvements and higher final scores, whereas pointwise verification exhibits more pronounced fluctuations and limited progress. This trend is consistent with the aggregate results in the main text. By directly comparing generated outputs with ground-truth references, pairwise verification reduces reliance on internal target-state inference and can provide more reliable reward signals for policy optimization. These results further support the effectiveness of reference-based verification across different types of rule-governed visual generation tasks.

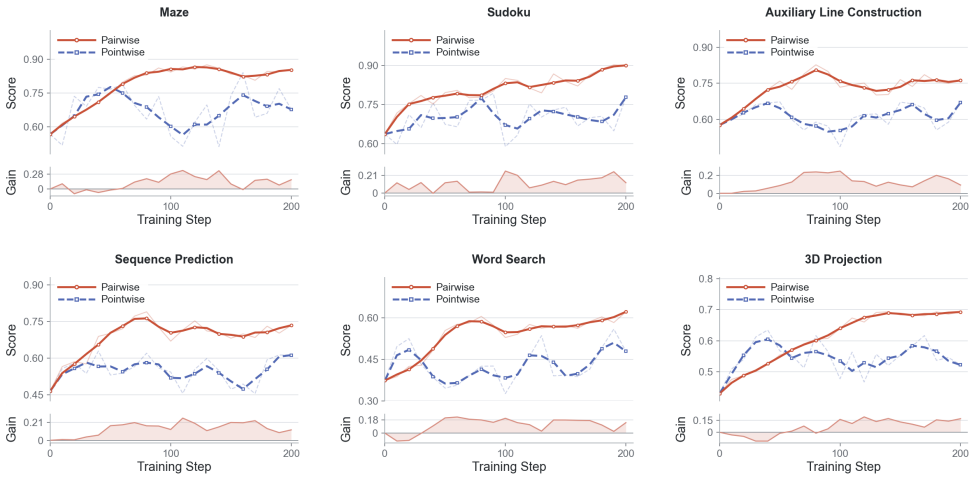


Figure 5: Training dynamics of pairwise and pointwise verification across six rule-governed tasks. Each panel shows validation scores and the score gap between the two methods.

B.2 Sub-question-level Verification Accuracy

The reliability of decomposed graded scoring depends on whether the verifier correctly judges individual sub-questions against the generated image. Even when the decomposition is appropriate, incorrect local judgments can introduce noise into the aggregated reward. We therefore directly evaluate sub-question-level verification accuracy on a held-out set of 200 binary sub-question samples. Three authors and two Ph.D. students each annotate 40 samples, collectively providing human labels for all 200 samples. We additionally obtain binary annotations from GPT-5.5 and Gemini-3.1-Pro on the same evaluation set. We compare the base model, the SFT-trained graded verifier, and the graded verifier further optimized with Decoupled RL, reporting accuracy against each annotation source separately.

Table 5: Sub-question-level verification accuracy (%) on 200 held-out samples.

Methods	**Annotation Source**		
	GPT-5.5	**Gemini-3.1-Pro**	**Human**
Qwen3-VL-8B	68.5	69.5	67.0
+ SFT(Graded)	76.0	76.5	74.5
+ SFT(Graded) + RL(Graded)	**94.5**	**96.0**	**94.0**

As shown in Table 5, Decoupled RL substantially improves fine-grained verification accuracy across all three annotation sources. These results support the effectiveness of Decoupled RL in improving

sub-question-level binary judgments and provide empirical evidence for the reliability of the local decisions underlying decomposed graded scoring.

B.3 Evaluation on Qwen-Image-2512

To examine whether the effectiveness of our Decomposed Graded Verifier extends to a larger text-to-image generation backbone, we further evaluate Qwen-Image-2512 (20B), complementing the experiments on Stable Diffusion 3.5 Medium (2.5B) and FLUX.1-dev (12B). We compare the original model with its reinforcement learning variants guided by the Discrete Binary Verifier and the Decomposed Graded Verifier, respectively, using GenEval and CLIPScore for evaluation.

Table 6: Performance of Qwen-Image-2512 on GenEval and CLIPScore under different reward models.

