

DreamOmni3: Scribble-based Editing and Generation

Bin Xia, Bohao Peng, Jiyang Liu, Sitong Wu, Jingyao Li,
Junjia Huang, Xu Zhao, Yitong Wang, Ruihang Chu, Bei Yu, Jiaya Jia, *IEEE Fellow*

Abstract—Recently unified generation and editing models have achieved remarkable success with their impressive performance. These models rely mainly on text prompts for instruction-based editing and generation, but language often fails to capture users’ intended edit locations and fine-grained visual details. To this end, we propose two tasks: scribble-based editing and generation, which enables more flexible creation on a graphical user interface (GUI) combining user textual, images, and freehand scribbles. We introduce DreamOmni3, tackling two challenges: data creation and framework design. Our data synthesis pipeline includes two parts: scribble-based editing and generation. For scribble-based editing, we define four tasks: scribble and instruction-based editing, scribble and multimodal instruction-based editing, image fusion, and doodle editing. Based on DreamOmni2’s dataset, we extract editable regions and overlay hand-drawn boxes, circles, doodles, or cropped images to construct training data. For scribble-based generation, we define three tasks: scribble and instruction-based generation, scribble and multimodal instruction-based generation, and doodle generation, following similar data creation pipelines. For the framework, instead of using binary masks, which struggle with complex edits involving multiple scribbles, images, and text instructions, we propose a joint input scheme that feeds both the original and scribbled source images into the model, using different colors to distinguish regions and simplify processing. By applying the same index and position encodings to both images, the model can precisely localize scribbled regions while maintaining accurate editing. Furthermore, we introduce joint training with the VLM and our generation/editing model to better understand the user’s abstract scribbles. Finally, we establish comprehensive DreamOmni3 benchmarks for these tasks to promote further research. Extensive experimental results show that DreamOmni3 achieves outstanding performance.

Index Terms—Image Editing, Image Generation

I. INTRODUCTION

The recent success of unified generation and editing models [1]–[3] can be attributed to three key factors: (1) Unified training for generation and editing not only improves performance on existing tasks through mutual enhancement but also gives rise to a wide range of new editing and generation capabilities. (2) It significantly lowers the user barrier, enabling all functions within a single model without the need to choose specialized ones. (3) They exhibit strong

multimodal understanding, responding to real-world visuals, marking progress toward world models and AGI.

The current unified generation and editing models focus on generation and editing based on images and text as instructions. However, this approach often falls short of meeting users’ interactive needs. For example, certain editing or generation positions in the image are difficult to describe with language, some objects may be hard for users to identify by name, or there may be multiple identical objects that are hard to distinguish. In these cases, manual annotation is necessary. Furthermore, users may wish to make more flexible and creative additions, deletions, or modifications to the content in the image, such as through drawing. In these scenarios, it’s essential for the model to not only understand images and language but also understand the editing and generation intentions behind the user’s manual scribbles.

To create a more intelligent and comprehensive unified creation tool, we present DreamOmni3, a model that integrates our proposed scribble-based editing and generation with the existing unified editing and generation framework. First, we define the tasks of scribble-based editing and generation to guide our data creation. Specifically, as shown in Fig. 2 (a), we categorize scribble-based editing into four types: scribble and instruction-based editing, scribble and multimodal instruction-based editing, image fusion, and doodle editing. Additionally, as shown in Fig. 2 (b), we categorize scribble-based generation into three types: scribble and instruction-based generation, scribble and multimodal instruction-based generation, and doodle generation.

To train DreamOmni3, the main challenge is the lack of training data. To address this, we introduce a comprehensive data synthesis pipeline for scribble-based editing and generation. **Scribble-based editing data creation:** (1) For multimodal instruction-based editing, we use the Referseg service [4] on the DreamOmni2 dataset to obtain the coordinates and size of the edited objects in both the source and reference images, marking the edited positions with hand-drawn circles or boxes. (2) For instruction-based editing, we use the same data as in step (1) but without the reference image. (3) For image fusion, we extract the edited objects from the reference image and paste them onto the corresponding position on the source image. (4) For doodle editing, we crop the edited objects from the target image, generate sketches, and place them back into the source image. **Scribble-based generation data creation:** (1) For multimodal instruction-based generation, we use Referseg to locate the edited objects in the image and mark the circles or boxes on the blank

Bin Xia, Bohao Peng, Sitong Wu, Jingyao Li, Ruihang Chu, and Bei Yu are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong SAR, China.

Jiyang Liu, Junjia Huang, Xu Zhao, and Yitong Wang are with ByteDance Inc., Shenzhen, China.

Jiaya Jia is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong SAR, China.



Fig. 1: The gallery of DreamOmni3, which has scribble-based editing and generation capabilities. The gallery of DreamOmni3, which has scribble-based editing and generation capabilities.

canvas. (2) For instruction-based generation, we remove the reference image from the data created in step (1). (3) For doodle generation, we extract the edited objects from the target image, generate sketches, and place them on a blank canvas at the same position.

For the framework, we chose to input the source image with scribbles rather than using a binary mask for two reasons: (1) It avoids the complexity and computational cost of binary masks when editing multiple areas. Scribbles, distinguished by color, are easier to describe with instructions. (2) It is compatible

with existing unified editing and generation models, which require RGB inputs, not binary masks.

Based on the above analysis, we propose the DreamOmni3 framework. Specifically, we propose a joint input scheme for editing by feeding both the original source image and the scribbled source image. This allows us to retain both the scribbled information and the pixels covered by the scribbles. To better align the source image with its scribbled version, both share the same index and position encodings. For the reference image, to distinguish it from the source images

and avoid pixel confusion, we adopt the same position shift and index encoding scheme as in DreamOmni2 (Fig. 2 (c)). Additionally, we propose a joint training scheme for the generation/editing model and VLM. Since user scribbles are often diverse and VLM possesses strong reasoning capabilities, we train it on our scribbles dataset to recognize and provide contextual explanations of the user’s scribbles. This aids the generation/editing model in better understanding the user’s intent and producing more accurate results. Furthermore, for these tasks, we construct a DreamOmni3 benchmark using real-world image data, enabling a more accurate evaluation of the model’s generalization and real-world performance. Our main contributions are fourfold:

- We introduce two highly useful tasks for unified generation and editing models: scribble-based editing and scribble-based generation. These tasks can be combined with text and image instructions, enhancing the creative usability of unified models and providing clear definitions for targeted optimization and future research.
- We propose a data synthesis pipeline to create a high-quality, comprehensive training dataset for scribble-based editing and generation.
- We present DreamOmni3, a framework that supports text, image, and scribble inputs with complex logic. It takes both the source text, image and scribble as input, maintaining editing consistency while accurately interpreting the scribble’s intent. We design position and index encoding schemes to differentiate between the scribble and source image, ensuring compatibility with existing unified generation and editing architectures. Additionally, we propose joint training of the generation/editing model and VLM to enhance scribble understanding.
- For these tasks, we build the DreamOmni3 benchmark using real-world image data. Experiments demonstrate the effectiveness of DreamOmni3 in real scenarios.

