# A. Appendix

In the appendix, we present details omitted from the main paper due to the page limit. In § A.1, we first present additional image generation results of Instruct-Imagen, both quantitatively and qualitatively. In § A.2, we then discuss the details of model architecture, the training related details of Instruct-Imagen and the inference specification. In § A.3, we provide additional details about the retrieval-augmented pre-training and multimodal instruction tuning dataset.

## A.1. Additional Experimental Results

### A.1.1 Complete Quantitative Evaluation

Table 4 shows the complete evaluation results, including the breakdown of semantic consistency and perceptual quality. In addition to the numbers shown in the main paper, we also report the additional average performance over all methods on in-domain tasks and zero-shot tasks. We observe that Instruct-Imagen is better than both the prior methods and our proposed baseline methods in most tasks.

### A.1.2 More Qualitative Evaluation

Besides the qualitative comparison shown in the main paper, we provide additional qualitative visualization on more diverse and sophisticated multi-modal instructions, to explore the limit of Instruct-Imagen. Particularly, Figure 10 presents the in-domain generation outputs, and Figure 11, Figure 12, Figure 13, and Figure 14 jointly presents complex tasks that is unseen during the training. Note that we do not provide qualitative results on face generation due to lack of consent from the original dataset owner.

Figure 9 further presents Instruct-Imagen's generation on instruction with even higher complexity. The left figure presents how good Instruct-Imagen is in handling tasks that contains three multi-modal conditions; The right example showcases the task with detailed text understanding in the condition. As we can see, Instruct-Imagen could handle the case where reference images presents a iconic text content; However, when there is a dense text content, Instruct-Imagen could no longer faithfully reconstruct the output.

## A.2. Implementation Details

### A.2.1 Model Architecture

Our base model design is similar to the Imagen [38], with a few key modifications. First, we've shifted from a three-stage cascaded diffusion model to a two-stage cascaded diffusion model. Concretely, in this setup, the text-to-image generation model first produces 128×128 images (instead of the 64×64 in [38]), and then subsequently upsampled to 1024×1024 by only one super-resolution model. This adjustment allows more detailed and information-rich outputs from the image generation model. As aforementioned, the focus of this work is to adapt and fine-tune the text-to-image generation model to comprehend multimodal instruction. Secondly, rather than employing one DBlock / UBlock per-resolution with multiple ResNetBlocks in each DBlock / UBlock, we've opted for multiple DBlock / UBlock for each resolution, consistently using numResNetBlocksPerBlock=1. This design choice enables us to incorporate more attention layers, a critical aspect for our model. Finally, we've increased the model size, as elaborated below.

To process the multi-modal instruction, we repurpose the downsample network within the text-to-image model as an encoder to extract latent features from the multi-modal instruction. These features, derived from the final layer, are integrated into the text-to-image generation model by introducing a cross-attention layer into each DBlock / UBlock, similar to the text cross-attention in Imagen [38]. Comprehensive architectural details for both the text-to-image and super-resolution models can be found in Table 5.

### A.2.2 Optimization & Inference

The model is trained to predict $\mathbf{v}$ utilizing the standard L2 loss in accordance with [39]. For all experiments, the Adafactor optimizer [40] with $\beta_1$=0.9 and $\beta_2$=0.999 is employed, maintaining a consistent learning rate of $10^{-4}$, along with a warm-up phase comprising $10,000$ steps. The model undergoes training for 500k steps in retrieval-augmented training and 400k steps in multi-modal instruction tuning, utilizing a batch size of 512. Following [12], we utilize the moving average (with weight decay rate 0.9999) for the model weights used in inference. We use the PaxML[2] framework and train the models on 64 TPUv4. During inference, the sampling schedule requires 256 timesteps, employing DDPM and cosine noise schedule. We employ an oscillating classifier-free guidance schedule, alternating between a guidance at the scale of 25.0 and no guidance every consecutive step.

## A.3. More Details on the Training Dataset

### A.3.1 Retrieval-augmented Training Dataset

In the retrieval-augmented training, there are two data situations being presented to the model: (1) the model receives an input of text and a multi-modal context consists of several relevant (image, text) pairs, and outputs the target image. (2) the model receives an input text and outputs the target image (with multi-modal context dropped at 10% of chances). The former data situation represents the task of synthesising a given visual concept, using the

