Omni-Attribute: Open-vocabulary Attribute Encoder for Visual Concept Personalization

Supplementary Material

A. Implementation Details

A.1. Training Datasets

To learn high-quality representations for open-vocabulary attributes, we collect semantically linked image pairs from two complementary sources, resulting in nine datasets, as illustrated in Fig. 10.

First, we collect 23.7M image pairs from an in-house image collection dataset, where images are organized into thematic collections. As shown in Fig. 10(a), images within the same collection are typically captured during a single photo session, exhibiting both shared and distinct characteristics across multiple visual aspects. We randomly sample two images from each theme to form pairs, producing diverse combinations of positive and negative attributes. In total, the dataset contains 600K unique attribute labels. We qualitatively demonstrate the richness and diversity of these labels through a word cloud visualization in Fig. 2. To further enhance representations for "*person identity*," we additionally sample an identity-centric subset consisting of 2.21M image pairs, where paired images depict the same individual(s), as shown in Fig. 10(b).

While these *image collection datasets* are large-scale and rich in attribute diversity, their image pairs often exhibit multiple entangled positive attributes, making it challenging to isolate one attribute. To address this limitation, we construct seven additional datasets, each focusing on a specific attribute (*e.g.*, *facial expression*, *background*, or *lighting*). As illustrated in Fig. 10(c-i), image pairs in these datasets are designed to share only one or a few positive attributes, facilitating the learning of attribute-specific representations. The detailed curation process for these datasets is summarized as follows:

Dataset	Scale	Curation Process
Facial Expression	51.0K	We employ the expression editing model, LivePortrait [23], to manipulate two human images initially with *neutral facial expressions* (according to the detailed descriptions labeled during the attribute annotation). For each pair, we randomly sample a set of editing parameters to generate two images exhibiting the same facial expression.
Hairstyle	8.77K	We use an in-house hairstyle editing model to process two face-centric images. Given a reference image with a target hairstyle, the model generates two edited images sharing the same hairstyle.
Pose	106K	We extract face, body, and hand keypoints from a human image, and synthesize its paired image using ControlNet [73] to ensure the same pose as the source image.
Background	35.1K	We first sample clean background images filtered by Qwen-VL [66] (by asking whether images depict clear backgrounds). Then, we use Qwen-Image-Edit [68] to randomly insert foreground objects and modify contextual factors (*e.g.*, time of day, weather, and camera angle). Each background image is edited twice to form a pair with consistent background.
Camera Angle	98.7K	We collect 2,081 panoramic images from multiple sources [38, 49, 60, 74] and apply PreciseCam [3] to crop views corresponding to specific camera angles. For each pair, cropping hyperparameters are randomly sampled to ensure both images share the same camera angle.
Lighting and Tone	159K	We construct a set of detailed prompts describing various image lighting conditions, along with a separate set of prompts specifying identities and actions. By fixing the lighting prompts while varying the identity prompts, we synthesize paired images via FLUX [36], ensuring consistent lighting across each pair.
Style and Material	27.5K	Similar to the *Lighting and Tone* dataset, we design descriptive prompts focusing on style and material properties. Images are synthesized using Stable Diffusion XL [52], where the same style or material description is preserved across pairs.

During training, we assign a sampling weight of 100 to both image collection datasets, and a weight of 1 to each attribute-specific dataset.