II. RELATED WORK

Mask-based Editing refers to editing operations performed on selected regions or items of an image. It can generally be divided into three categories: (1) Image inpainting [5]–[9] is a common technique where users paint over the area they wish to modify. The model then regenerates that region based on the instruction while preserving the rest of the image. (2) Since traditional inpainting completely regenerates the masked region and cannot preserve its original structure or color, MagicQuill [10] introduces auxiliary inputs such as edge maps or low-resolution images to help maintain contour and color consistency. (3) Some approaches [11]–[13] inject compressed object IDs from a reference image into the masked region to insert specific objects. Compared to these methods, DreamOmni3 offers several advantages: (a) Traditional inpainting datasets are created by simply masking parts of an image, which prevents the model from reasoning about environmental effects such as lighting or shadow changes (the third row of Fig. 5). Our dataset, built through instruction-based editing, allows the model to respond more coherently to contextual changes. (b) Inpainting often suffers from inaccurate masks — regions may be over- or under-edited. Even

MagicQuill, though guided by edge maps, still requires precise manual masking. In contrast, our instruction-based approach is far more robust: users can roughly circle a region, and the model accurately understands both the intent and the editing scope, providing a much more user-friendly experience. (c) The unified generation and editing model is the way forward. However, previous methods relied on inconsistent input formats, making them hard to unify under a single framework. DreamOmni3 establishes a unified architecture that simplifies use and integration. (d) Benefiting from this unified design, DreamOmni3 not only extends existing editing capabilities but also introduces new ones. For example, in image fusion, prior methods [12] could only merge a single reference image into a masked area, whereas our model supports multimodal inputs (multiple images, text, scribbles, and more) for richer and more precise edits. Moreover, DreamOmni3 bridges the gap between image generation and editing, enabling interactive experiences like direct sketching on tablets.

Mask-based Generation incorporates masks into the image generation process to control content within specific regions. It can be broadly categorized into three types. For mask-guided text-to-image generation, Multi-diffusion [14] first explored a training-free approach for multi-region generation on text-to-image models, but the results were unstable. GLIGEN [15] improved this by introducing an attention module and encoding bounding boxes through Fourier embeddings. Later works [16]–[18] further optimized attention mechanisms and refined the granularity of input information. For mask-guided subject-driven generation, MS-diffusion [19] employs a grounding resampler to associate visual information with specific entities and spatial constraints. Nevertheless, existing mask-based generation frameworks are often overly complex in both input format and inference process. In contrast, DreamOmni3 allows users to paint a mask and combine it with image and language instructions to generate more complex outputs. Its unified design aligns with existing frameworks, bridging mask-based generation and image editing within a single system, greatly enhancing usability.

III. METHODOLOGY

A. Synthetic Data

The biggest challenge in scribble-based editing and generation is the lack of data. We need to construct a dataset that incorporates language, images, and scribbles as instructions, and develop the ability to combine these three instruction types for complex editing tasks, enabling smarter editing tools. We found that DreamOmni2 effectively unified language and image instructions and introduced the multimodal instruction editing and generation task, along with corresponding training data. Therefore, we directly use DreamOmni2’s multimodal instruction editing and generation data as the base, and further extend it to create a dataset that includes scribbles as instructions.

As shown in Fig. 2 (a), we divide scribble-based editing into four tasks: (1) Scribble and multimodal instruction-based editing: We use Referseg [4] service to locate the editing objects in both the reference and target images. Since users

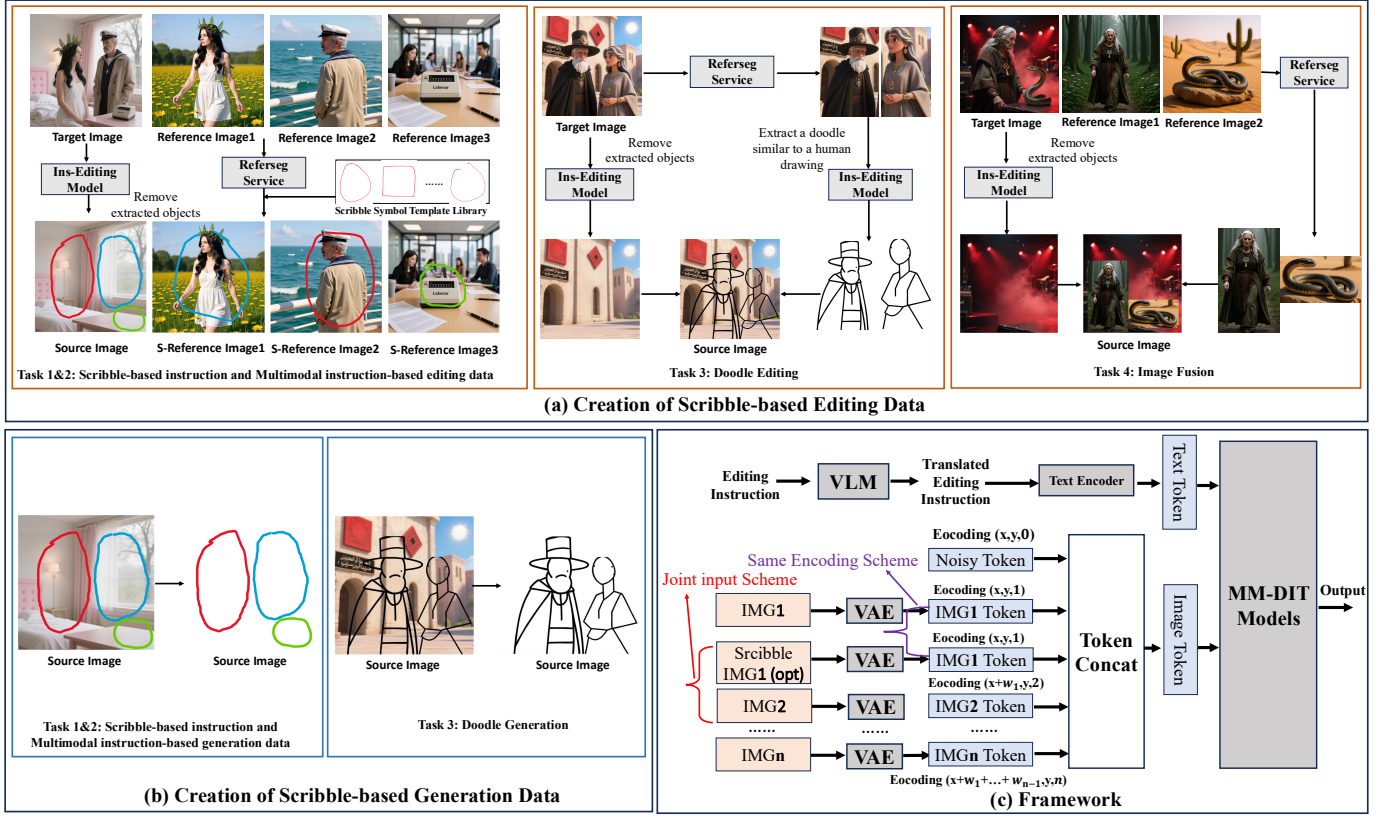


Fig. 2: The overview of DreamOmni3’s training data construction and framework. The overview of DreamOmni3’s training data construction and framework: **(a)** We create scribble-based editing training data. For scribble and multimodal instruction-based editing, we use Referseg [4] to locate edit objects and paste corresponding scribbles onto the source and reference images to create training pairs. For scribble and instruction-based editing, we omit the reference image. For doodle editing, we use a dedicated model to convert the edit objects into abstract sketches and paste them back into the source image. For image fusion, we crop objects from the reference image and paste them into the corresponding position on the source image to build training pairs. **(b)** Scribble-based generation training data is created similarly to editing, except the source image is replaced with a blank white canvas. **(c)** DreamOmni3 builds on the framework of DreamOmni2 [20], introducing a joint input scheme for scribble inputs. We also apply the same encoding scheme to both the source and scribbled images, ensuring better pixel alignment and perfect compatibility with previous image and language instruction editing.

often draw imperfect shapes, we manually created 30 different squares and circles for a scribble symbol template library. After determining the object’s position in the reference image, we randomly select and resize a symbol from the library, then paste it to create a scribble-based reference image (S-Reference Image). For the source image, we modify the target image by either removing or altering the object, then add a corresponding scribble symbol, resulting in the scribble-based source image (S-source image). This process generates target images based on either the reference image (or S-reference image) and source image. The reference image can include or exclude scribbles depending on the instruction. Our editing types encompass a wide range of both concrete objects and abstract attributes, as demonstrated in DreamOmni2 [20]. **(2)** Scribble-based instruction editing: This task requires no special data preparation. We simply remove the reference image from (1) and add an object description to the instruction. **(3)** Doodle editing: We use Referseg to locate the editing object and the instruction-based editing model to turn it into a

simple abstract doodle. We avoid using Canny edge detection due to users’ imperfect drawings, which require aesthetic corrections from the model. Instead, we use the GPT-Image-1 model, which is better suited for generating doodles with non-strict pixel consistency. **(4)** Image fusion: In this task, users can extract objects from one image and insert them into the target image. We first remove the target object’s corresponding section using the instruction-based editing model. Next, we use Referseg to locate and crop the object from the reference image, then paste it into the target image after resizing it to fit. This results in the source image.