[2] https://github.com/google/paxml

|  | Single-Task |  |  | Multi-Task |  |  | Prior Mtd. |  |  | Instruct-Imagen |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
|  | $SC_{avg}$ | $PQ_{avg}$ | Overall | $SC_{avg}$ | $PQ_{avg}$ | Overall | $SC_{avg}$ | $PQ_{avg}$ | Overall | $SC_{avg}$ | $PQ_{avg}$ | Overall |
| In-domain Evaluation |  |  |  |  |  |  |  |  |  |  |  |  |
| Depth2Img | 0.09 | 0.65 | 0.24 | 0.51 | 0.37 | 0.44 | 0.64 | 0.55 | 0.59 | 0.86 | 0.66 | **0.75** |
| Mask2Img | 0.79 | 0.60 | 0.68 | 0.67 | 0.53 | 0.60 | 0.50 | 0.41 | 0.45 | 0.87 | 0.70 | **0.78** |
| Edge2Img | 0.73 | 0.51 | 0.61 | 0.46 | 0.33 | 0.39 | 0.48 | 0.58 | 0.53 | 0.84 | 0.71 | **0.77** |
| Sty Gen. | 0.44 | 0.46 | 0.45 | 0.60 | 0.70 | 0.65 | 0.64 | 0.71 | 0.67 | 0.85 | 0.92 | **0.88** |
| Sub Gen. | 0.69 | 0.66 | 0.67 | 0.53 | 0.59 | 0.56 | 0.69 | 0.70 | 0.70 | 0.81 | 0.82 | **0.81** |
| Txt2Img | 0.68 | 0.68 | 0.68 | 0.58 | 0.51 | 0.55 | 0.64 | 0.71 | 0.67 | 0.77 | 0.76 | **0.76** |
| Face Gen. | 0.18 | 0.77 | 0.37 | 0.45 | 0.34 | 0.39 | 0.66 | 0.80 | 0.72 | 0.69 | 0.86 | **0.77** |
| Sty Trans. | 0.43 | 0.43 | 0.43 | 0.00 | 0.49 | 0.00 | 0.58 | 0.56 | **0.57** | 0.55 | 0.50 | 0.53 |
| Average | 0.50 | 0.59 | 0.52 | 0.48 | 0.48 | 0.45 | 0.60 | 0.63 | 0.61 | 0.78 | 0.74 | **0.76** |
| Zero-shot Evaluation |  |  |  |  |  |  |  |  |  |  |  |  |
| Sty+Sub | - | - | - | 0.72 | 0.32 | 0.48 | 0.61 | 0.18 | 0.33 | 0.79 | 0.43 | **0.58** |
| Multi Sub | - | - | - | 0.73 | 0.40 | **0.54** | 0.65 | 0.29 | 0.43 | 0.77 | 0.36 | 0.53 |
| Ctrl+Sub | - | - | - | 0.54 | 0.24 | 0.36 | 0.46 | 0.23 | 0.32 | 0.61 | 0.59 | **0.60** |
| Ctrl+Sty | - | - | - | 0.59 | 0.22 | 0.36 | 0.18 | 0.06 | 0.11 | 0.74 | 0.54 | **0.63** |
| Average | - | - | - | 0.64 | 0.30 | 0.44 | 0.48 | 0.19 | 0.30 | 0.73 | 0.48 | **0.59** |

Table 4. Full evaluation results.

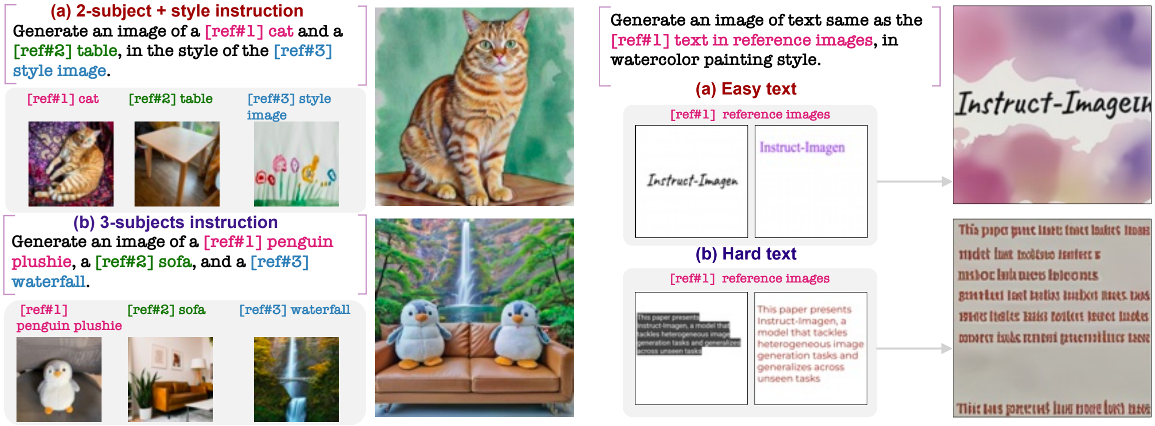

(a) 2-subject + style instruction
Generate an image of a [ref#1] cat and a [ref#2] table, in the style of the [ref#3] style image.
[ref#1] cat
[ref#2] table
[ref#3] style image
(b) 3-subjects instruction
Generate an image of a [ref#1] penguin plushie, a [ref#2] sofa, and a [ref#3] waterfall.
[ref#1] penguin plushie
[ref#2] sofa
[ref#3] waterfall
Generate an image of text same as the [ref#1] text in reference images, in watercolor painting style.
(a) Easy text
[ref#1] reference images
Instruct-Imagen
Instruct-Imagen
Instruct-Imagen
(b) Hard text
[ref#1] reference images
This paper presents Instruct-Imagen, a model that tackles heterogeneous image generation tasks and generalizes across unseen tasks

Figure 9. **Left**: Examples of complex instruction & Outputs. **Right**: Styled text generation results, using text images as input.

text and context, whereas the later situation presents the conventional text-to-image synthesis. As an outcome, the trained `Instruct-Imagen` can preserve the capability of text-to-image generation, while learning the new context-dependent image generation skill. Please refer to Figure 15 for concrete examples from these two learning situations.

### A.3.2 Multi-modal Instruction-tuning Datasets

Subsequent to the retrieval-augmented training, we perform instruction-tuning using multi-modal instructions. In this work, we adopt 9 different tasks, which divides into five general categories.

**Text-to-image Generation.** We require the model to generate both natural and art images to balance its learning of the two domains. To achieve this, we use two datasets for instructed text-to-image generation: an internal high-quality natural image dataset with manual caption; and an art specific dataset crawled from WikiArt (using the pipeline in [44]), with the caption generated by PaLI [8]. Note that the goal of art generation is to not only learn the alignment with content description, but also learn the alignment between art style description. Figure 16 presents the examples from both datasets, which are augmented with a sampled text instruction that summarize the goal of the generation (whether it is natural image or art generation).

