Can We Generate Images with CoT? Let's Verify and Reinforce Image Generation Step by Step

Renrui Zhang*†[1], Chengzhuo Tong*[4], Zhizheng Zhao*[3], Ziyu Guo*[2], Haoquan Zhang[4]
Manyuan Zhang[1], Peng Gao[4], Hongsheng Li‡[1]

CUHK [1]MMLab & [2]MiuLar Lab [3]Peking University [4]Shanghai AI Lab

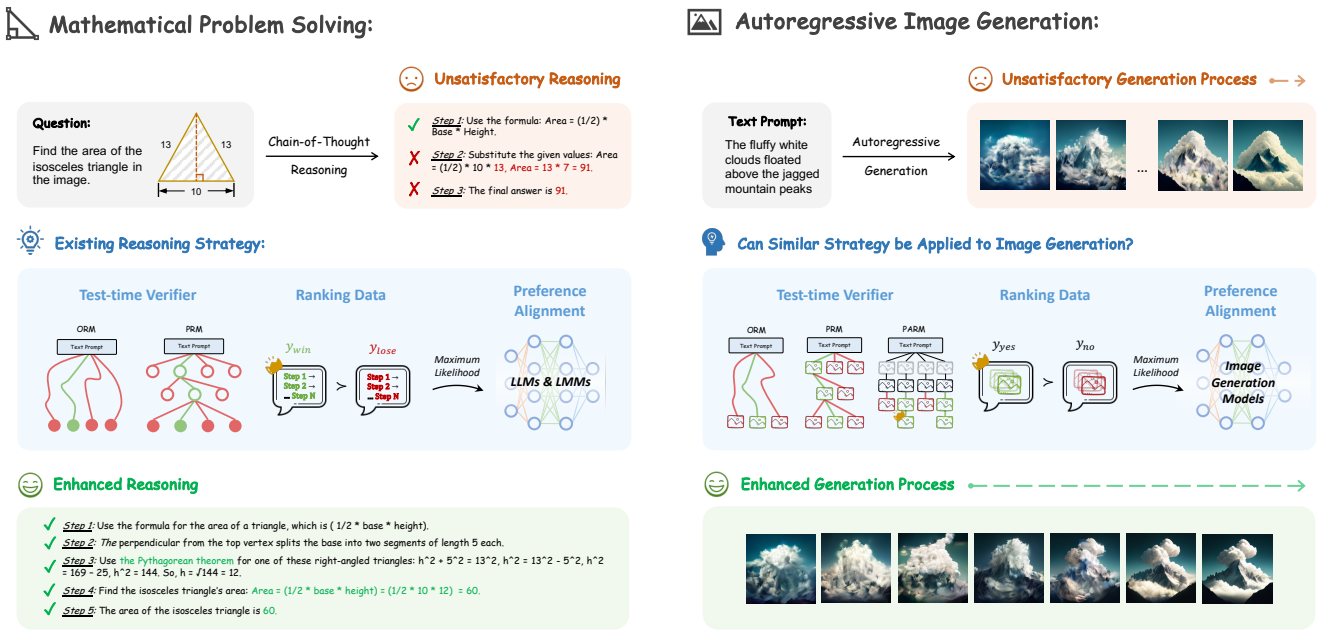

Figure 1. ***Can We Verify and Reinforce Image Generation with Chain-of-Thought (CoT) Reasoning Strategies?*** Given the success of mathematical CoT reasoning in LLMs [36, 74] and LMMs [83, 84] (Left), we provide *the first* investigation to comprehensively explore the potential of applying current reasoning techniques to autoregressive image generation (Right), including test-time verification and preference alignment.

Abstract

*Chain-of-Thought (CoT) reasoning has been extensively explored in large models to tackle complex understanding tasks. However, it still remains an open question whether such strategies can be applied to verifying and reinforcing image generation scenarios. In this paper, we provide the first comprehensive investigation of the potential of CoT reasoning to enhance autoregressive image generation. We focus on three techniques: scaling test-time computation for verification, aligning model preferences with Direct Preference Optimization (DPO), and integrating these techniques for complementary effects. Our results demonstrate that these approaches can be effectively adapted and combined to significantly improve image generation performance. Furthermore, given the pivotal role of reward models in our findings, we propose the **P**otential **A**ssessment **R**eward **M**odel (**PARM**) specialized for autoregressive image generation. PARM adaptively assesses each generation step through a potential assessment mechanism, merging the strengths of existing reward models. Using our investigated reasoning strategies, we enhance a baseline model, Show-o, to achieve superior results, with a significant +24% improvement on the GenEval benchmark, surpassing Stable Diffusion 3 by +15%. We hope our study provides unique insights and paves a new path for integrating CoT reasoning with autoregressive image generation.*

* Equal Contribution † Project Leader ‡ Corresponding Author

Figure 2. **Autoregressive Image Generation without (Top) and with (Bottom) Our Reasoning Strategies.** We adopt Show-o [75] as the baseline model that produces unsatisfactory text-to-image generation. After using our investigated reasoning strategies (integrating PARM with iterative DPO for both reward model guidance and test-time verification), the generation process and results are effectively enhanced.

1. Introduction

Large Language Models (LLMs) [4, 67, 68] and Large Multimodal Models (LMMs) [17, 47, 82] have gained remarkable achievements across language [10, 45], 2D image [14, 38], video [15, 26], and 3D [23, 25]. Building upon general understanding skills, recent efforts have been made toward enhancing LLMs and LMMs with complex Chain-of-Thought (CoT) reasoning capabilities [30, 72, 74, 86], e.g., OpenAI o1 [48], contributing to superior performance in mathematics [39, 83], science [24, 61], and coding [22, 89].

Despite the success in multimodal understanding, it remains under-explored *whether multi-step reasoning strategies can be effectively applied to image generation*. Considering the discrepancy between two tasks, we observe that, autoregressive image generation [6, 64, 73, 75] shares a similar output manner to the nature of LLMs and LMMs. Specifically, they both quantize the target data (language and image) into discrete tokens, and iteratively predict partial content conditioned on previously generated tokens.

As illustrated in Figure 1, LMMs leverage CoT to break down complex mathematical problems into manageable steps, which enables scaling test-time computation with reward models [36, 42, 63, 70] and reinforcement learning for preference alignment [27, 33, 40, 84]. Likewise, autoregressive image generation through step-by-step decoding can produce intermediate images, potentially allowing for similar verification and reinforcement techniques. This raises the question: *Can we verify and reinforce image generation step-by-step with strategies revealed by OpenAI o1?*

To this end, we conduct a systematic investigation into the potential of CoT reasoning for autoregressive image generation. We adopt Show-o [75], a latest discrete generative model, as our baseline, and evaluate on a challenging text-to-image generation benchmark: GenEval [20]. Specifically, we focus on examining two key perspectives: *1) Scaling test-time computation with Outcome/Process Reward Model (ORM/PRM) as verifiers*; and *2) Reinforced preference alignment with Direct Preference Optimization (DPO)*. The specifics are as follows:

- **ORM *vs* PRM as Test-time Verifiers.** As the top-1 result may not always be reliable, reward models are employed to score sampled candidates and perform outcome selection, where ORM is instance-level and PRM is process-level. In our settings, the score assesses whether each candidate image is inherently reasonable and aligns with the given textual prompt. We prompt LLaVA-OneVision (7B) [34] as a zero-shot reward model, and then curate text-to-image ranking data for reward fine-tuning. We apply a best-of-N selection approach in the comparison of zero-shot and fine-tuned reward models.

  Observation: *ORM demonstrates significant improvement, while PRM offers minimal benefit.*

- **Test-time Verifiers *vs* Preference Alignment.** Exploring the trade-off between inference-time and post-training offers valuable insights into the model's attainable performance. Preference alignment are adopted to elicit the implicit reasoning capabilities from their widely learned knowledge. In this study, we construct ranking preference data and apply DPO alignment with iterative training [53] to optimize the generation decoding process, comparing its effectiveness to test-time verification.

  Observation: *DPO alignment with iterative training attains stronger results to the fine-tuned ORM verifier.*

- **Preference Alignment *plus* Test-time Verifiers.** After investigating the individual impact, we integrate the two techniques to highlight their complementary potential in autoregressive image generation. We consider three approaches: *1) DPO with reward model guidance*, i.e., integrating DPO's policy with reward models' objectives for alignment; *2) Verification after DPO alignment*, i.e., ap-

plying reward models for best-of-N selection on DPO-aligned models; and *3) Verification after DPO with reward model guidance*, i.e., a combination of *1* and *2*.

Observation: *All integration methods lead to greater improvements, indicating complementary characteristics.*

Through our experiments, we demonstrate the promsing potential of applying CoT reasoning strategies to image generation scenarios, uncovering their adaptation methods and compatibility. Furthermore, we identify significant room for improvement in reward models tailored for autoregressive image generation. For ORM, the global-level assessments are unable to capture the nuanced step-wise information, leading to inaccurate reward judgments. For PRM, the early-stage images tend to appear blurry, while later-stage images across different paths often converge to visually similar outputs, limiting its discrimination ability.

To alleviate these issues, we propose a specialized reward model for autoregressive image generation, termed **P**otential **A**ssessment **R**eward **M**odel (**PARM**). PARM adaptively verifies the generation process step by step with three delicately designed tasks: *1) Judge which step is clear and convincing enough to be evaluated*, given that most early-stage images are typically blurry; *2) Assess whether the current step has the potential to yield a high-quality final image*, since later-stage images generally do not change too much; and *3) Score the remaining final paths for selecting the best one*, similar to an ORM. In this way, PARM can adaptively conduct assessment at appropriate steps (overcoming PRM's scoring challenges), while effectively capturing fine-grained step-by-step cues (complementing the coarse evaluation of ORM). Our experiments showcase that PARM significantly outperforms both ORM and PRM, which finally improves the baseline model by +24% on GenEval, as visualized in Figure 2, surpassing the advanced Stable Diffusion 3 [13] by +15%.

Our main contributions are summarized as follows:

- We present *the first* comprehensive empirical study of applying CoT reasoning strategies to autoregressive image generation domains, providing unique insights into the future advancement of this field.
- We investigate specific adaption methods of techniques, including test-time verification, preference alignment, and ranking data curation, to autoregressive image generation, indicating their performance and complementarity.
- We further introduce PARM, a new reward model tailored for image generation scenarios, which adaptively performs step-wise potential assessment and path selection, significantly enhancing text-to-image generation quality.