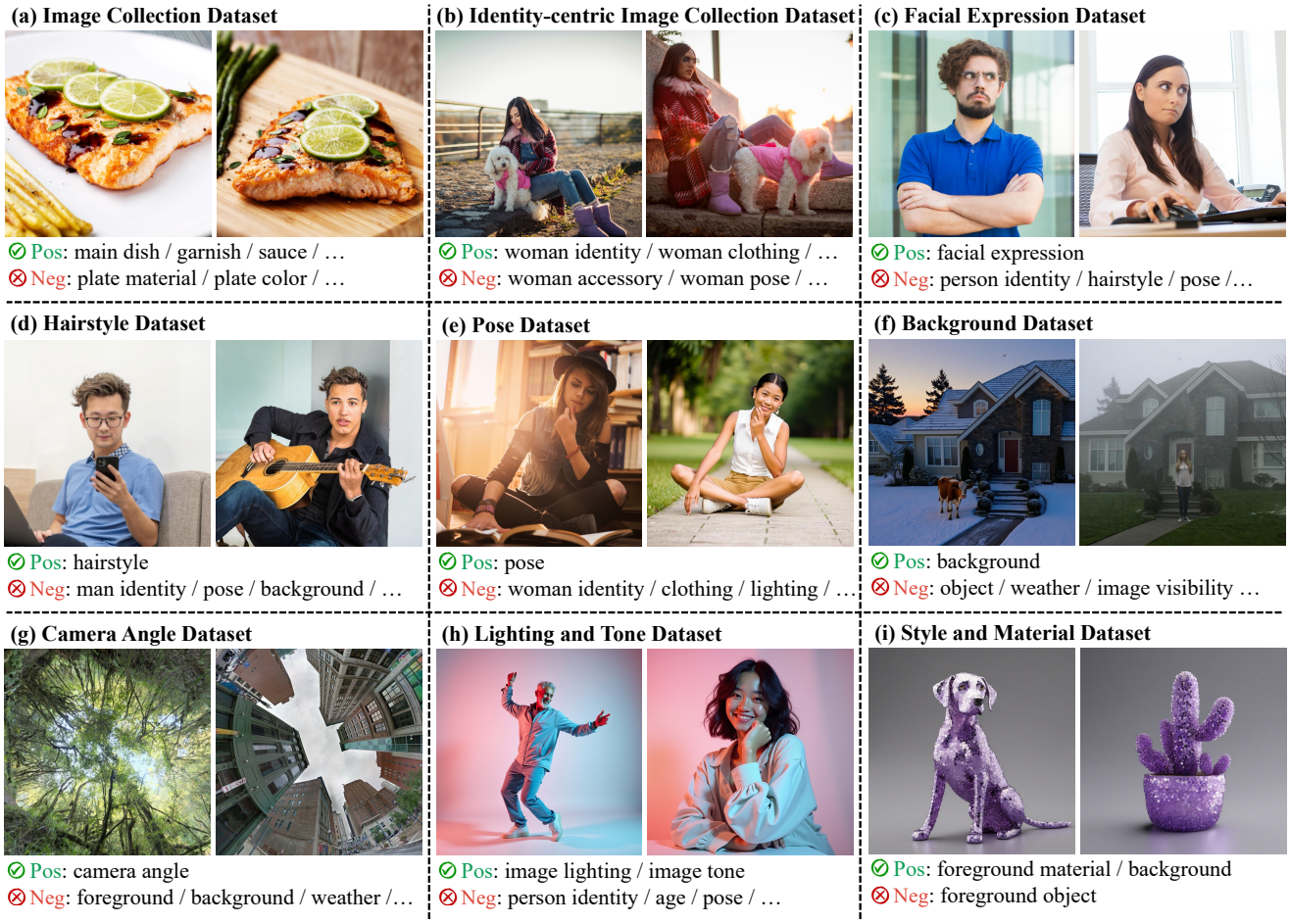

Figure 10. **Training datasets.** Our training image pairs are constructed from two sources: (i) *image collection datasets* (a–b), with image pairs captured during the same photo session, exhibiting varying positive and negative attributes; and (ii) *attribute-specific datasets* (c–i), with image pairs synthesized via generative or editing models that differ mainly in one or a few positive attributes.

A.2. Annotation of Positive and Negative Attributes

As described in Sec. 3.1, we adopt a two-stage annotation pipeline to balance annotation quality and computational cost.