**Control-to-Image Generation.** For control-related tasks (Figure 17), we use the widely-adopted conditions – mask, edge, and depth. This allows the trained the model to control the outputs based on the aforementioned conditions. Specifically, we use MiDas [34] for depth estimation, HED [46] for edge extraction, and salient object [30]

| | | Text-to-Image | Super-Resolution |
|---|---|---|---|
| Model size | | 2.76$B$ | 581$M$ |
| DBlock-1 | Resolution | 128 → 64 | 1024 → 512 |
| | #Blocks | 8 | 2 |
| | OutChannels | 512 | 128 |
| | Attention | - | - |
| DBlock-2 | Resolution | 64 → 32 | 512 → 256 |
| | #Blocks | 8 | 4 |
| | OutChannels | 1024 | 256 |
| | Attention | - | - |
| DBlock-3 | Resolution | 32 → 16 | 256 → 128 |
| | #Blocks | 8 | 8 |
| | OutChannels | 2048 | 512 |
| | Attention | Text Instr + Multi-modal Ctx | - |
| DBlock-4 | Resolution | \| | 128 → 64 |
| | #Blocks | \| | 8 |
| | OutChannels | \| | 1024 |
| | Attention | ↓ | Text Instr. |
| UBlock-4 | Resolution | \| | 64 → 128 |
| | #Blocks | \| | 8 |
| | OutChannels | \| | 512 |
| | Attention | ↓ | Text Instr. |
| UBlock-3 | Resolution | 16 → 32 | 128 → 256 |
| | #Blocks | 8 | 8 |
| | OutChannels | 1024 | 256 |
| | Attention | Text Instr + Multi-modal Ctx | - |
| UBlock-2 | Resolution | 32 → 64 | 256 → 512 |
| | #Blocks | 8 | 4 |
| | OutChannels | 512 | 128 |
| | Attention | - | - |
| UBlock-1 | Resolution | 64 → 128 | 512 → 1024 |
| | #Blocks | 8 | 2 |
| | OutChannels | 3 | 3 |
| | Attention | - | - |

Table 5. Model architecture of the Backbone U-Network. Note that the Text-to-Image network do not have DBlock-4 and UBlock-4.

for mask extraction. We also employed edge-to-image data from a sketch dataset [23] as additional edge signals. Since edge is a very loose definition and can present at many different granularity, we perform the edge augmentation during the training. Particularly, we applied edge extraction on the original image, the depth map, and the mask, to obtain both coarse-grained and fine-grained contour images. Additionally, we perform image dilation (with random configurations) on the edge map to simulate the edge image data with different thickness. Finally, for different control signals, we add different text intructions as prefixes to hint the model about the scope of the task to the text description of the image content.

**Subject-driven Generation.** As aforementioned, we employ two subject-driven datasets for general objects and face generation. Particularly, we use the subject-driven dataset introduced in SuTI [7] for general object learning, and the celebrity face datasets [19, 25] to learn face rendering. For face rendering, we group the faces of the same person and caption them with PaLI [8], then we use one sampled (image, text) example as the input text and target image, and using the rest as multi-modal context. Both datasets then join the instruction templates, with reference markers inserted to refer the multi-modal context. Figure 18 provides a qualitative example of these two constructed datasets.

**Styled Generation.** We apply the recipe of StyleDrop [42] to fine-tune our backbone cascaded diffusion model (500 steps on the 128 × 128 model) and create data for styled image generation. The outcome model are used to sample with a styled image for a set of text prompts, which gives as the triplet of (style image, text prompts, and styled image) in return for `Instruct-Imagen` training. Note that the text prompts used here are sampled from the manual prompts of the aforementioned internal natural image dataset, and the style images used for fine-tuning is sampled from WikiArt. We employ a CLIP model to filter out examples that fails the alignment with either style image or text content, which provides a total of 100K data in total. Then we create the multi-modal instructions via combining the instruction template with style image and the manual caption, such that the style image is correctly referred. Figure 19 (a) presents an example of the style-to-image generation data.

**Style Transfer.** Similarly, we construct the style transfer dataset via combining style images from our WikiArt crawl and content images from the internal dataset (with the captions discarded). We use a style transfer model [13] based on the backbone cascaded diffusion model, which allows fast and large-scale generation, to blend the style image with the content image. Note that in the style transfer task, language is not providing any information about the content of the target image, so the model needs to referring fully to the content image to extract semantic information of the target image output. Figure 19 (b) presents an example of the style transfer data.