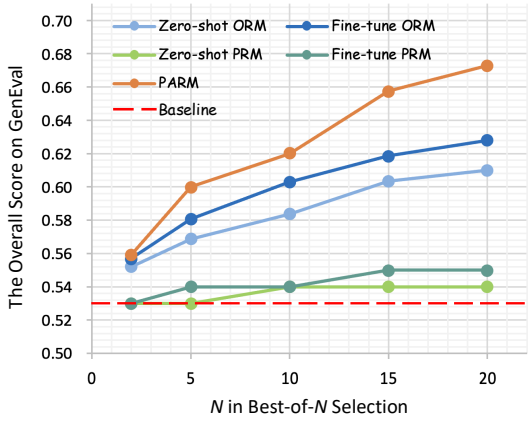

Figure 3. **Comparison of Reward Models as Test-time Verifiers.** We adopt Show-o [75] as the 'Baseline' and evaluate Best-of-N selection on the GenEval [20] benchmark.

2. Our Investigation

Chain-of-Thought (CoT) reasoning has been widely exploited to solve complex problems for language and multimodal understanding. In this study, we conduct a systematic investigation aiming to find out, whether we can verify and reinforce image generation step-by-step.

2.1. Overview

Task Formulation. To enable the applicability of current reasoning techniques, we focus on autoregressive image generation tasks, demonstrated by models such as MaskGiT [6] and LlamaGen [64]. This task employs a data representation and output paradigm akin to those used in LLMs and LMMs, while achieving comparable performance to continuous diffusion models [54, 57, 59]. Specifically, it leverages quantized autoencoders [12] to transform images into discrete tokens, allowing for the classification loss of Direct Preference Optimization (DPO) [55] in post-training. Additionally, it iteratively predicts one or more tokens at each step, conditioned on prior outputs, thereby creating reasoning paths suitable for step-wise verification.

Experimental Settings. We select Show-o [75] as our baseline model for investigation, a latest autoregressive image generation model with advanced capabilities. To comprehensively evaluate different strategies, we assess the text-to-image generation performance on a rigorous benchmark: GenEval [20]. This scenario challenges the model to produce images with not only high visual quality and image-text alignment, but also accurate object attribute and co-occurrence. In the subsequent sections, we explore three strategies to improve the step-by-step decoding of image generation: test-time verification (Sec. 2.2), preference alignment (Sec. 2.3), and their combination (Sec. 2.4).

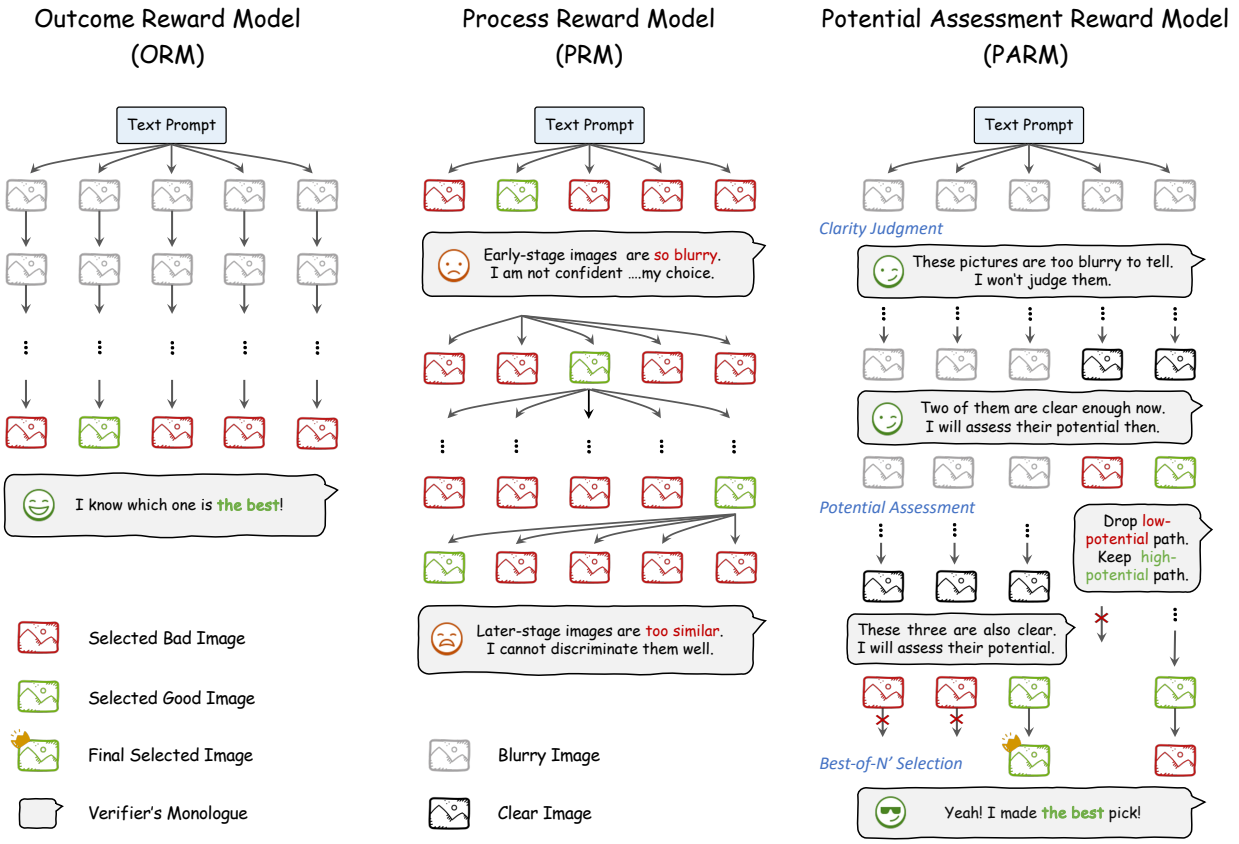

Figure 4. **Investigation of Reward Models in Autoregressive Image Generation.** For test-time verification, we implement Outcome Reward Model (ORM) and Process Reward Model (PRM), and introduce a new Potential Assessment Reward Model customized for image generation scenarios, which progressively performs three tasks (highlighted in blue) to enhance the reasoning of generation process.

2.2. ORM *vs* PRM as Test-time Verifiers

Scaling test-time computation [36, 42, 63, 70] to enhance reasoning capabilities has emerged as an effective alternative to scaling training costs. Current approaches often employ reward models as test-time verifiers within CoT reasoning paths, typically using two main categories: Outcome Reward Model (ORM) and Process Reward Model (PRM). Drawing inspiration from these methods, we respectively implement and evaluate them within the context of autoregressive image generation, as illustrated in Figure 4.

ORM. Based on multiple complete reasoning outputs, ORM assigns each candidate a reword score and select the most confident one using a best-of-N strategy. In our study, we adopt ORM solely to evaluate the generated image at the final step, rather than the entire CoT process in mathematical reasoning tasks. Specifically, we begin with a zero-shot ORM, followed by curating a text-to-image ranking dataset to fine-tune the ORM for enhancement, as outlined below:

- ***Zero-shot ORM:*** We leverage a pre-trained LLaVA-OneVision (7B) [34], an LMM with superior generalization, as our zero-shot ORM. We input the text prompt along with the generated image into LLaVA-OneVision, and devise a prompt template (detailed in the Supplementary Material) to activate its visual understanding capabilities. This model assesses the quality of candidate images, providing binary responses, 'yes' (good quality) or 'no' (low quality). The candidate image with the highest probability of 'yes' is then selected as the final output.

- ***ORM Ranking Data Curation:*** To enhance the accuracy of outcome rewards, we curate a dataset of 288K text-to-image ranking examples for fine-tuning ORM. First, we prompt GPT-4 [46] to generate a list of 200 countable daily object names with specific colors. Using these objects, we apply the six object-centered prompt templates from GenEval, constructing a diverse set of 13K text prompts. We perform a strict filtering to ensure that these prompts do not overlap with the GenEval test samples. Then, using our baseline model, Show-o, we synthesize around 50 images per prompt at a high temperature. After that, we label each image with a binary annotation of 'yes' or 'no' using the evaluation metric in GenEval.

Table 1. **Test-time Verifiers (ORM *vs* PRM) *vs* Preference Alignment.** We evaluate text-to-image generation on the GenEval [20] benchmark and adopt Show-o [75] as the autoregressive baseline model. 'ORM/PRM' and 'DPO' denote Outcome/Process Reward Model and Direct Preference Optimization [55], respectively. We adopt the best-of-N selection for test-time verifiers, setting $N = 20$, and highlight the better-performed variant of each reasoning strategy in green.

Reasoning Strategy	Method	Setting	Single object	Two object	Counting	Colors	Position	Attribute binding	**Overall**
Baseline	-	-	0.95	0.52	0.49	0.82	0.11	0.28	0.53
Test-time Verifier	ORM	Zero-Shot	0.99	0.63	0.63	0.84	0.19	0.39	0.61
		Fine-tuned	0.99	0.72	0.65	0.84	0.25	0.33	0.63
	PRM	Zero-Shot	0.98	0.51	0.54	0.82	0.11	0.23	0.53
		Fine-tuned	0.98	0.55	0.54	0.83	0.13	0.29	0.55
Preference Alignment	DPO	-	0.96	0.70	0.50	0.82	0.30	0.43	0.62
		Iterative	0.98	0.72	0.53	0.84	0.40	0.46	0.65

- ***Fine-tuned ORM:*** Using the curated ranking dataset, we fine-tune LLaVA-OneVision to enhance its capability for assessing image quality and cross-modal alignment. The training data format is consistent with the prompt template used in the zero-shot ORM, incorporated with our constructed 288K text prompts and associated images. The model is fine-tuned for one epoch, using a batch size of 8 and a learning rate of $1e^{-5}$. This fine-tuning process enables the ORM to capture more intricate aspects of object composition and nuanced visual-text relationships, resulting in more reliable reward scoring.

PRM. Different from ORM that evaluates only the final output, we utilize PRM to provide a reward score to each candidate with different steps throughout the generation process. Similar to our previous investigation, we start from a zero-shot PRM, LLaVA-OneVision, and then curate 10K step-wise text-to-image ranking data to obtain a fine-tuned PRM. We refer to the Supplementary Material for detailed implementation of PRM.

Experiments and Insights. As showcased in the middle of Table 1, we compare the test-time verification results between ORM and PRM with a best-of-20 strategy. The observations are summarized below:

- ***Test-time verification can significantly boost generation performance.*** Compared to the baseline scores of 53% on GenEval, the fine-tuned ORM as a test-time verifier achieves the highest gains of +10%. This finding suggests that current autoregressive image generation models, similar to LLMs and LMMs, face challenges with inconsistent and unstable decoding paths. Consequently, a test-time verification strategy is essential to identify and follow the most reliable reasoning path.
- ***ORM exhibits stronger enhancement capabilities than PRM.*** In contrast to ORM providing a clear benefit, PRM yields only marginal improvements, e.g., +2% on GenEval after fine-tuning. This discrepancy arises from the unique characteristics of the autoregressive image generation task in two key ways: *1)* Images at early steps are too blurry for PRM to effectively interpret their visual features and image-text alignment. *2)* Images at later steps tend to exhibit minimal differences, making it challenging for PRM to discriminate. Whereas, ORM evaluates images at the final step, which provides sufficient visual and semantic information for accurate judgment.
- ***Fine-tuning by ranking data enhances verification results and demonstrates improved scaling performance.*** As illustrated in Figure 3, both fine-tuned ORM and PRM outperform their zero-shot counterparts, achieving higher scores with larger N values in the best-of-N selection. Additionally, as N increases, fine-tuned reward models show greater improvements, indicated by steeper curves, reflecting a better scaling response to test-time computation. This highlights the effectiveness of our curated ranking dataset in refining reward accuracy and benefiting scalability across a broader range of candidates.

