OmniGen2: Towards Instruction-Aligned Multimodal Generation

Supplementary Material

8. More Qualitative Results

In this section, we present more qualitative results of OmniGen2. OmniGen2 demonstrates unified and versatile capabilities across a wide range of tasks, including text-to-image generation, image editing, in-context generation, and others. Moreover, it exhibits strong generalization abilities, allowing it to effectively tackle complex generation scenarios with remarkable consistency and fidelity.

8.1. Text to Image

Figure 1 showcases OmniGen2's robust capabilities in text-to-image (T2I) synthesis. The model demonstrates high fidelity across a wide spectrum of conceptual and thematic prompts, from fantasy scenes like a celestial staircase to photorealistic portraits and dynamic actions. OmniGen2 excels in rendering intricate details, such as the water droplets on the blue rose, and displays a sophisticated understanding of complex lighting and composition, as seen in the dramatic glow of the girl wielding lightning and the serene ambiance of the underwater scene. Crucially, these examples also highlight the model's native support for arbitrary aspect ratios, generating high-quality portrait, landscape, and square images without distortion. This combination of conceptual diversity, high fidelity, and flexible aspect ratio support validates OmniGen2 as a powerful and versatile T2I generator.

8.2. Image Editing

Figure 2 demonstrates OmniGen2's comprehensive suite of image editing capabilities, showcasing its ability to interpret a wide range of user instructions with high fidelity. The model adeptly handles localized object manipulations, including precisely adding (a hat), removing (a cat), replacing (a sword with a hammer), and extracting subjects from their backgrounds. Beyond object-level changes, OmniGen2 excels at nuanced semantic alterations, such as modifying facial expressions (adding a smile) and character motion (changing a pose to waving). Furthermore, the model is capable of executing complex, global modifications that affect the entire image, from changing backgrounds to performing complete stylistic transformations (converting a photograph into a 3D figurine). A key strength observed across all examples is the model's ability to preserve the identity of the subject and the integrity of unmodified regions, ensuring coherent and believable results.

8.3. In context generation

Figure 3 showcases OmniGen2's advanced capabilities in in-context generation and editing, a challenging task requiring the model to comprehend and manipulate subjects provided in reference images. The model adeptly performs compositional tasks, such as seamlessly integrating a subject into a new environment (`OBJECT + SCENE`, `PERSON + SCENE`) or combining multiple distinct subjects into a coherent new image (`ANIME + ANIME`). In these examples, OmniGen2 successfully preserves the high-fidelity identity of each subject while harmonizing them with the new context through appropriate adjustments in lighting, scale, and placement. Furthermore, the model handles complex in-context editing instructions. This includes replacing a subject within an existing scene (`OBJECT REPLACE`, `PERSON REPLACE`), where it not only swaps the main element but also intelligently adapts its appearance (e.g., color and accessories) to fit the new setting. These results demonstrate a sophisticated level of visual reasoning, where the model goes beyond simple generation to perform compositional and conditional editing based on multiple visual inputs.

8.4. Limitations

we also find several limitations of OmniGen2:

1) Performance Disparity Between English and Chinese Prompts. As shown in the first row of Figure 4, prompts in English generally yield better results than those in Chinese. For instance, when using a Chinese prompt, the generated image exhibits a minor inconsistency between input image and edited image.

2) Limited Generalization to Certain Instructions. The second row highlights OmniGen2's difficulty in modifying human body shapes, likely due to the scarcity of real-world data capturing such variations.

3) Sensitivity to Input Image Quality. As illustrated in Figure 4, the quality of the generated output is highly sensitive to the quality of the input image. When we input a low-quality image (generated by adding noise to the raw image), the resulting images exhibit significant degradation, with details becoming notably blurred. Furthermore, downsampling the input image to a maximum dimension of 256 pixels leads to further loss of clarity and detail, and the model's ability to accurately follow generation instructions is substantially reduced.

4) Ambiguity in Multi-Image Inputs. The third row of Figure 4 demonstrates that the model's performance improves when the prompt explicitly specifies the correspon-

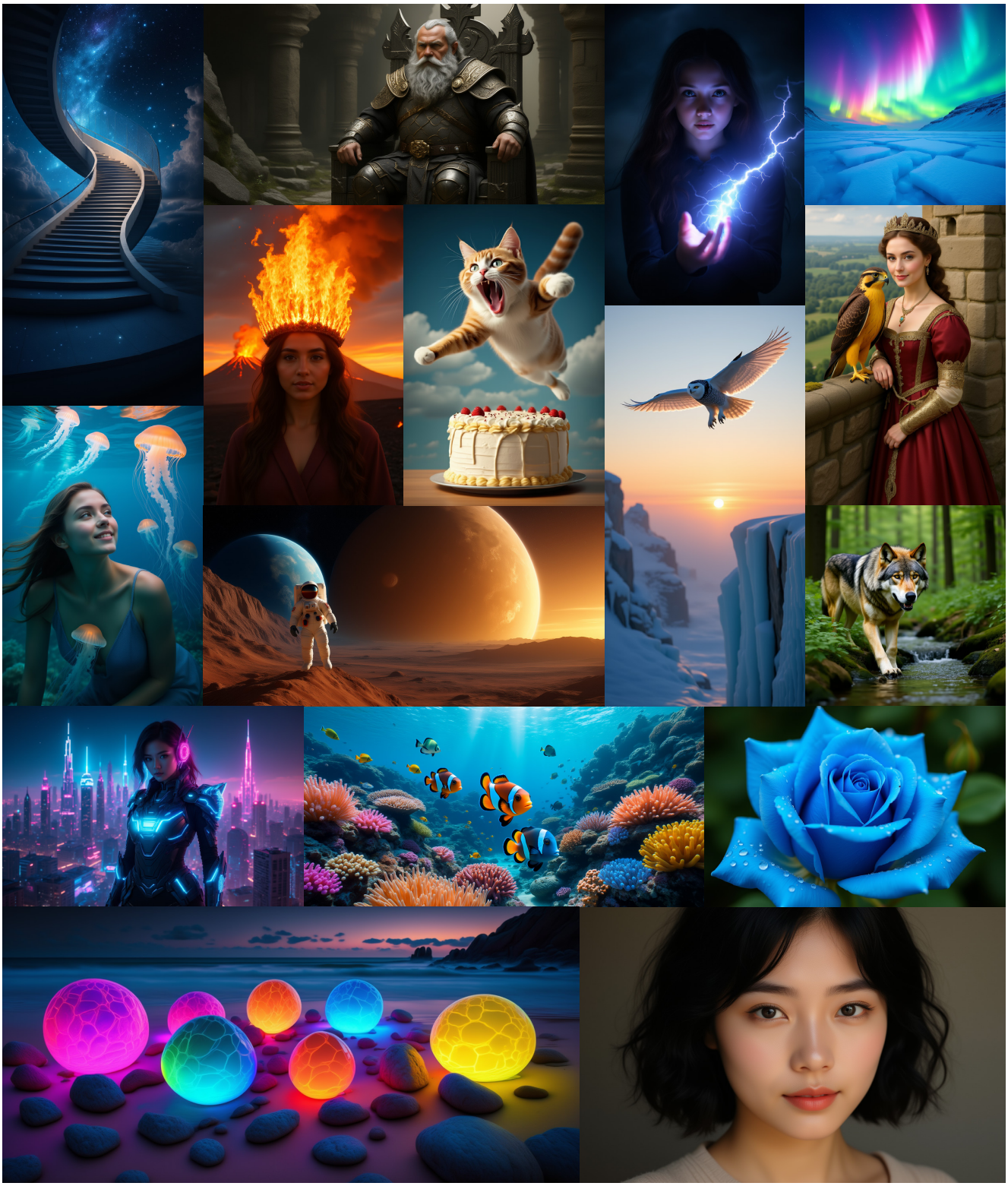

Figure 1. **Qualitative text-to-image generation by OmniGen2.** Examples showcasing the model's high fidelity to various text prompts and its support for diverse aspect ratios.

