Supplementary Material:
Let's Verify and Reinforce Image Generation Step by Step

Renrui Zhang*†1, Chengzhuo Tong*4, Zhizheng Zhao*3, Ziyu Guo*2, Haoquan Zhang4
Manyuan Zhang1, Jiaming Liu3, Peng Gao4,5, Hongsheng Li‡1,6

CUHK 1MMLab & 2MiuLar Lab 3Peking University
4Shanghai AI Lab 5SIAT 6CPII under InnoHK

A. 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 [? ? ? ?], code synthesis [? ? ?], and workflow generation [? ? ?]. One line of research adapts the input space to leverage Chain-of-Thought (CoT) capabilities, using approaches like in-context CoT examples [?] or zero-shot CoT prompts [?]. Another branch modifies or integrates reasoning paths within the output space, utilizing strategies such as self-consistency [?], CoT decoding [?], and verifier-based selection [? ? ?]. Among these, test-time verifiers have demonstrated generality and robustness in enhancing reasoning performance. For example, early work [?] trains an Outcome Reward Model (ORM) to evaluate final outputs and select the best-of-N candidates for optimal results. Later, Lightman et al. [? ?] adopt the Process Reward Model (PRM) to evaluate intermediate reasoning steps, achieving greater effectiveness. Snell et al. [?] further highlights that scaling test-time computation is often more impactful than scaling model parameters during training. Recently, OpenAI o1 [?] 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 [? ? ?] or Chain-of-Thought (CoT) reasoning [? ? ?]. Traditional approaches [? ? ? ?] 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 [?]), and then fine-tuning the target LLM to maximize this reward using techniques like proximal policy optimization (PPO) [?]. However, RL-based methods often encounter issues related to complexity and instability. To overcome these challenges, Rafailov et al. introduced Direct Preference Optimization (DPO) [?], 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 [? ?] and code generation [? ? ?]. Further advancements have extended DPO with step-wise preference data [? ?] for more granular supervision and multi-modality learning [? ?] 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 [? ? ? ? ? ? ?] have demonstrated a remarkably successful modeling approach in language and multi-modality. Motivated by such progress, a series of work, e.g., DALL-E [?], LlamaGen [?], and Chameleon [?], utilizes such autoregressive modeling with casual attention to learn the dependency within image pixels for image generation tasks, rather than popular diffusion models [? ? ? ?]. However,

*Equal Contribution †Project Lead ‡Corresponding Author 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 [? ? ? ?]. To address the challenges, MaskGiT [?] 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 [?]) and its integration with diffusion techniques (MAR [?]). 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.

B. Data and Implementation Details

B.1. ORM

Zero-shot ORM. To implement a zero-shot ORM in image generation, we adopt a pre-trained LLaVA-OneVision (7B) [?] 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 [?] 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”*

B.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 [?], 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.

B.3. PARM

In Figure 1, 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 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 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"*

C. Additional Results

Quantitative Results. In Table 1, we present a comprehensive performance comparison on GenEval [?] 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 2 and 3, we present the performance of test-time verification integrated with DPO [?] 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.

Qualitative Results. In Figures 4, 5, 6, 7, and 8, we showcase qualitative examples of text-to-image generation comparing the baseline, Show-o, and our best-performing configuration, which integrates PARM with iterative DPO for both reward model guidance and test-time verification. Our results demonstrate that this approach significantly improves the generation quality, achieving stronger alignment between the generated images and the input text prompts.