As shown in Fig. 2 (b), we divide scribble-based generation into three tasks: **(1)** For scribble-based multimodal instruction generation, the method is similar to that for scribble-based multimodal instruction editing, with the key difference being the placement of scribble symbols on a white canvas to create the source image. This enables the model to refer to attributions or objects of the reference image and generate them at corresponding symbol positions. **(2)** For scribble-based

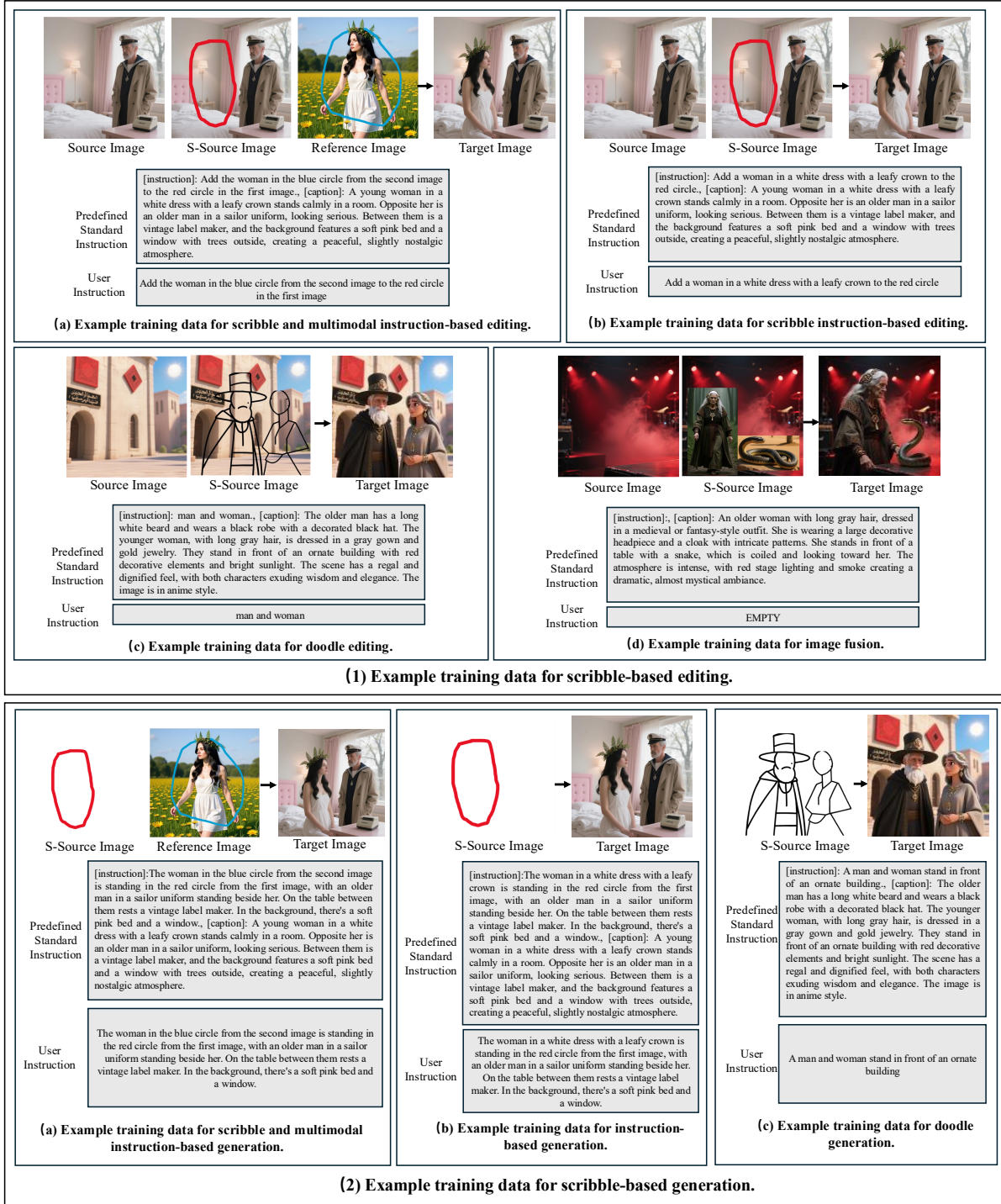


Fig. 3: Example training data for scribble-based editing and generation.

instruction generation, we remove the reference image from step (1) and add descriptions of the removed objects in the instructions. (3) For doodle generation, the method is similar to doodle editing, with the final sketch placed on a white canvas to allow the model to generate the corresponding objects and background based on the sketch and instructions.

Our data is created based on DreamOmni2's multi-reference image generation and editing training datasets. In our created dataset, the scribble-based editing dataset contains several types of data. The scribble-based multimodal instruction

editing includes 32K training samples, while scribble-based instruction editing has about 14K training samples. The image fusion dataset consists of 16K training samples, and the doodle editing dataset contains 8K training samples. Notably, both scribble-based multimodal instruction editing and scribble-based instruction editing cover a wide range of editing categories. These include edits related to abstract properties, such as design style, color scheme, and hairstyle, as well as edits of concrete objects, such as various objects, people, and animals, with the ability to add, remove, or modify them. In contrast,

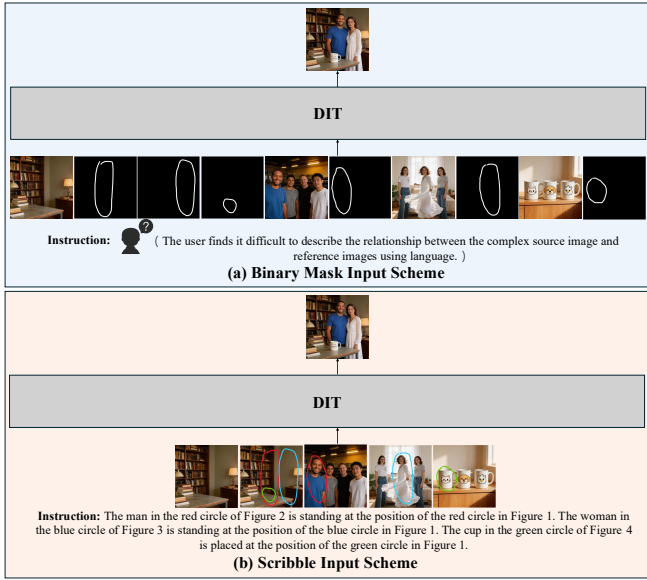


Fig. 4: Comparison of binary mask input scheme and scribble input scheme.

image fusion and doodle editing focus on the task of adding concrete objects into the image.