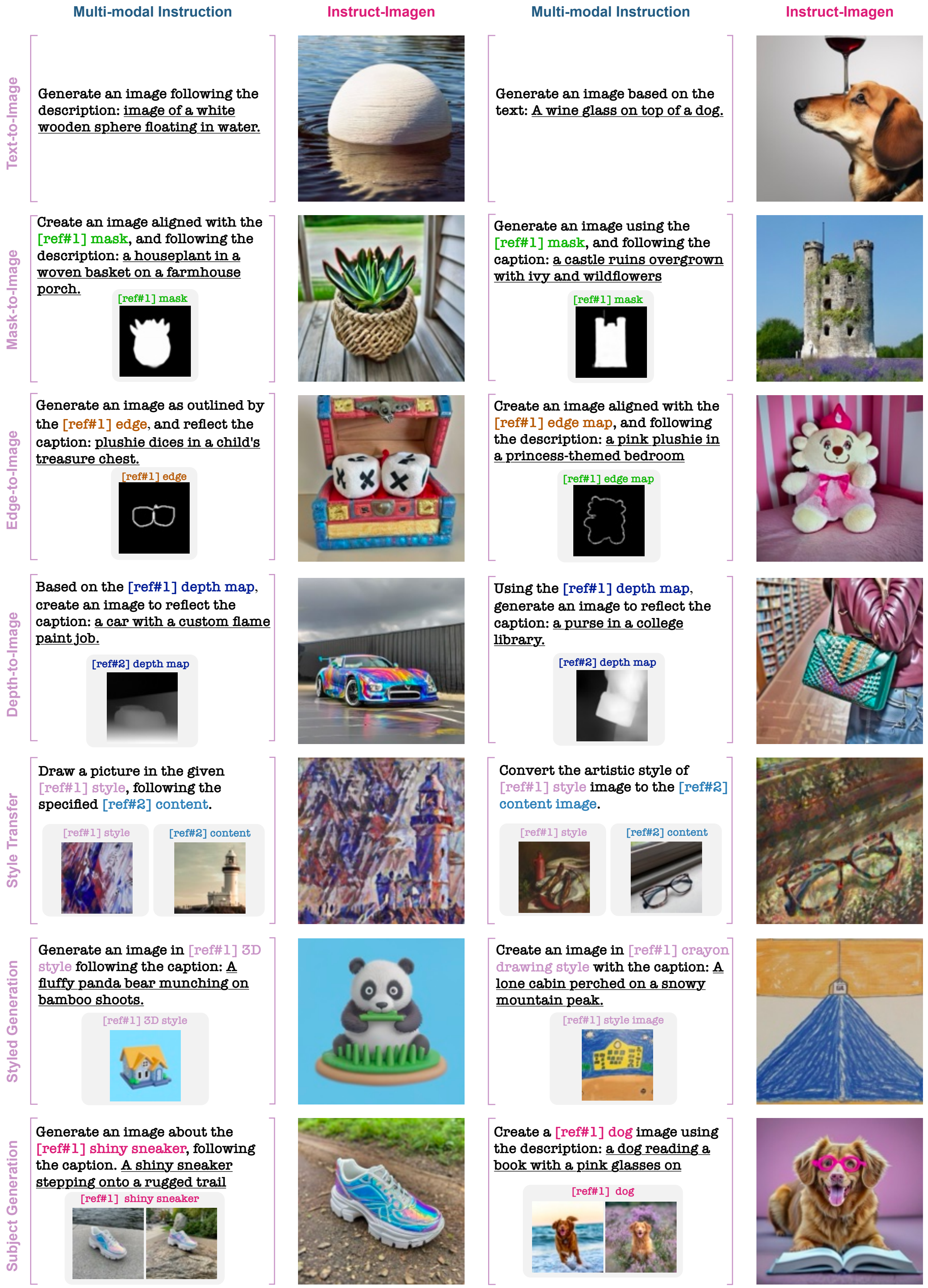
**Instruction Template Generation.** As aforementioned, we prompted the GPT-4 [27] to generate 100 rephrased instruction templates with high variation, and validated the semantic correctness of them manually. During the instruction creation, we use the placeholders in the place where multi-modal contexts are suppose to be inserted, and populate the reference marker (and its associative short prompt) when the instruction is going to be added to each particular data. For example, in subject driven generation, one template could be “`Generate an image of [placeholder], using` `the caption:`", where the placeholder would be substituted with the subject prompt and reference "`[ref#1] a dog`". Note that the reference marker corresponds to a special tokens in the language embedding model.

### A.4. Acknowledgement

We thank Zhiwei Deng, Jason Baldridge, Nando de Freitas for reviewing an early version of this paper in depth, with valuable comments and suggestions. Special thanks to Han Zhang for project idea discussion in the early stage of this project. We also thank Irina Blok for providing a style image used in our evaluation.

### A.5. Broader Impact

Text-to-image generation models like Imagen [38] and Stable Diffusion [29] present ethical concerns, including social bias. `Instruct-Imagen`, using similar Web-scale datasets, faces these same issues. `Instruct-Imagen`'s retrieval-augmented training and multi-modal instruction-tuning have notably enhanced image controllability and attribution. This control can be beneficial or harmful. A risk is using `Instruct-Imagen` for malicious activities, such as creating misleading images of people. Conversely, it offers advantages, like reducing image hallucination and improving relevance to user intent. It also benefits minority communities by effectively generating images of less-known landmarks, foods, and cultural artifacts, addressing the bias in AI systems. To mitigate public risks, we'll be careful with code and API releases. Future work will focus on a responsible use framework, weighing the benefits of research transparency against the dangers of open access, ensuring safe and beneficial usage.



Generate an image following the description: image of a white wooden sphere floating in water.

Generate an image based on the text: A wine glass on top of a dog.

Create an image aligned with the [ref#1] mask, and following the description: a houseplant in a woven basket on a farmhouse porch.

Generate an image using the [ref#1] mask, and following the caption: a castle ruins overgrown with ivy and wildflowers

Generate an image as outlined by the [ref#1] edge, and reflect the caption: plushie dices in a child's treasure chest.

Create an image aligned with the [ref#1] edge map, and following the description: a pink plushie in a princess-themed bedroom

Based on the [ref#1] depth map, create an image to reflect the caption: a car with a custom flame paint job.

Using the [ref#1] depth map, generate an image to reflect the caption: a purse in a college library.

Draw a picture in the given [ref#1] style, following the specified [ref#2] content.

Convert the artistic style of [ref#1] style image to the [ref#2] content image.

