ImageGen-CoT: Enhancing Text-to-Image In-context Learning with Chain-of-Thought Reasoning

Supplementary Material

Overview

In this supplementary material, we present more details and more experimental results that are not included in the main paper. The contents include:

- A detailed introduction to CoBSAT [2] and Dreambench++[1] in Sec. S-A.
- Additional details on the experimental setup in Sec. S-B.
- More experimental results in Sec. S-C.
- Effectiveness of Iterative Refinement Strategy in Data Construction in Sec. S-D.
- Computational cost of the proposed test-time scaling method in Sec. S-E.
- Automatic Dataset Construction Pipeline On CoBSAT S-F.

S-A. Dataset Details

CoBSAT: CoBSAT [2] is a comprehensive benchmark dataset designed specifically to evaluate Text-to-Image In-Context Learning (T2I-ICL) capabilities of Multimodal Large Language Models (MLLMs). The dataset consists of ten distinct tasks across five thematic areas, with each task carefully structured to assess different aspects of T2I-ICL performance. The benchmark is organized into two main categories: object-inference tasks and attribute-inference tasks. In object-inference tasks, models must infer the correct object from demonstrations while being given explicit attributes in the text prompt. Conversely, in attribute-inference tasks, models are provided with the object in the text prompt and must infer the appropriate attribute from the demonstrations. This dual structure enables a thorough evaluation of MLLMs' ability to learn and generalize from multimodal in-context examples.

Dreambench++: Dreambench++ [1] is a comprehensive benchmark for evaluating personalized text-to-image generation models. It features three key advantages: 1) Human-aligned evaluation through carefully designed GPT prompting that achieves over 79% agreement with human assessments; 2) Fully automated evaluation process that eliminates the need for time-consuming manual evaluation; and 3) A diverse dataset containing 150 images and 1,350 prompts across various categories including animals, humans, objects and styles. The benchmark evaluates two fundamental aspects of personalized image generation: concept preservation and prompt following capabilities.

S-B. Detailed Experimental Setup

S-B.1. CoBSAT

Data Split. Following CoBSAT's default settings, we split the predefined lists of text inputs (X) and latent variables (Θ) into training and testing subsets with a 1:1 ratio, ensuring the test set contains completely unseen prompts and attributes. For training, we generate 300 samples per task by enumerating all possible combinations of $\theta \in \Theta_{\text{train}}$ and $(x_n)_{n=1}^{N+1} \in X_{\text{train}}^{N+1}$, resulting in 3,000 training samples across 10 tasks. For evaluation, we randomly sample 250 prompts per task from $\theta \in \Theta_{\text{test}}$ and $(x_n)_{n=1}^{N+1} \in X_{\text{test}}^{N+1}$, yielding a total of 2,500 test samples.

Training Strategy. For model training, we fine-tune both SEED-LLaMA and SEED-X using LoRA. Specifically, SEED-LLaMA is fine-tuned with rank=64, α=16, learning rate=1e-4 for 1 epoch, while SEED-X uses rank=64, α=64, learning rate=1e-4 for 1 epoch.

S-B.2. Dreambench++

Data Split. To prevent subject overlap in evaluation, we split the dataset by subjects, with 60% subjects (90 subjects, resulting in 810 samples) for training and 40% subjects (60 subjects, resulting in 540 samples) for testing.

Training Strategy. For Dreambench++, SEED-LLaMA is fine-tuned using LoRA with rank=64, α=16, learning rate=1e-4 for 5 epochs, while SEED-X uses rank=64, α=64, learning rate=1e-4 for 1 epoch.

S-C. More experimental results

As described in the main paper, the ImageGen-CoT dataset comprises two distinct components. The first component focuses on training the model to generate ImageGen-CoT text, while the second component teaches the model to generate images based on the generated ImageGen-CoT text. While our main paper primarily focused on training using Part I of the dataset, here we extend our experiments by utilizing the complete dataset for comprehensive evaluation. As presented in Tables S-1 and S-2, we conducted comprehensive experiments using both parts of the ImageGen-CoT dataset. On the CoBSAT benchmark, SEED-LLaMA fine-tuned with the complete ImageGen-CoT dataset achieved a significant performance

Table S-1. **Main results on CoBSAT benchmark.** "FT w/ GT Image" denotes fine-tuning with ground truth images, while "FT w/ ImageGen-CoT" represents fine-tuning with our ImageGen-CoT dataset. The results demonstrate that ImageGen-CoT significantly improves model performance, with relative improvements over baseline model shown in red.