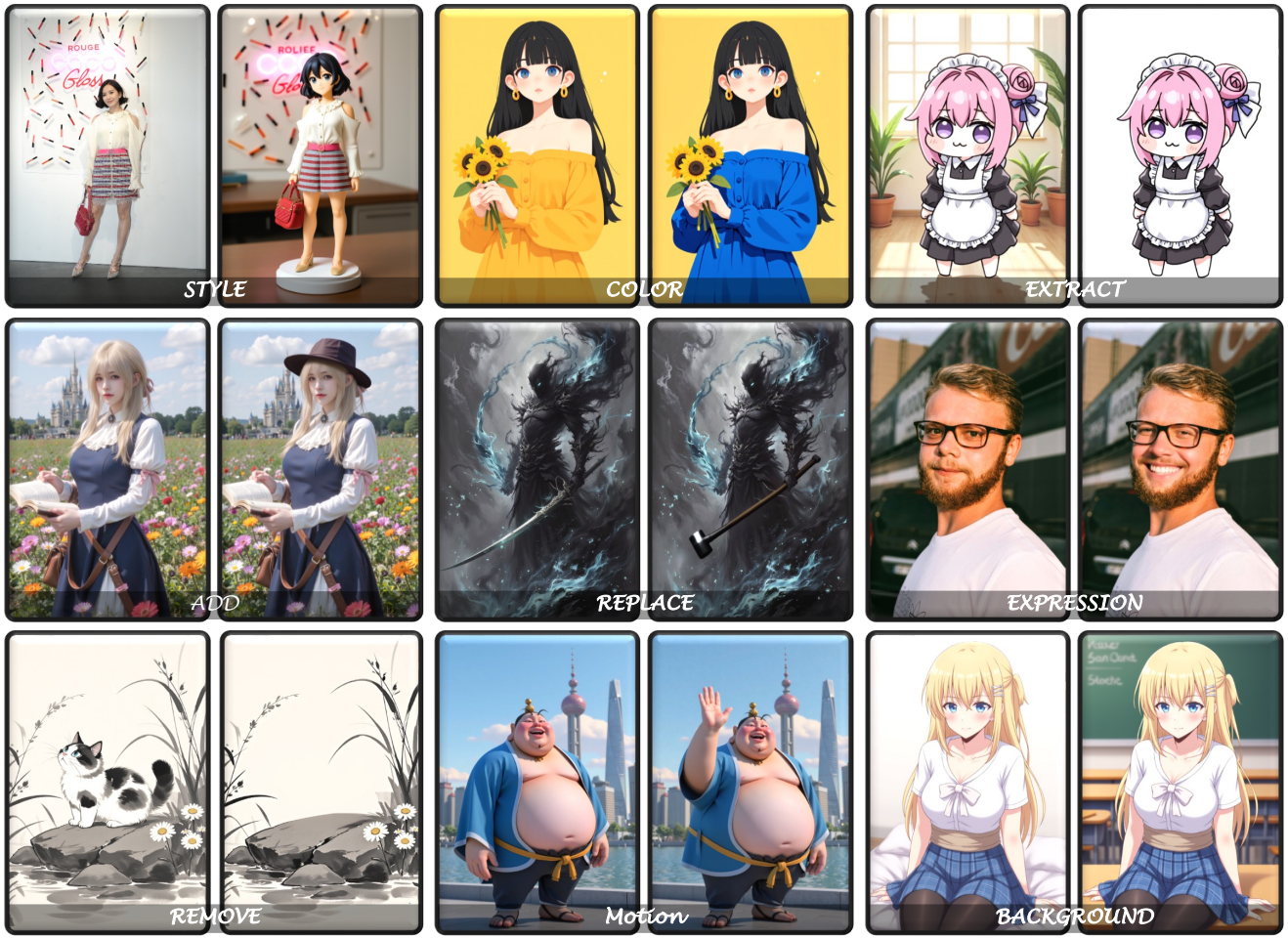

Figure 2. **Versatile image editing with OmniGen2.** The model skillfully handles a wide variety of instructions, from simple object modifications to complex motion change and stylistic alterations.