In the first stage, we employ a powerful vision-language model, Qwen2.5-VL-72B [66], to generate high-quality attribute annotations based on a comprehensive instruction prompt (see Fig. 11). Due to the substantial computational overhead, this stage is applied only to a subset of 200K samples. We introduce two key design choices to further improve annotation effectiveness: (i) Inspired by Chain-of-Thoughts [67], we prompt the model to explicitly reason the detailed similarities and differences behind each positive and negative attribute. Empirically, this approach enhances annotation quality. (ii) We design the instruction prompt, ending with the initial segment of the target output format (*i.e.*, "`{positive:[`" at the bottom of Fig. 11) and activate the "`continue_final_message`" setting. This guides the model to continue generation in the intended structured format and can improve the rate of syntactically valid outputs.

In the second stage, we finetune a Qwen2.5-VL-32B [66] into a task-dependent model tailored for this annotation task, eliminating the need for a lengthy instruction prompt. During training, the input is a concise multimodal prompt: "*<image_1> <image_2> What are the positive attributes shared by the two images and the negative attributes that differentiate them? For the similarities and differences, explain the reasons in brackets.*" The output is a string following a structured dictionary-style format. This design reduces the input token length by 3.1× and lowers the per-sample forward latency by 6.3×. The model is finetuned on 200K annotated samples using 32× 80GB H100 GPUs, with a learning rate of 2e-7 (with a linear warm-up and cosine decay strategy), a batch size of 512, and 15 training epochs.

For large-scale inference, we enhance computational efficiency by constraining image dimensions so that the total pixel count does not exceed $1280 \times 28 \times 28$. This constraint yields maximum resolutions of 1336×752 for $16:9$ images,

 # System:

You need to compare an image pair and answer the positive attributes shared by two images and the negative attributes differentiating two images.

The attributes can relate to foreground, background, visual characteristics, or camera information. Here are some examples:

1. Foreground:

a. If foreground is a human or animate character: [(list of example attributes and their descriptions)]
b. If foreground is an animal or plant: [(list of example attributes and their descriptions)]
c. If foreground is an inanimate object: [(list of example attributes and their descriptions)]

2. Background: [(list of example attributes and their descriptions)]

3. Visual Characteristics: [(list of example attributes and their descriptions)]

4. Camera Information: [(list of example attributes and their descriptions)]

Be advised of three notes:

- These are just examples. You can answer any attributes. Some other attributes include number of objects, subject interaction, subject position, subject shape, subject size, person age, person gender, background clarity, background complexity, image symmetry, image sharpness, image exposure, image reflection, aspect ratio.

- Mention only up to 10 most perceivable attributes that primarily align or differ the image pair.

- After mentioning the positive / negative attributes, try to briefly explain the similarity / difference in brackets.

Here is an example:

You can answer: {
"positive": ["woman identity", "hairstyle (medium length with natural waves)", "clothing (textured blazer)", "action (talk on the phone)"],
"negative": ["pose (stand straight / stand with slight angle)", "facial expression (smile / concerned)", "camera distance (medium shot / close shot)"]
}

Here is another example:

You can answer: {
"positive": ["tree species (coastal redwood)", "background environment (forest)"],
"negative": ["image structure", "image lighting", "camera angle (worm's-eye view / low-angle)", "aspect ratio (landscape / portrait)"]
}

 # User:

Answer the positive and negative attributes between these two images:

<image_1> <image_2>

 # Assistant:

{"positive": ["

Figure 11. **Instruction prompt for the first stage of attribute annotation.**

1157×868 for $4:3$ images, and 1002×1002 for square images. We perform inference on 80GB H100 GPUs, where each image pair takes approximately 2.54 seconds to annotate.

A.3. Model Architecture

As described in Sec. 3.3, our attribute encoder consists of a LoRA-finetuned multimodal large language model (MLLM) followed by a fully trainable connector module, while our image generator builds upon a frozen image generator equipped with trainable IP-adapter modules. We detail the model architecture as follows:

Module	Description
MLLM	We adopt Qwen2.5-VL-7B [66] as the base MLLM, where all model parameters are frozen, and LoRA adapters [31] are inserted into every linear projection within both the vision encoder and the MLLM modules. Each LoRA module uses a rank of 256 and an alpha value of 512. The token dimensionality is 3584, and we restrict the maximum number of image tokens at 1000.
Connector	To bridge the MLLM with the image generator, we follow the Step1X-Edit [41] design and incorporate a linear projection layer followed by eight self-attention layers as the connector. The linear layer projects the token dimensionality from 3584 to 4096 to align with the image generator.
Image Generator	We use FLUX.1-dev [36] as the base model. Following the observations of Goyal *et al*. [21], we note that finetuning the model can diminish FLUX's ability to perform distillation guidance (*i.e*., it tends to overfit to the non-distillation-guidance setting used during training). To mitigate this, we adopt their proposed *Shortcut-Rerouted Adapter*, which helps preserve the model's prior for distillation guidance.
IP-Adapter	We follow the implementation of the InstantX team [32], where each DiT block [50] in FLUX includes two MLP modules (*i.e*., `to_k` and `to_v`) that compute key and value embeddings, respectively. These embeddings are then injected into query image tokens through multi-head cross-attention, enabling the generator to incorporate conditioning signals for attribute personalization.

A.4. Model Training

We train the model in two stages to enhance training efficiency. In the first stage, the model is optimized solely with the generative loss for 100K steps; in the second stage, we introduce an additional contrastive loss and continue training for 10K more steps. This two-stage design is due to the computational overhead of the contrastive loss, which requires four additional forward passes through the MLLM for each training sample (*i.e*., two images cross-paired with the positive and negative attributes). Therefore, it could substantially slow down convergence if we optimize the contrastive loss from the start.

We conduct all experiments using $64\times$ 80 GB H100 GPUs with a total batch size of 256. Training is performed in mixed precision with parameter and reduction data types set to `bf16` and `fp32`, respectively. We apply gradient clipping with a maximum gradient norm of 1.0, and employ Distributed Data Parallel (DDP) and Fully Sharded Data Parallel (FSDP) [75] strategies for efficient large-scale training. We use the AdamW optimizer [43] with a learning rate of 1e-5, weight decay of 0.01, and β of [0.9, 0.99]. A linear warmup is applied during the first 1K steps of both stages. During the first stage, we only finetune the connector and IP-Adapter modules for the first 10K steps while keeping the MLLM parameters frozen to prevent disruption of pretrained representations.

For image preprocessing, we resize each reference image such that its total pixel count does not exceed $1000 \times 28 \times 28$. To improve robustness to low-resolution inputs during inference, we apply a 10% probability of downsampling augmentation. The target images are resized and center-cropped to 512×512 resolution. For the generative loss, we adopt the flow-matching objective [39] with λ_{gen} of 1. We adjust the balance between generative and contrastive losses through λ_{con}, as ablated in Sec. 4.4 and Tab. 1.

B. Evaluation Details

B.1. Open-vocabulary Attribute Personalization

Sec. 4.1 compares *Omni-Attribute* with the existing models for personalization across 15 attributes, grouped into two categories. We list all evaluation attributes below:

- *Concrete objects*: man identity, woman identity, object identity, clothing, and background.
- *Abstract concepts*: hairstyle, facial expression, makeup, pose, foreground material, texture, camera angle, image lighting, image tone, and artistic style.

To complement the qualitative and quantitative evaluation shown in Figs. 5 and 6, we list the full evaluation prompts used in Fig. 5 in Tab. 2, and report the original numerical values visualized in Fig. 6 in Tab. 3.

As described in Sec. 4.1, we apply both MLLM-based and human evaluations for a comprehensive assessment. For the MLLM evaluation, we query GPT-4o [47] three times using the prompts shown in Fig. 12 to measure *text fidelity*, *image fidelity*, and *image naturalness*. For the user study, participants are presented with the reference image-attribute pair, the prompt, and the generated image for each sample, as shown in Fig. 13. They are then asked to rate the three evaluation metrics on a scale from 1 (poor) to 5 (excellent). All scores are subsequently normalized to the range of [0,1].

