Pacific Graphics 2024
R. Chen, T. Ritschel, and E. Whiting
(Guest Editors)

Volume 43 (2024), Number 7

CustomSketching: Sketch Concept Extraction for Sketch-based Image Synthesis and Editing
Supplemental Material

Chufeng Xiao[1,2] and Hongbo Fu[3,†]

[1] HKGAI, Hong Kong University of Science and Technology, Hong Kong, China
[2] School of Creative Media, City University of Hong Kong, Hong Kong, China
[3] Division of Arts and Machine Creativity, Hong Kong University of Science and Technology, Hong Kong, China
† Corresponding Author (hongbofu@ust.hk)

1. Dataset

Figure 22 shows a thumbnail of our created dataset, covering diverse object categories. For each object regarded as one concept, we invited three normal users without any professional training in drawing to trace separate contour lines S_C and details lines S_D over the reference images. One training sketch traced over an image generally cost 30s-2min for an amateur, while a testing one cost less than 1min. The sketch-image pairs are with purple borders in Figure 22. Note that each concept has 1-6 image-sketch pair(s) for training, where the concepts of human portrait and clothing only have a single pair. Then, the users were asked to create 3-5 edited dual sketches (with yellow borders in Figure 22) initialized from one of the traced sketches or drawn from scratch. In this way, we created the concepts with different fine-grained attributes (shape, pose, details) from the reference images, represented by the edited sketches. For each traced or edited sketch, we used a polygon filling method (implemented via OpenCV v3) to automatically generate a foreground mask following S_C. The automatically generated masks were generally accurate but the annotators were allowed to manually refine the masks if necessary. Finally, we obtained 35 groups of concept data, with 102 traced sketches with paired images, 159 edited sketches, as well as foreground masks corresponding to both sketches. Similar to [AAF*23], we set up ten prompt templates with the learned textual token $[v]$ as follows:

- *"A photo of* $[v]$ *at the beach"*
- *"A photo of* $[v]$ *in the jungle"*
- *"A photo of* $[v]$ *in the snow"*
- *"A photo of* $[v]$ *in the street"*
- *"A photo of* $[v]$ *on top of a wooden floor"*
- *"A photo of* $[v]$ *with a city in the background"*
- *"A photo of* $[v]$ *with a mountain in the background"*
- *"A photo of* $[v]$ *with the Eiffel tower in the background"*
- *"A photo of* $[v]$ *floating on top of water"*
- *"A photo of* $[v]$ *in an office"*

Therefore, we have 2,610 = (102+159)×10 sketch-text pairs for evaluation.

2. Implementation Details

Our method and all the compared baselines were based on Stable Diffusion v1.5 [RBL*22]. A training image and its corresponding sketch were both resized to 512×512. The sketch features extracted from a sketch encoder $\mathcal{F}$ were injected into four layers of the encoder of the denoising U-net, with resolutions of 64, 32, 16, and 8, following the settings of [MWX*23]. For the optimization of Stage I, we only fine-tuned a newly added textual token $[v]$ with a learning rate of $5e^{-4}$. The token was initialized using the class name of the target concept, e.g., "toy" for the toy object. The sketch encoder for Stage I is a pre-trained model (*t2iadapter_sketch_sd15v2*) from [MWX*23] with frozen weights during optimization. For Stage II, we jointly optimized the token $[v]$ and two sketch encoders with a small learning rate of $2e^{-6}$, similar to [AAF*23]. The weights of the two sketch encoders were initialized with those of the pre-trained one [MWX*23] used in Stage I. During training, a text prompt as input was randomly selected from the list of text templates in [GAA*22], while during testing, the prompt was picked from our created dataset. Empirically, we trained each stage in our experiment for 400 steps (batch size=16) using the Adam solver via the PyTorch framework. We randomly augmented (with the probability of 0.5) the training data by translating each sketch-image pair in the range of [-0.2,0.2], rotating it in the range of [-45°,45°], and horizontal flip. We trained and tested our method *CustomSketching* on a PC with Intel i9-13900K, 128GB RAM, and a single NVIDIA GeForce RTX 4090. The two-stage optimization took around 30 mins, while one pass inference (DDPM sampling with 50 steps) cost around 3s.