Figure 3. **Qualitative results of in-context generation and in-context edit.**

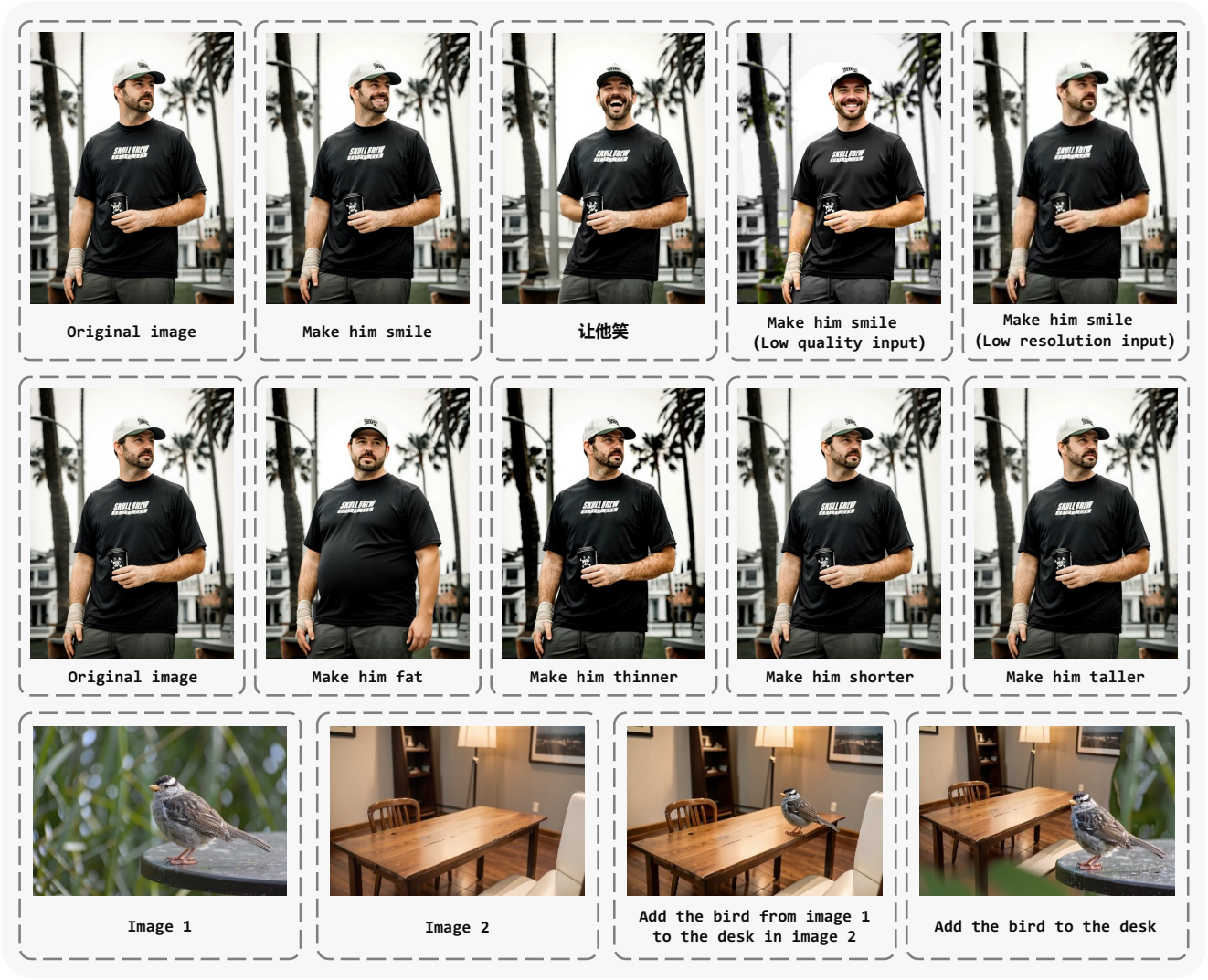

Figure 4. **Visualization of OmniGen2's Limitations. Line 1:** The model performs poorly when processing Chinese prompts and low-quality images. **Line 2:** The model often struggles to modify human body shapes accurately. **Line 3:** The model is sensitive to ambiguous instructions involving multiple image sources.

dence between objects and their source images (e.g., "the bird from image 1 and the desk from image 2"), indicating a sensitivity to ambiguous multi-source instructions.

5) In in-context generation tasks, the model occasionally fails to perfectly reproduce objects from the provided context. Increasing the guidance scale of image can partially alleviate this issue; however, it does not offer a complete solution. We hypothesize that significant improvements on such complex tasks may require further scaling of the model size.

9. Other Experimental Details

9.1. Toy Experiment Verification for Omni-RoPE

Figure 5 provides the full loss curves for the toy reconstruction experiment introduced in Section 3.3. The trends are consistent with the quantitative results in Table 1 of the main paper: both Omni-RoPE variants reduce the reconstruction loss much faster than the prior positional encoding designs. In particular, Qwen2-VL's RoPE shows a noticeable optimization plateau before converging, while Lumina-Image-2.0's design remains unstable for a substantially longer period and converges to a worse final solution.

The zoomed-in view further highlights the late-stage behavior. Omni-RoPE already achieves a low and stable loss floor, and adding the image index embedding mainly improves the final fidelity and variance in the last stage of training, yielding the best overall reconstruction quality. These results support our key design choice of disentangling image identity from local spatial coordinates: it preserves patch-wise correspondence across images while avoiding the optimization difficulty introduced by entangled

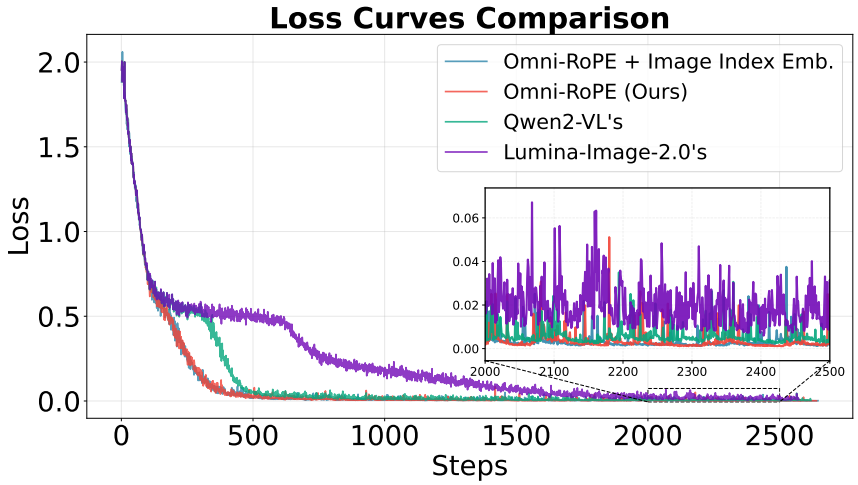

Figure 5. **Full loss curves for the Omni-RoPE toy reconstruction experiment.** Omni-RoPE converges substantially faster than prior positional encoding schemes. The inset shows late-stage optimization, where adding image index embeddings yields the lowest and most stable final loss.

global offsets.

9.2. Data Construction Pipeline

For multimodal understanding tasks, we utilize the dataset provided by LLaVA-OneVision [21]. For T2I generation, our training corpus comprises approximately 140 million open-source images sourced from Recap-DataComp [22], SAM-LLaVA [5], ShareGPT4V [7], LAION-Aesthetic [37], ALLaVA-4V [4], DOCCI [27], DenseFusion [23], JourneyDB [38], and BLIP3-o [6]. Furthermore, we incorporate 10 million proprietary images, for which we generate synthetic annotations using the Qwen2.5-VL-72B [2]. For image editing tasks, we collect publicly available datasets, including SEED-Data-Edit [13], UltraEdit [49], OmniEdit [41], PromptFix [48], and ImgEdit [47]. However, these open-source resources often suffer from suboptimal image quality, limited instruction accuracy, and insufficient task diversity. To overcome these constraints and better serve our research objectives, we have meticulously constructed a new comprehensive training dataset for this study. The subsequent sections provide a detailed account of our data construction pipeline.

9.3. In-Context Data

The in-context image generation task [20, 39, 43, 46] focuses on extracting a visual concept—such as a specific object, identity or individual—from input images and accurately reproducing it within newly generated images. This task, also known as subject-driven generation [36], parallels in-context learning in large language models: the image generation model produces personalized outputs in real time based solely on the provided context, without the need for additional fine-tuning. While in-context image generation has been extensively explored due to its broad range of applications, the community still faces a notable shortage of high-quality datasets tailored to this task.