Table 2. **Full evaluation prompts used in Fig. 5.**

Reference Attribute	Full Prompt
Person Identity	"*A woman in a side view that looks festive and blowing a party horn with a laughing expression, surrounded by colorful streamers and balloons.*"
Dog Identity	"*A dog playing in a bright, minimalist art gallery with polished floors and white walls displaying modern paintings. Natural light filters through glass doors, highlighting a clean, contemporary aesthetic of calm and creativity.*"
Clothing	"*A joyful young woman with long brunette hair leaping midair in a white space, frozen in motion. Her athletic form radiates freedom, energy, and the graceful strength of dance.*"
Hairstyle	"*A woman wearing a denim jacket over a white top. Her ears are adorned with large hoop earrings, and she has bold makeup featuring defined eyebrows and bright lipstick. The background is a soft, solid pink hue.*"
Facial Expression	"*A woman with dark hair styled in a messy bun. She is wearing a plain white shirt, and the background is a clean, neutral white.*"
Makeup	"*A woman with long, wavy brown hair, wearing a white t-shirt featuring bold red and blue graphics. The background includes a bright turquoise frame against a neutral-colored wall, creating a striking contrast.*"
Pose	"*A low-angle view a joyful young woman midair against a bright blue sky. Wearing orange trousers and green sneakers, she embodies freedom, exuberance, and carefree spontaneity.*"
Foreground Material	"*A handbag equipped with a chain strap resting on a clean white block. The background is a soft gradient of light blue, creating a calm and sophisticated setting.*"
Texture	"*A sleek Mclaren sports car parked on a paved street. The vehicle's aerodynamic design and shiny exterior reflect its high-performance nature. In the background, a brick building with large windows and various utility fixtures stands, along with a striped awning and some greenery.*"
Image Lighting	"*A pristine armchair with a tufted backrest and decorative buttons standing against a clean, minimalist backdrop. Its elegant design features slender, spindle-like legs that add a touch of classic charm to its modern aesthetic. The seat cushion appears soft and inviting, complementing the chair's overall refined look.*"
Artistic Style	"*A graceful white horse galloping across a dirt path, its mane and tail flowing in the breeze. The background is a lush, dense forest bathed in soft, golden light, creating a serene and natural setting.*"

Table 3. **Numerical results of open-vocabulary attribute personalization.** We complement the quantitative comparison graphs (Fig. 6) by providing the exact measurements of *text fidelity* (Text-F), *attribute fidelity* (Attr-F), *image naturalness* (Natural), and their average.

Method	*Concrete Objects*				*Abstract Concepts*			
	Text-F↑	Attr-F↑	Natural↑	Average↑	Text-F↑	Attr-F↑	Natural↑	Average↑
MLLM Evaluation								
CLIP [54]	0.9000	0.6550	0.8400	0.7983	0.9504	0.3120	0.8056	0.6893
DINOv2 [48]	0.8460	0.7747	0.8160	0.8122	0.9168	0.3568	0.8073	0.6936
Qwen-VL [66]	0.8820	0.6848	0.8460	0.8043	0.9760	0.2736	0.8072	0.6856
OmniGen2 [70]	0.9140	0.7890	0.7810	0.8280	0.9512	0.2863	0.7435	0.6603
FLUX-Kontext [37]	0.8540	0.8910	0.7091	0.8180	0.9304	0.3363	0.7440	0.6702
Qwen-Image-Edit [68]	0.5910	0.8980	0.8091	0.7660	0.3872	0.7515	0.6895	0.6074
Omni-Attribute	0.9381	0.7634	0.8540	0.8518	0.8539	0.5181	0.8079	0.7267
Human Evaluation								
CLIP [54]	0.9300	0.6050	0.8477	0.7942	0.9159	0.4331	0.8576	0.7356
DINOv2 [48]	0.9087	0.6844	0.8434	0.8122	0.9179	0.4363	0.8684	0.7409
Qwen-VL [66]	0.9242	0.5505	0.8542	0.7763	0.9445	0.3957	0.8754	0.7385
OmniGen2 [70]	0.9376	0.7575	0.8110	0.8354	0.9462	0.4392	0.8533	0.7463
FLUX-Kontext [37]	0.9054	0.8785	0.8032	0.8624	0.9323	0.4688	0.8910	0.7641
Qwen-Image-Edit [68]	0.7055	0.8913	0.8426	0.8131	0.6133	0.8622	0.8618	0.7344
Omni-Attribute	0.9564	0.7691	0.8680	0.8645	0.9374	0.7031	0.9584	0.8663

(a) Text Fidelity Score

You are an expert in visual–semantic alignment. Given the following text prompt: "<prompt>", your task is to judge whether the image content accurately matches the meaning of the text.