2.3. Test-time Verifiers *vs* Preference Alignment

Post-training has been widely utilized in existing LLMs and LMMs to align model outputs with human preferences. Common techniques include reinforcement learning with reward models, e.g., Proximal Policy Optimization (PPO) [62], and its streamlined counterpart with classification objectives, e.g., Direct Preference Optimization (DPO) [55]. Given that most autoregressive image generation models inherently function within a classification framework, we leverage the simplicity of DPO alignment to enhance the quality of generated images.

Table 2. **Test-time Verifiers *plus* Preference Alignment.** We evaluate text-to-image generation on the GenEval [20] benchmark and adopt Show-o [75] as the autoregressive baseline model. 'Ft. ORM' and 'It. DPO' denote the fine-tuned ORM and iterative DPO [55]. We explore three combination approaches ('1*st*, 2*nd*, and 3*rd* Integration') of reward models and preference alignment, comparing 'individual' results. We adopt the best-of-N selection for test-time verifiers, setting $N = 20$, and highlight the best-performing integration in green.

Reasoning Strategy	Test-time Verifier	Preference Alignment	Reward Guidance	Single object	Two object	Counting	Colors	Position	Attribute binding	**Overall**
Baseline	-	-	-	0.95	0.52	0.49	0.82	0.11	0.28	0.53
Individual	Ft. ORM	-	-	0.99	0.72	0.65	0.84	0.25	0.33	0.63
	-	It. DPO	-	0.98	0.72	0.53	0.84	0.40	0.46	0.65
1*st* Integration	-	It. DPO	Ft. ORM	0.98	0.78	0.44	0.81	0.50	0.48	0.67
2*nd* Integration	Ft. ORM	It. DPO	-	0.98	0.80	0.62	0.83	0.59	0.54	0.72
3*rd* Integration	Ft. ORM	It. DPO	Ft. ORM	0.98	0.84	0.64	0.85	0.66	0.52	0.75

DPO Ranking Data Curation. To bypass reinforcement learning, DPO leverages an implicit rewarding mechanism by training on a ranking dataset of paired preferred and dispreferred responses, corresponding to well-generated and poor-quality images in our case. Fortunately, we have already constructed a substantial amount of ranking data for training ORM, annotated with 'yes' and 'no' labels to indicate the generation quality. Building on this, we utilize the 13K unique text prompts from the ORM training dataset and, for each prompt, randomly pair two generated images, one labeled 'yes' and the other 'no', yielding a total of 10K paired data for DPO alignment.

DPO for Autoregressive Image Generation. Since autoregressive image generation models are also trained using a cross-entropy loss, we can directly apply the maximum likelihood objective in DPO to our setting. In detail, the parameterized policy is initialized from Show-o and optimized during training, while the reference policy is also initialized from Show-o but kept frozen. The objective encourages the model to assign a higher likelihood to preferred images over dispreferred images, aligning with the curated preference structure. The DPO training is conducted over one epoch with a batch size of 10 and a learning rate of $1e^{-5}$.

DPO with Iterative Training. Following the initial stage of DPO alignment, the model has learned to generate images that better align with the preferred responses. Inspired by iterative DPO [52], we further refine this alignment by applying the newly aligned model to generate updated ranking data based on the text prompts in $\mathcal{D}$. We annotate these new images with 'yes' or 'no' labels using the same method in Sec. 2.2. For each prompt, we collect paired images labeled y_{yes} and y_{no}, and exclude samples where all images receive the same label, resulting in a refined DPO ranking dataset of 7K samples. By conducting another round of DPO, the model can be further improved by learning from more informative preference relations. We iterate the DPO training process once with the same training configurations.

Experiments and Insights. In the bottom of Table 1, we present the evaluation results of DPO alignment and compare it with the performance with test-time verification. The observations are summarized below:

- ***DPO alignment can effectively reinforce the generation performance.*** On GenEval, initial DPO alignment improves the baseline model's performance by +9%. With iterative training, these gains are further extended as +11%, highlighting the effectiveness of a refined preference dataset in strengthening model alignment with desired outputs. This demonstrates that DPO alignment can serve as a powerful method for enhancing autoregressive image generation models, especially in scenarios where explicit preference data is available to guide training.
- ***Initial DPO matches test-time verification, while iterative DPO surpasses.*** After the initial alignment, the model achieves performance comparable to that of the fine-tuned ORM, the top-performing variant for verification. However, with iterative alignment on refined ranking data, the model outperforms all test-time verifiers, i.e., +2% over the fine-tuned ORM. This demonstrates the potential of iterative DPO to progressively reinforce image generation capabilities through updated ranking data.

2.4. DPO Alignment *plus* Test-time Verifiers

The investigations above have demonstrated the individual effectiveness of test-time verification and DPO alignment. Next, we explore three approaches to integrate these two techniques to assess their potential for complementary enhancement in image generation, leveraging both the adaptability of verifiers and the reinforcement of DPO.

Table 3. **Performance Comparison on the GenEval [20] Benchmark.** Compared to existing diffusion and autoregressive models, we investigate the potential of Chain-of-Thought (CoT) reasoning strategies in text-to-image generation. 'Ft. ORM' and 'It. DPO' denote the fine-tuned ORM and iterative DPO [55]. **PARM** refers to our Potential Assessment Reward Model specialized for autoregressive image generation. We adopt the best-of-20 selection for test-time verifiers, and highlight the best and second-best overall scores in green and red.

Model	Test-time Verifier	Preference Alignment	Reward Guidance	Single object	Two object	Counting	Colors	Position	Attribute binding	**Overall**
PixArt-α [7]	-	-	-	0.98	0.50	0.44	0.80	0.08	0.07	0.48
SD v2.1 [60]	-	-	-	0.98	0.51	0.44	0.85	0.07	0.17	0.50
DALL-E 2 [57]	-	-	-	0.94	0.66	0.49	0.77	0.10	0.19	0.52
SDXL [54]	-	-	-	0.98	0.74	0.39	0.85	0.15	0.23	0.55
SD 3 (d=24) [13]	-	-	-	0.98	0.74	0.63	0.67	0.34	0.36	0.62
LlamaGen [64]	-	-	-	0.71	0.34	0.21	0.58	0.07	0.04	0.32
Chameleon [66]	-	-	-	-	-	-	-	-	-	0.39
LWM [37]	-	-	-	0.93	0.41	0.46	0.79	0.09	0.15	0.47
SEED-X [18]	-	-	-	0.97	0.58	0.26	0.80	0.19	0.14	0.49
Show-o [75]	-	-	-	0.95	0.52	0.49	0.82	0.11	0.28	0.53
	Ft. ORM	-	-	0.99	0.72	0.65	0.84	0.25	0.33	0.63
	-	It. DPO	-	0.98	0.72	0.53	0.84	0.40	0.46	0.65
	Ft. ORM	It. DPO	Ft. ORM	0.98	0.84	0.64	0.85	0.66	0.52	0.75
	PARM	-	-	0.99	0.77	0.68	0.86	0.29	0.45	0.67
	-	It. DPO	**PARM**	0.97	0.75	0.60	0.83	0.54	0.53	0.69
	PARM	It. DPO	-	0.98	0.83	0.64	0.84	0.59	0.62	0.74
	PARM	It. DPO	**PARM**	0.99	0.86	0.67	0.84	0.66	0.64	0.77

DPO with Reward Model Guidance. As discussed in previous works [77], DPO can struggle with out-of-distribution responses due to distribution shifts from the ranking dataset. A potential solution [1, 79] is to incorporate prompt-only datasets during post-training and leverage a reward model to provide online preference guidance. Following this approach, we adopt our fine-tuned ORM as the explicit reward model to offer more generalized preference feedback, and add the online objectives with the original DPO loss We maintain the same training data and configurations as in the initial DPO alignment stage.

Verification after DPO Alignment. We observe that verification and DPO techniques may naturally complement each other in two key ways: *1)* They operate independently at different stages of implementation, i.e., post-training and test-time; and *2)* DPO refines the internal knowledge distribution within the model to enhance reasoning, while verification focuses on selecting the optimal reasoning path within this refined distribution. Therefore, we apply the fine-tuned ORM for best-of-N selection directly on the model after DPO alignment.

Verification after DPO with Reward Model Guidance. In this approach, we combine the strengths of both DPO with reward model guidance and test-time verification. Our goal is to achieve optimal alignment, enhancing the model's generalization capabilities during training, while also ensuring reliable image generation paths at inference time.

Experiments and Insights. Table 2 reports the text-to-image generation scores with different integration methods. The observations are summarized below:

- ***Verification and alignment perform expected complementary characteristics.*** Across all three approaches, verification and alignment complement each other effectively. For instance, the third integration method outperforms DPO alignment alone by +10%, and surpasses the verification alone by +12%. These results highlight the potential of combining verification and alignment techniques in future autoregressive image generation tasks, enabling the production of high-quality outputs that are both preference-aligned and test-time reliable.
- ***Applying verifiers to both training and test time yields maximum enhancement.*** We observe the third combination approach delivers the most significant gains, outperforming the first approach by +8%, and the second by +3%. This suggests that, even with a model already aligned to preferences, the fine-tuned ORM can play complementary roles in the test-time decoding. These dual functions reinforce each other, highlighting the versatility of reward models in autoregressive image generation.

3. Potential Assessment Reward Model

From our comprehensive investigation, reward models prove valuable by enabling both decoding path selection and preference reward guidance. However, we still observe considerable room for enhancing reward models.

Limitation of ORM and PRM. *1)* ORM showcases strong performance by selecting optimal final outputs, yet it lacks the capacity to provide fine-grained, step-wise evaluation at each generation step. *2)* While PRM has demonstrated effectiveness in understanding tasks such as mathematics, it is less suitable for autoregressive image generation. As analyzed in Sec. 2.2, PRM struggles with early-stage images that are too blurry for reliable evaluation, given that only a few regions are decoded. In later stages, images derived from similar previous steps lack sufficient distinction, challenging for PRM to discriminate.