In the scribble-based generation dataset, scribble-based multimodal instruction generation has 29K training samples, scribble-based instruction generation has 10K training samples, and doodle generation has 8K training samples. Both scribble-based multimodal instruction generation and scribble-based instruction generation involve generating concrete objects as well as referencing abstract attributions. However, doodle generation is mainly focused on generating concrete objects.

B. Framework and Training

The current unified generation and editing models primarily focus on instruction-based editing and subject-driven generation. Recently, DreamOmni2 extended this model to multi-reference generation and editing. However, the input format for doodle instructions still requires exploration. In DreamOmni3, we considered two input schemes: one using binary masks, similar to inpainting, and another using a joint input of the source image and the source image with doodles. Since doodles inevitably alter parts of the source image, but editing requires consistency in the non-edited areas, we need to input doodle information while preserving the source image details. As shown in Fig. 4, there are two possible approaches: the traditional inpainting method with binary masks and our proposed joint input of the source image and the doodle-modified source image. Our approach offers two several advantages over binary masks in inpainting: **(1)** In Fig. 4, compared to Binary Mask, Scribble input allows users to easily map complex logical relationships between the reference image and the source image using the prompt, based on the color of their brush. In other words, it enables users to clearly describe task requirements in natural language. On the other hand, Binary Mask is limited to black and white, making it difficult to

describe complex tasks. **(2)** In Fig. 4, with Scribble input, users can paint on a single image and distinguish different scribbles by color. In contrast, Binary Mask requires a separate binary mask layer for each scribble, which significantly increases the computational load when dealing with multiple scribbles. **(3)** Scribble input integrates better with existing unified generation and editing models. These models are trained on RGB natural images and have powerful image understanding capabilities. Since Scribble input operates on natural images, it can leverage these pre-trained capabilities effectively, whereas Binary Mask, being limited to black and white, cannot fully utilize the base model’s strengths.

Additionally, we found that user scribbles are often messy, requiring the model to have strong reasoning abilities to interpret their intent. To address this, we introduce a VLM with enhanced reasoning capabilities. However, since existing VLMs are not specifically trained to predict the use of scribbles, they need to be trained on our scribble dataset to accurately infer user intent and output a predefined annotation format. To this end, we propose joint training of the VLM and generation/editing models, allowing the VLM to interpret complex text, image, and scribble instructions and improve the generation/editing model’s understanding of user intent.

Our framework design is shown in Fig. 2 (c), building upon the DreamOmni2 framework with additional adaptation for scribble instruction input. Our joint input scheme is optional. Specifically, when the source image in the editing task contains scribbles, we input both the source image and the scribbled source image into the MM-DIT models. However, if the reference images contain scribbles, we do not use the joint input scheme, as there is no need to maintain pixel-level consistency in the non-edited areas of the reference image, and adding an extra input would unnecessarily increase computational cost. For scribble-based generation tasks, we also avoid using the joint input scheme since there is no need for pixel-level preservation. Using the joint input scheme does introduce two challenges: **(1)** it adds an extra image, which affects the indexing of subsequent input images and could potentially confuse users, and **(2)** the model must correctly map pixel relationships between the source image and the doodle-modified source image. To address these, we use the same index encoding and position encoding scheme for both the source image and the scribbled source image. Experiments show that this encoding effectively resolves these issues, seamlessly integrating doodle editing capabilities into the existing unified generation and editing model framework.

During training, we fine-tune Qwen2.5-VL [21] 7B to learn the predefined standard output format, with a learning rate of 1×10^{-5} , using approximately 10 A100 hours. For more details, please refer to the supplementary. We then train the editing and generation models using LoRA with a rank of 256. Notably, by leveraging LoRA for training, we retain the original instruction-editing capabilities of Kontext. When a user inputs an image with scribbles into the model, our LoRA is activated, seamlessly integrating multimodal instruction-based editing and generation into the unified model. Additionally, since the multi-reference generation and editing capabilities in the previous DreamOmni2 model were trained separately with

TABLE I: Quantitative comparison of scribble-based editing and generation. We use Gemini [27] and Doubao [28] to evaluate the success editing ratio of different models on scribble-based editing and generation, respectively. In addition, “Human” refers to professional engineers assessing the editing success rates of all models.

Method	Scribble-based Editing			Scribble-based Generation		
	Gemini↑	Doubao↑	Human↑	Gemini↑	Doubao↑	Human↑
GPT-4o [2]	0.6176	0.5441	0.5956	0.6023	0.5114	0.4318
Nano Banana [1]	0.4779	0.3897	0.4118	0.4318	0.3977	0.2727
Omnigen2 [29]	0.0735	0.0588	0.0294	0.0909	0.0795	0.0341
Kontext [26]	0.0515	0.1029	0.0147	0.0682	0.0227	0.0909
Qwen-image-Edit-2509 [30]	0.1912	0.1691	0.1397	0.1364	0.0568	0.0795
DreamOmni2 [20]	0.1765	0.2059	0.1471	0.1477	0.0795	0.0568
DreamOmni3 (Ours)	0.5294	0.4632	0.5588	0.5341	0.4773	0.5455

two LoRAs, our generation and editing models are also trained with separate LoRAs to ensure compatibility. The training process took approximately 400 A100 hours.

C. Benchmark and Evaluation

We propose scribble-based editing and generation that integrates language, images, and scribble instructions. Currently, there is no dedicated benchmark to evaluate these tasks. To foster the development of this valuable direction, we introduce the DreamOmni3 benchmark. Our benchmark is comprehensive, consisting of real images to accurately assess model performance in real-world scenarios. The test cases cover a wide range of editing and generation tasks, including the four editing tasks and three generation tasks proposed in our paper. The editing categories are also diverse, encompassing both abstract property edits and concrete object edits. More details about the DreamOmni3 benchmark can be found in the supplementary materials.

Traditional metrics, such as DINO [22] and CLIP [23], are not sufficient to accurately evaluate the complex and diverse instruction-based editing and generation tasks we propose [24]. Recent works like Step1x-Edit [25] and Kontext [26] rely on VLMs and human evaluations for instruction editing. Given the increased complexity of our Scribble-based Editing and Generation tasks, evaluation must be based on VLMs. We propose a comprehensive set of evaluation standards for these new tasks, focusing on three key aspects: (1) accuracy in following the instructions in the generated edits, (2) consistency in human appearance, objects, and abstract attributes, (3) avoidance of severe visual artifacts, and (4) alignment of the generated or edited content with the specified doodle regions. Only when all these criteria are met do we consider the editing or generation task successful, ensuring relevance to real-world use cases. We apply the same standards for human evaluations, and our results show that VLM-based assessments align closely with human evaluations. The system prompt used for VLM evaluation can be found in the supplementary materials.

TABLE II: Comparison of consistency under CLIP and DINO.