Consider aspects like object presence, actions, attributes, and overall scene composition.

Please rate how well this image matches the text prompt on a scale of 0 to 10, where:
- 0 means the no match (image and text describe entirely different things)
- 10 means the perfect match (image fully and clearly depicts what the text describes)

Respond with ONLY a single number between 0 and 10.

(b) Attribute Fidelity Score

You are an expert in visual attribute analysis. Given two images, your task is to judge whether the second image successfully incorporates the "<reference attribute>" from the first (reference) image?

Please rate on a scale of 0 to 10, where:
- 0 means no transfer (the attribute is completely absent or totally different)
- 10 means perfect transfer (the attribute is perfectly preserved/transferred)

Respond with ONLY a single number between 0 and 10.

(c) Image Naturalness Score

You are an expert in visual quality assessment. Given an image, your task is to rate the naturalness and quality of this image.

Consider aspects like:
- Visual coherence and realism
- Absence of artifacts or distortions
- Natural lighting and shadows
- Realistic textures and proportions

Please rate on a scale of 0 to 10, where:
- 0 means the very unnatural (image looks completely unnatural, artificial, or has severe artifacts)
- 10 means the completely natural (image looks perfectly natural and photorealistic)

Respond with ONLY a single number between 0 and 10.

Figure 12. **Instruction prompt for MLLM evaluation.**

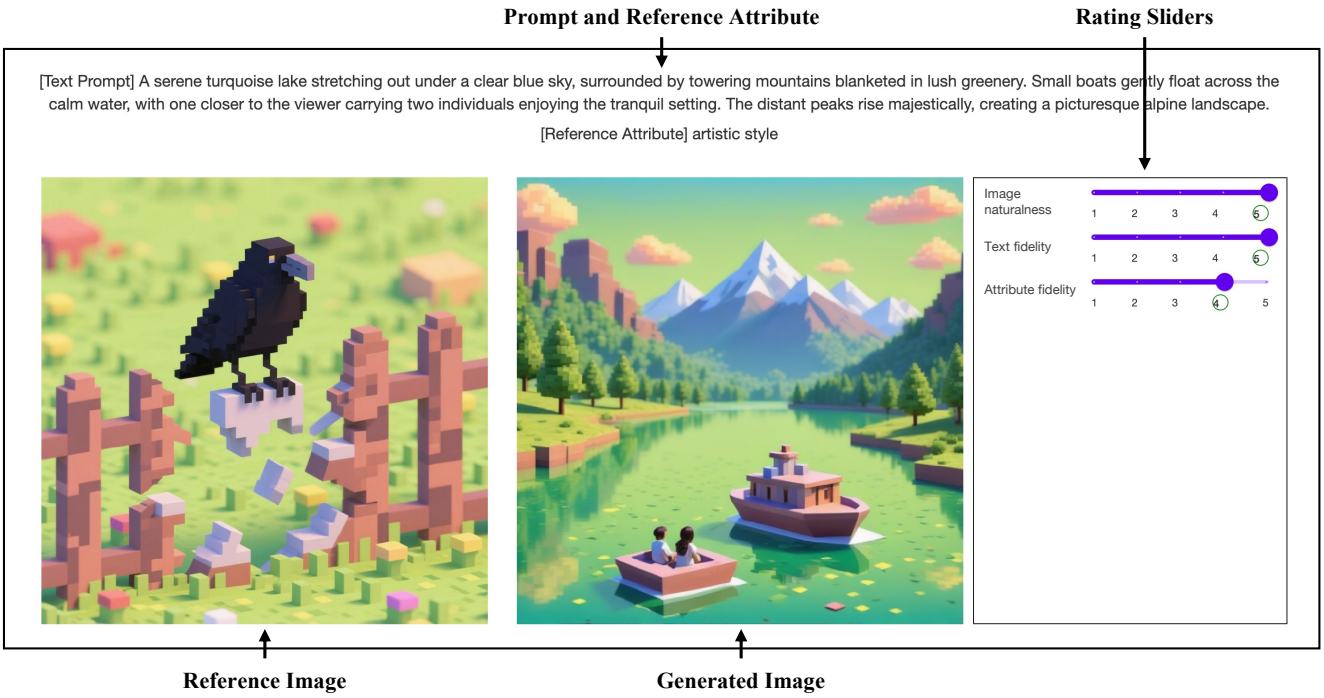

Figure 13. **Interface of the user study.** Given the input conditions (top and right) and the generated image (center), participants are asked to rate three aspects: *image naturalness*, *text fidelity*, and *attribute fidelity* on a 1 (poor) to 5 (excellent) scale using the sliders (left).

B.2. Attribute-oriented Image Retrieval

Since there is no existing model directly supporting attribute-oriented image retrieval, we construct a text-guided baseline using GPT-4o [47] and CLIP [54]. Specifically, we first prompt GPT-4o to generate descriptive texts of approximately 60 words for each target attribute. These descriptions are then converted into text embeddings using CLIP, which are subsequently used to retrieve the most semantically similar images corresponding to the given attribute.

C. Additional Results

Fig. 1(a) illustrates that *Omni-Attribute* can extract high-fidelity, attribute-specific information while suppressing irrelevant visual details. This helps reduce "*copy-and-paste*" artifacts and leads to a more coherent synthesis of the user-specified attribute in new contexts. Additional results demonstrating such attribute disentanglement are shown in Fig. 14.