PARM. Motivated by these observations, we propose the Potential Assessment Reward Model (PARM), a specialized reward model tailored for autoregressive image generation, as illustrated in Figure 4. PARM combines the best of both worlds: *1)* it operates adaptively in a step-wise manner, using a potential assessment mechanism to overcome PRM's evaluation challenges; and *2)* it performs a best-of-N' selection across N' ($N' \leq N$) high-potential reasoning paths, thus inheriting ORM's advantage. Specifically, the methodology of PARM contains three progressive tasks:

1. **Clarity Judgment.** In the best-of-N setting, we first sample N different reasoning paths for image generation. Then, at each intermediate step, PARM evaluates whether the partially generated image contains enough visual clarity to be meaningfully assessed, assigning a binary label If labeled 'no', the model skips to the next step. If labeled 'yes', the model proceeds to the next task for potential assessment. This pre-judgment prevents scoring on early, blurry images that lack informative content (as seen in PRM), ensuring only sufficiently clear steps are considered for rewarding.
2. **Potential Assessment.** For each clear step that passes the clarity judgment, PARM assesses the potential of the current step to determine whether it can lead to a high-quality final image, again using a binary label. If labeled 'no', the generation path is truncated immediately. If labeled 'yes', the path is preserved to produce the final image. This approach is based on the observation that, once an image at a given step is clear enough to evaluate, its overall layout and structure are unlikely to change significantly in subsequent steps, making it a reliable candidate for potential assessment. This task helps identify promising intermediate steps accurately, effectively pruning low-potential candidates during inference.
3. **Best-of-N' Selection.** After completing the above two tasks, suppose there are N' high-potential paths remaining to produce the final images ($N' \leq N$). PARM then performs a best-of-N' selection to identify the most promising image candidates as the output. If $N' = 0$, the model defaults to selecting the reasoning path with the lowest probability of a 'no' label as the output. This final task leverages ORM's global selection capabilities to ensure a high-quality generated image.

PARM Ranking Data Curation. To empower PARM with robust capabilities, we curate a new ranking dataset with 400K instances by re-annotating the 13K text prompts from ORM ranking data. The dataset is structured into three subsets corresponding to the three evaluation tasks, containing 120K, 80K, and 200K instances, respectively. Please refer to the Supplementary Material for detailed data formats.

Experiments and Insights. With the new reward model, we revisit our previous investigation by applying PARM to different approaches enhancing autoregressive image generation. The observations are summarized below:

- ***PARM demonstrates the best-performing reward model across different strategies.*** Table 3 and Figure 3 present the effectiveness of PARM as test-time verifiers, significantly outperforming other reward models, e.g., +6% to the fine-tuned ORM. Additionally, PARM scales effectively with increasing N, highlighting its potential for further improvement with larger test-time computation. PARM also outperforms iterative DPO, the enhanced preference alignment strategy with refined data. Furthermore, PARM better harnesses the complementary strengths with post-training, consistently attaining higher integration scores than the fine-tuned ORM. These results underscore PARM's capability as a versatile and robust reward model for autoregressive image generation.
- ***With PARM, our baseline model (Show-o) is enhanced to achieve leading generation performance.*** Compared to other image generation models in Table 3, our best-performing configuration, i.e., integrating PARM with iterative DPO in both post-training and test-time, achieves a score of 77%, improving the baseline by +24% and surpassing the advanced Stable Diffusion 3 [13] by +15%. In particular, substantial gains are observed in 'Two Obj.', 'Colors', 'Position', and 'Attribute binding' emphasizing the robustness in handling challenging compositional generation. In Figures 5, 6, 7, 8, and 9, we showcase extensive qualitative examples. We observe that the baseline model often generates inaccurate spatial relationships and strange appearances, or fail to precisely reflect object attributes. In contrast, our approach consistently mitigates such issues, ensuring that the spatial relations, object features, and overall fidelity to the text prompt are preserved.

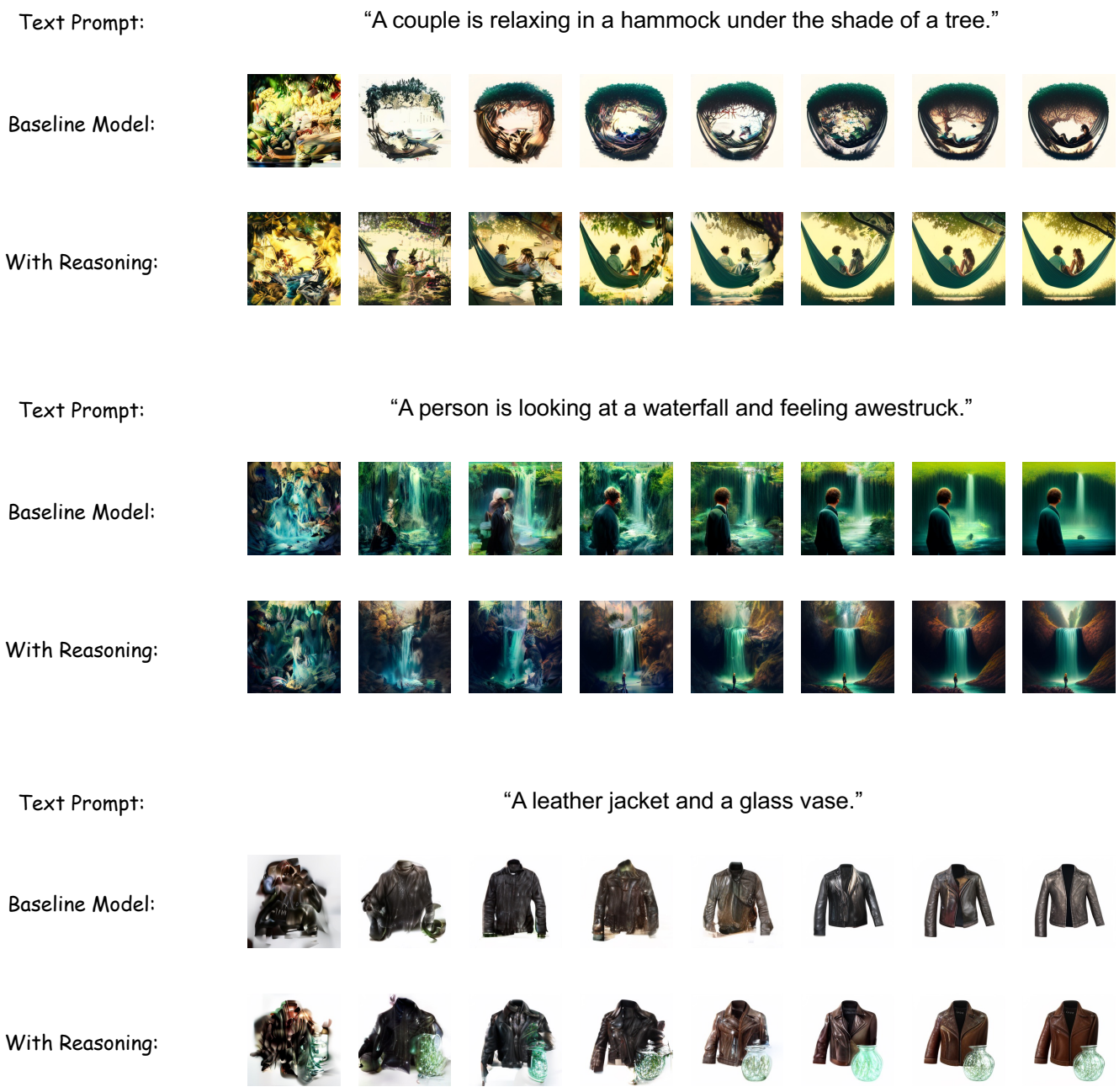

Figure 5. **Qualitative Results using Our Reasoning Strategies.** Show-o [75] is adopted as the baseline model, and compared to our best-performing reasoning strategy: integrating PARM with iterative DPO for both reward model guidance and test-time verification.

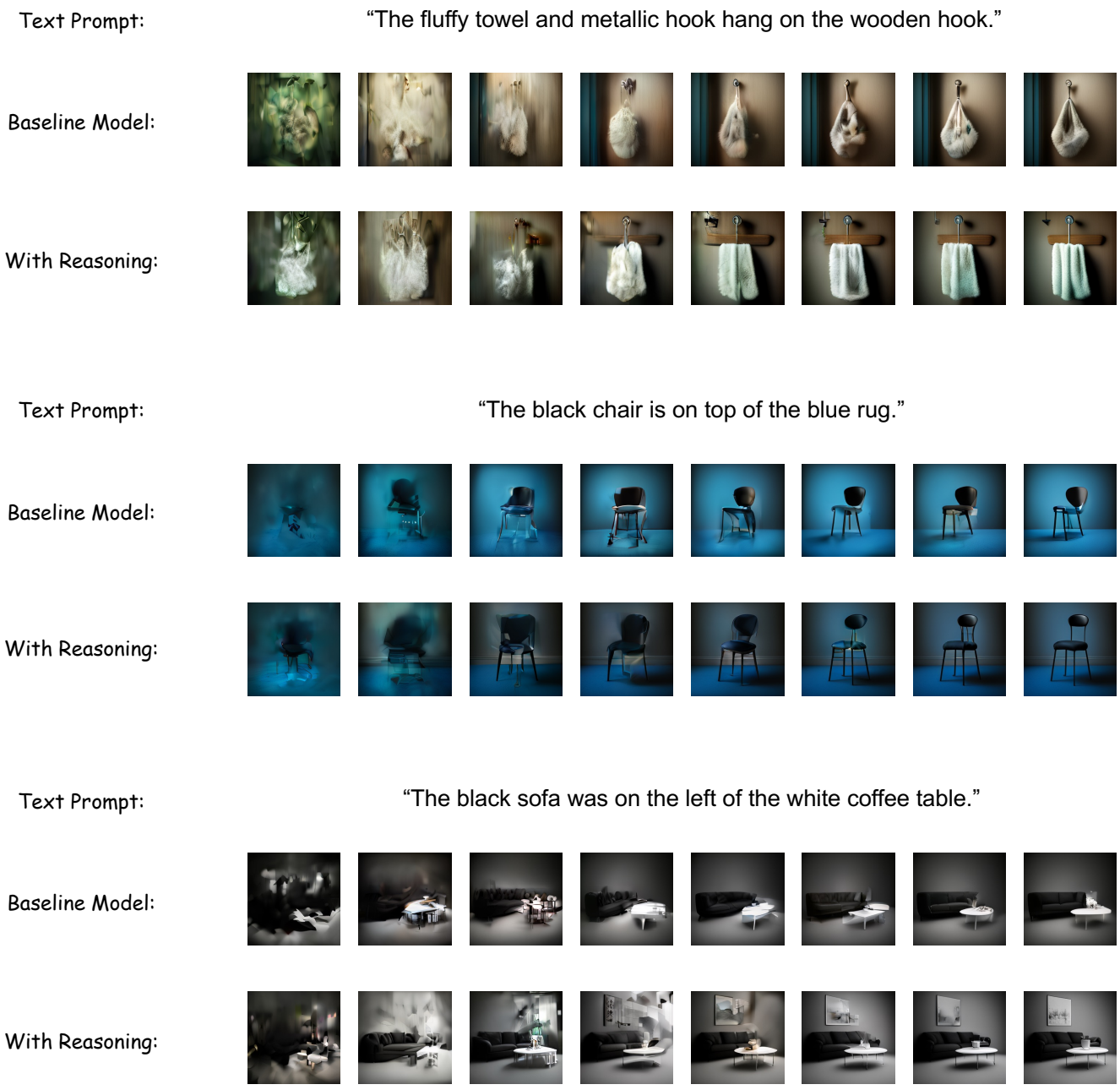

Figure 6. **Qualitative Results using Our Reasoning Strategies.** Show-o [75] is adopted as the baseline model, and compared to our best-performing reasoning strategy: integrating PARM with iterative DPO for both reward model guidance and test-time verification.