9.3.1. In-Context Generation

In-context generation tasks require modeling the diverse appearances of an object across different scenarios. To address this, we leverage video data, which inherently capture the same subjects under varying conditions across frames. This temporal diversity enables the construction of training pairs in which subjects remain semantically consistent but exhibit differences in pose, viewpoint, and illumination. As illustrated in Figure 6, our data pipeline begins by extracting keyframes from each video and designating a base frame. Using Qwen2.5-VL-7B-Instruct [2], we identify the primary subjects within the base frame, capitalizing on the model's vision-language capabilities to focus on semantically salient entities while filtering out irrelevant background objects. The subject bounding boxes are then obtained via Grounding DINO [26], conditioned on the tags generated by the vision-language model. Subsequently, SAM2 [34] is employed to segment and track identified subjects in subsequent frames, with the last valid frame containing all subjects selected to maximize appearance variation. To mitigate tracking errors—such as the inclusion of visually similar but incorrect objects—we introduce a VLM-based filtering step to ensure subject consistency. To further enhance visual diversity, FLUX.1-Fill-dev[1] is used to outpaint the subject with a novel background in the input frame. We apply DINO [3]-based similarity filtering to discard samples where the subject's appearance deviates significantly, and Qwen2.5-VL-7B-Instruct is leveraged to assess both the semantic quality and consistency of the generated samples. Additionally, Qwen2.5-VL-7B-Instruct is

[1]https://huggingface.co/black-forest-labs/FLUX.1-Fill-dev

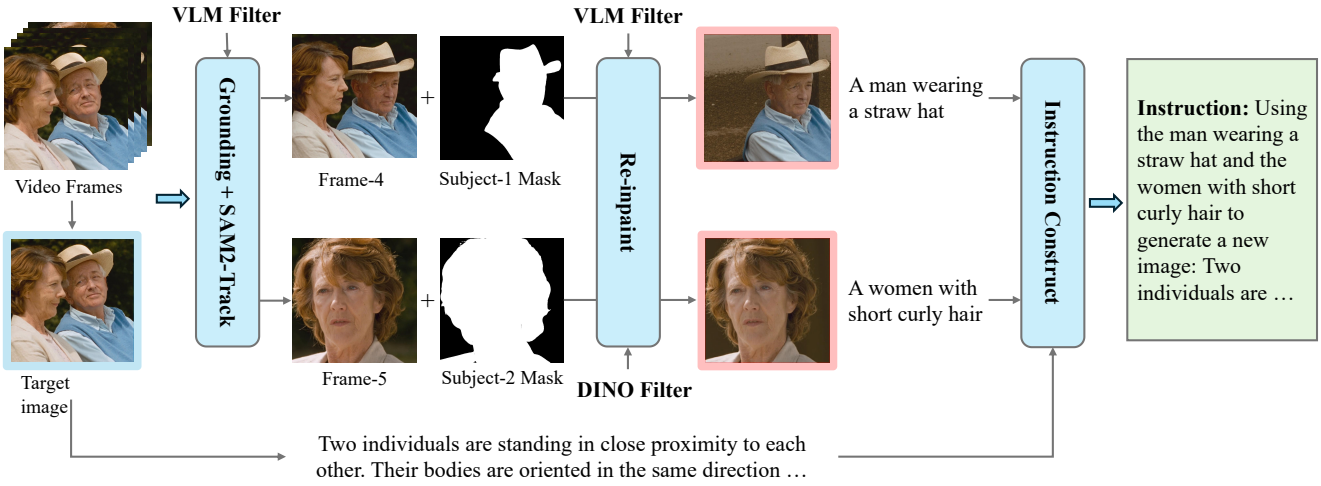

Figure 6. In-Context Generation Dataset Construction Pipeline. The final input images are outlined with a red border and the target image is marked by a blue boundary.

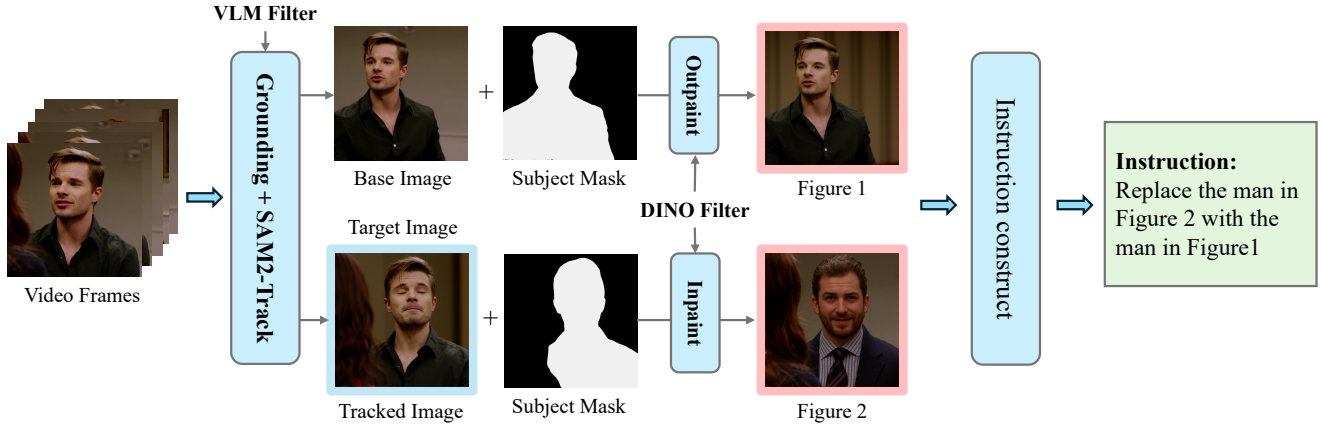

Figure 7. In-Context Editing Dataset Construction Pipeline. The final input and target images are outlined by red and blue consistent with Figure 6.

used to generate concise object descriptions and detailed captions for the base image, which are then integrated into natural language instructions. The final training triplet comprises the instruction, the repainted image as input, and the original image as output, providing semantically rich and visually diverse supervision for multi-subject generation tasks.

9.3.2. In-Context Edit

We further extend the in-context generation paradigm to editing tasks, introducing a new task termed in-context editing, as illustrated in Figure 7. Here, the model extracts relevant elements from a context image and utilizes them to edit a target input image.

The data source for in-context editing mirrors that of in-context generation: two frames containing the same object are selected, with one serving as the context clip and the other as the target clip. Initially, object masks for both frames are obtained using SAM2 [34]. For the context image, FLUX.1-Fill-dev is applied to generate a new background for the object via outpainting, encouraging the model to focus on object-specific features. Subsequently, FLUX.1-Fill-dev is used to inpaint the target clip, removing the object while preserving the original background to create the input clip. Finally, Qwen2.5-VL-72B-Instruct [2] generates a natural language description of the transformation from the input clip to the target clip, which is combined with the object description from the context clip to produce comprehensive natural language instructions.