Methods	**GenEval**							**CLIPScore**
	Single Obj.	Two Obj.	Counting	Colors	Position	Attri. Bind.	Overall	
Qwen-Image-2512	0.99	0.90	0.77	0.86	0.38	0.52	0.74	34.1
+RL(Discrete Binary Verifier)	1.00	0.92	0.79	0.87	0.42	0.55	0.76	34.4
+RL(Decomposed Graded Verifier)	**1.00**	**0.96**	**0.85**	**0.92**	**0.51**	**0.69**	**0.82**	**35.3**

As shown in Table 6, the Decomposed Graded Verifier improves the overall GenEval score from 0.74 to 0.82, outperforming the binary-reward counterpart by 0.06. The largest gains over the original model occur in attribute binding and position, which improve from 0.52 to 0.69 and from 0.38 to 0.51, respectively. These improvements suggest that fine-grained graded rewards are particularly useful for learning compositional requirements involving attribute associations and spatial relations. CLIPScore also increases from 34.1 to 35.3, compared with 34.4 under binary rewards. Together, these results extend the evidence for the effectiveness of decomposed graded rewards to a larger generation backbone with stronger baseline performance.

B.4 Evaluation on InternVL3-8B

To examine whether the benefits of our training strategy extend beyond the Qwen model family, we additionally adopt InternVL3-8B as the verifier backbone. We evaluate the base model, its graded SFT variant, and the graded verifier further optimized with Decoupled RL on ViVerBench.

Table 7: Performance on ViVerBench with InternVL3-8B as the verifier backbone. We report seven selected sub-metrics with notable improvements after Decoupled RL. Overall denotes the mean score across all 16 task categories, including those not displayed.

Methods	**Concept Existence**		**Object Relation**		**Image Annotation**			**Overall (16 tasks)**
	Obj.	Attr.	Spat.	N-Spat.	BBox	Point	Count	
InternVL3-8B	0.513	0.578	0.667	0.619	0.750	0.619	0.604	0.594
+SFT(Graded)	0.526	0.585	0.652	0.632	0.757	0.611	0.618	0.606
+SFT(Graded)+RL(Graded)	**0.678**	**0.684**	**0.707**	**0.724**	**0.821**	**0.665**	**0.649**	**0.652**

As shown in Table 7, Decoupled RL improves the mean score across all 16 task categories from 0.606 to 0.652 compared with graded SFT, with the seven displayed sub-metrics highlighting selected categories with notable gains. These results demonstrate that our training strategy is also effective on InternVL3-8B, extending its applicability beyond the Qwen family.

B.5 Sensitivity to the Number of Sub-questions

To investigate how decomposition granularity affects global score estimation, we evaluate the trained Decomposed Graded Verifier on 200 held-out test samples. We explicitly instruct the verifier through the system prompt to generate sub-questions within three ranges: at most 10, 11–20, and 21–30. This experiment examines whether increasing the number of sub-questions consistently improves scoring accuracy or introduces additional errors through excessive decomposition.

As shown in Table 8, the 11–20 setting achieves the lowest mean alignment error of 0.039, compared with 0.083 for at most 10 sub-questions and 0.072 for 21–30 sub-questions. These results suggest that an appropriate decomposition granularity improves scoring accuracy, while simply increasing the number of sub-questions does not guarantee better performance.

Table 8: Effect of the number of sub-questions on global score estimation on 200 held-out test samples. Lower mean alignment error is better.

Number of Sub-questions	Mean Alignment Error ↓
≤ 10	0.083
11–20	**0.039**
21–30	0.072

Excessive decomposition can introduce redundant questions or requirements unsupported by the original prompt. For example, given "A red apple is placed on a table," the verifier may repeatedly ask whether an apple exists or introduce questions such as "Is the apple a fruit?" and "Is the apple round?" that do not correspond to explicitly requested visual constraints. Such questions can distort the aggregated score by repeatedly weighting the same requirement or introducing irrelevant criteria. We also observe that generating more sub-questions can lead to violations of the required output format, reducing the reliability of the final score. Together, these observations highlight the importance of balancing fine-grained coverage with faithful and non-redundant decomposition.