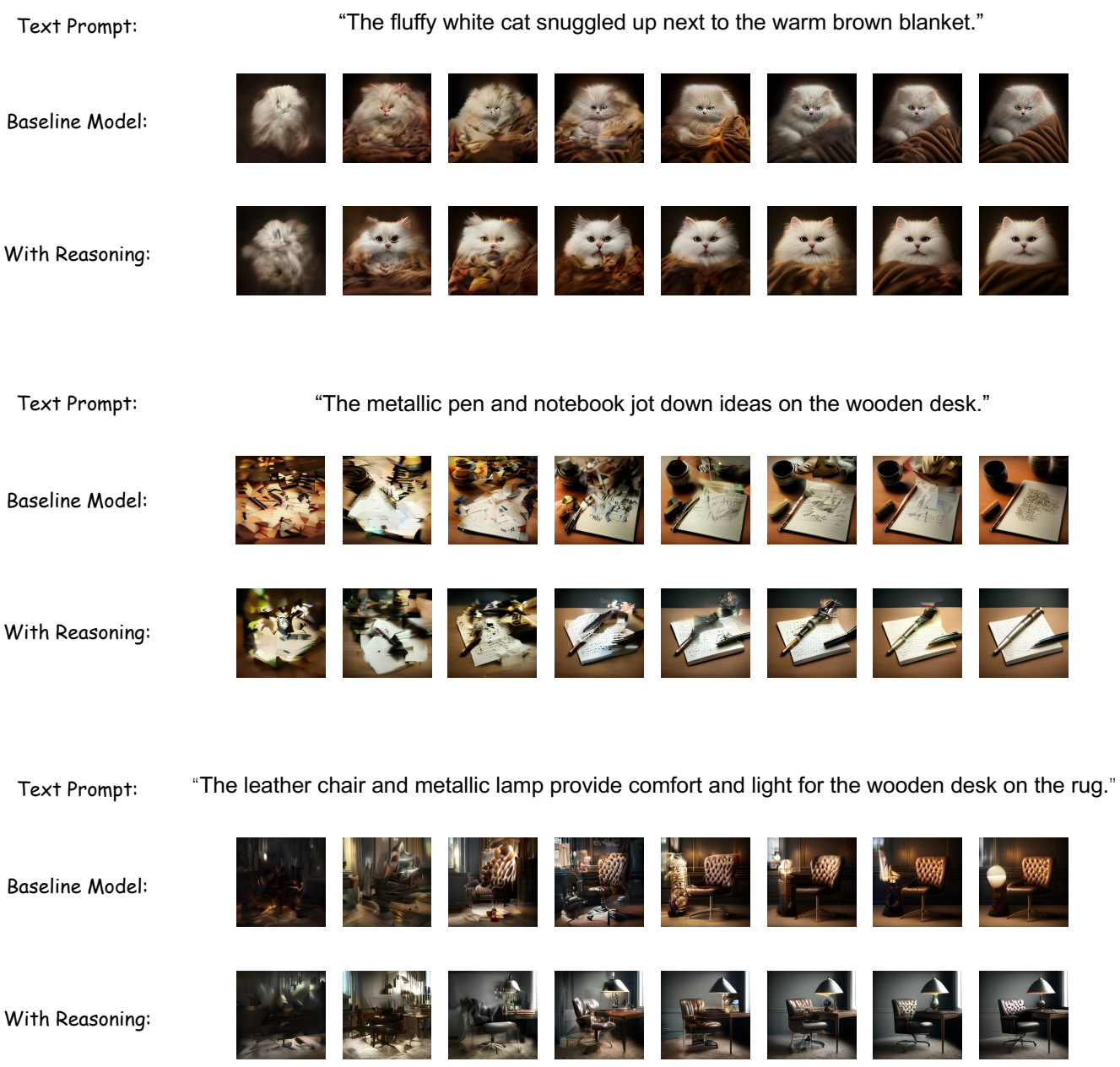

Figure 7. **Qualitative Results using Our Reasoning Strategies.** Show-o [75] is adopted as the baseline model, and compared to our best-performing reasoning strategy: integrating PARM with iterative DPO for both reward model guidance and test-time verification.

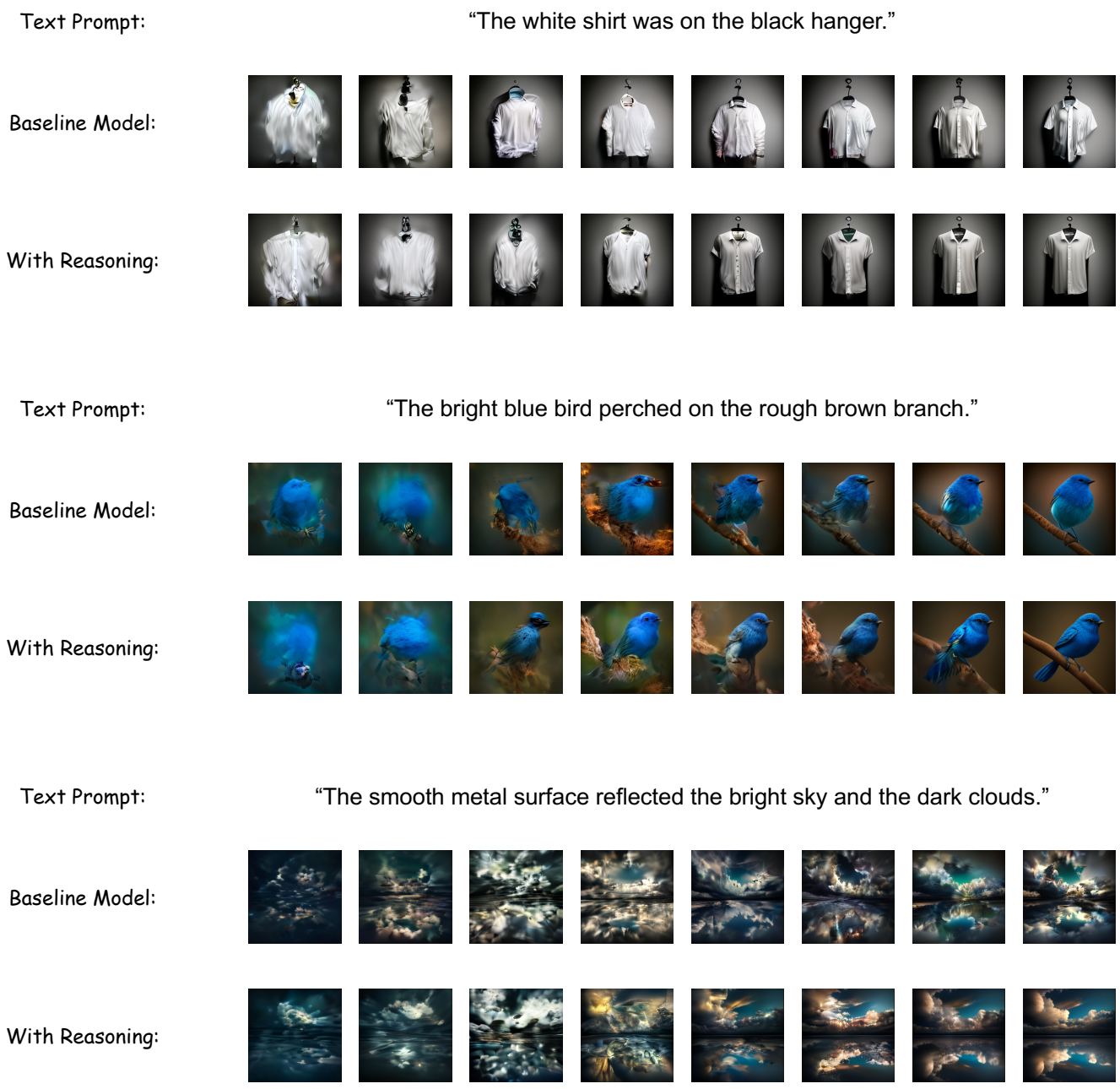

Figure 8. **Qualitative Results using Our Reasoning Strategies.** Show-o [75] is adopted as the baseline model, and compared to our best-performing reasoning strategy: integrating PARM with iterative DPO for both reward model guidance and test-time verification.