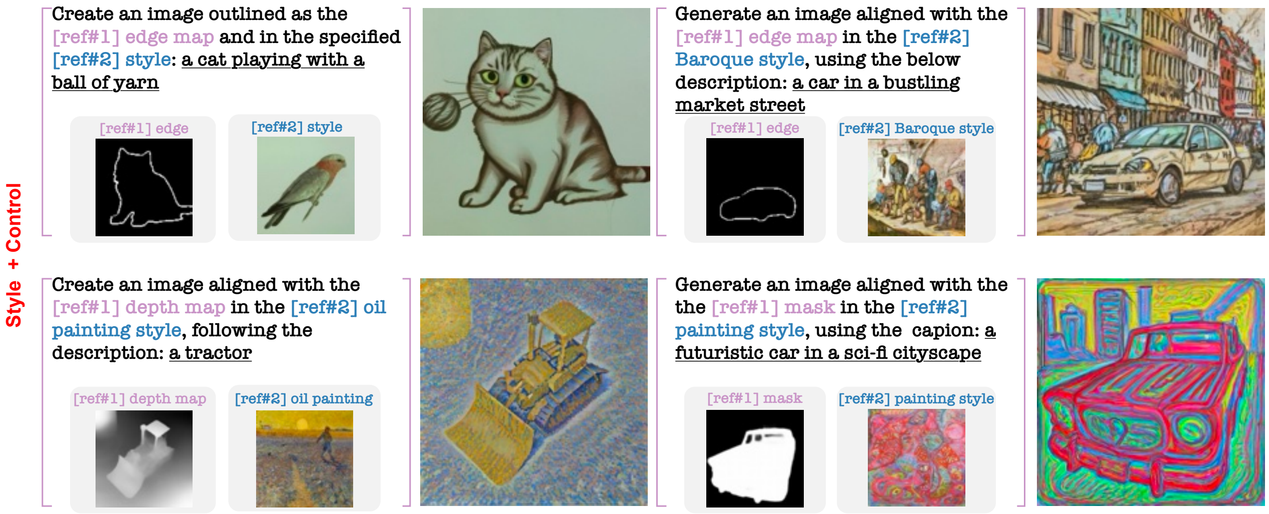
Generate an image in [ref#1] 3D style following the caption: A fluffy panda bear munching on bamboo shoots.

Create an image in [ref#1] crayon drawing style with the caption: A lone cabin perched on a snowy mountain peak.

Generate an image about the [ref#1] shiny sneaker, following the caption. A shiny sneaker stepping onto a rugged trail

Create a [ref#1] dog image using the description: a dog reading a book with a pink glasses on

Figure 10. Additional Qualitative Evaluation on `Instruct-Imagen` for In-domain Tasks. We do not visualize the outputs of the `face generation` task due to lack of consent from the original persons.



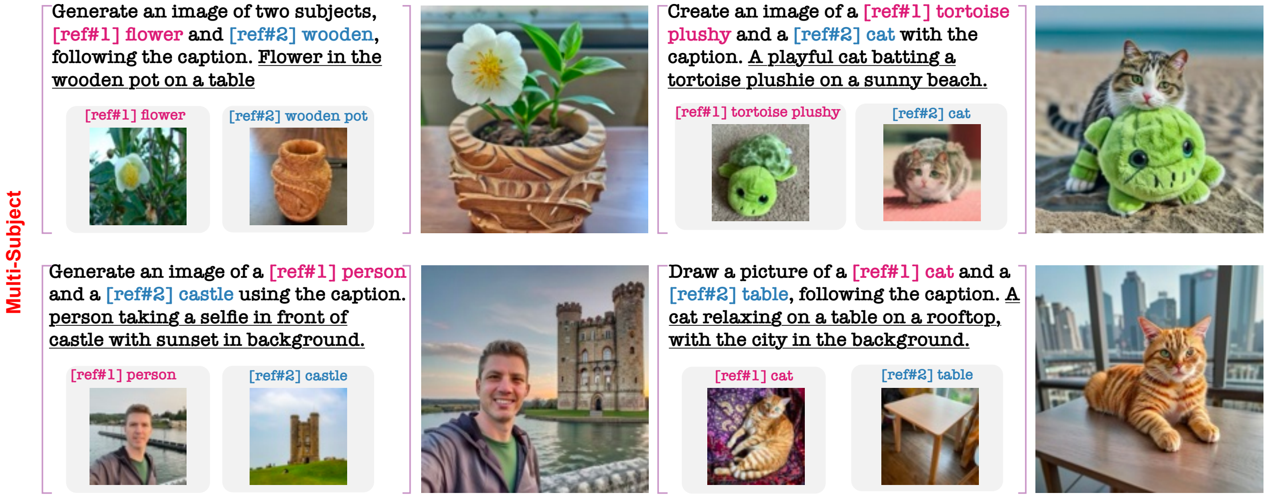
Create an image outlined as the [ref#1] edge map and in the specified [ref#2] style: a cat playing with a ball of yarn

Generate an image aligned with the [ref#1] edge map in the [ref#2] Baroque style, using the below description: a car in a bustling market street

Create an image aligned with the [ref#1] depth map in the [ref#2] oil painting style, following the description: a tractor

Generate an image aligned with the the [ref#1] mask in the [ref#2] painting style, using the capion: a futuristic car in a sci-fi cityscape

Figure 11. Additional Qualitative Evaluation of `Instruct-Imagen` on Control + Style Generation.