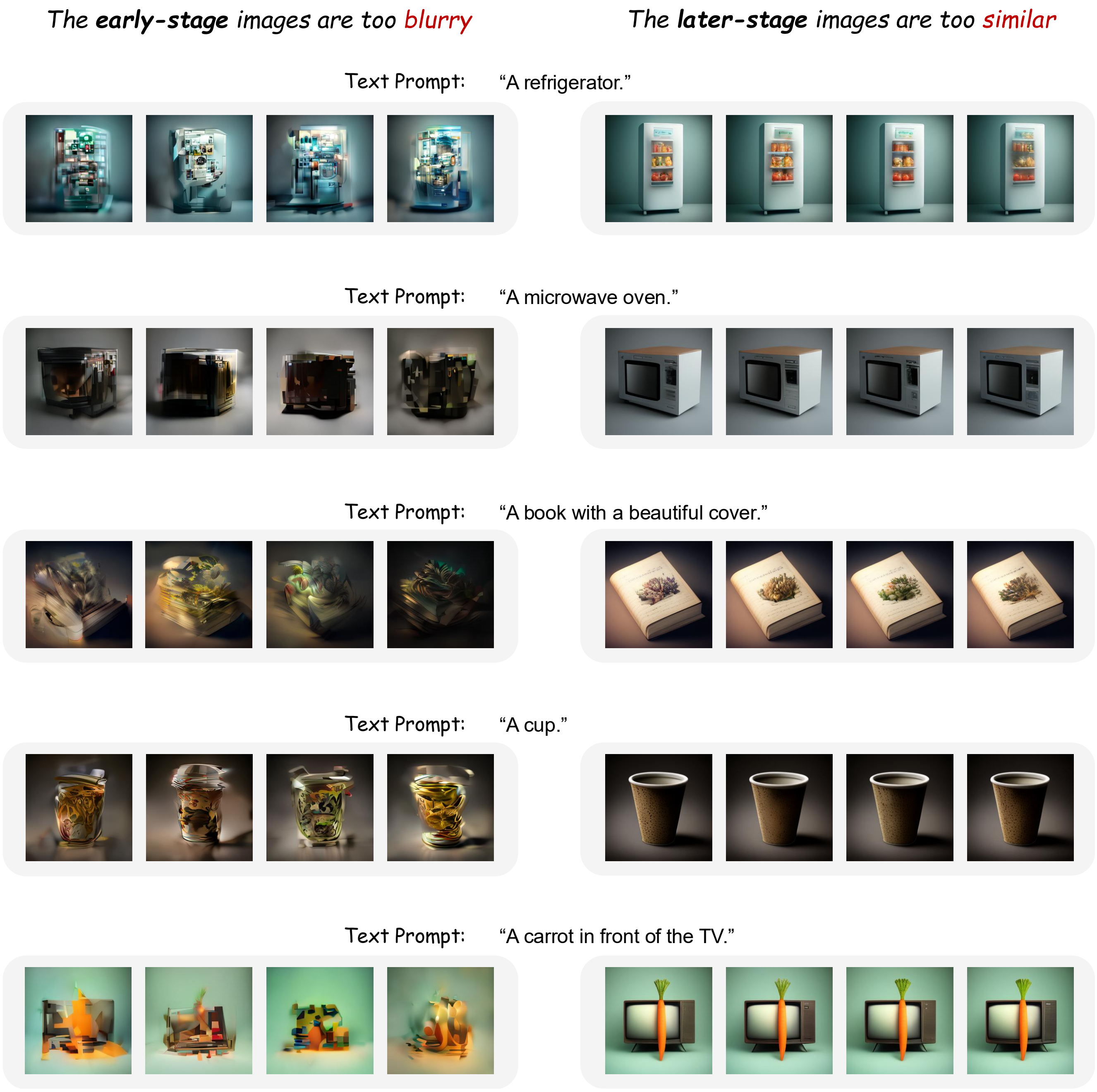
The early-stage images are too blurry
The later-stage images are too similar
Text Prompt: "A refrigerator."
Text Prompt: "A microwave oven."
Text Prompt: "A book with a beautiful cover."
Text Prompt: "A cup."
Text Prompt: "A carrot in front of the TV."

Figure 1. **Visualization of Early-stage and Later-stage Images.** We visualize the generated images in the intermediate steps of Show-o [**?**], 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.

Specifically, we observe that baseline models often generates inaccurate spatial relationships between objects, produce 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.