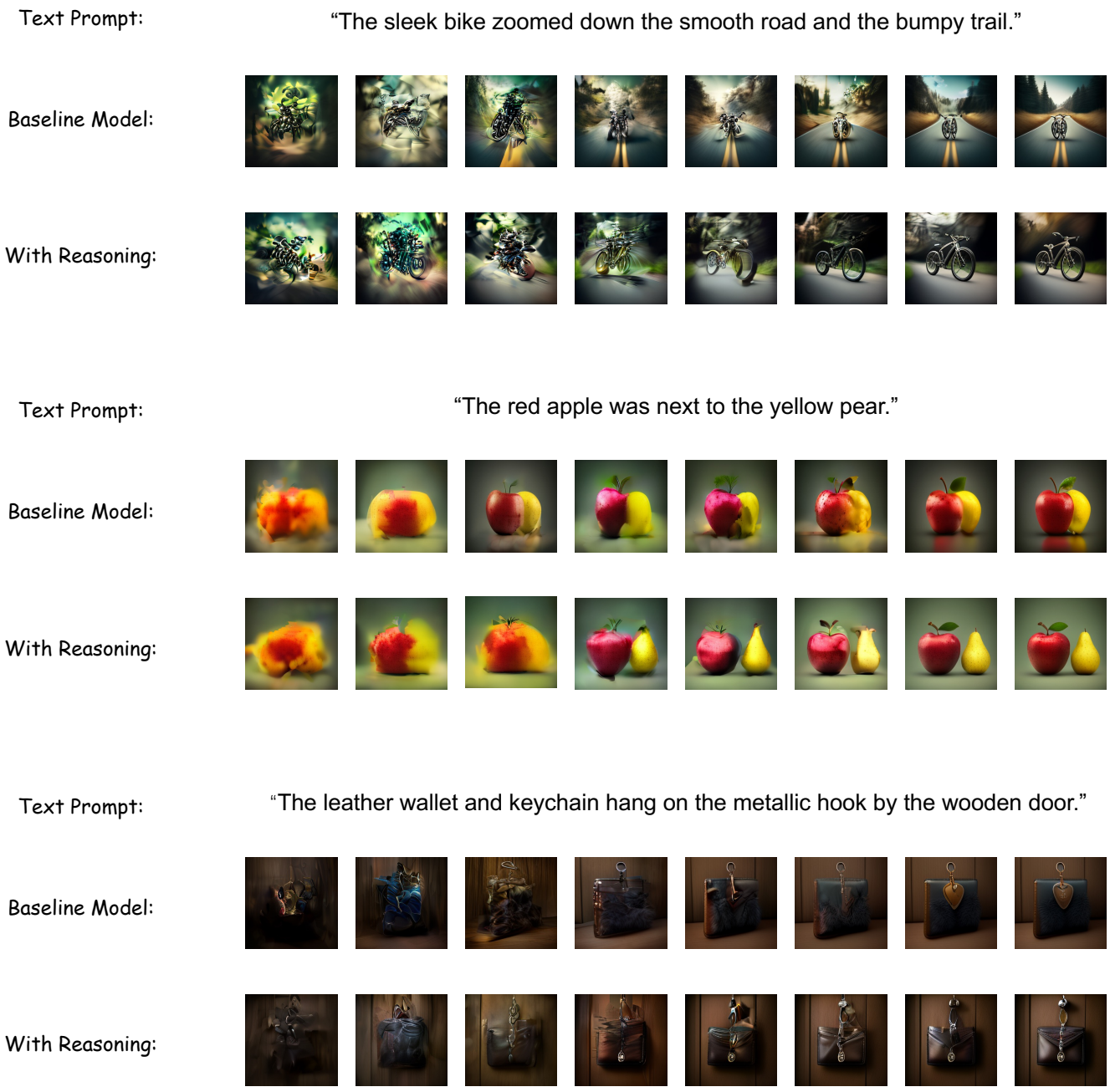

Figure 9. **Qualitative Results using Our Reasoning Strategies.** Show-o [75] is adopted as the baseline model, and compared to our best-performing reasoning strategy: integrating PARM with iterative DPO for both reward model guidance and test-time verification.

4. Related Work

Scaling Test-time Computation. Humans often dedicate significant time and effort to solve complex problems. Inspired by this, many efforts have focused on scaling test-time computation for Large Language Models (LLMs) to tackle reasoning tasks such as mathematical problem-solving [64, 70, 76, 80], code synthesis [11, 44, 89], and workflow generation [16, 78, 81]. One line of research adapts the input space to leverage Chain-of-Thought (CoT) capabilities, using approaches like in-context CoT examples [74] or zero-shot CoT prompts [31]. Another branch modifies or integrates reasoning paths within the output space, utilizing strategies such as self-consistency [71], CoT decoding [71], and verifier-based selection [9, 36, 63]. Among these, test-time verifiers have demonstrated generality and robustness in enhancing reasoning performance. For example, early work [9] trains an Outcome Reward Model (ORM) to evaluate final outputs and select the best-of-N candidates for optimal results. Later, Lightman et al. [36, 42] adopt the Process Reward Model (PRM) to evaluate intermediate reasoning steps, achieving greater effectiveness. Snell et al. [63] further highlights that scaling test-time computation is often more impactful than scaling model parameters during training. Recently, OpenAI o1 [50] has demonstrated exceptional reasoning capabilities across a variety of complex and challenging scenarios, underscoring the potential of this approach. Building on these advancements in understanding tasks, we conduct a comprehensive investigation into whether verifier-based strategies can also enhance image generation tasks, and propose a new Potential Assessment Reward Model (PARM), specifically designed for this domain.

Reinforced Preference Alignment. After robust pre-training and fine-tuning, LLMs often acquire substantial knowledge. However, a post-training alignment stage is typically required to align their output preferences to meet specific targets, such as human feedback [2, 8, 32] or Chain-of-Thought (CoT) reasoning [33, 40, 70]. Traditional approaches [5, 28, 51, 87] often leverage reinforcement learning (RL) to address this challenge. These methods usually involve two steps: first, optimizing a neural-network-based reward function within a preference model (e.g., the Bradley-Terry model [3]), and then fine-tuning the target LLM to maximize this reward using techniques like proximal policy optimization (PPO) [62]. However, RL-based methods often encounter issues related to complexity and instability. To overcome these challenges, Rafailov et al. introduced Direct Preference Optimization (DPO) [55], which parameterizes the reward model to enable the derivation of the optimal policy through a closed-form solution. This approach has been effectively applied to enhance CoT capabilities in mathematical reasoning [41, 70] and code generation [19, 43, 77]. Further advancements have extended DPO with step-wise preference data [33, 40] for more granular supervision and multi-modality learning [84, 85] to support visual reasoning. In this study, we apply DPO-based preference alignment to autoregressive image generation, demonstrating its effectiveness in improving image quality during step-by-step decoding.

Autoregressive Image Generation. The transformer architectures with autoregressive output schemes [1, 34, 46, 47, 49, 68] have demonstrated a remarkably successful modeling approach in language and multi-modality. Motivated by such progress, a series of work, e.g., DALL-E [56], LlamaGen [65], and Chameleon [66], utilizes such autoregressive modeling with casual attention to learn the dependency within image pixels for image generation tasks, rather than popular diffusion models [7, 13, 29, 54, 57, 88]. However, such raster-order autoregression suffers from severe time consumption and performance constraints when synthesizing high-resolution and high-fidelity images, attributed to the growing number of discrete tokens compressed by VQ-VQE [12, 21, 58, 69]. To address the challenges, MaskGiT [6] proposes to learn a bidirectional autoregressive transformer with a parallel iterative decoding strategy, benefiting both the generation performance and efficiency. Recently, this approach has been effectively extended, primarily focusing on two aspects: the unification of visual understanding and generation (Show-o [75]) and its integration with diffusion techniques (MAR [35]). Considering that such generation paradigm is quite similar to that of LLMs, representing data with discrete tokens and predicting iteratively conditioned on previous tokens, we explore the potential of applying CoT reasoning techniques within LLMs to autogressive image generation. Through our thorough investigation, we demonstrate its promising effectiveness for enhanced image generation capabilities.

5. Conclusion

In this work, we investigate the adaption and potential of CoT reasoning strategies in autoregressive image generation. Through a systematic investigation, we demonstrate that different reasoning strategies can effectively improve image generation, e.g., test-time verification, preference alignment, and their integration. Given our observation, we further introduce a tailored reward model for autoregressive image generation, termed Potential Assessment Reward Model (PARM), which evaluates the step-wise potential of image generation for adaptive reward scoring with superior results. Our experiments underscore the promise of CoT reasoning in autoregressive image generation, advancing this field in new directions.

References

[1] AI@Meta. Llama 3 model card, 2024.

[2] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[3] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in neural information processing systems*, pages 1877–1901, 2020.

[5] Róbert Busa-Fekete, Balázs Szörényi, Paul Weng, Weiwei Cheng, and Eyke Hüllermeier. Preference-based reinforcement learning: evolutionary direct policy search using a preference-based racing algorithm. *Machine learning*, 97: 327–351, 2014.

[6] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022.

[7] Junsong Chen, Jincheng YU, Chongjian GE, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024.

[8] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

[9] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[10] OpenCompass Contributors. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass, 2023.

[11] Matthew DeLorenzo, Animesh Basak Chowdhury, Vasudev Gohil, Shailja Thakur, Ramesh Karri, Siddharth Garg, and Jeyavijayan Rajendran. Make every move count: Llm-based high-quality rtl code generation using mcts. *arXiv preprint arXiv:2402.03289*, 2024.

[12] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[13] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024.

[14] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023.

[15] Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.

[16] Rinon Gal, Adi Haviv, Yuval Alaluf, Amit H Bermano, Daniel Cohen-Or, and Gal Chechik. Comfygen: Prompt-adaptive workflows for text-to-image generation. *arXiv preprint arXiv:2410.01731*, 2024.

[17] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, Hongsheng Li, and Yu Qiao. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023.

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

[19] Leonidas Gee, Milan Gritta, Gerasimos Lampouras, and Ignacio Iacobacci. Code-optimise: Self-generated preference data for correctness and efficiency. *arXiv preprint arXiv:2406.12502*, 2024.

[20] 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, 2024.

[21] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10696–10706, 2022.

[22] Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yu Wu, YK Li, et al. Deepseek-coder: When the large language model meets programming–the rise of code intelligence. *arXiv preprint arXiv:2401.14196*, 2024.

[23] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, et al. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv preprint arXiv:2309.00615*, 2023.

[24] Ziyu Guo, Renrui Zhang, Hao Chen, Jialin Gao, Peng Gao, Hongsheng Li, and Pheng-Ann Heng. Sciverse. *https://sciverse-cuhk.github.io*, 2024.

[25] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Chengzhuo Tong, Peng Gao, Chunyuan Li, and Pheng-Ann Heng. Sam2point: Segment any 3d as videos in zero-shot and promptable manners. *arXiv preprint arXiv:2408.16768*, 2024.

[26] Jiaming Han, Renrui Zhang, Wenqi Shao, Peng Gao, Peng Xu, Han Xiao, Kaipeng Zhang, Chris Liu, Song Wen, Ziyu Guo, et al. Imagebind-llm: Multi-modality instruction tuning. *arXiv preprint arXiv:2309.03905*, 2023.

[27] Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. Teaching large language models to reason with reinforcement learning. *arXiv preprint arXiv:2403.04642*, 2024.

[28] Ashesh Jain, Brian Wojcik, Thorsten Joachims, and Ashutosh Saxena. Learning trajectory preferences for manipulators via iterative improvement. *Advances in neural information processing systems*, 26, 2013.

[29] Dongzhi Jiang, Guanglu Song, Xiaoshi Wu, Renrui Zhang, Dazhong Shen, Zhuofan Zong, Yu Liu, and Hongsheng Li. Comat: Aligning text-to-image diffusion model with image-to-text concept matching. *arXiv preprint arXiv:2404.03653*, 2024.

[30] Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanmin Wu, Jiayi Lei, Pengshuo Qiu, Pan Lu, Zehui Chen, Guanglu Song, Peng Gao, et al. Mmsearch: Benchmarking the potential of large models as multi-modal search engines. *arXiv preprint arXiv:2409.12959*, 2024.

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

[32] Andras Kupcsik, David Hsu, and Wee Sun Lee. Learning dynamic robot-to-human object handover from human feedback. *Robotics Research: Volume 1*, pages 161–176, 2018.

[33] Xin Lai, Zhuotao Tian, Yukang Chen, Senqiao Yang, Xiangru Peng, and Jiaya Jia. Step-dpo: Step-wise preference optimization for long-chain reasoning of llms. *arXiv preprint arXiv:2406.18629*, 2024.