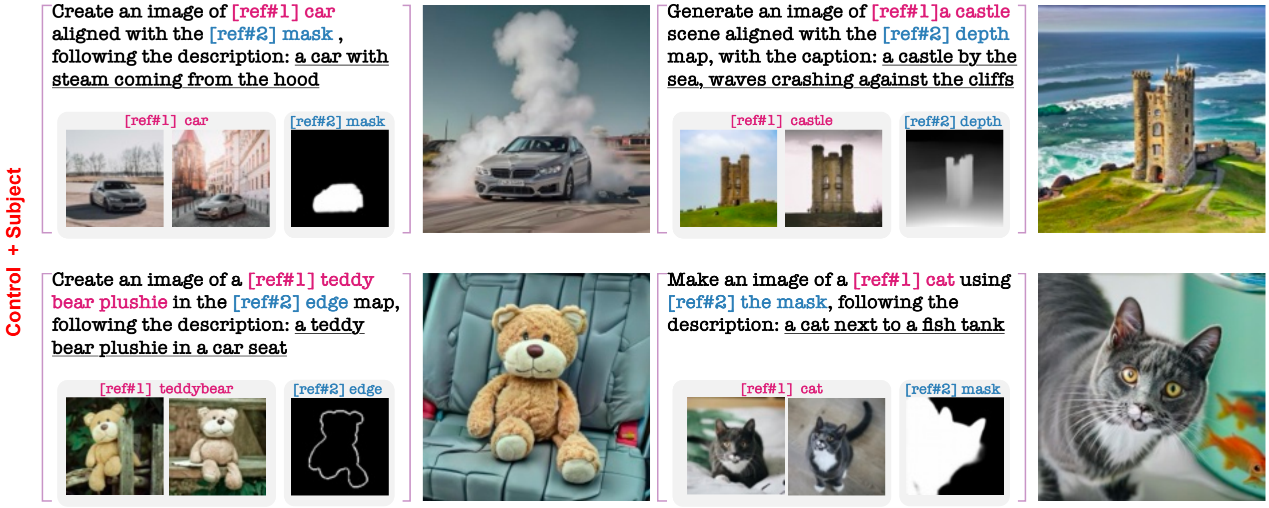
Generate an image of two subjects, [ref#1] flower and [ref#2] wooden, following the caption. Flower in the wooden pot on a table

Create an image of a [ref#1] tortoise plushy and a [ref#2] cat with the caption. A playful cat batting a tortoise plushie on a sunny beach.

Generate an image of a [ref#1] person and a [ref#2] castle using the caption. person taking a selfie in front of castle with sunset in background.

Draw a picture of a [ref#1] cat and a [ref#2] table, following the caption. A cat relaxing on a table on a rooftop, with the city in the background.

Figure 12. Additional Qualitative Evaluation of `Instruct-Imagen` on Multi-Subject Generation.



Create an image of [ref#1] car aligned with the [ref#2] mask , following the description: a car with steam coming from the hood

Generate an image of [ref#1]a castle scene aligned with the [ref#2] depth map, with the caption: a castle by the sea, waves crashing against the cliffs

Create an image of a [ref#1] teddy bear plushie in the [ref#2] edge map, following the description: a teddy bear plushie in a car seat

Make an image of a [ref#1] cat using [ref#2] the mask, following the description: a cat next to a fish tank

Figure 13. Additional Qualitative Evaluation of `Instruct-Imagen` on Control + Subject Generation.

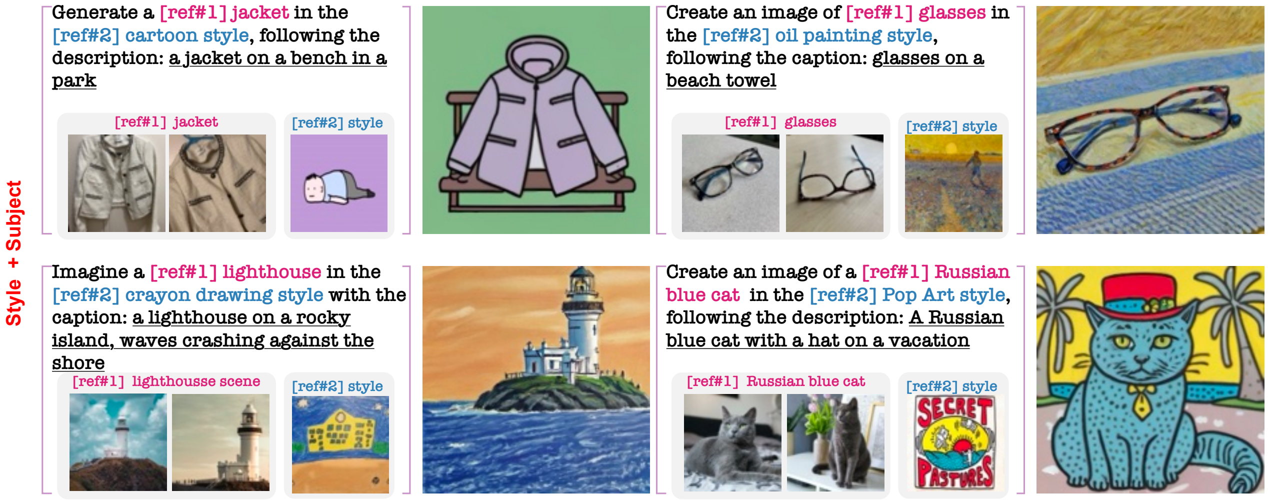

Style + Subject
Generate a [ref#1] jacket in the [ref#2] cartoon style, following the description: a jacket on a bench in a park
[ref#1] jacket
[ref#2] style
Create an image of [ref#1] glasses in the [ref#2] oil painting style, following the caption: glasses on a beach towel
[ref#1] glasses
[ref#2] style
Imagine a [ref#1] lighthouse in the [ref#2] crayon drawing style with the caption: a lighthouse on a rocky island, waves crashing against the shore
[ref#1] lighthousse scene
[ref#2] style
Create an image of a [ref#1] Russian blue cat in the [ref#2] Pop Art style, following the description: A Russian blue cat with a hat on a vacation
[ref#1] Russian blue cat
[ref#2] style

Figure 14. Additional Qualitative Evaluation of `Instruct-Imagen` on Styled Subject Generation.