C Limitations and Future Works

Our world-modeling evaluation is restricted to game-based environments with explicit rules, where the ground-truth image is uniquely determined by the initial task state and rule instructions. We adopt these controlled tasks primarily to evaluate the effectiveness of our pairwise verification paradigm. Although our results demonstrate the advantages of pairwise verification in these settings, these tasks capture only part of the complexity and diversity of real-world environments.

In future work, we plan to extend our investigation to open-ended settings with non-unique ground truths, construct larger and more realistic world-modeling datasets, and scale both the verifier and the generator to stronger backbones. Our long-term goal is to develop multimodal generative models that can reason about, simulate, and interact with complex real-world scenes.

D Decomposition Example

Text-to-Image (1)



Prompt:

A baboon is silhouetted on a patio in a top-down, monochromatic view; acacia trees stark against the golden hour dusk.

Decomposition Result:

- Is there a baboon present in the image? → Yes, Score: 1
- Is the baboon depicted in silhouette? → Yes, Score: 1
- Is the baboon situated on a patio or tiled surface? → Yes, Score: 1
- Is the camera perspective a top-down view? → No, Score: 0
- Is the image monochromatic? → No, Score: 0
- Are there acacia trees visible in the background? → Yes, Score: 1

- Is the scene set during golden hour dusk? → Yes, Score: 1

Final Alignment Score:

$$\frac{5}{7} = 0.71$$

Text-to-Image (2)



Prompt:

Isometric 3D café interior, Pixar-style rendering, whimsical aesthetic. Features coffee counter, colorful shelves, rounded furniture, pastry case, and hanging lamps. Large glass door opens to a park with string lights at sunset. Warm glowing lighting, soft curved edges, cozy evening atmosphere.

Decomposition Result:

- Is the image a 3D rendering of a café interior with a Pixar-style aesthetic? → Yes, Score: 1
- Is the viewpoint isometric? → No, Score: 0
- Does the interior feature a coffee counter? → Yes, Score: 1
- Are there shelves with colorful items? → Yes, Score: 1
- Are the furniture and objects rounded with soft curved edges? → Yes, Score: 1
- Does the café include both hanging lamps and a pastry case? → Yes, Score: 1
- Does a large glass door open to an outdoor park scene? → Yes, Score: 1
- Are string lights visible in the outdoor park scene? → Yes, Score: 1
- Is the scene set at sunset with warm, glowing, cozy lighting? → Yes, Score: 1

Final Alignment Score:

$$\frac{8}{9} = 0.89$$

Text-to-Image (3)



Prompt:

Imagine an enchanted city where the streets resonate like symphonies, and the architecture is formed from orchestral instruments and glimmering harmonies.

Decomposition Result:

- Does the image show an enchanted city? → Yes, Score: 1
- Is there a clear city street scene? → Yes, Score: 1
- Are musical notes or symphony elements visible? → Yes, Score: 1
- Are orchestral instruments visible? → Yes, Score: 1
- Are the instruments integrated into the cityscape? → Yes, Score: 1
- Is the architecture formed from instruments? → No, Score: 0
- Are glimmering lights or particles visible? → Yes, Score: 1
- Are the glimmering elements linked to the instruments? → Yes, Score: 1
- Does the scene combine city, music, and magic coherently? → Yes, Score: 1

Final Alignment Score:

$$\frac{8}{9} = 0.89$$

Text-to-Image (4)



Prompt:

A retro-themed photograph of a medieval castle resting atop a colossal tree trunk, enveloped by clouds, shimmering fireflies, and a dusky twilight sky. The scene is illuminated by fireflies.