To further showcase the practical utility of *Omni-Attribute*, we design four real-world application scenarios: (i) advertisement image synthesis, (ii) hairstyle customization, (iii) storytelling visualization, and (iv) creative content generation. The corresponding results are shown in Fig. 15.

D. Limitations

We identify the following limitations of our work:

Attribute-specific Embeddings. Our attribute embeddings are designed to capture image information related to one or a few specific attributes. This inherently constrains the applicability to tasks such as image editing, where most of the visual content must remain unchanged, and only a limited set of attributes should be modified.

Entanglement of Correlated Attributes. We observe that the model occasionally struggles to disentangle attributes that are often correlated, such as *person identity* and *hairstyle*. For example, as illustrated in Fig. 1, while we attempt to transfer the *identity* of *Vincent van Gogh* to new contexts, the generated images mostly preserve his hairstyle, indicating information leakage. One potential solution is to increase the sampling weight of the *hairstyle dataset* (as depicted in Fig. 10(d)) to better learn how to separate these factors. However, it remains an open question whether certain attributes can ever be perfectly disentangled (*e.g.*, whether *hairstyle* is inherently part of *identity*).

Sensitivity to Contrastive Learning Hyperparameters. Prior contrastive learning studies [7, 8] noted that the hyperparameters of contrastive loss, such as temperature, typically have a strong and dataset-dependent impact on model performance. In this work, we also notice that the selection of these hyperparameters has a huge impact on the quality of the learned attribute embeddings, as shown in Tab. 1.

We leave the study of these limitations for future work.

Figure 14. **Additional results of attribute disentanglement.** Each row shows three generated images (right), which are conditioned on the same reference image (left) and the same textual prompt, but with different attribute inputs (colored boxes). As seen, given the same reference image, *Omni-Attribute* effectively extracts attribute-specific representations, enabling the coherent synthesis of the user-specified attribute in new contexts while reducing the leakage of irrelevant visual information from the reference image.