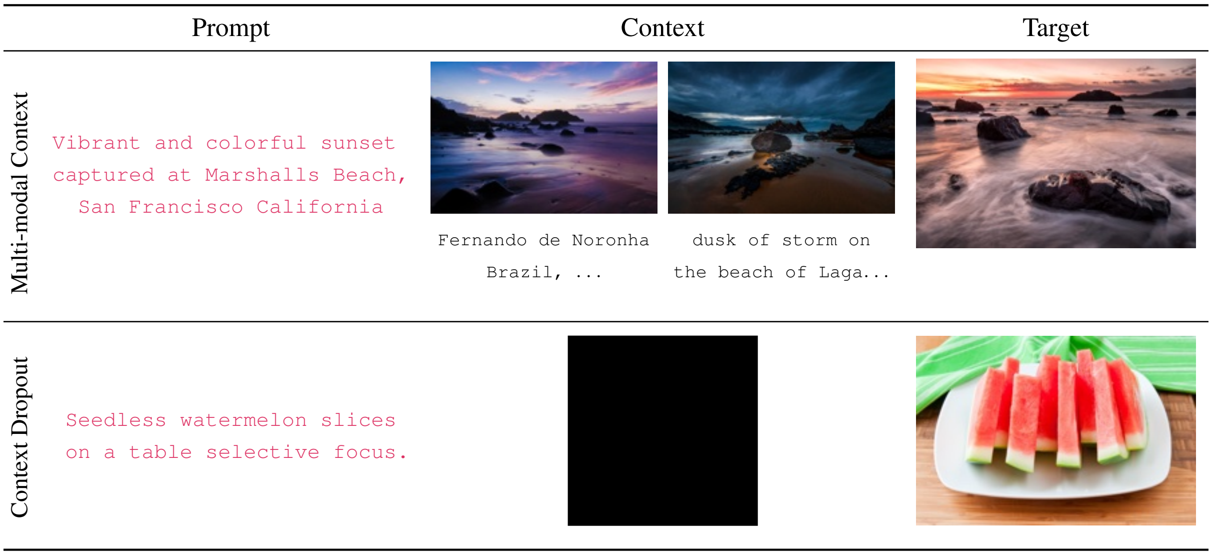

Prompt
Context
Target
Multi-modal Context
Vibrant and colorful sunset captured at Marshalls Beach, San Francisco California
Fernando de Noronha Brazil, ...
dusk of storm on the beach of Laga...
Context Dropout
Seedless watermelon slices on a table selective focus.

Figure 15. **Data for Retrieval-Augmented Training.** We present two training situations: (1) the case where multi-modal context are presented to the model when generating the image; and (2) the case where multi-modal context are dropped during the training.

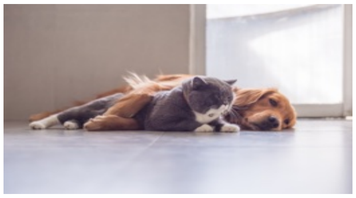
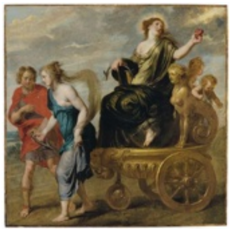
|  | Instruction | Target |
|---|---|---|
| (a) Natural Images | Bring forth an image based on the caption, British short hair cat and golden retriever. |  |
| (b) Art Images | Generate this artwork: The Triumph of Hope, an allegorical painting by Erasmus Quellinus The Younger in the Baroque style. |  |

Figure 16. **Text-to-Image Data for Instruction-Tuning.**.