Method	CLIP-I↑	DINO↑
GPT-4o [2]	0.8368	0.7585
Nano Banana [1]	0.8669	0.8186
Omnigen2 [29]	0.8611	0.8029
Kontext [26]	0.8278	0.7443
Qwen-image-Edit-2509 [30]	0.8565	0.8155
DreamOmni2 [20]	0.8076	0.7026
DreamOmni3 (Ours)	0.8702	0.8172

IV. EXPERIMENTS

Evaluation on Scribble-based Editing. As shown in Tab. I, we compared several competitive unified generation and editing models, such as Omnigen2 [29], Qwen-image-Edit-2509 [30], and DreamOmni2 [20]. In addition, although Kontext does not natively support multi-image input, we applied a method from Diffusers [31] that combines multiple images into one for input. We also compared closed-source commercial models, Nano Banana [1] and GPT-4o [2]. We tested the performance of all models on the DreamOmni3 benchmark, which we constructed using real images for doodle editing, and evaluated the model outputs’ pass rates using Gemini 2.5 [27] and Doubao 1.6 [28]. Additionally, we had several people manually evaluate the results from different models and derive the pass rates. The evaluation standards are outlined in Sec. III-C. For manual assessment, each case is evaluated by 5 reviewers, and a case is deemed successful if it receives approval from more than 3 evaluators. As can be seen, DreamOmni3 achieved the best results in human evaluations. For large language model evaluations, DreamOmni3 also outperformed open-source models and showed results comparable to commercial models. Notably, GPT-4o frequently exhibits a yellowing issue in the images, and the pixels in the non-edited areas of the output often do not match those in the input image. Meanwhile, Nano Banana showed several issues, such as copy-and-paste effects and incorrect object proportions. These issues

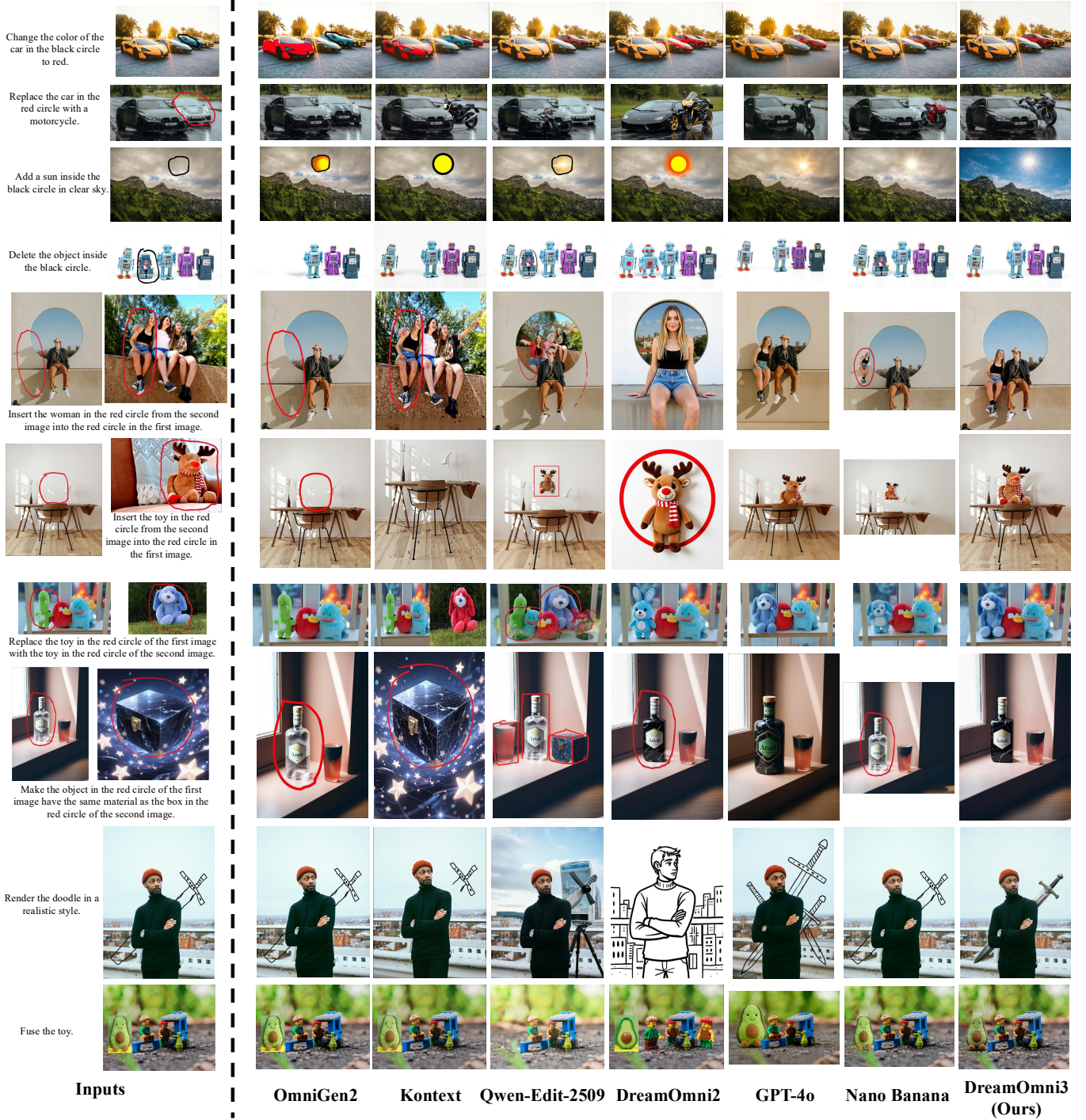


Fig. 5: Visual comparison of scribble-based editing. Compared to other competitive methods and even closed-source commercial models (GPT-4o and Nano Banana), DreamOmni3 shows more accurate editing results and better consistency.

are difficult for Vision Language Models (VLMs) to detect accurately.

Moreover, as shown in Tab. II, we compared traditional CLIP-I and DINO scores between the source image and the edited results for scribble-based editing, evaluating consistency. Our DreamOmni3 outperforms open-source models and is close to closed-source models. However, these metrics only reflect basic consistency and cannot measure editing success, as the highest score would occur by simply outputting the original image.

The qualitative results are shown in Fig. 5. We present a range of editing outcomes, including scribble-based instruction edits in the first four rows, scribble-based multimodal instruction edits in rows 5 to 8, and a comparison of doodle editing results in row 9, followed by image fusion results in row 10. As observed, DreamOmni3 generates more precise edits with enhanced consistency.

Evaluation on Scribble-based Generation. As shown in Tab. I, our method outperforms Nano Banana in both human evaluations and assessments by Doubao and Gemini,