Table 1. **Performance Comparison on the GenEval [?] 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 [?], 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 [?]	-	-	-	0.98	0.50	0.44	0.80	0.08	0.07	0.48
SD v2.1 [?]	-	-	-	0.98	0.51	0.44	0.85	0.07	0.17	0.50
DALL-E 2 [?]	-	-	-	0.94	0.66	0.49	0.77	0.10	0.19	0.52
SDXL [?]	-	-	-	0.98	0.74	0.39	0.85	0.15	0.23	0.55
SD 3 (d=24) [?]	-	-	-	0.98	0.74	0.63	0.67	0.34	0.36	0.62
LlamaGen [?]	-	-	-	0.71	0.34	0.21	0.58	0.07	0.04	0.32
Chameleon [?]	-	-	-	-	-	-	-	-	-	0.39
LWM [?]	-	-	-	0.93	0.41	0.46	0.79	0.09	0.15	0.47
SEED-X [?]	-	-	-	0.97	0.58	0.26	0.80	0.19	0.14	0.49
Show-o [?]	-	-	-	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

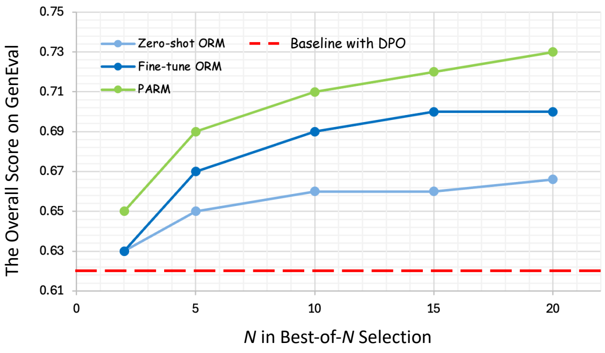

Figure 2. **Comparison of Reward Models as Test-time Verifiers with DPO Alignment.** We adopt Show-o [?] with DPO alignment as the 'Baseline with DPO' and evaluate Best-of-N selection on the GenEval [?] benchmark.

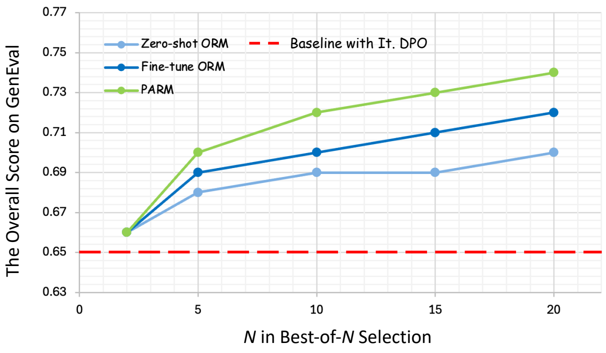

Figure 3. **Comparison of Reward Models as Test-time Verifiers with Iterative DPO Alignment.** We adopt Show-o [?] with iterative DPO alignment as the 'Baseline with It. DPO' and evaluate Best-of-N selection on the GenEval [?] benchmark.