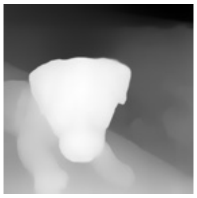
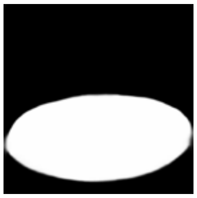
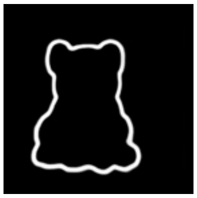
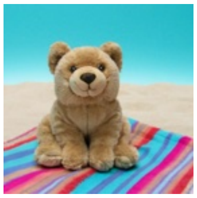
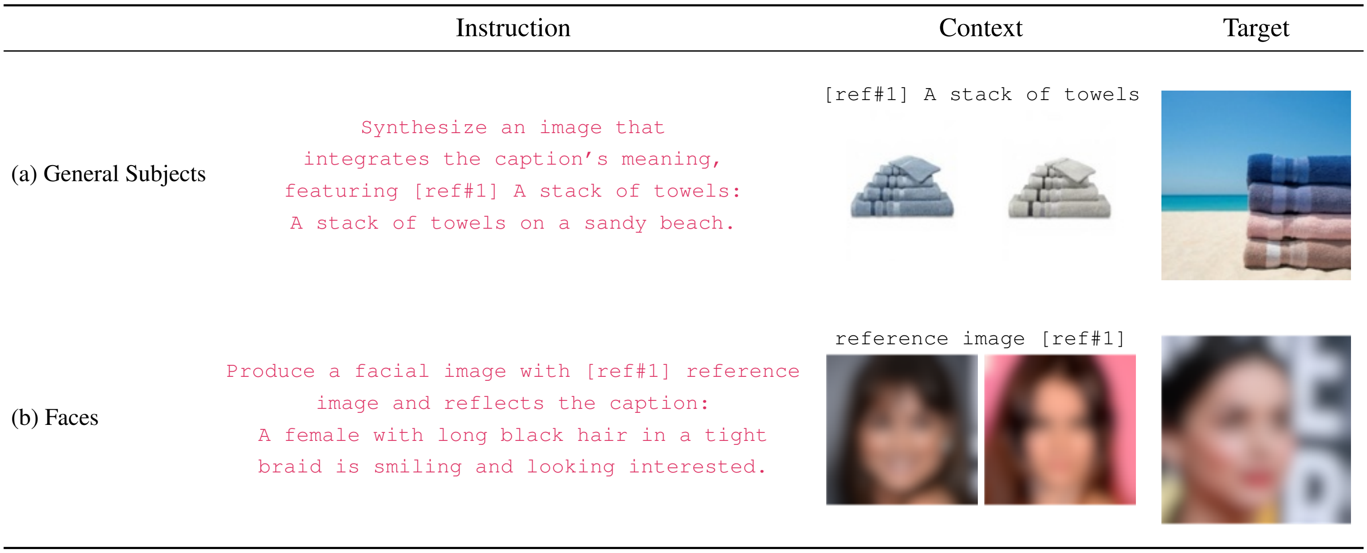
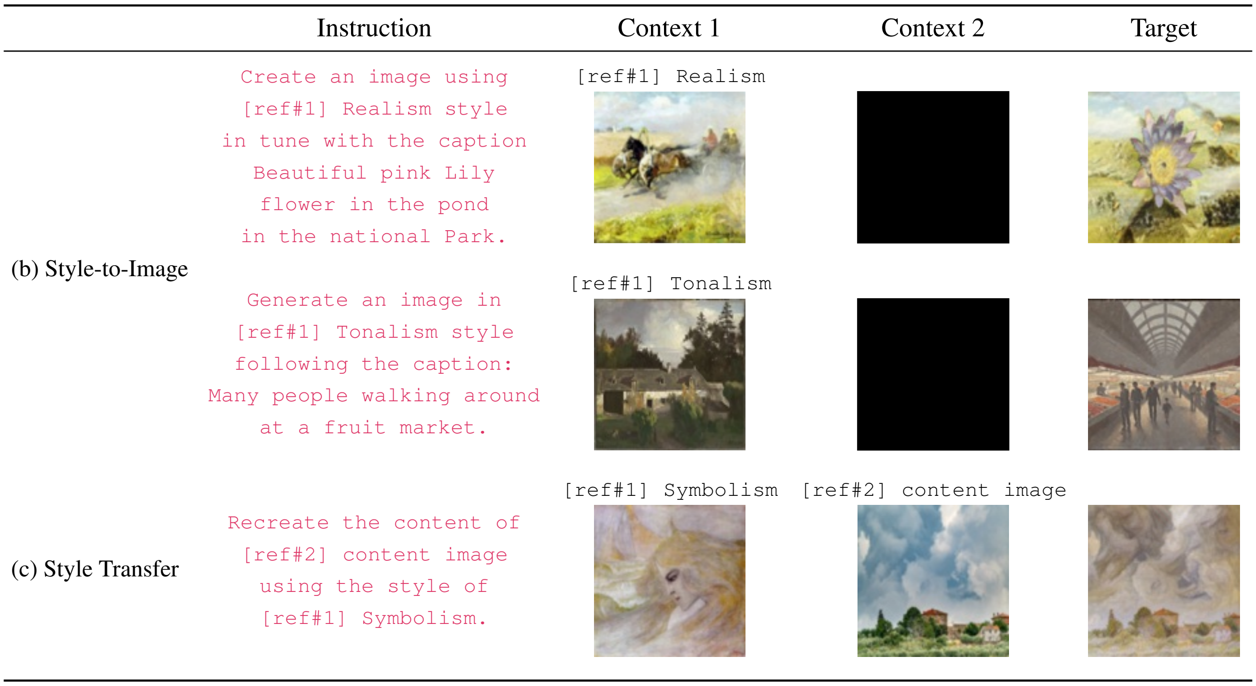
|  | Instruction | Context | Target |
|---|---|---|---|
| (a) Depth | Create an image using [ref#1] depth map as a reference and following the below description: A black and white puppy in a sunflower field. | [ref#1] depth map<br> | |
| (b) Mask | Generate an image by taking cues from [ref#1] object mask as a reference and following this caption: A pizza on top of a mountain peak. | [ref#1] object mask<br> | |
| (b) Edge | Let [ref#1] edge image guide you in crafting an image that fulfills this description - A stuffed animal on a beach blanket. | [ref#1] edge image<br> |  |

Figure 17. **Control-Related Data for Instruction-Tuning.**



| | Instruction | Context | Target |
|---|---|---|---|
| (a) General Subjects | Synthesize an image that integrates the caption’s meaning, featuring [ref#1] A stack of towels: A stack of towels on a sandy beach. | [ref#1] A stack of towels | |
| (b) Faces | Produce a facial image with [ref#1] reference image and reflects the caption: A female with long black hair in a tight braid is smiling and looking interested. | reference image [ref#1] | |

Figure 18. **Subject-Related Data for Instruction-Tuning.** The face image is anonymized to protect the privacy.



| | Instruction | Context 1 | Context 2 | Target |
|---|---|---|---|---|
| (b) Style-to-Image | Create an image using [ref#1] Realism style in tune with the caption Beautiful pink Lily flower in the pond in the national Park. | [ref#1] Realism | | |
| | Generate an image in [ref#1] Tonalism style following the caption: Many people walking around at a fruit market. | [ref#1] Tonalism | | |
| (c) Style Transfer | Recreate the content of [ref#2] content image using the style of [ref#1] Symbolism. | [ref#1] Symbolism | [ref#2] content image | |

Figure 19. **Style-Related Data for Instruction-Tuning.**