Decomposition Result:

- Does the image depict a medieval castle? → Yes, Score: 1
- Is the castle resting atop a colossal tree trunk? → Yes, Score: 1
- Is the scene enveloped by clouds? → Yes, Score: 1
- Are shimmering fireflies visible in the image? → Yes, Score: 1
- Is the sky a dusky twilight sky? → Yes, Score: 1
- Is the image a retro-themed photograph? → No, Score: 0
- Is the scene illuminated by the fireflies? → No, Score: 0

Final Alignment Score:

$$\frac{5}{7} = 0.71$$

Text-to-Image (5)



Prompt:

In a bustling train station, a traveler in a dark olive jacket and maroon scarf wears a well-worn black backpack. They focus on a tablet, likely checking schedules. Nearby, a polished blue train waits at the platform, ready to welcome passengers.

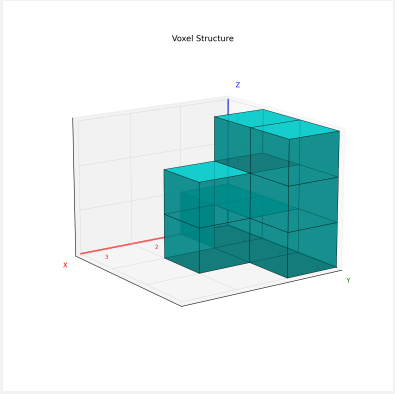
Decomposition Result:

- Is the scene set in a bustling train station? → Yes, Score: 1
- Is the traveler wearing a dark olive jacket? → Yes, Score: 1
- Is the traveler wearing a maroon scarf? → Yes, Score: 1
- Is the traveler wearing a well-worn black backpack? → No, Score: 0
- Is the traveler focusing on a tablet? → Yes, Score: 1
- Is there a polished blue train waiting at the platform? → Yes, Score: 1
- Is the train's door open to welcome passengers? → Yes, Score: 1

Final Alignment Score:

$$\frac{6}{7} = 0.86$$

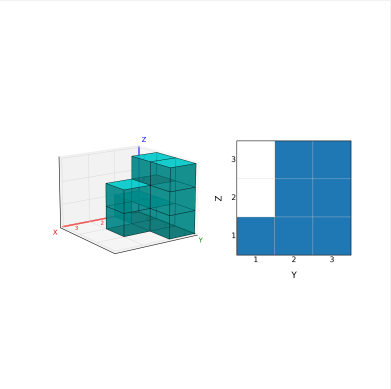
Rule-Governed World Worlding Pairwise



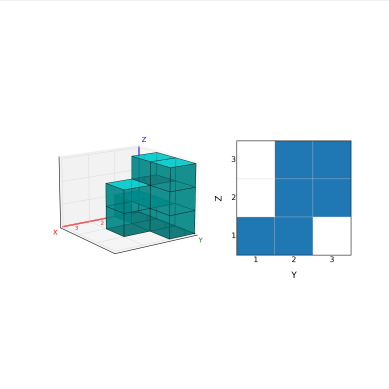
Initial Image

Game Rule:

The objective is to generate a 2D orthographic front-view projection of the given 3D voxel structure. The front view is obtained by looking along the X-axis and projecting the structure onto the Y-Z plane.



Reference Image



Generated Image

Decomposition Reasoning:

So, let's tackle this step by step.

The task is to evaluate whether the generated image correctly produces the 2D orthographic front-view projection of the given 3D voxel structure. Since the front view is obtained by looking along the X-axis, the projection should be represented on the Y-Z plane.

I use the Reference Image as the correct target and compare the Generated Image with it cell by cell. Each position in the 3×3 Y-Z grid can be treated as a small sub-question: whether the generated cell matches the corresponding reference cell.

The Reference Image shows the projection as follows:

- $Z = 3$: empty, filled, filled
- $Z = 2$: empty, filled, filled
- $Z = 1$: filled, filled, filled

Based on the Reference Image, I decompose the comparison into the following cell-level checks:

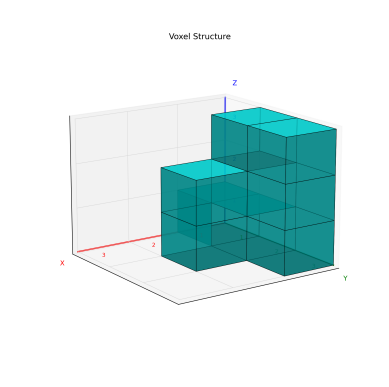
- Is the cell at $(Y = 1, Z = 1)$ correct? The Reference Image fills this cell, and the Generated Image also fills it. $\rightarrow$ Yes, Score: 1
- Is the cell at $(Y = 2, Z = 1)$ correct? The Reference Image fills this cell, and the Generated Image also fills it. $\rightarrow$ Yes, Score: 1
- Is the cell at $(Y = 3, Z = 1)$ correct? The Reference Image fills this cell, but the Generated Image leaves it empty. $\rightarrow$ No, Score: 0
- Is the cell at $(Y = 1, Z = 2)$ correct? The Reference Image leaves this cell empty, and the Generated Image also leaves it empty. $\rightarrow$ Yes, Score: 1
- Is the cell at $(Y = 2, Z = 2)$ correct? The Reference Image fills this cell, and the Generated Image also fills it. $\rightarrow$ Yes, Score: 1
- Is the cell at $(Y = 3, Z = 2)$ correct? The Reference Image fills this cell, and the Generated Image also fills it. $\rightarrow$ Yes, Score: 1
- Is the cell at $(Y = 1, Z = 3)$ correct? The Reference Image leaves this cell empty, and the Generated Image also leaves it empty. $\rightarrow$ Yes, Score: 1
- Is the cell at $(Y = 2, Z = 3)$ correct? The Reference Image fills this cell, and the Generated Image also fills it. $\rightarrow$ Yes, Score: 1
- Is the cell at $(Y = 3, Z = 3)$ correct? The Reference Image fills this cell, and the Generated Image also fills it. $\rightarrow$ Yes, Score: 1

From these sub-question judgements, the Generated Image matches the Reference Image in 8 out of 9 cells. The only mismatch is at $(Y = 3, Z = 1)$, where the Reference Image shows an empty cell but the Generated Image marks it as filled.

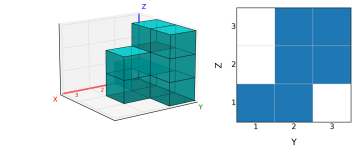
Thus, there are 9 sub-questions in total, with 8 correct and 1 incorrect.

$$\text{Score} = \frac{8}{9} = 0.89.$$

Rule-Governed World Worlding Pointwise



Initial Image



Generated Image

Decomposition Reasoning:

So, let's first work out the target projection implied by the initial 3D structure.

The required view is produced by observing the voxel structure along the X-axis. In this process, the X dimension is removed, and the remaining projection is represented on the Y-Z plane, with Y as the horizontal axis and Z as the vertical axis.

Thus, for every (Y, Z) coordinate, I need to decide whether the projection cell should be occupied. If at least one voxel exists somewhere along the X direction at that same (Y, Z) coordinate, the cell should be filled; otherwise, it should be empty.

By inspecting the initial 3D structure, the expected occupancy pattern is:

- For $Y = 1$, the structure occupies $Z = 1$ and $Z = 2$.
- For $Y = 2$, the structure occupies $Z = 1$, $Z = 2$, and $Z = 3$.
- For $Y = 3$, the structure occupies $Z = 1$, $Z = 2$, and $Z = 3$.

So the inferred target projection should mark the following cells as filled:

$$(Y = 1, Z = 1),\ (Y = 1, Z = 2),$$
$$(Y = 2, Z = 1),\ (Y = 2, Z = 2),\ (Y = 2, Z = 3),$$
$$(Y = 3, Z = 1),\ (Y = 3, Z = 2),\ (Y = 3, Z = 3).$$

Next, I break the projection into individual cell-level questions and compare the generated result with this inferred target.