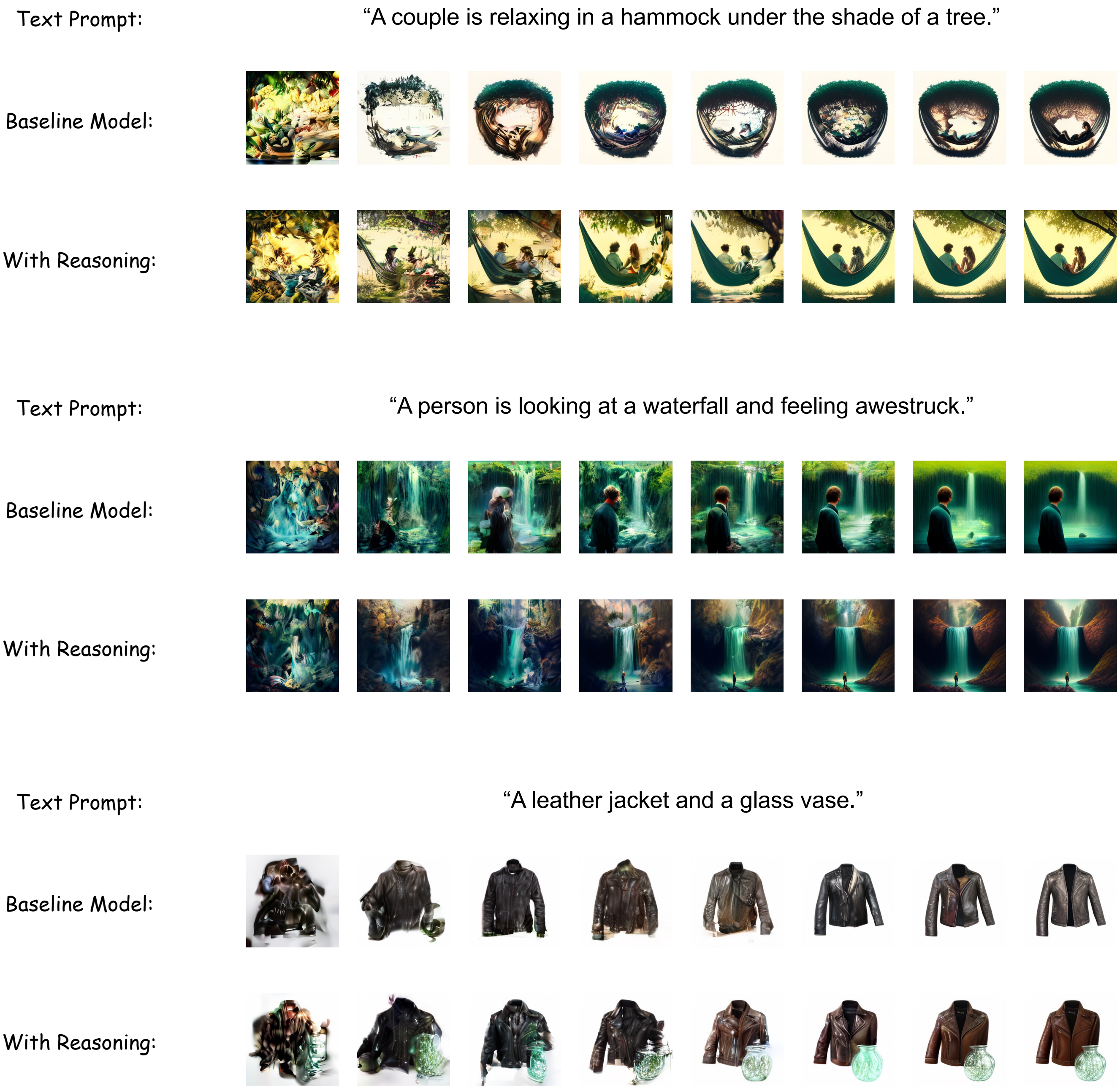
Text Prompt:
“A couple is relaxing in a hammock under the shade of a tree.”
Baseline Model:
With Reasoning:
Text Prompt:
“A person is looking at a waterfall and feeling awestruck.”
Baseline Model:
With Reasoning:
Text Prompt:
“A leather jacket and a glass vase.”
Baseline Model:
With Reasoning:

Figure 4. **Qualitative Results using Our Reasoning Strategies.** Show-o [**?**] 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.