9.4. Image Editing Data

9.4.1. Inpaint Data

Although most existing editing datasets are constructed through inpainting techniques, they suffer from two primary limitations: (1) substandard image quality, stemming from both inherent low resolution and post-processing degrada-

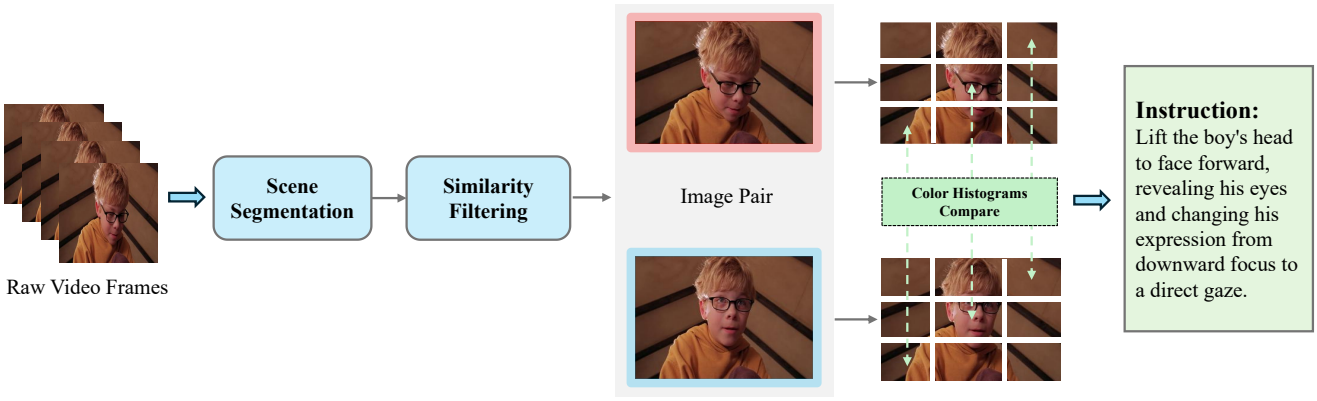

Figure 8. Create image edit pairs from videos. We first filter out frames belonging to different scenes to ensure contextual consistency, and then remove frames that exhibit significant changes in viewpoint.

tion during inpainting. (2) Editing instructions are inaccurate: previous work predefines editing instructions and uses inpainting models to generate images based on these instructions, but inpainting models have poor instruction-following capabilities, causing a mismatch between editing instructions and original-inpainted image pairs.

In this work, we select a small set of high-quality images from text-to-image data as our data source, applying FLUX.1-Fill-dev for inpainting. We use the inpainted images as inputs and the original images as targets to ensure high-quality target images. Additionally, we do not input instructions to the inpainting model, allowing it to fill content randomly. After obtaining image pairs, we employ a MLLM to write editing instructions based on these pairs. We find that the latest MLLM(e.g., Qwen2.5-VL) excels at writing editing instructions for original-inpainted image pairs, resulting in a high-accuracy editing dataset.

9.4.2. Video Data

Traditional inpainting methods are inherently limited in their capacity to construct diverse types of data, rendering them inadequate for tasks such as action modification, object movement, or expression changes. To address these limitations, we additionally extract editing pairs from video sources.

We show the pipeline in Figure 8. Image editing tasks typically require localized modifications while preserving the integrity of the surrounding context. To construct suitable image editing pairs from videos, it is essential to identify frame pairs that exhibit only local changes. We begin by segmenting videos into distinct scenes to avoid pairing frames across discontinuous contexts. Scene boundaries are detected by analyzing average RGB pixel intensities, while a rolling average of differences in the HSV color space enhances robustness to rapid motion. Within each identified scene, we extract multiple frame pairs and evaluate their differences using both DINOv2 [29] and CLIP [33]. Pairs exhibiting substantial differences—indicative of viewpoint changes—or negligible differences are filtered out.

Since camera viewpoints in videos often change even within a single scene, further refinement is necessary. Existing approaches, such as vision-language models, are computationally expensive and prone to inaccuracies, while methods based on color histograms or pixel-level similarity are either insensitive to spatial structure or overly susceptible to noise. To address these challenges, we divide each image into multiple blocks and compare the color histograms of corresponding blocks to assess their similarity, effectively reducing the impact of noise. The proportion of similar blocks is then computed to impose spatial constraints, serving as a reliable indicator of viewpoint consistency. This strategy efficiently filters out frame pairs with viewpoint changes while maintaining computational efficiency.

Finally, for each retained image pair with a consistent camera viewpoint, we employ Qwen2.5-VL-72B-Instruct [2] to generate precise editing instructions, thereby facilitating the construction of high-quality image editing datasets.

9.5. Interleave Data

9.5.1. Interleaved Frames

We initially segment videos based on detected scene transitions and extract key frames from each segment. Subsequently, we construct two types of video frame sequences, each comprising up to five frames: intra-scene interleaved sequence composed of frames within identical scene and inter-scene interleaved sequence composed of frames across different scenes. Following frame sequence extraction, we annotate each pair of consecutive frames with descriptive captions using an MLLM to describe changes in object actions and behaviors, variations in environment and background, and differences in object appearances. Given the substantial volume of required annotations, we employ

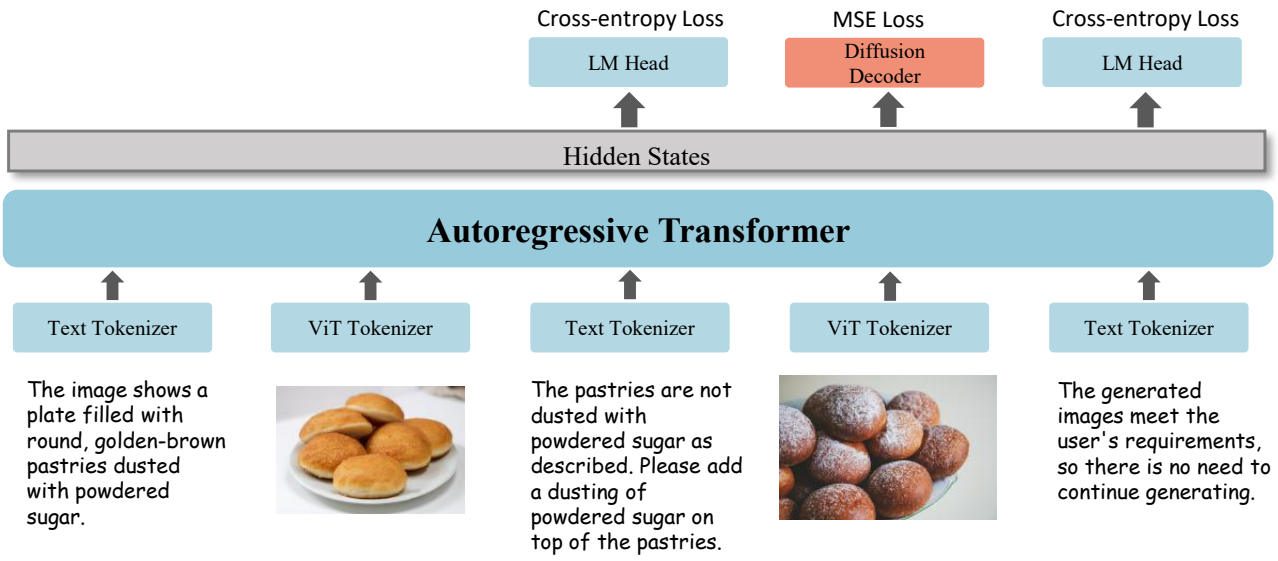

Figure 9. Multimodal Reflection for image generation.