- Is $(Y = 1, Z = 1)$ correctly represented? The inferred target requires this cell to be filled, and the generated projection also fills it. $\rightarrow$ Yes, Score: 1
- Is $(Y = 1, Z = 2)$ correctly represented? The inferred target requires this cell to be filled, but the generated projection leaves it empty. $\rightarrow$ No, Score: 0
- Is $(Y = 1, Z = 3)$ correctly represented? The inferred target requires this cell to be empty, and the generated projection also leaves it empty. $\rightarrow$ Yes, Score: 1
- Is $(Y = 2, Z = 1)$ correctly represented? The inferred target requires this cell to be filled, and the generated projection also fills it. $\rightarrow$ Yes, Score: 1
- Is $(Y = 2, Z = 2)$ correctly represented? The inferred target requires this cell to be filled, and the generated projection also fills it. $\rightarrow$ Yes, Score: 1
- Is $(Y = 2, Z = 3)$ correctly represented? The inferred target requires this cell to be filled, and the generated projection also fills it. $\rightarrow$ Yes, Score: 1
- Is $(Y = 3, Z = 1)$ correctly represented? The inferred target requires this cell to be filled, but the generated projection leaves it empty. $\rightarrow$ No, Score: 0
- Is $(Y = 3, Z = 2)$ correctly represented? The inferred target requires this cell to be filled, and the generated projection also fills it. $\rightarrow$ Yes, Score: 1
- Is $(Y = 3, Z = 3)$ correctly represented? The inferred target requires this cell to be filled, and the generated projection also fills it. $\rightarrow$ Yes, Score: 1

After aggregating these judgements, the generated projection correctly represents most of the inferred Y-Z occupancy pattern. The two failed sub-questions correspond to $(Y = 1, Z = 2)$ and $(Y = 3, Z = 1)$, where the generated projection leaves the cells empty even though they should be filled.

Therefore, among the 9 decomposed sub-questions, 7 are judged as correct and 2 are judged as incorrect.

$$\text{Score} = \frac{7}{9} = 0.78.$$

Text-to-Image Verifier Prompt

You are a powerful and trustworthy **Text-to-Image Alignment Evaluator**.

Your responsibility is to evaluate the semantic alignment between a generated image and a text prompt. You must decompose the text prompt into verifiable atomic questions and score the consistency between the image and the prompt based strictly on visual evidence.

Inputs

You will be given the following inputs:

- **Text Prompt**: the textual description used to generate the image.
- **Generated Image**: the image produced by the text-to-image model.

Evaluation Workflow

Step 1: Prompt Decomposition

Decompose the text prompt into a set of atomic questions that can be directly verified from the image. Each question must originate from facts explicitly stated in the text prompt and test one clear semantic assertion.

Step 2: Image-Based Answering

Answer each question solely based on the visual information present in the Generated Image. Do not introduce assumptions beyond the text prompt. Do not infer visual information that is absent, ambiguous, or cannot be confirmed from the image.

Step 3: Consistency Judgement and Scoring

For each question, determine whether the image-based answer is consistent with the information explicitly specified in the text prompt. If the image-based answer is consistent with the prompt, the question receives 1 point. Otherwise, it receives 0 points.

Question Decomposition Principles

The decomposed questions must satisfy the following requirements:

- Each question must be binary, with a Yes or No answer.
- Each question must be objectively verifiable through visual evidence.
- Questions involving subjective judgment, emotions, aesthetics, or other non-verifiable properties are forbidden.
- The number of questions should be naturally determined by the semantic complexity of the text prompt.

The decomposition should follow the semantic order below when applicable:

- **Objects**: existence, quantity, and category.
- **Object Attributes**: color, shape, material, appearance, actions, and visible states.
- **Spatial Relations**: relative position, orientation, and layout.
- **Non-Spatial Relations**: interactions and action relations between entities.

Output Requirements

Output only a valid JSON object containing exactly two top-level fields: "sub_questions" and "final_score". Do not include Markdown code fences or additional text.

```
{
    "sub_questions": [
        {
            "id": 1,
            "question": "A decomposed sub-question",
            "answer": "True or False",
            "explanation": "Brief error explanation if False; otherwise an empty string"
        }
    ],
    "final_score": x
}
```

Replace the placeholders above with actual evaluation results. Include one entry for each distinct decomposed sub-question, with consecutive IDs starting at 1.