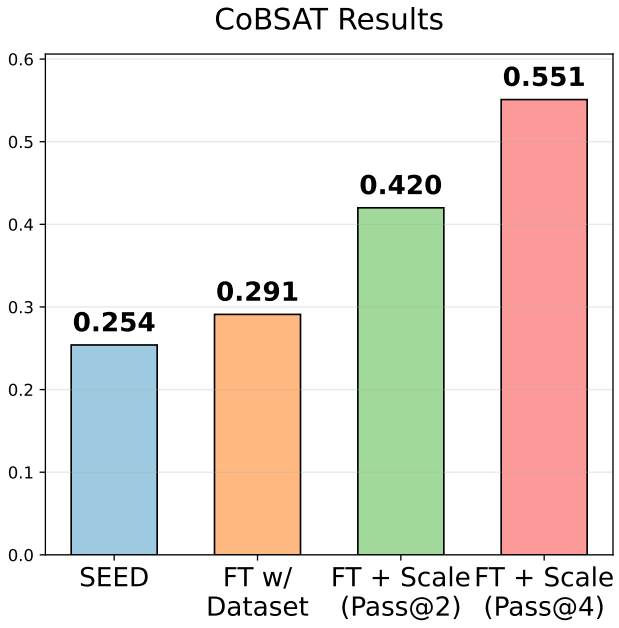
Method	Object-Inference Task					Attribute-Inference Task					Avg.↑
	Color-I	Bkg-I	Style-I	Action-I	Texture-I	Color-II	Bkg-II	Style-II	Action-II	Texture-II	
SEED-LLaMA	.616	.216	.272	.592	.112	.088	.168	.192	.220	.056	.254
+ ImageGen-CoT (via Prompt)	.700	.276	.300	.408	.084	.176	.292	.272	.192	.132	.283
+ FT w/ GT Image	.632	.272	.352	.540	.128	.164	.200	.256	.172	.112	.283
+ FT w/ ImageGen-CoT Dataset (Part1)	.620	.368	.384	.424	.060	.192	.288	.208	.216	.148	.291
+ FT w/ ImageGen-CoT Dataset (All Part)	.716	.432	.436	.420	.200	.168	.380	.256	.216	.248	**.347** ↑36.6%
SEED-X	.796	.412	.316	.596	.240	.176	.344	.260	.252	.104	.349
+ ImageGen-CoT (via Prompt)	.724	.440	.660	.784	.216	.312	.472	.228	.320	.240	.439
+ FT w/ GT Image	.936	.712	.896	.860	.468	.280	.324	.388	.636	.424	.592
+ FT w/ ImageGen-CoT Dataset (Part1)	.884	.692	.928	.936	.420	.504	.612	.660	.524	.424	**.658**
+ FT w/ ImageGen-CoT Dataset (All Part)	.832	.596	.840	.892	.484	.384	.548	.572	.608	.500	.626 ↑79.4%

Table S-2. **Evaluation results on Dreambench++ benchmark.** CP refers to concept preservation and PF refers to prompt following metrics. "FT" stands for fine-tuning. The relative gains over baseline model are shown in red.

Method	Concept Preservation					Prompt Following				CP·PF↑
	Animal	Human	Object	Style	Overall	Photorealistic	Style	Imaginative	Overall	
SEED-LLaMA	.436	.315	.288	.381	.358	.306	.202	.154	.218	.078
+ ImageGen-CoT (via Prompt)	.390	.241	.262	.346	.317	.291	.211	.170	.222	.078
+ FT w/ ImageGen-CoT Dataset (Part1)	.399	.290	.271	.318	.325	.348	.355	.210	.310	.101
+ FT w/ ImageGen-CoT Dataset (All Part)	.414	.269	.243	.328	.319	.408	.317	.199	.334	**.107** ↑37.2%
SEED-X	.647	.420	.526	.571	.559	.346	.342	.303	.337	.188
+ ImageGen-CoT (via Prompt)	.547	.293	.369	.424	.427	.862	.775	.737	.817	.347
+ FT w/ ImageGen-CoT Dataset (Part1)	.549	.410	.403	.432	.458	.922	.851	.846	.881	**.403**
+ FT w/ ImageGen-CoT Dataset (All Part)	.511	.358	.424	.303	.421	.926	.910	.870	.906	.384 ↑104.2%

gain of +36.6% (0.254 → 0.347) compared to the baseline model. Similarly, SEED-X demonstrated remarkable improvement with a +79.4% increase (0.349 → 0.626) over its baseline performance. For the Dreambench++ benchmark, training with the complete dataset resulted in even more pronounced improvements. SEED-LLaMA showed a +37.2% gain (0.078 → 0.107) in CP·PF score, while SEED-X achieved a substantial +104.2% improvement (0.188 → 0.384). These comprehensive results demonstrate that utilizing the complete ImageGen-CoT dataset can still significantly improve model performance.

Due to page constraints, the main paper only presents the results of the hybrid scaling strategy on SEED-X. Here, we extend the analysis by using a different model, SEED. As shown in the figure below, the hybrid scaling strategy enhances the performance of SEED + FT with ImageGen-CoT, increasing it from 0.291 to 0.551, thereby highlighting its effectiveness.



S-D. Effectiveness of Iterative Refinement Strategy in Data Construction

We evaluate the effectiveness of our iterative refinement strategy on both CoBSAT and Dreambench++ datasets. As demonstrated in Table S-3, the proposed strategy yields consistent improvements across all evaluation metrics. Specifically, on the CoBSAT dataset, our method achieves improvements of 1.1%, 2.9%, and 2.0% in object inference, attribute inference, and overall score, respectively. For Dreambench++, the refinement strategy enhances prompt

Table S-3. Performance comparison of data construction with and without iterative refinement.