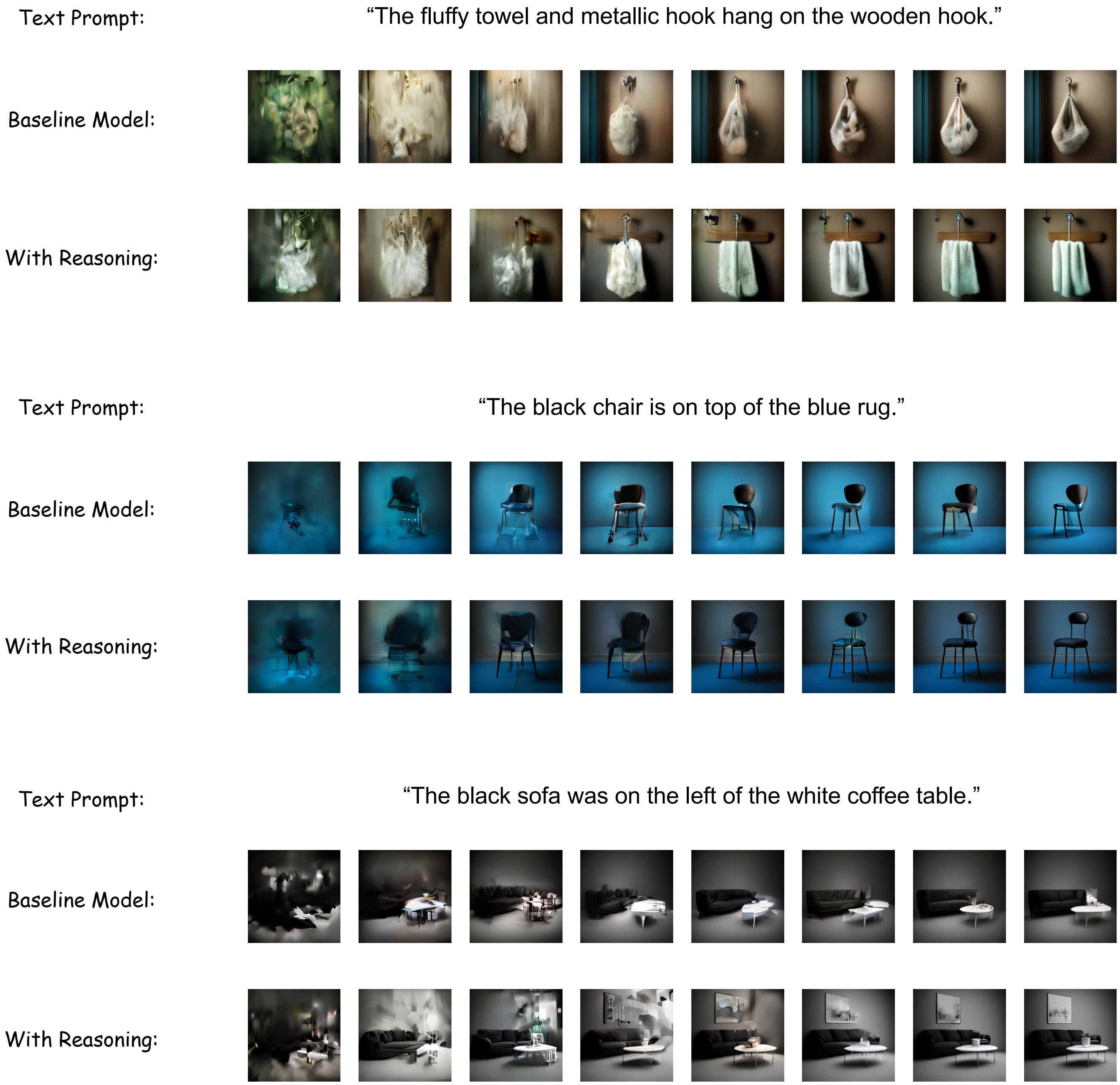
Text Prompt:
“The fluffy towel and metallic hook hang on the wooden hook.”
Baseline Model:
With Reasoning:
Text Prompt:
“The black chair is on top of the blue rug.”
Baseline Model:
With Reasoning:
Text Prompt:
“The black sofa was on the left of the white coffee table.”
Baseline Model:
With Reasoning:

Figure 5. **Qualitative Results using Our Reasoning Strategies.** Show-o [**?**] 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.