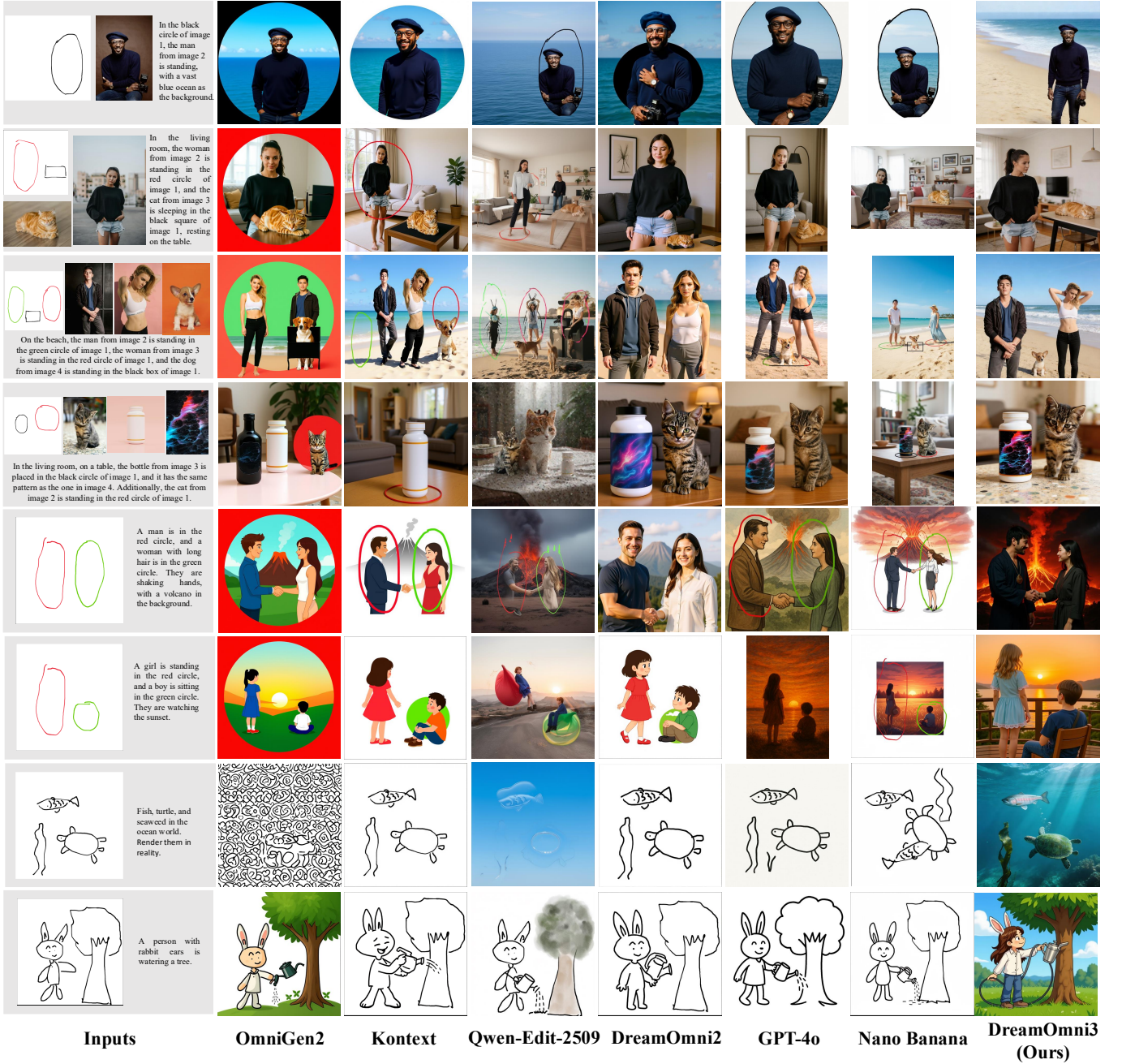


Fig. 6: Visual comparison of scribble-based generation. Our DreamOmni3 significantly outperforms current open-source models and achieves generation results comparable to closed-source commercial models (GPT-4o and Nano Banana).

with performance similar to GPT-4o. In fact, we found that GPT-4o and Nano Banana are not specifically optimized for scribble-based scenarios, and even when explicitly instructed not to generate scribble marks, these models still often output scribbles along with the generated results. The evaluation standards are outlined in Sec. III-C, and further details can be found in the supplementary. Compared to open-source models, the current state-of-the-art (SOTA) models, DreamOmni2 and Qwen-image-edit-2509, also do not focus on or optimize for these new tasks, resulting in lower success rates on scribble-related tasks. This further demonstrates the effectiveness and necessity of our DreamOmni3 for scribble-based generation.

Quantitative results are shown in Fig. 6. We found that open-source models often retain scribbles in their outputs, which is related to the lack of task-specific optimization. Commercial models, Nano Banana and GPT-4o, perform relatively better, but Nano Banana exhibits some unnatural collage-like artifacts.

Alignment of VLM-based and human assessments.

As shown in Tab. III. It indicates that VLM-based assessments can align with human judgments when criteria are clearly defined (Supp Sec. 4). Since the evaluation outcomes are binary, Pearson's r is not an appropriate metric.

Joint Input and Joint Training. As shown in Tab. IV, we

TABLE III: VLM-judge validation.

Tasks	Scribble-based Editing				Scribble-based Generation		
	Doodle Editing	Scribble and Ins-based Editing	Scribble and Multimodal Ins-based Editing	Image Fusion	Doodle Generation	Scribble and Ins-based Generation	Scribble and Multimodal Ins-based Generation
Cohen’s κ (Gemini) $\uparrow$	0.4792	0.7872	0.6257	0.4595	0.5385	0.6479	0.5970
Cohen’s κ (Doubao) $\uparrow$	0.5192	0.7902	0.5714	0.4521	0.5385	0.7235	0.5844

TABLE IV: Joint input scheme and joint training scheme for scribble-based generation and editing.

Method	Generation or Editing Model Training	Joint Input	Joint Training	Editing	Generation
Scheme 1	$\times$	$\times$	$\times$	0.0882	0.0341
Scheme 2	$\checkmark$	$\times$	$\times$	0.2868	0.3409
Scheme 3	$\times$	$\checkmark$	$\times$	0.0956	0.0341
Scheme 4	$\times$	$\times$	$\checkmark$	0.0882	0.0455
Scheme 5	$\times$	$\checkmark$	$\checkmark$	0.1176	0.0455
Scheme 6	$\checkmark$	$\times$	$\checkmark$	0.3897	0.4432
Scheme 7	$\checkmark$	$\checkmark$	$\times$	0.3456	0.3636
Scheme 8 (Ours)	$\checkmark$	$\checkmark$	$\checkmark$	0.4632	0.4773

compare the impact of training with our custom scribbled dataset versus using joint inputs of source images and scribbled source images, along with the joint training of VLM, on model performance. In Scheme 1, we use the base model, Kontext, which only takes the scribbled source image as input. In Scheme 2, we train the model on our dataset, but input only the scribbled source image, without using the trained VLM to assist in understanding the scribble. In Scheme 3, we combine the source image and the scribbled source image as joint input, but without training on our dataset or using the trained VLM. In Scheme 4, we use the base model Kontext (not trained on our dataset) and input only the scribbled source image, but with a trained VLM to assist in understanding. Following the same pattern, in Scheme 8, we use joint input, train the generation/editing models on our dataset, and leverage the trained VLM to assist in understanding the scribbles.

Based on experiments, we can draw three key conclusions. (1) Comparing Scheme 2 with Scheme 1, we observe that our dataset significantly improves the model’s scribble-based editing and generation capabilities. (2) Comparing Scheme 6 with Scheme 8, joint input significantly boosts editing performance, while the improvement in generation tasks is less pronounced. This is because scribbles can obscure edited regions of the source image, so joint input ensures the model can see the original image information, improving editing consistency. In contrast, generation tasks don’t require high pixel-level consistency, making the improvement less significant. Based on these results, we only use joint input for editing tasks when the source image contains scribbles. For other reference images or generation tasks with scribbles, we input the scribbled image directly. (3) Comparing Scheme 7 and Scheme 8, we find that the trained VLM effectively helps the generation/editing models understand the user’s scribble intent, resulting in more accurate outputs.

Index and Position Encoding. As shown in Tab. V, we compare different encoding schemes for the source image and the scribbled source image as joint inputs. The results show that using the same index encoding and position encoding for both inputs yields the best performance. We believe this is due to two main reasons: (1) Matching the position and index

TABLE V: Different encoding schemes for scribble inputs.

Method	Same Index Encoding	Same Position Encoding	Editing	Generation
Scheme 1	$\times$	$\times$	0.3824	0.3068
Scheme 2	$\times$	$\checkmark$	0.4118	0.4318
Scheme 3	$\checkmark$	$\times$	0.4338	0.3636
Scheme 4 (Ours)	$\checkmark$	$\checkmark$	0.4632	0.4773

TABLE VI: Fine-Grained DreamOmni3 Evaluation on Scribble-Based Editing.

Scribble-based Editing Tasks	Consistency of Non-Edited Regions	Instruction Accuracy	Visual Quality	Output Scribbles	Consistency of Edited Elements	Overall
Doodle Editing	0.7500	0.6000	0.5500	0.8500	0.7500	0.4500
Scribble and Instruction-based Editing	0.7083	0.5625	0.9167	0.7292	0.8333	0.5208
Scribble and Multimodal Instruction-based Editing	0.6042	0.4792	0.8125	0.6458	0.4167	0.3333
Image Fusion	1.0000	0.8500	0.7500	/	0.9500	0.6500

encodings allows for better pixel-level alignment between the two images, enabling the model to more accurately locate the scribbles and preserve pixel-level information, leading to more consistent and precise editing. (2) Using the same encodings means that subsequent reference images do not need to modify their encodings, maintaining consistency with the original training format and allowing the model to better leverage its pre-trained editing capabilities for more accurate results.