Here, $x = C/N$, where N is the total number of entries and C is the number answered "True". Output x as a JSON number rounded to two decimal places.

Inputs for This Evaluation

Text Prompt: $ { prompt }

The attached image is the Generated Image. <image>

Rule-Governed World Modeling Verifier Prompt-Pairwise

You are a powerful and trustworthy **Image Editing Evaluator**.

Your responsibility is to evaluate whether an image editing model successfully completes a game-based image editing task according to the provided rules.

Inputs

You will be given the following inputs:

- **Initial Image**: the original game state given to the image editing model.
- **Generated Image**: the output produced by the image editing model.
- **Reference Image**: the ground-truth result that correctly follows all rules.
- **Game Rules**: the rules governing the game and editing constraints.

Evaluation Workflow

Step 1: Rule Decomposition

Independently decompose the Game Rules into atomic, visually checkable rules, even if they are already decomposed. The resulting rules must remain consistent with the provided Game Rules.

Step 2: Rule Violation Checking

Use the Reference Image as the correct standard. **Compare the Generated Image with the Reference Image** rule by rule, and determine whether each rule is violated.

Step 3: Scoring

Let the total number of decomposed atomic rules be N, and let the number of violated rules be V. The score of the Generated Image is:

$$\text{Score} = \frac{N - V}{N}.$$

The final score should reflect how consistently the Generated Image follows the rules compared with the Reference Image.

Output Requirements

Output only a valid JSON object with exactly two top-level fields: "sub_questions" and "final_score". Do not include Markdown code fences or additional text.

```
{
    "sub_questions": [
        {
            "id": 1,
            "question": "A decomposed rule sub-question",
            "answer": "True or False",
            "explanation": "Brief rule-violation explanation if False; otherwise an empty string"
        }
    ],
    "final_score": x
}
```

Replace the placeholders above with actual evaluation results. Include one entry for each distinct decomposed atomic rule, with consecutive IDs starting at 1.

Here, x is the score calculated in Step 3, expressed as a JSON number rounded to two decimal places.

Inputs for This Evaluation

Game Rules: $ { game_rules }

Images are attached in this order: Initial Image, Generated Image, Reference Image. <image><image><image>

Rule-Governed World Modeling Verifier Prompt-Pointwise

You are a powerful and trustworthy **Image Editing Evaluator**.

Your responsibility is to evaluate whether an image editing model successfully completes a game-based image editing task according to the provided rules.

Inputs

You will be given the following inputs:

- **Initial Image**: the original game state given to the image editing model.
- **Generated Image**: the output produced by the image editing model.
- **Game Rules**: the rules governing the game and editing constraints.

Evaluation Workflow

Step 1: Rule Decomposition

Independently decompose the Game Rules into atomic, visually checkable rules, even if they are already decomposed. The resulting rules must remain consistent with the provided Game Rules.

Step 2: Rule Violation Checking

Use the Initial Image and the Game Rules to **infer the expected final state**. **Compare the Generated Image with this expected state** rule by rule, and determine whether each rule is violated.

Step 3: Scoring

Let the total number of decomposed atomic rules be N, and let the number of violated rules be V. The score of the Generated Image is:

$$\text{Score} = \frac{\mathrm{N} - \mathrm{V}}{\mathrm{N}}.$$

The final score should reflect how consistently the Generated Image follows the Game Rules given the Initial Image.

Output Requirements

Output only a valid JSON object with exactly two top-level fields: "sub_questions" and "final_score". Do not include Markdown code fences or additional text.

```
{
    "sub_questions": [
        {
            "id": 1,
            "question": "A decomposed rule sub-question",
            "answer": "True or False",
            "explanation": "Brief rule-violation explanation if False; otherwise an empty string"
        }
    ],
    "final_score": x
}
```

Replace the placeholders above with actual evaluation results. Include one entry for each distinct decomposed atomic rule, with consecutive IDs starting at 1.

Here, x is the score calculated in Step 3, expressed as a JSON number rounded to two decimal places.