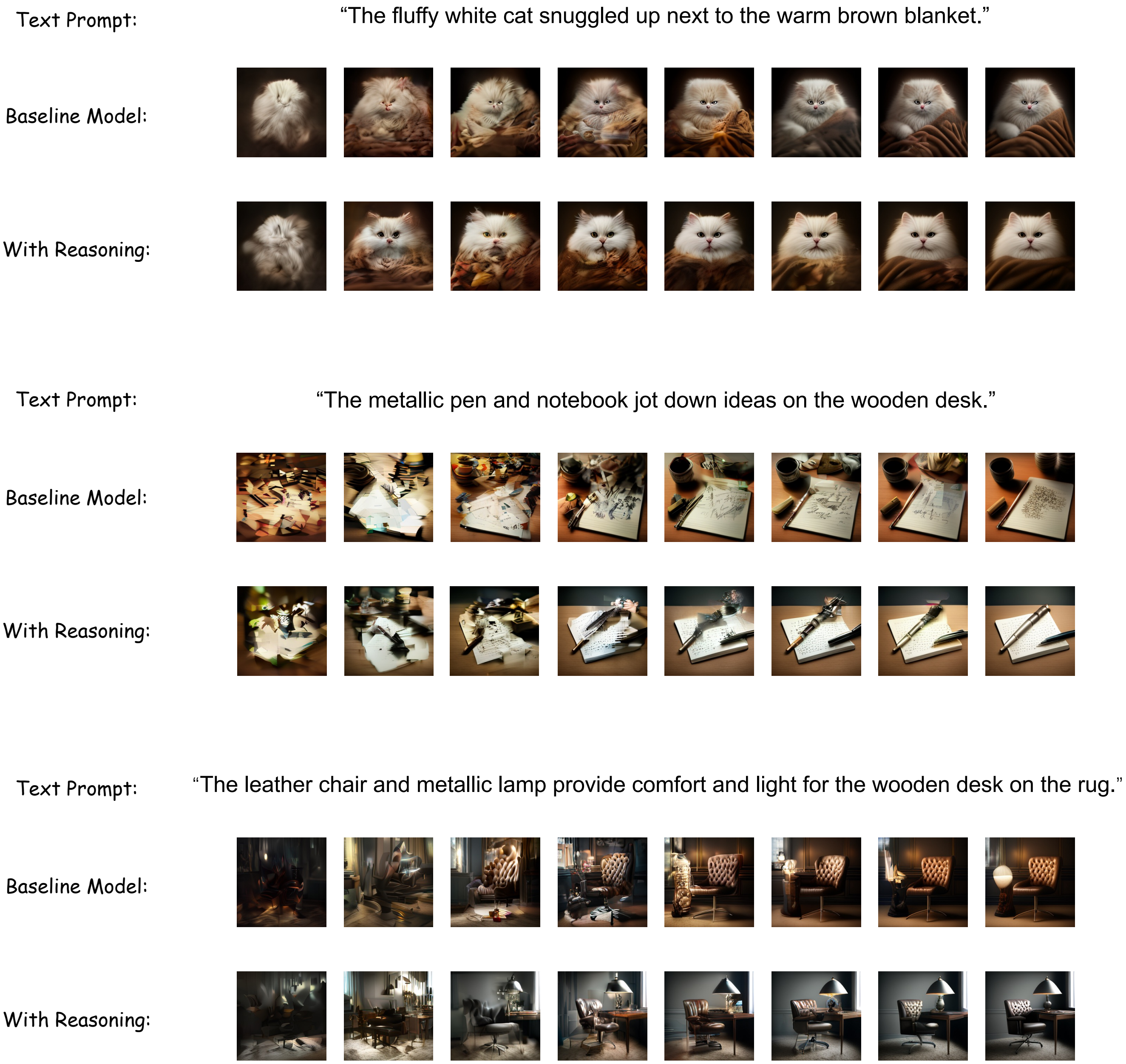
Text Prompt:
"The fluffy white cat snuggled up next to the warm brown blanket."
Baseline Model:
With Reasoning:
Text Prompt:
"The metallic pen and notebook jot down ideas on the wooden desk."
Baseline Model:
With Reasoning:
Text Prompt:
"The leather chair and metallic lamp provide comfort and light for the wooden desk on the rug."
Baseline Model:
With Reasoning:

Figure 6. **Qualitative Results using Our Reasoning Strategies.** Show-o [?] 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.