Fine-Grained Evaluation on Scribble-Based Editing. As shown in Tab. VI, we report DreamOmni3’s performance across five evaluation metrics on four scribble-based editing tasks. These evaluation metrics include Consistency of Non-Edited Regions measures the proportion of cases where non-edited areas remain unchanged between the source image and result; Instruction Accuracy, which evaluates how well text and scribble instructions are followed; Visual Quality, which assesses the naturalness of the result; Output Scribble, which measures the proportion of results without scribbles; and Consistency of Edited Elements, which evaluates the appearance consistency of edited elements. For more details, please refer to the supplementary materials. We can draw five conclusions: (1) Among the four tasks, Scribble and Multimodal Instruction-based Editing has the lowest overall success rate, indicating that it is the most challenging and requires more training data to improve model performance. (2) For Doodle Editing, the main bottleneck lies in the naturalness of the generated results. (3) For Scribble and Instruction-based Editing, the key issue is unresponsiveness to instructions. (4) In Scribble and Multimodal Instruction-based Editing, failure mainly occurs due to the inconsistent appearance of reference objects or humans. (5) For Image Fusion, the main issue is the poor fusion of the collage image, leading to unnatural visual quality.

Fine-Grained Evaluation on Scribble-Based Generation. As shown in Tab. VII, we report DreamOmni3’s performance across four evaluation metrics on three scribble-based generation tasks. These metrics are: Consistency, which measures the ratio of cases where the result aligns with the reference image or doodle; Instruction Accuracy, which evaluates how well the result follows the text instructions; Visual Quality, which assesses the naturalness of the generated result; and

TABLE VII: Fine-Grained DreamOmni3 Evaluation on Scribble-Based Generation.

Scribble-based Generation Tasks	Consistency	Instruction Accuracy	Visual Quality	Output Scribbles	Overall
Doodle Generation	0.7778	0.7222	0.6111	0.7222	0.5556
Scribble and Instruction-based Generation	/	0.6333	0.8667	0.7667	0.5667
Scribble and Multimodal Instruction-based Generation	0.4750	0.5500	0.7750	0.6500	0.3750

Output Scribble, which measures the proportion of results without scribbles. We can draw four conclusions: (1) Scribble and Multimodal Instruction-based Generation has the lowest overall success rate among the three tasks, making it the most challenging. (2) For Doodle Generation, the main difficulty is improving the naturalness and aesthetics of the output. (3) For Scribble and Instruction-based Generation, the key challenge is unresponsiveness to instructions. (4) For Scribble and Multimodal Instruction-based Generation, the main issue is maintaining consistency in the appearance of people and objects in the reference image.

V. CONCLUSION

The current unified generation and editing model performs image edits based on text instructions. However, language struggles to accurately describe edit locations and capture all the user’s intended details. To enhance this, we propose two tasks: scribble-based editing and generation, allowing users to simply use a brush in the GUI to make edits. This method can combine language, image, and scribble instructions, offering more flexibility. Building on this, we introduce DreamOmni3, addressing the challenge of limited training data. Using DreamOmni2 data, we developed a data creation scheme based on Referseg to generate high-quality, accurate datasets that integrate scribbles, text, and image instructions. We also tackled the model framework issue, as binary masks are inadequate for complex real-world needs. When multiple masks are present, they are hard to distinguish and describe with language. To solve this, we propose a scribble-based approach, where different masks are easily differentiated by brush color, allowing for the handling of any number of masks. Since scribbles may obscure some image details, we introduce a joint input scheme, feeding both the original and scribbled images into the model. We further optimized this by using the same index and position encoding to preserve the details while maintaining accurate correspondence to reference images. We have also introduced joint training of VLM and generation/editing models to improve the model’s understanding of user text, images, and irregular scribbles, enabling more accurate results. In addition, we developed a DreamOmni3 benchmark and evaluation framework that combines scribbles, text, and image input instructions, fostering the advancement of this new field.

REFERENCES

- [1] Google, “Nano banana,” <https://aistudio.google.com/models/gemini-2-5-flash-image>, 2025. 1, 7
- [2] OpenAI, “Gpt-4o image generation,” <https://openai.com/index/introducing-4o-image-generation/>, 2025. 1, 7
- [3] C. Deng, D. Zhu, K. Li, C. Gou, F. Li, Z. Wang, S. Zhong, W. Yu, X. Nie, Z. Song *et al.*, “Emerging properties in unified multimodal pretraining,” *arXiv preprint arXiv:2505.14683*, 2025. 1
- [4] X. Lai, Z. Tian, Y. Chen, Y. Li, Y. Yuan, S. Liu, and J. Jia, “Lisa: Reasoning segmentation via large language model,” in *CVPR*, 2024. 1, 3, 4
- [5] B. Xia, Y. Zhang, S. Wang, Y. Wang, X. Wu, Y. Tian, W. Yang, and L. Van Gool, “Diffir: Efficient diffusion model for image restoration,” in *ICCV*, 2023. 3
- [6] S. Xie, Z. Zhang, Z. Lin, T. Hinz, and K. Zhang, “Smartbrush: Text and shape guided object inpainting with diffusion model,” in *CVPR*, 2023. 3
- [7] B. Yang, S. Gu, B. Zhang, T. Zhang, X. Chen, X. Sun, D. Chen, and F. Wen, “Paint by example: Exemplar-based image editing with diffusion models,” in *CVPR*, 2023. 3
- [8] B. Xia, Y. Zhang, J. Li, C. Wang, Y. Wang, X. Wu, B. Yu, and J. Jia, “Dreamomni: Unified image generation and editing,” in *CVPR*, 2025. 3
- [9] B. Xia, S. Wang, Y. Tao, Y. Wang, and J. Jia, “Llmg: Multimodal large language model based generation assistant,” in *ECCV*, 2024. 3
- [10] Z. Liu, Y. Yu, H. Ouyang, Q. Wang, K. L. Cheng, W. Wang, Z. Liu, Q. Chen, and Y. Shen, “Magicquill: An intelligent interactive image editing system,” in *CVPR*, 2025. 3
- [11] J. Singh, J. Zhang, Q. Liu, C. Smith, Z. Lin, and L. Zheng, “Smartmask: Context aware high-fidelity mask generation for fine-grained object insertion and layout control,” in *CVPR*, 2024. 3
- [12] C. Mao, J. Zhang, Y. Pan, Z. Jiang, Z. Han, Y. Liu, and J. Zhou, “Ace++: Instruction-based image creation and editing via context-aware content filling,” *arXiv preprint arXiv:2501.02487*, 2025. 3
- [13] W. Song, H. Jiang, Z. Yang, R. Quan, and Y. Yang, “Insert anything: Image insertion via in-context editing in dit,” *arXiv preprint arXiv:2504.15009*, 2025. 3
- [14] X. Wang, S. Fu, Q. Huang, W. He, and H. Jiang, “Ms-diffusion: Multi-subject zero-shot image personalization with layout guidance,” *arXiv preprint arXiv:2406.07209*, 2024. 3
- [15] Y. Li, H. Liu, Q. Wu, F. Mu, J. Yang, J. Gao, C. Li, and Y. J. Lee, “Gligen: Open-set grounded text-to-image generation,” in *CVPR*, 2023. 3
- [16] D. Zhou, Y. Li, F. Ma, X. Zhang, and Y. Yang, “Migc: Multi-instance generation controller for text-to-image synthesis,” in *CVPR*, 2024. 3
- [17] X. Wang, T. Darrell, S. S. Rambhatla, R. Girdhar, and I. Misra, “Instancediffusion: Instance-level control for image generation,” in *CVPR*, 2024. 3
- [18] H. Zhang, Z. Duan, X. Wang, Y. Chen, and Y. Zhang, “Eligen: Entity-level controlled image generation with regional attention,” *arXiv preprint arXiv:2501.01097*, 2025. 3
- [19] X. Wang, S. Fu, Q. Huang, W. He, and H. Jiang, “Ms-diffusion: Multi-subject zero-shot image personalization with layout guidance,” *arXiv preprint arXiv:2406.07209*, 2024. 3
- [20] B. Xia, B. Peng, Y. Zhang, J. Huang, J. Liu, J. Li, H. Tan, S. Wu, C. Wang, Y. Wang *et al.*, “Dreamomni2: Multimodal instruction-based editing and generation,” *arXiv preprint arXiv:2510.06679*, 2025. 4, 7
- [21] P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge *et al.*, “Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution,” *arXiv preprint arXiv:2409.12191*, 2024. 6
- [22] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, “Dino: Detr with improved denoising anchor boxes for end-to-end object detection,” *arXiv preprint arXiv:2203.03605*, 2022. 7
- [23] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, “Learning transferable visual models from natural language supervision,” in *ICML*, 2021. 7
- [24] Y. Peng, Y. Cui, H. Tang, Z. Qi, R. Dong, J. Bai, C. Han, Z. Ge, X. Zhang, and S.-T. Xia, “Dreambench++: A human-aligned benchmark for personalized image generation,” *arXiv preprint arXiv:2406.16855*, 2024. 7
- [25] S. Liu, Y. Han, P. Xing, F. Yin, R. Wang, W. Cheng, J. Liao, Y. Wang, H. Fu, C. Han *et al.*, “Step1x-edit: A practical framework for general image editing,” *arXiv preprint arXiv:2504.17761*, 2025. 7
- [26] S. Batifol, A. Blattmann, F. Boesel, S. Consul, C. Diagne, T. Dockhorn, J. English, Z. English, P. Esser, S. Kulal *et al.*, “Flux.1 kontext: Flow matching for in-context image generation and editing in latent space,” *arXiv e-prints*, pp. arXiv–2506, 2025. 7
- [27] Google, “Gemini,” <https://deepmind.google/models/gemini/>, 2025. 7
- [28] ByteDance, “Doubao,” <https://www.doubao.com/>, 2025. 7
- [29] C. Wu, P. Zheng, R. Yan, S. Xiao, X. Luo, Y. Wang, W. Li, X. Jiang, Y. Liu, J. Zhou *et al.*, “Omnigen2: Exploration to advanced multimodal generation,” *arXiv preprint arXiv:2506.18871*, 2025. 7