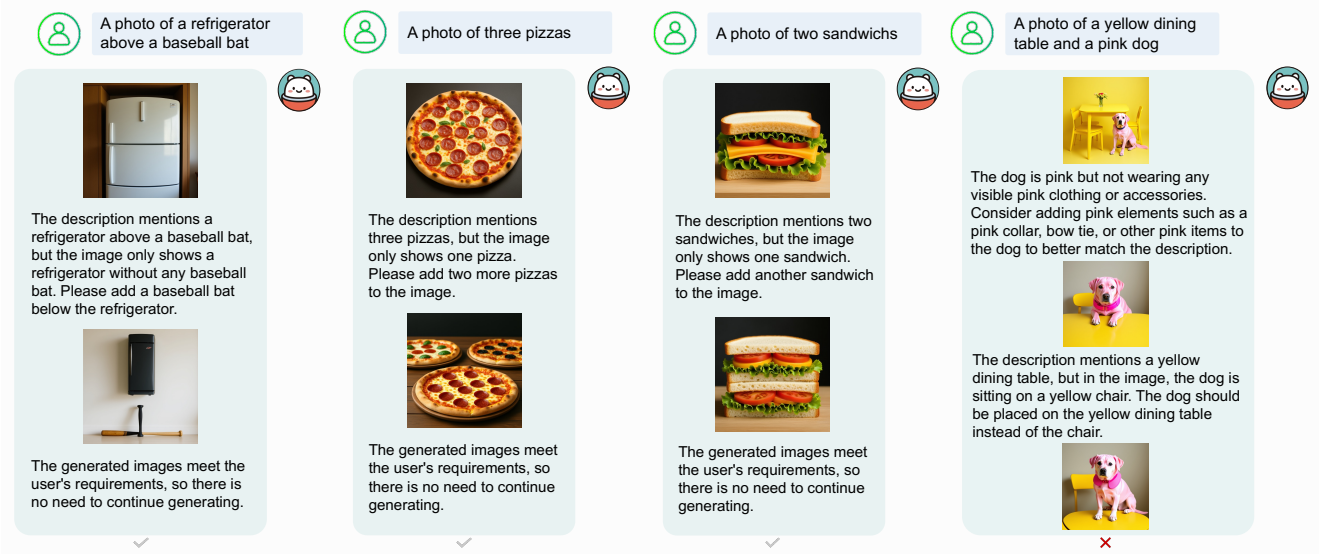

Figure 10. Example of generation with reflection using OmniGen2. **Left** and **middle:** Successful correction via one round of reflection. **Right:** an example of failed reflection, where the correct answer is incorrectly judged as wrong due to over-reflection.

Qwen2.5-VL-7B-Instruct for this process. Consequently, we obtain 0.8 million interleaved data samples from video sources, which serve to pretrain the model's capacity for processing continuous multimodal sequences.

9.5.2. Reflection Data

Inspired by previous advances in test-time scaling and self-reflection of large language models [16, 17, 24], we further explore the integration of reflection capabilities into multimodal generation models and demonstrate how test-time scaling can enhance the quality of image generation. In this section, we focus on describing the construction of the reflection data for subsequent model fine-tuning. The reflection data comprise an interleaved sequence of text and images, beginning with a user instruction, followed by the multimodal model's generated image, and step-by-step reflections on the previous generated outputs. Each reflection addresses two key aspects: 1) an analysis of the deficiencies or unmet requirements in relation to the original instruction, and 2) proposed solutions to address the previous image's limitation.

To construct self-reflection data, we select a small subset from the training data (in the current experiment, we only use data from the text-to-image task) and generate im-

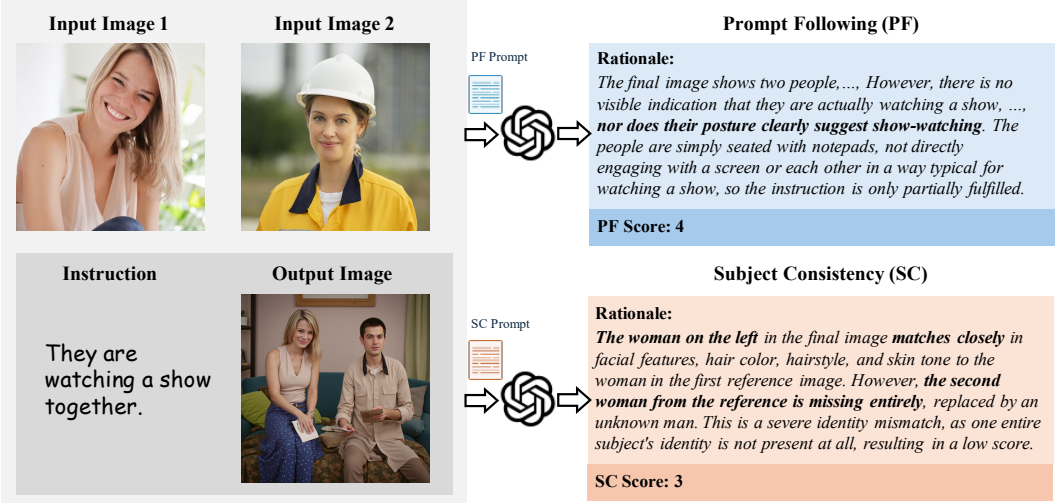

Figure 11. An illustrative example of evaluating the output image in the OmniContext benchmark.

ages through the model. Subsequently, we use an MLLM to assess whether the generated images meet the instruction requirements. If the images fail to adequately follow instructions or exhibit other quality issues, the model identifies specific errors and suggests modifications. Initially, we experimented with the DSG [9] evaluation framework to assess instruction-image alignment. However, this approach frequently led to hallucinations. Later, we discovered that powerful multimodal models could handle this task directly, so we employed Doubao-1.5-pro [12] to output issues and modification suggestions. After obtaining the first round of reflections, we append the generated images and corresponding reflections to the original instructions and fine-tune the model on these data. Once training is complete, we continue inferring data (using the first round of reflection data) to obtain a second round of images and corresponding reflective data. This iterative process yields multiple rounds of self-reflection data.

There is currently limited research on employing reflection mechanisms to enhance image generation tasks within multimodal generative models. We hope that our present work will contribute to advancing the development of reasoning capabilities in the field of multimodal generation. After the model acquires initial reflective capabilities through training with the current data, online reinforcement learning algorithms can further enhance these capabilities, which we leave for future exploration.

9.6. Reflection Fine-Tuning

We fine-tune OmniGen2 on reflection dataset following the illustrated in Figure 9. The model's enhanced reflection capabilities are demonstrated through the examples in Figure 10. In the successful cases, the model effectively reflects on the initial generated image, identifies its shortcomings, makes appropriate corrections and terminate the generation process at an appropriate point. However, it still faces challenges in reflection and correction. The model may over-reflect on simple instructions, generating unnecessary requirements or fails to revise the image. These issues arise from the limited perception of the 3B-scale MLLM and insufficient reflection data. In future work, we plan to scale up the model and employ reinforcement learning to improve reflection quality.

9.7. Training pipeline details

Detailed configurations for each stage are summarized in Table 1.

Stage	Resolution	Task Type	Steps
1. Pre-training	256×256	T2I-only	50k
		Mixed-Task	50k
	512×512	T2I-only	30k
		Mixed-Task	30k
	1024×1024	T2I-only	50k
		Mixed-Task	50k
2. SFT	1024×1024	Mixed-Task	100k
3. RL Alignment	512×512	Mixed-Task	2.4k

Table 1. **Details of the OmniGen2 staged training pipeline.** The curriculum progresses from general pre-training to general instruction alignment, with increasing task complexity and resolution.