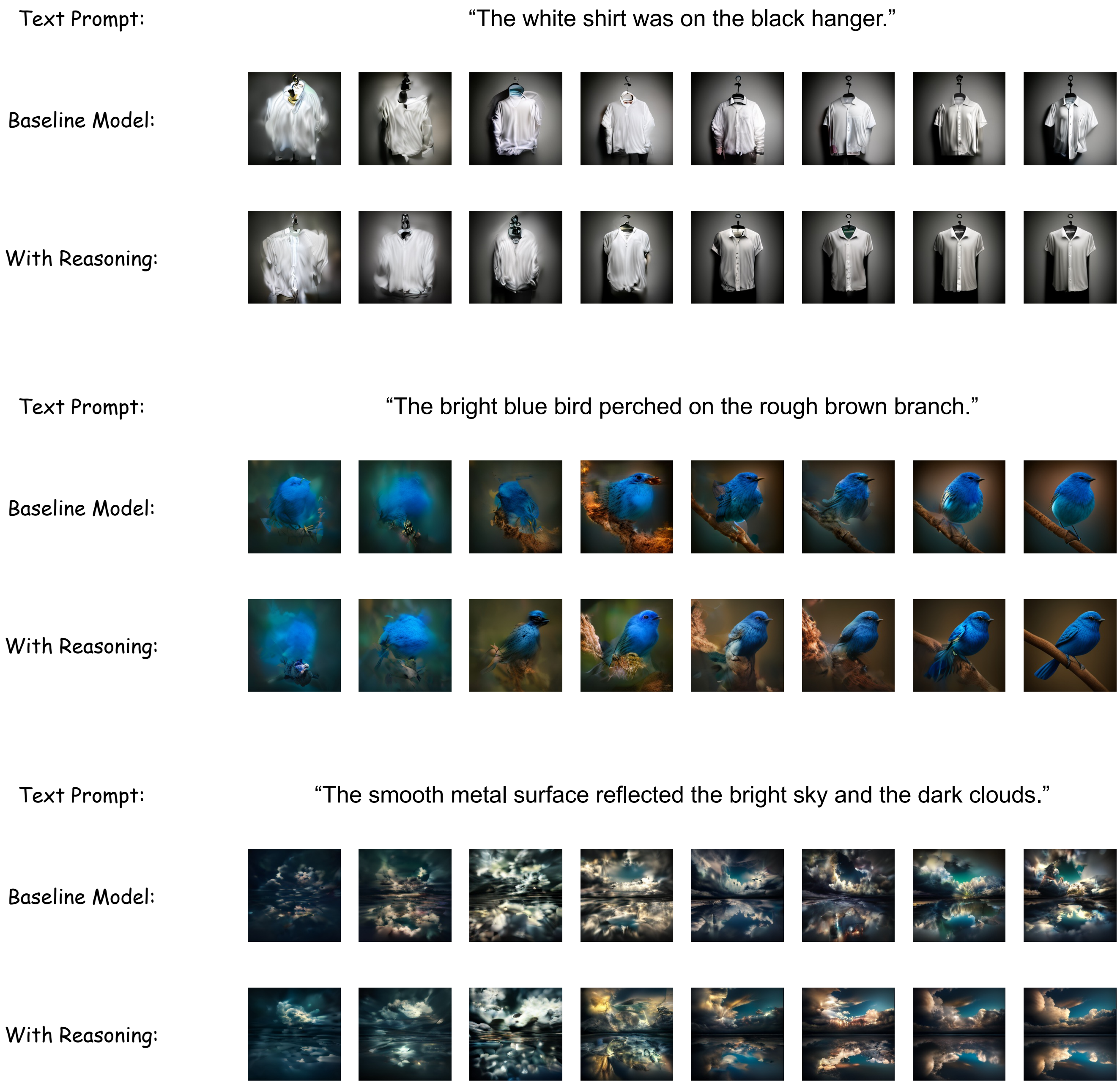
Text Prompt:
"The white shirt was on the black hanger."
Baseline Model:
With Reasoning:
Text Prompt:
"The bright blue bird perched on the rough brown branch."
Baseline Model:
With Reasoning:
Text Prompt:
"The smooth metal surface reflected the bright sky and the dark clouds."
Baseline Model:
With Reasoning:

Figure 7. **Qualitative Results using Our Reasoning Strategies.** Show-o [?] 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.