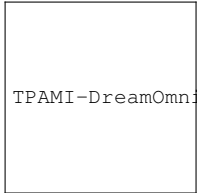
- [30] C. Wu, J. Li, J. Zhou, J. Lin, K. Gao, K. Yan, S.-m. Yin, S. Bai, X. Xu, Y. Chen *et al.*, “Qwen-image technical report,” *arXiv preprint arXiv:2508.02324*, 2025. 7
- [31] P. von Platen, S. Patil, A. Lozhkov, P. Cuenca, N. Lambert, K. Rasul, M. Davaadorj, D. Nair, S. Paul, W. Berman, Y. Xu, S. Liu, and T. Wolf, “Diffusers: State-of-the-art diffusion models,” <https://github.com/huggingface/diffusers>, 2022. 7



Bin Xia received the B.E degree in space communication science and technology from Xidian University, Xian, China, in 2020. He received the M.E. degree in communication engineering at Tsinghua University, Shenzhen, China. He is currently pursuing the PhD degree in computer science at the Chinese University of HongKong. His research interests include image processing, computer vision and model compression.



Bohao Peng is a Ph.D. student in the Department of Computer Science and Engineering at The Chinese University of Hong Kong. His research interests mainly focus on controllable generation, large language models, vision-language models, and computer vision. He has published papers in leading conferences including CVPR and ICLR, with research spanning 3D understanding, semantic segmentation, and multimodal learning. He has also achieved multiple awards in programming competitions, including gold medals in ACM-ICPC.



Jiyang Liu received the BS and MS degrees in the Department of Automation from Tsinghua University in 2017 and 2020, respectively. He is currently a computer vision researcher with ByteDance Intelligent Creation Lab. His research interests mainly cover deep learning, computer vision, dense image understanding, and AIGC-based image editing.



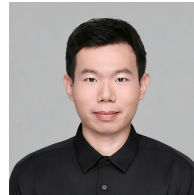
Sitong Wu is currently a Ph.D student at the Chinese University of Hong Kong. She has also had internship experiences at institutions such as Baidu Research, Shanghai AI Laboratory, Alibaba DAMO Academy, and Tencent. Her current research interests include LLM & Agent Post-Training and Self-Evolving. She has published more than 20 papers at top journals and conferences, including TPAMI, NeurIPS, ICLR, CVPR, ICCV, ACL, AAAI, etc.



Jingyao Li received the B.Eng. degree from Xi'an Jiaotong University. She is currently a Ph.D. student at Department of Computer Science and Engineering of the Chinese University of Hong Kong (CUHK), under the supervision of Prof. Jiaya Jia. Her research interests include foundation large language models, especially mid-training and agentic reinforcement learning.



Junjia Huang received the BEng degree from Sun yat-sen University, in 2022. He is currently working toward the PhD degree with Sun yat-sen University. His current research interest including computer vision and image processing.



Xu Zhao received the bachelor's degree from Nanjing University and the master's degree from Shanghai Jiaotong university. He is a senior engineer with Bytedance Inc. His research interests include computer vision and deep learning, particularly in image generative models.



Yitong Wang received the BS and MS degrees in computer science from Peking University, Beijing, China, in 2013 and 2016, respectively. From 2017 to 2019, he was formally a researcher with Tencent AI Laboratory, Shenzhen, China. He is currently with ByteDance Inc. His research interests include image processing, computer vision, and machine learning.



Ruihang Chu received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong, Hong Kong, in 2024. In 2017 and 2020, he received the B.E. degree and M.E degree in mechanical engineering and automation from Beihang University, Beijing, respectively. His research interests include visual generation and multi-modal large language models.



Bei Yu (M'15-SM'22) received the Ph.D. degree from The University of Texas at Austin in 2014. He is currently a Professor in the Department of Computer Science and Engineering, The Chinese University of Hong Kong. He has served as TPC Chair of ACM/IEEE Workshop on Machine Learning for CAD (ISED) in 2019, International Symposium of EDA (ISED) in 2025, and in many journal editorial boards and conference committees. He received eleven Best Paper Awards, ten ICCAD/ISPD/IWLS contest awards, IEEE CEDA Ernest S. Kuh Early Career Award in 2021, DAC Under-40 Innovator Award in 2024, and Hong Kong RGC Research Fellowship Scheme (RFS) Award in 2024.



Jiaya Jia received the Ph.D. degree in Computer Science from Hong Kong University of Science and Technology in 2004 and is currently a full professor in Department of Computer Science and Engineering at the Chinese University of Hong Kong (CUHK). He assumes the position of Associate Editor-in-Chief of IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) and is in the editorial board of International Journal of Computer Vision (IJCV). He continuously served as area chairs for ICCV, CVPR, AAAI, ECCV, and several other conferences for the organization. He was on program committees of major conferences in graphics and computational imaging, including ICCP, SIGGRAPH, and SIGGRAPH Asia. He is a Fellow of the IEEE.