9.8. OmniContext Details

The detailed metrics for each subtask are presented in Tables 2, 3 and 4. The pipeline to evaluate models on OmniContext as shown in Figure 11.

Method	SINGLE↑								
	Character			Object			Average		
	PF	SC	Overall	PF	SC	Overall	PF	SC	Overall
Flux.1 Kontext max [20]	7.98	**9.24**	8.48	8.78	**8.76**	8.68	8.38	**9.00**	8.58
Gemini-2.0-flash [15]	5.54	5.98	5.06	6.17	5.89	5.17	5.86	5.93	5.11
GPT-4o [28]	**8.89**	9.03	**8.90**	**9.40**	8.74	**9.01**	**9.14**	8.88	**8.95**
InfiniteYou [18]	7.81	5.15	6.05	-	-	-	-	-	-
UNO [43]	7.56	6.48	6.60	7.78	6.65	6.83	7.67	6.56	6.72
BAGEL [10]	7.72	4.86	5.48	8.56	6.06	7.03	8.14	5.46	6.25
OmniGen [44]	7.12	7.58	7.21	7.66	5.04	5.71	7.39	6.31	6.46
OmniGen2	**7.92**	**8.68**	**8.19**	**8.98**	**8.44**	**8.63**	**8.45**	**8.56**	**8.41**

Table 2. Comparison on task type SINGLE from OmniContext. Prompt Following (PF), Subject Consistency (SC), and Overall scores are reported (higher is better, ↑).

Method	MULTIPLE↑											
	Character			Object			Char. + Obj.			Average		
	PF	SC	Overall	PF	SC	Overall	PF	SC	Overall	PF	SC	Overall
Gemini-2.0-flash [15]	3.65	3.02	2.91	2.50	5.02	2.16	4.26	5.80	3.80	3.47	4.62	2.96
GPT-4o [28]	**9.17**	**9.03**	**9.07**	**9.06**	**8.90**	**8.95**	**8.34**	**8.89**	**8.54**	**8.86**	**8.94**	**8.86**
UNO [43]	3.88	2.38	2.54	7.46	5.86	6.51	5.10	4.10	4.39	5.48	4.11	4.48
BAGEL [10]	6.14	4.86	5.17	7.54	6.10	6.64	6.74	6.28	6.24	6.81	5.75	6.02
OmniGen [44]	5.92	6.18	5.65	5.60	5.46	5.44	4.64	4.96	4.68	5.39	5.53	5.26
OmniGen2	**7.30**	**8.10**	**7.45**	**7.98**	**7.74**	**7.80**	**7.60**	**8.34**	**7.93**	**7.63**	**8.06**	**7.73**

Table 3. Comparison on task type MULTIPLE from OmniContext. Prompt Following (PF), Subject Consistency (SC), and Overall scores are reported (higher is better, ↑).

Method	SCENE↑											
	Character			Object			Char. + Obj.			Average		
	PF	SC	Overall	PF	SC	Overall	PF	SC	Overall	PF	SC	Overall
Gemini-2.0-flash [15]	3.76	3.33	3.02	4.02	5.22	3.89	2.89	4.63	2.92	3.56	4.39	3.28
GPT-4o [28]	**9.05**	**8.88**	**8.90**	**8.33**	**8.62**	**8.44**	**8.71**	**8.57**	**8.60**	**8.70**	**8.69**	**8.65**
UNO [43]	2.74	2.50	2.06	5.62	3.52	4.33	5.22	3.86	4.37	4.53	3.29	3.59
BAGEL [10]	4.56	3.94	4.07	6.12	5.50	5.71	5.90	5.30	5.47	5.53	4.91	5.08
OmniGen [44]	4.14	3.42	3.59	5.24	3.72	4.32	5.56	4.84	5.12	4.98	3.99	4.34
OmniGen2	**8.02**	**7.64**	**7.75**	**8.10**	**7.80**	**7.91**	**8.08**	**7.88**	**7.93**	**8.07**	**7.77**	**7.86**

Table 4. Comparison on task type SCENE from OmniContext. Prompt Following (PF), Subject Consistency (SC), and Overall scores are reported (higher is better, ↑).

9.9. RL Generalization to Out-of-Distribution Benchmarks

To further evaluate the generalization ability of our multi-stage RL curriculum, we conduct experiments on out-of-distribution (OOD) benchmarks that are not directly aligned with our training rewards.

Specifically, we evaluate on Emu-Edit and OneIG-Bench, which assess editing fidelity and general image generation quality under diverse and challenging conditions. These benchmarks differ from our training objectives and thus provide a reliable measure of transferability.

As shown in Table 5, our full curriculum (Edit → GenEval → IC) consistently outperforms the base model and alternative training orders across all OOD metrics. This demonstrates that our alignment strategy does not overfit to specific reward signals, but instead learns generalizable capabilities that transfer across tasks.

Strategy	Emu-Edit			OneIG
	CLIP-I↑	CLIP-Out↑	DINO↑	Align.↑
Base	0.877	0.309	0.823	0.7870
GenEval+Edit	0.909	0.311	0.894	0.8160
GenEval+Edit+IC	0.886	**0.312**	0.858	0.8212
Edit+IC+GenEval	0.868	0.310	0.826	0.8242
Edit+GenEval+IC	**0.896**	0.311	**0.876**	**0.8289**

Table 5. RL curriculum ablation on out-of-distribution (OOD) benchmarks. Base denotes the model without RL.

Method	Single object↑	Two object↑	Counting↑	Colors↑	Position↑	Color attribution↑	Overall↑
SDv2.1 [35]	0.98	0.51	0.44	0.85	0.07	0.17	0.50
SDXL [31]	0.98	0.74	0.39	0.85	0.15	0.23	0.55
IF-XL	0.97	0.74	0.66	0.81	0.13	0.35	0.61
LUMINA-Next [50]	0.92	0.46	0.48	0.70	0.09	0.13	0.46
SD3-medium [1]	0.99	0.94	0.72	0.89	0.33	0.60	0.74
FLUX.1-dev [19]	0.99	0.81	0.79	0.74	0.20	0.47	0.67
NOVA [11]	0.99	0.91	0.62	0.85	0.33	0.56	0.71
OmniGen [44]	0.98	0.84	0.66	0.74	0.40	0.43	0.68
TokenFlow-XL [32]	0.95	0.60	0.41	0.81	0.16	0.24	0.55
Janus [42]	0.97	0.68	0.30	0.84	0.46	0.42	0.61
Janus Pro [8]	0.99	0.89	0.59	0.90	0.79	0.66	0.80
Emu3-Gen† [40]	0.99	0.81	0.42	0.80	0.49	0.45	0.66
Show-o [45]	0.98	0.80	0.66	0.84	0.31	0.50	0.68
MetaQuery-XL† [30]	-	-	-	-	-	-	0.80
BLIP3-o† 4B [6]	-	-	-	-	-	-	0.81
BLIP3-o† 8B [6]	-	-	-	-	-	-	0.84
BAGEL [10]	0.99	0.94	0.81	0.88	0.64	0.63	0.82
BAGEL† [10]	0.98	0.95	0.84	0.95	0.78	0.77	0.88
UniWorld-V1 [25]	0.99	0.93	0.79	0.89	0.49	0.70	0.80
UniWorld-V1† [25]	0.98	0.93	0.81	0.89	0.74	0.71	0.84
OmniGen2	0.99	1	0.93	0.91	1	0.86	**0.95**

Table 6. Evaluation of text-to-image generation ability on GenEval [14] benchmark. † refers to the methods using LLM rewriter.