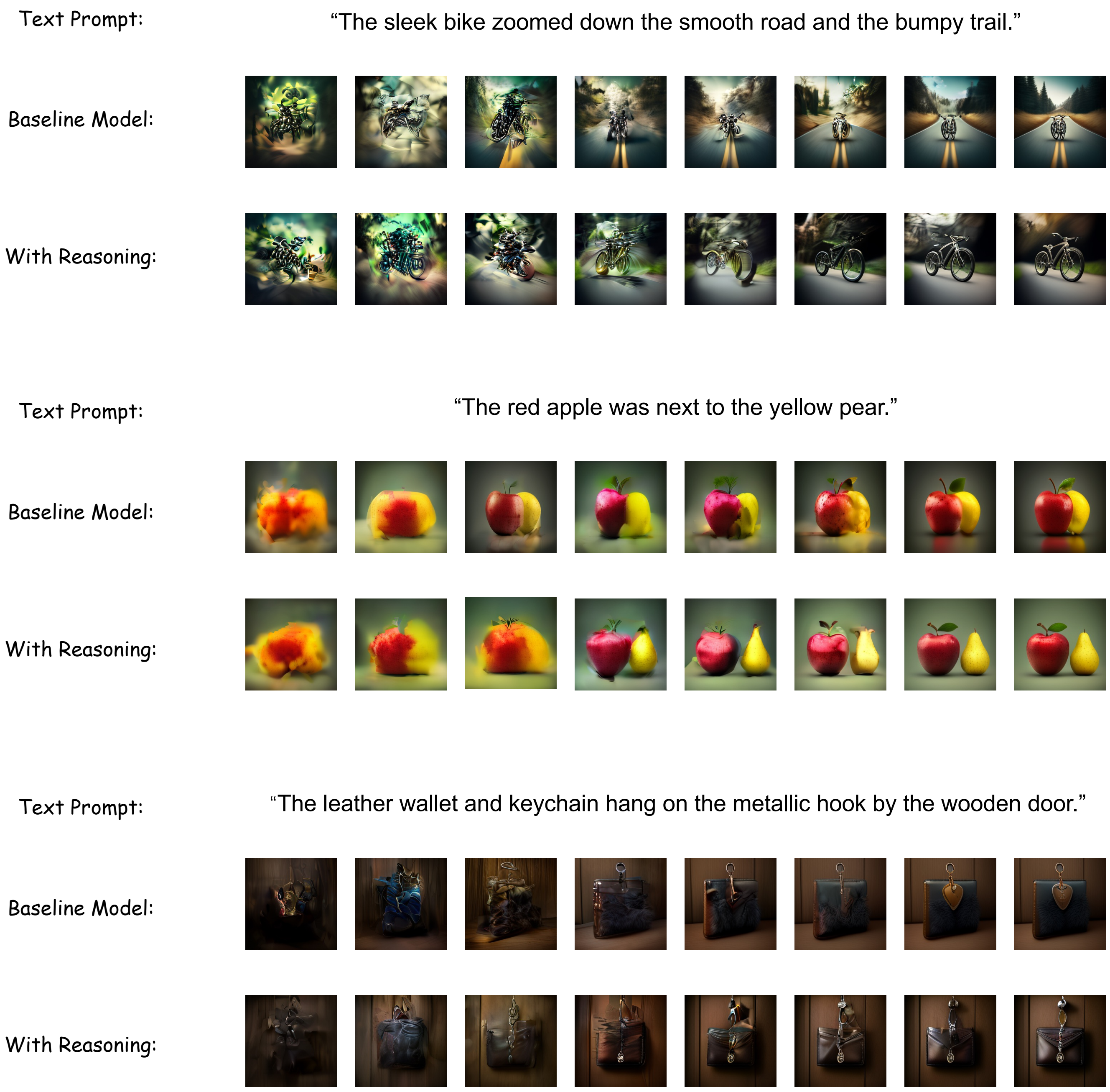
Text Prompt:
“The sleek bike zoomed down the smooth road and the bumpy trail.”
Baseline Model:
With Reasoning:
Text Prompt:
“The red apple was next to the yellow pear.”
Baseline Model:
With Reasoning:
Text Prompt:
“The leather wallet and keychain hang on the metallic hook by the wooden door.”
Baseline Model:
With Reasoning:

Figure 8. **Qualitative Results using Our Reasoning Strategies.** Show-o [?] 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.