[34] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[35] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *arXiv preprint arXiv:2406.11838*, 2024.

[36] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.

[37] Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with blockwise ringattention, 2024.

[38] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*, 2023.

[39] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chun yue Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating math reasoning in visual contexts with gpt-4v, bard, and other large multimodal models. *ArXiv*, abs/2310.02255, 2023.

[40] Zimu Lu, Aojun Zhou, Ke Wang, Houxing Ren, Weikang Shi, Junting Pan, and Mingjie Zhan. Step-controlled dpo: Leveraging stepwise error for enhanced mathematical reasoning. *arXiv preprint arXiv:2407.00782*, 2024.

[41] Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. *arXiv preprint arXiv:2308.09583*, 2023.

[42] Qianli Ma, Haotian Zhou, Tingkai Liu, Jianbo Yuan, Pengfei Liu, Yang You, and Hongxia Yang. Let's reward step by step: Step-level reward model as the navigators for reasoning. *arXiv preprint arXiv:2310.10080*, 2023.

[43] Yibo Miao, Bofei Gao, Shanghaoran Quan, Junyang Lin, Daoguang Zan, Jiaheng Liu, Jian Yang, Tianyu Liu, and Zhijie Deng. Aligning codellms with direct preference optimization. *arXiv preprint arXiv:2410.18585*, 2024.

[44] Ansong Ni, Srini Iyer, Dragomir Radev, Veselin Stoyanov, Wen-tau Yih, Sida Wang, and Xi Victoria Lin. Lever: Learning to verify language-to-code generation with execution. In *International Conference on Machine Learning*, pages 26106–26128. PMLR, 2023.

[45] OpenAI. Chatgpt. https://chat.openai.com, 2023.

[46] OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023.

[47] OpenAI. GPT-4V(ision) system card, 2023.

[48] OpenAI. Openai o1. [Online], 2024. https://openai.com/index/learning-to-reason-with-llms/.

[49] OpenAI. Hello gpt-4o. https://openai.com/index/hello-gpt-4o/, 2024.

[50] OpenAI. Learning to reason with llms. https://openai.com/index/learning-to-reason-with-llms/, 2024. Accessed: 2024-11-06.

[51] Aldo Pacchiano, Aadirupa Saha, and Jonathan Lee. Dueling rl: reinforcement learning with trajectory preferences. *arXiv preprint arXiv:2111.04850*, 2021.

[52] Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative reasoning preference optimization, 2024. *URL https://arxiv.org/abs/2404.19733*.

[53] Richard Yuanzhe Pang, Weizhe Yuan, Kyunghyun Cho, He He, Sainbayar Sukhbaatar, and Jason Weston. Iterative reasoning preference optimization. *arXiv preprint arXiv:2404.19733*, 2024.

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

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

[56] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

[57] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.

[58] Ali Razavi, Aäron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2 (supplementary material).

[59] 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.

[60] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

[61] Tanik Saikh, Tirthankar Ghosal, Amish Mittal, Asif Ekbal, and Pushpak Bhattacharyya. Scienceqa: A novel resource for question answering on scholarly articles. *International Journal on Digital Libraries*, 23(3):289–301, 2022.

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

[63] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters, 2024.

[64] Linzhuang Sun, Hao Liang, and Wentao Zhang. Beats: Optimizing llm mathematical capabilities with backverify and adaptive disambiguate based efficient tree search. *arXiv preprint arXiv:2409.17972*, 2024.

[65] Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

[66] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models, 2024.

[67] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[68] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[69] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[70] Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9426–9439, 2024.

[71] Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting. *arXiv preprint arXiv:2402.10200*, 2024.

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

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

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

[75] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

[76] Huajian Xin, ZZ Ren, Junxiao Song, Zhihong Shao, Wanjia Zhao, Haocheng Wang, Bo Liu, Liyue Zhang, Xuan Lu, Qiushi Du, et al. Deepseek-prover-v1. 5: Harnessing proof assistant feedback for reinforcement learning and monte-carlo tree search. *arXiv preprint arXiv:2408.08152*, 2024.

[77] Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is dpo superior to ppo for llm alignment? a comprehensive study. *arXiv preprint arXiv:2404.10719*, 2024.

[78] Xiangyuan Xue, Zeyu Lu, Di Huang, Wanli Ouyang, and Lei Bai. Genagent: Build collaborative ai systems with automated workflow generation–case studies on comfyui. *arXiv preprint arXiv:2409.01392*, 2024.

[79] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

[80] Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. Rest-mcts*: Llm self-training via process reward guided tree search. *arXiv preprint arXiv:2406.03816*, 2024.

[81] Jiayi Zhang, Jinyu Xiang, Zhaoyang Yu, Fengwei Teng, Xionghui Chen, Jiaqi Chen, Mingchen Zhuge, Xin Cheng, Sirui Hong, Jinlin Wang, et al. Aflow: Automating agentic workflow generation. *arXiv preprint arXiv:2410.10762*, 2024.

[82] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. LLaMA-adapter: Efficient fine-tuning of large language models with zero-initialized attention. In *The Twelfth International Conference on Learning Representations*, 2024.

[83] Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei

Chang, Peng Gao, et al. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? *ECCV 2024*, 2024.

[84] Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Yichi Zhang, Ziyu Guo, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, Shanghang Zhang, et al. Mavis: Mathematical visual instruction tuning. *arXiv preprint arXiv:2407.08739*, 2024.

[85] Ruohong Zhang, Bowen Zhang, Yanghao Li, Haotian Zhang, Zhiqing Sun, Zhe Gan, Yinfei Yang, Ruoming Pang, and Yiming Yang. Improve vision language model chain-of-thought reasoning. *arXiv preprint arXiv:2410.16198*, 2024.

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

[87] Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425*, 2023.

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

[89] Qihao Zhu, Daya Guo, Zhihong Shao, Dejian Yang, Peiyi Wang, Runxin Xu, Y Wu, Yukun Li, Huazuo Gao, Shirong Ma, et al. Deepseek-coder-v2: Breaking the barrier of closed-source models in code intelligence. *arXiv preprint arXiv:2406.11931*, 2024.

A. Data and Implementation Details

A.1. ORM

Zero-shot ORM. To implement a zero-shot ORM in image generation, we adopt a pre-trained LLaVA-OneVision (7B) [34] for test-time verification. We adopt a simple prompt to elicit its capability for text-to-image evaluation, which we observe performs well in most cases, as below:

> **Prompt:** *"**<image>** This image is generated by a prompt: **<prompt>**. Does this image accurately represent the prompt? Please answer yes or no without explanation."*

The '***<image>***' and '***<prompt>***' denote and generated image from Show-o [75] and the input textual prompt.

ORM Ranking Data Curation. To obtain the fine-tuned ORM from LLaVA-OneVision, we curate 288K text-to-image ranking examples as specified in the main paper. We adopt the same prompt in the instruction as the zero-shot ORM, and label 'yes' or 'no' in the response to denote the positive or negative instance, as showcased below:

> **Instruction:** *"**<image>** This image is generated by a prompt: **<prompt>**. Does this image accurately represent the prompt? Please answer yes or no without explanation."*
>
> **Response:** *"Yes"* or *"No"*

A.2. PRM

Zero-shot PRM. We also utilize the pre-trained LLaVA-OneVision (7B) as our zero-shot PRM, applying similar prompt template used in ORM as:

> **Prompt:** *"**<image>** This is an intermediate image in the generation process by a prompt: **<prompt>**. Does this intermediate image accurately represent the prompt? Please answer yes or no without explanation."*

At each intermediate step in the generation process, the zero-shot PRM assesses each candidate image with a binary response, 'yes' or 'no'. We then adopt a step-level best-of-N strategy, selecting the most confident candidate and following this path for subsequent decoding. By iteratively employing the PRM at each step, the generation process is guided step by step towards the final output.

PRM Ranking Data Curation. We observe that the images generated at intermediate steps tend to appear very blurry, as only partial visual tokens in specific regions are decoded while others remain unresolved. Since LLaVA-OneVision is pre-trained only on natural images (similar to those generated at the final step), the zero-shot PRM has limited capability for precise step-wise evaluation. To address this issue, we curate a 300K step-wise text-to-image ranking dataset to fine-tune an improved PRM. We adopt the same prompt in the instruction as the zero-shot PRM, formulated as:

> **Instruction:** *"**<image>** This is an intermediate image in the generation process by a prompt: **<prompt>**. Does this intermediate image accurately represent the prompt? Please answer yes or no without explanation."*
>
> **Response:** *"Yes"* or *"No"*

First, we utilize the 13K unique text prompts from our ORM ranking dataset, generating 18 intermediate-step images per prompt using Show-o. Inspired by Math-Shepherd [70], we employ an automated annotation approach to obtain accurate step-wise labels, eliminating the need for costly human labor or GPT assistance. For instance, to label the image at step i ($1 \leq i \leq 18$), we condition Show-o on that image and then produce four different paths for the remaining 18 - i steps. By evaluating the final images from each of these paths, if any path receives a 'yes' score, it indicates that step i has a high potential to lead to a correct final output, and thus it is labeled as 'yes'; otherwise, it is labeled as 'no'. This automated approach allows us to efficiently obtain step-wise annotations for assessing the generation.

Fine-tuned PRM. With the step-wise ranking data, the LLaVA-OneVision is fine-tuned to boost the visual comprehension of intermediate-step images. The data format and training configurations are the same as those used for fine-tuning the ORM. After training, the PRM becomes more capable of interpreting blurry images within the decoding process for more accurate step-by-step selection.