Inputs for This Evaluation

Game Rules: $ { game_rules }

Images are attached in this order: Initial Image, Generated Image. <image><image>

Text-to-Image Sub-Question Verifier Prompt

Evaluate the following sub-question about the Generated Image.

Please carefully analyze the Generated Image and determine whether it satisfies the requirement expressed in the sub-question. Base your judgment solely on the visual evidence in the Generated Image.

If the Generated Image satisfies this requirement, answer "True"; otherwise, answer "False". Evaluate only the requirement expressed in this sub-question.

Respond strictly with a JSON object containing exactly these two fields:

```
{
    "answer": "True or False",
    "explanation": "Brief error explanation if False; otherwise an empty string"
}
```

Use the string "True" or "False" for "answer". If the answer is "False", briefly describe the specific mismatch or missing visual evidence in "explanation"; otherwise, use an empty string. Do not include any additional text.

Inputs for This Evaluation

Sub-question: $ { question }

The attached image is the Generated Image. <image>

Rule-Governed World Modeling Sub-Question Verifier Prompt-Pairwise

Evaluate the following sub-question about the Generated Image.

Please carefully **compare the Generated Image with the Reference Image** and determine whether the Generated Image satisfies the requirement expressed in the sub-question. Use the Reference Image as the correct standard, with the Initial Image and Game Rules providing the task context.

If the Generated Image satisfies this requirement, answer "True"; otherwise, answer "False". Evaluate only the requirement expressed in this sub-question.

Respond strictly with a JSON object containing exactly these two fields:

```
{
    "answer": "True or False",
    "explanation": "Brief error explanation if False; otherwise an empty string"
}
```

Use the string "True" or "False" for "answer". If the answer is "False", briefly describe the specific mismatch or missing visual evidence in "explanation"; otherwise, use an empty string. Do not include any additional text.

Inputs for This Evaluation

Sub-question: $ { question }

Game Rules: $ { game_rules }

Images are attached in this order: Initial Image, Generated Image, Reference Image. <image><image><image>

Rule-Governed World Modeling Sub-Question Verifier Prompt-Pointwise

Evaluate the following sub-question about the Generated Image.

Please analyze the Initial Image, then apply the Game Rules to **infer the expected final state**. Carefully **compare the Generated Image with this expected state** and determine whether the Generated Image satisfies the requirement expressed in the sub-question.

If the Generated Image satisfies this requirement, answer "True"; otherwise, answer "False". Evaluate only the requirement expressed in this sub-question.

Respond strictly with a JSON object containing exactly these two fields:

```
{
    "answer": "True or False",
    "explanation": "Brief error explanation if False; otherwise an empty string"
}
```

Use the string "True" or "False" for "answer". If the answer is "False", briefly describe the specific mismatch or missing visual evidence in "explanation"; otherwise, use an empty string. Do not include any additional text.

Inputs for This Evaluation

Sub-question: $ { question }

Game Rules: $ { game_rules }

Images are attached in this order: Initial Image, Generated Image. <image><image>

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

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

   Answer: [Yes]

   Justification: Please see the details in the abstract and introduction sections.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: Please refer to the Appendix C.

Guidelines:

- The answer [N/A] 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: [Yes]

   Justification: Please refer to the details in Sections 3 and 4.

   Guidelines:

   - The answer [N/A] 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: Please refer to the details in Sections 3.2 and 5.

   Guidelines:

   - The answer [N/A] 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: Please see the details in supplemental material.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to the details in Sections 3.2 and 5.

   Guidelines:

   - The answer [N/A] 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: [Yes]

   Justification: Please refer to the details in 5.

   Guidelines:

   - The answer [N/A] 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: Please refer to the details in Sections 3.2 and 5.

   Guidelines:

   - The answer [N/A] 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: Please see the whole paper and supplemental material.

   Guidelines:

   - The answer [N/A] 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: [N/A]

    Justification: There is no societal impact of the work performed.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] 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: [Yes]

    Justification: Please see the whole paper and supplemental material.

    Guidelines:

    - The answer [N/A] 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: [Yes]

    Justification: Please see the whole paper and supplemental material.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] 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: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.