9.10. GenEval Results

As shown in the Table 6, OmniGen2 excels at generating images from complex, compositional prompts. Our model achieves an impressive overall score of 0.95. This result surpasses other powerful unified models like UniWorld-V1 (0.84) and BAGEL (0.88). It is crucial to note that this SOTA performance is achieved with exceptional efficiency. OmniGen2 utilizes only 4B trainable parameters and was trained on 15M T2I pairs and 50k prompts used in RL.

References

[1] Stability AI. Sd3-medium. https://stability.ai/news/stable-diffusion-3-medium, 2024. 11

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 5, 6, 7

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 5

[4] Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. Allava: Harnessing gpt4v-synthesized data for a lite vision-language model, 2024. 5

[5] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 5

[6] Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, et al. Blip3-o: A family of fully open unified multimodal models-architecture, training and dataset. *arXiv preprint arXiv:2505.09568*, 2025. 5, 11

[7] Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023. 5

[8] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025. 11

[9] Jaemin Cho, Yushi Hu, Jason Baldridge, Roopal Garg, Peter Anderson, Ranjay Krishna, Mohit Bansal, Jordi Pont-Tuset, and Su Wang. Davidsonian scene graph: Improving reliability in fine-grained evaluation for text-to-image generation. In *ICLR*, 2024. 9

[10] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025. 10, 11

[11] Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. *arXiv preprint arXiv:2412.14169*, 2024. 11

[12] Doubao. Doubao-1.5-pro. https://seed.bytedance.com/zh/special/doubao_1_5_pro, 2025. 9

[13] Yuying Ge, Sijie Zhao, Chen Li, Yixiao Ge, and Ying Shan. Seed-data-edit technical report: A hybrid dataset for instructional image editing. *arXiv preprint arXiv:2405.04007*, 2024. 5

[14] 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. 11

[15] Google. Gemini 2.0 flash. https://developers.googleblog.com/en/experiment-with-gemini-20-flash-native-image-generation, 2025. 10

[16] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025. 8

[17] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024. 8

[18] Liming Jiang, Qing Yan, Yumin Jia, Zichuan Liu, Hao Kang, and Xin Lu. Infiniteyou: Flexible photo recrafting while preserving your identity. *arXiv preprint arXiv:2503.16418*, 2025. 10

[19] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 11

[20] Black Forest Labs. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space. 2025. 5, 10

[21] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 5

[22] Xianhang Li, Haoqin Tu, Mude Hui, Zeyu Wang, Bingchen Zhao, Junfei Xiao, Sucheng Ren, Jieru Mei, Qing Liu, Huangjie Zheng, Yuyin Zhou, and Cihang Xie. What if we recaption billions of web images with llama-3? *arXiv preprint arXiv:2406.08478*, 2024. 5

[23] Xiaotong Li, Fan Zhang, Haiwen Diao, Yueze Wang, Xinlong Wang, and Ling-Yu Duan. Densefusion-1m: Merging vision experts for comprehensive multimodal perception. *2407.08303*, 2024. 5

[24] 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. 8

[25] Bin Lin, Zongjian Li, Xinhua Cheng, Yuwei Niu, Yang Ye, Xianyi He, Shenghai Yuan, Wangbo Yu, Shaodong Wang, Yunyang Ge, et al. Uniworld: High-resolution semantic encoders for unified visual understanding and generation. *arXiv preprint arXiv:2506.03147*, 2025. 11

[26] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 5

[27] Yasumasa Onoe, Sunayana Rane, Zachary Berger, Yonatan Bitton, Jaemin Cho, Roopal Garg, Alexander Ku, Zarana Parekh, Jordi Pont-Tuset, Garrett Tanzer, et al. Docci: Descriptions of connected and contrasting images. *arXiv preprint arXiv:2404.19753*, 2024. 5

[28] OpenAI. Gpt-4o. https://openai.com/index/introducing-4o-image-generation, 2025. 10

[29] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 7

[30] Xichen Pan, Satya Narayan Shukla, Aashu Singh, Zhuokai Zhao, Shlok Kumar Mishra, Jialiang Wang, Zhiyang Xu, Jiuhai Chen, Kunpeng Li, Felix Juefei-Xu, et al. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025. 11

[31] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 11

[32] Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 2545–2555, 2025. 11

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 7

[34] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 5, 6

[35] 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. 11

[36] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 5

[37] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 5

[38] Keqiang Sun, Junting Pan, Yuying Ge, Hao Li, Haodong Duan, Xiaoshi Wu, Renrui Zhang, Aojun Zhou, Zipeng Qin, Yi Wang, et al. Journeydb: A benchmark for generative image understanding. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[39] Zhenxiong Tan, Songhua Liu, Xingyi Yang, Qiaochu Xue, and Xinchao Wang. Ominicontrol: Minimal and uni-

versal control for diffusion transformer. *arXiv preprint arXiv:2411.15098*, 2024. 5

[40] 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. 11

[41] Cong Wei, Zheyang Xiong, Weiming Ren, Xeron Du, Ge Zhang, and Wenhu Chen. Omniedit: Building image editing generalist models through specialist supervision. In *The Thirteenth International Conference on Learning Representations*, 2024. 5

[42] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12966–12977, 2025. 11

[43] Shaojin Wu, Mengqi Huang, Wenxu Wu, Yufeng Cheng, Fei Ding, and Qian He. Less-to-more generalization: Unlocking more controllability by in-context generation. *arXiv preprint arXiv:2504.02160*, 2025. 5, 10

[44] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. Omnigen: Unified image generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13294–13304, 2025. 10, 11

[45] 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. 11

[46] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 5

[47] Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. *arXiv preprint arXiv:2505.20275*, 2025. 5

[48] Yongsheng Yu, Ziyun Zeng, Hang Hua, Jianlong Fu, and Jiebo Luo. Promptfix: You prompt and we fix the photo. *arXiv preprint arXiv:2405.16785*, 2024. 5

[49] Haozhe Zhao, Xiaojian Shawn Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. Ultraedit: Instruction-based fine-grained image editing at scale. *Advances in Neural Information Processing Systems*, 37:3058–3093, 2024. 5

[50] Le Zhuo, Ruoyi Du, Han Xiao, Yangguang Li, Dongyang Liu, Rongjie Huang, Wenze Liu, Xiangyang Zhu, Fu-Yun Wang, Zhanyu Ma, et al. Lumina-next: Making lumina-t2x stronger and faster with next-dit. *Advances in Neural Information Processing Systems*, 37:131278–131315, 2024. 11