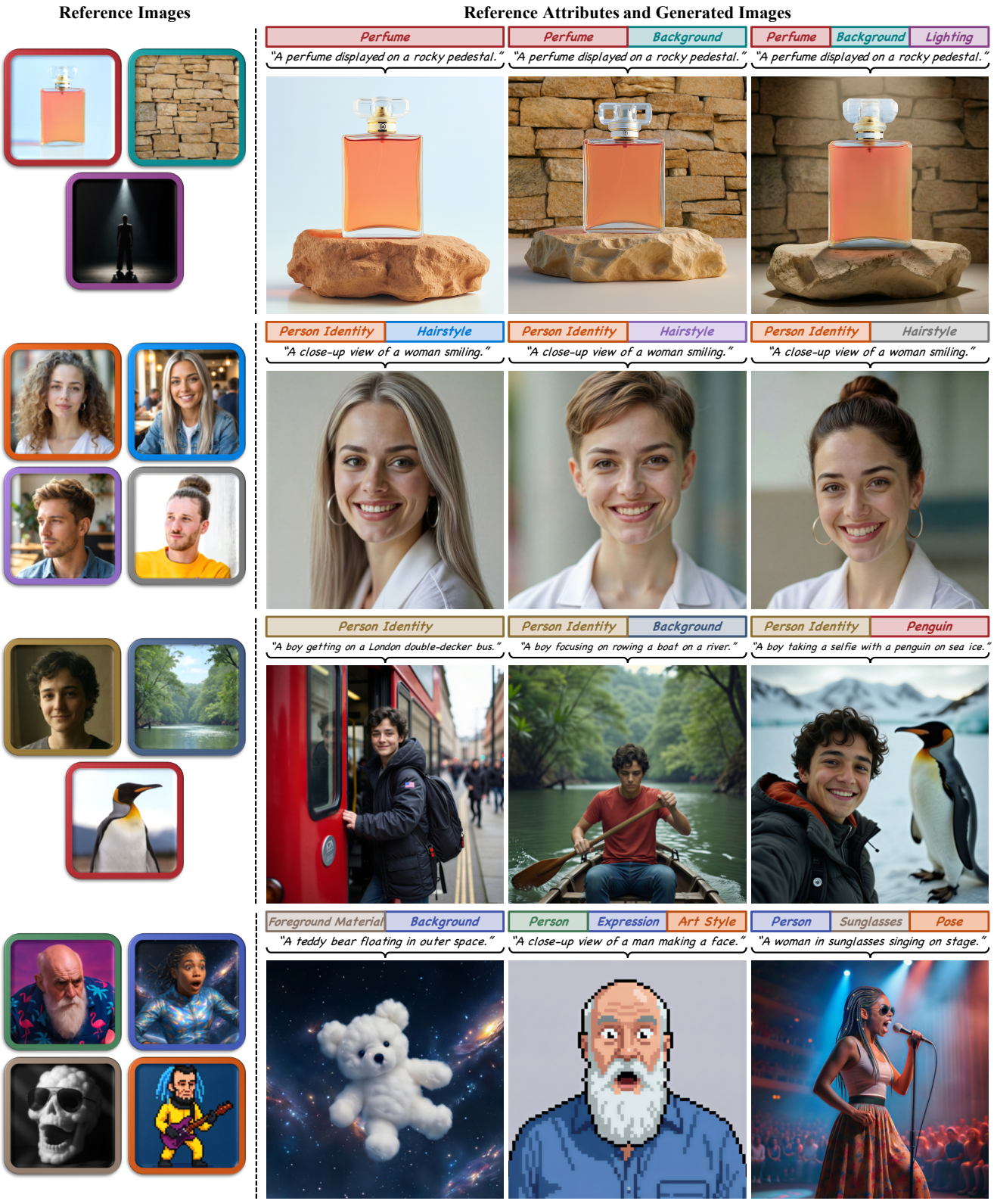

Figure 15. **Practical and creative applications of *Omni-Attribute*.** From top to bottom, each row demonstrates the practical utility of *Omni-Attribute* across four real-world applications: (i) advertisement image synthesis, (ii) hairstyle customization, (iii) storytelling visualization, and (iv) creative content generation.