(a) Results on CoBSAT

Method	Object	Attribute	Overall
w/o Iterative Refine	0.782	0.704	0.743
w/ Iterative Refine	0.793	0.733	**0.763**

(b) Results on Dreambench++

Method	PF	CP	PF*CP
w/o Iterative Refine	0.937	0.470	0.442
w/ Iterative Refine	0.946	0.517	**0.489**

following (PF) by 0.9% and concept preservation (CP) by 4.7%, resulting in a substantial 4.7% improvement in the combined PF*CP metric. These comprehensive results validate that our iterative refinement approach significantly enhances the quality of the constructed dataset.

S-E. Computational cost of the proposed test-time scaling method

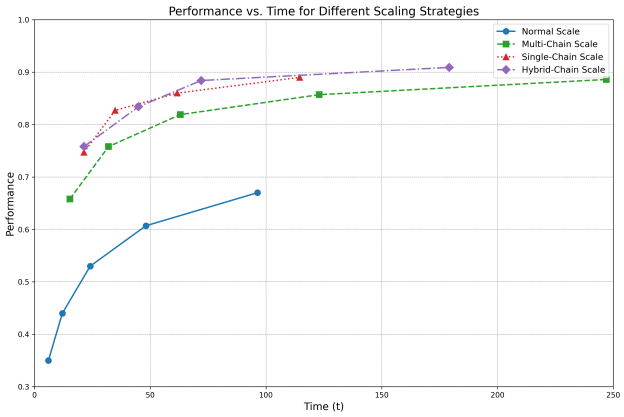


Figure S-1. Computational cost of the proposed test-time scaling method

As shown in the figure S-1, SEED-X + FT with ImageGen-CoT performs equally to SEED-X + Scale (Pass@16). Additionally, our proposed hybrid scaling method also performs the best with the same computational budget. These results demonstrate the practicality of our test-time scaling method.

S-F. Automatic Dataset Construction Pipeline On CoBSAT

On CoBSAT, we initially adopted the same method as DreamBench++. However, we found that the self-boosting approach underperformed due to the inherent complexity of CoBSAT, which requires the model to infer implicit visual-semantic relationships—posing a significant challenge to the model's reasoning capability. To solve this challenge, we sampled multiple text prompts from the MLLM and selected the best prompts using the self-consistency method. However, this method cannot be directly applied to CoBSAT. Self-consistency is commonly used in mathematical problem solving, where text answers are precise (e.g., numbers or options) and consistency can be directly evaluated using string matching. In contrast, CoBSAT involves long and complex text prompts, making direct string-based consistency evaluation infeasible.

The pipeline proceeds as follows: We first sample multiple chains of thought (CoT) from the MLLM. These CoTs are then used, along with the input sequence context, to generate multiple text prompts. Formally, let the CoT sampled from the MLLM be denoted as cot_t^i, where $i = 0, 1, \ldots, M-1$, and M is the number of sampled CoTs. Each CoT, combined with the input sequence x, is used to construct a corresponding text prompt p_t^i as:

$$p_t^i = \mathcal{F}(\text{cot}_t^i, x), \quad i = 0, 1, \ldots, M-1, \tag{1}$$

where $\mathcal{F}$ represents generating text prompts based on the CoT and input sequence context.

Next, we convert each text prompt into a vector representation using a text embedding model $\mathcal{E}$:

$$v_t^i = \mathcal{E}(p_t^i), \quad i = 0, 1, \ldots, M-1, \tag{2}$$

where $v_t^i \in \mathbb{R}^d$ is the embedding vector of the i-th prompt.

The similarity S_{ij} between two prompts is then measured using the inner product of their vector representations:

$$S_{ij} = \langle v_t^i, v_t^j \rangle = v_t^i \cdot v_t^j, \tag{3}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product.

The average similarity for each prompt p_t^i is computed as:

$$\bar{S}_i = \frac{1}{M-1} \sum_{\substack{j=0 \\ j \neq i}}^{M-1} S_{ij}. \tag{4}$$

Finally, the prompt $p_t^{i^*}$ with the highest average similarity is selected as the best candidate:

$$i^* = \arg\max_i \bar{S}_i. \tag{5}$$

The selected prompt $p_t^{i^*}$ is then used to generate the image, which is considered the best image. Simultaneously, its corresponding CoT is also identified as the best CoT. The CoT text and the generated image are then concatenated to form the ImageGen-CoT dataset.

References

[1] Yuang Peng, Yuxin Cui, Haomiao Tang, Zekun Qi, Runpei Dong, Jing Bai, Chunrui Han, Zheng Ge, Xiangyu Zhang, and Shu-Tao Xia. Dreambench++: A human-aligned benchmark for personalized image generation. *arXiv preprint arXiv:2406.16855*, 2024. 1

[2] Yuchen Zeng, Wonjun Kang, Yicong Chen, Hyung Il Koo, and Kangwook Lee. Can mllms perform text-to-image in-context learning? *arXiv preprint arXiv:2402.01293*, 2024. 1