A.3. PARM

In Figure 10, we illustrate why PRM is less suitable for autoregressive image generation. As shown, the early-stage images are too blurry for reliable evaluation, given that only a few regions are decoded, while the later-stage images derived from similar previous steps lack sufficient distinction, challenging for discrimination. To integrate the advantage of both ORM and PRM, we propose Potential Assessment Reward Model (PARM) and curate a new ranking dataset

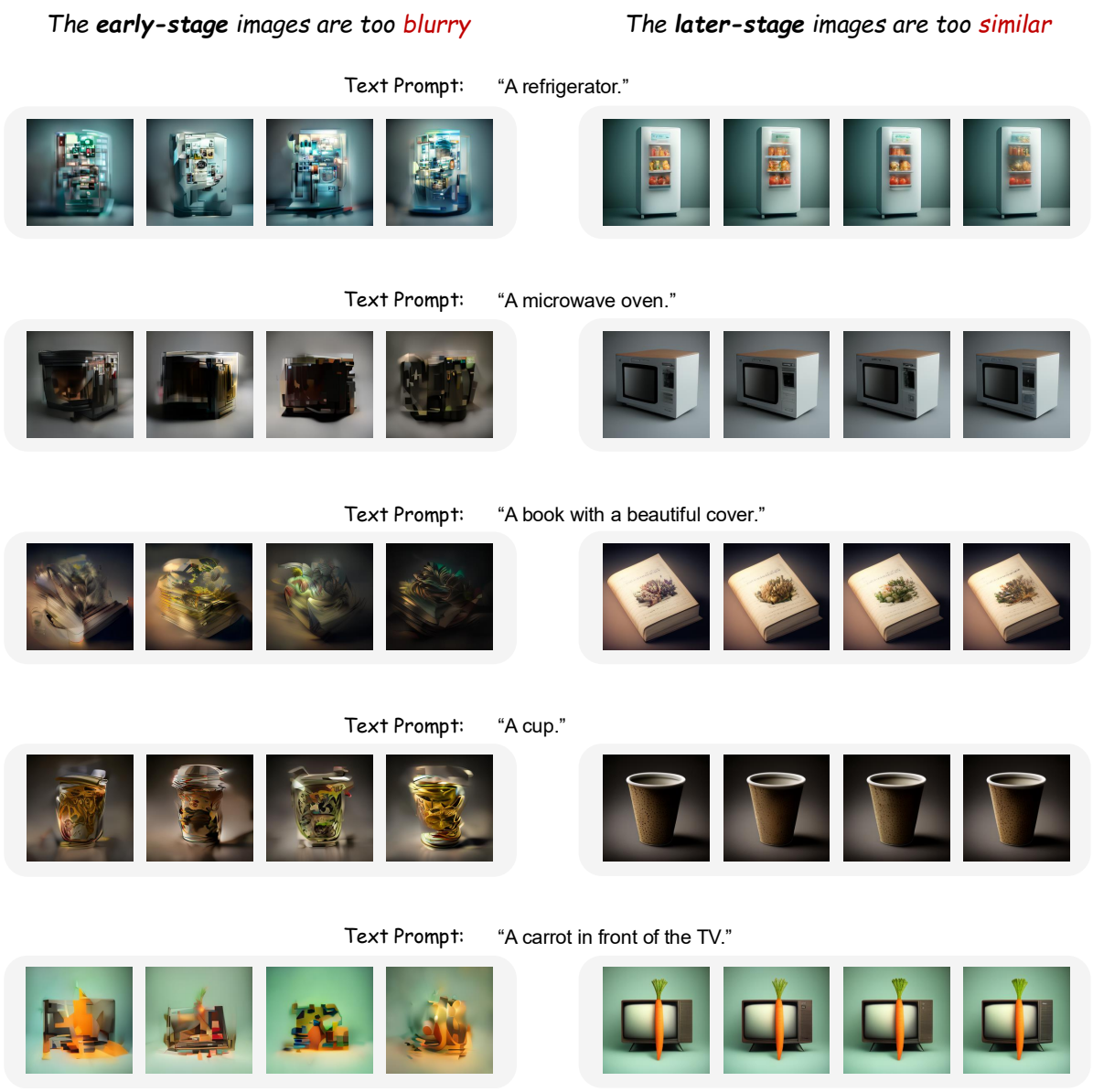

Figure 10. **Visualization of Early-stage and Later-stage Images.** We visualize the generated images in the intermediate steps of Show-o [75], where the early-stage images are too blurry to interpret, while the later-stage images are too similar to discriminate, posing great challenges for PRMs to effectively evaluate.

with 400K instances by re-annotating the 13K text prompts from ORM ranking data. The dataset is structured into three subsets corresponding to the three evaluation tasks:

Clarity Judgment Data (120K). Through comprehensive analysis, we observe that the baseline model (Show-o) typically produces its first clear image between steps 8 and 12 within the 18-step generation, qualifying it for potential assessment. Based on this, we simplify the annotation by labeling steps after 11 as 'yes' and those before 10 as 'no'. Although this approach is static, the trained PARM acquires generalization skills to adaptively identify the first 'yes' label within steps 8∼12 during inference. The data format is shown below:

> **Instruction:** *"**<image>** This image is a certain step in the text-to-image generation process with a prompt: **<prompt>**. It is not the final generated one, and will keep iterating better. Do you think this image can be used to judge whether it has the potential to iterate to the image satisfied the prompt? (The image, which needn't to be confused but can be clear and basically*

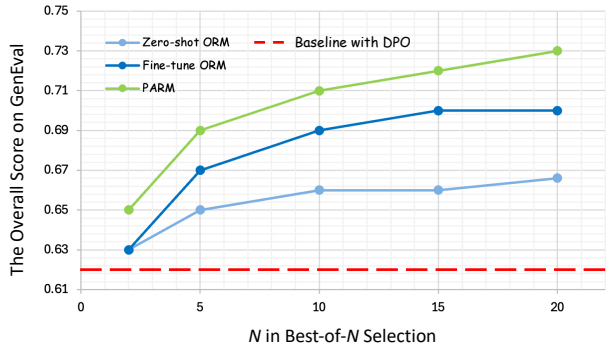

Figure 11. **Comparison of Reward Models as Test-time Verifiers with DPO Alignment.** We adopt Show-o [75] with DPO alignment as the 'Baseline with DPO' and evaluate Best-of-N selection on the GenEval [20] benchmark.

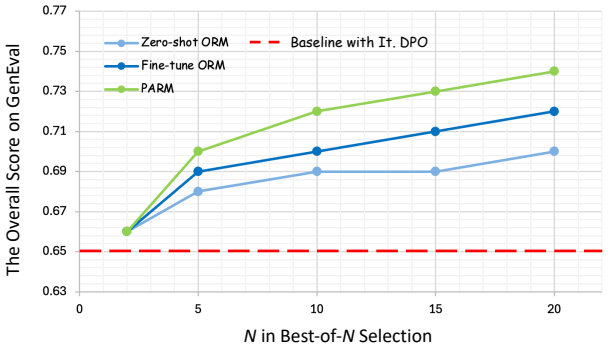

Figure 12. **Comparison of Reward Models as Test-time Verifiers with Iterative DPO Alignment.** We adopt Show-o [75] with iterative DPO alignment as the 'Baseline with It. DPO' and evaluate Best-of-N selection on the GenEval [20] benchmark.

> *judged the object, can be used to judge the potential) Answer yes or no without explanation."*
>
> **Response:** *"Yes"* or *"No"*

Potential Assessment Data (80K). We assign intermediate images from steps after 11 with a 'yes' or 'no' label, which is based on the final output label of that path in the ORM data annotation. In practice, if the previous clarity judgment task yields 'yes', the data of this task is organized as a follow-up question-answering within a multi-turn conversation. The data sample of this task is formulated as:

> **Instruction:** *"**<image>** Do you think whether the image has the potential to iterate to the image satisfied the prompt? Please answer yes or no without explanation."*
>
> **Response:** *"Yes"* or *"No"*

Best-of-N' Selection Data (200K). We directly utilize the labels in the ORM ranking dataset, with the format as

> **Instruction:** *"**<image>** This image is generated by a prompt: **<prompt>**. Does this image accurately represent the prompt? Please answer yes or no without explanation."*
>
> **Response:** *"Yes"* or *"No"*

B. Additional Results

Quantitative Results. In Table 4, we present a comprehensive performance comparison on GenEval [20] between previous diffusion and autoregressive models, and Shwo-o equipped with our investigated reasoning strategies. Substantial improvement for text-to-image generation are observed using different reasoning techniques. With PARM, the gains in complex attributes, such as 'Two Obj.', 'Counting', 'Position', and 'Attribute binding' emphasize the robustness of our approach in handling challenging aspects of compositional generation, setting a new standard in text-to-image performance. In Figures 11 and 12, we present the performance of test-time verification integrated with DPO [55] and iterative DPO, respectively, instead of the test-time verification only in Figure 2 of the main paper. As shown, our propose PARM both achieves the best results as the N increases for best-of-N selection.

Table 4. **Performance Comparison on the GenEval [20] Benchmark.** Compared to existing diffusion and autoregressive models, we investigate the potential of Chain-of-Thought (CoT) reasoning strategies in text-to-image generation. 'Zs.', 'Ft.', and 'It. DPO' denote the zero-shot, fine-tuned verifiers, and iterative DPO [55], repsectively. **PARM** refers to our proposed Potential Assessment Reward Model specialized for autoregressive image generation. We adopt the best-of-20 selection for test-time verifiers by default, and highlight the best and second-best overall scores in green and red.

Model	Test-time Verifier	Preference Alignment	Reward Guidance	Single object	Two object	Counting	Colors	Position	Attribute binding	**Overall**
PixArt-α [7]	-	-	-	0.98	0.50	0.44	0.80	0.08	0.07	0.48
SD v2.1 [60]	-	-	-	0.98	0.51	0.44	0.85	0.07	0.17	0.50
DALL-E 2 [57]	-	-	-	0.94	0.66	0.49	0.77	0.10	0.19	0.52
SDXL [54]	-	-	-	0.98	0.74	0.39	0.85	0.15	0.23	0.55
SD 3 (d=24) [13]	-	-	-	0.98	0.74	0.63	0.67	0.34	0.36	0.62
LlamaGen [64]	-	-	-	0.71	0.34	0.21	0.58	0.07	0.04	0.32
Chameleon [66]	-	-	-	-	-	-	-	-	-	0.39
LWM [37]	-	-	-	0.93	0.41	0.46	0.79	0.09	0.15	0.47
SEED-X [18]	-	-	-	0.97	0.58	0.26	0.80	0.19	0.14	0.49
Show-o [75]	-	-	-	0.95	0.52	0.49	0.82	0.11	0.28	0.53
	Zs. ORM	-	-	0.99	0.63	0.63	0.84	0.19	0.39	0.61
	Ft. ORM	-	-	0.99	0.72	0.65	0.84	0.25	0.33	0.63
	Zs. PRM	-	-	0.98	0.51	0.54	0.82	0.11	0.23	0.53
	Ft. PRM	-	-	0.98	0.55	0.54	0.83	0.13	0.29	0.55
	PARM	-	-	0.99	0.77	0.68	0.86	0.29	0.45	0.67
	-	DPO	-	0.96	0.70	0.50	0.82	0.30	0.43	0.62
	-	It. DPO	-	0.98	0.72	0.53	0.84	0.40	0.46	0.65
	Zs. ORM	It. DPO	-	0.99	0.79	0.63	0.85	0.44	0.50	0.70
	Ft. ORM	It. DPO	-	0.98	0.80	0.62	0.83	0.59	0.54	0.72
	PARM	It. DPO	-	0.98	0.83	0.64	0.84	0.59	0.62	0.74
	-	It. DPO	Ft. ORM	0.98	0.80	0.62	0.83	0.59	0.54	0.72
	-	It. DPO	**PARM**	0.97	0.75	0.60	0.83	0.54	0.53	0.69
	Ft. ORM	It. DPO	Ft. ORM	0.98	0.84	0.64	0.85	0.66	0.52	0.75
	PARM	It. DPO	**PARM**	0.99	0.86	0.67	0.84	0.66	0.64	0.77