We used cross-attention maps in each layer of the denoising U-Net to compute shape loss $\mathcal{L}_{shape}$. Following Hertz et al. [HMT*22], we combined and averaged all the cross-attention maps $A_\theta(z_t, v)$ of the token $[v]$. The different layers of the attention maps with diverse resolutions were resized to 16×16 for computation.

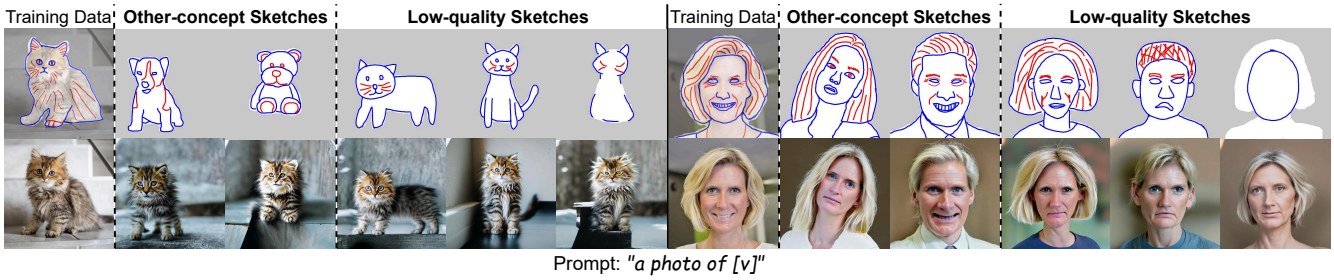

Figure 14: *Diverse results given different sketches with the same text prompt and sampling seed.*

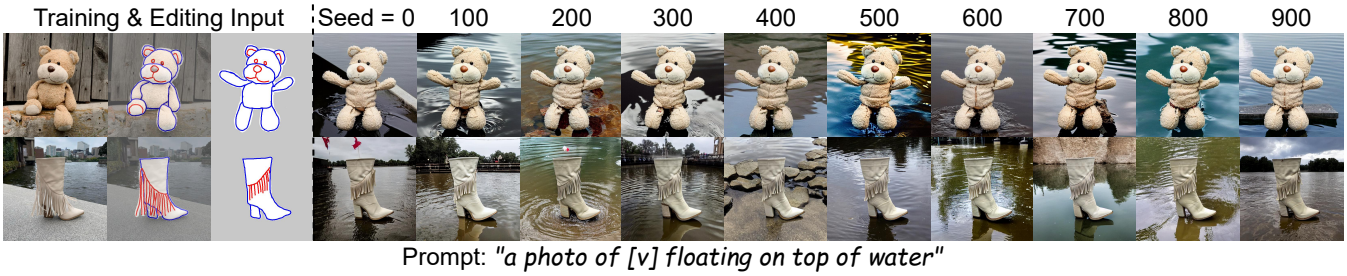

Figure 15: *Multiple random seeds with the same text and sketch for sampling diverse results.*

3. Experiments

3.1. Robustness Evaluation

We show the robustness of our method from two aspects: **1) Inputing sketches different from the training samples.** Our method can effectively avoid the T2I-adapter overfitting on training sketches, thanks to our two-stage optimization for embedding global semantics via text and local features via sketch into the pre-trained model. Thus, our method can tolerate sketches significantly from the training data. This is why our method can be successfully applied to concept transfer (see Main-text Figure 9 (b) & Figure 18). Figure 14 shows more results given two cases of different sketches, i.e., sketches from other concepts and low-quality sketches (even partial sketches). **2) Multiple Random Seeds.** We show diverse results given multiple random seeds with the same text and sketch (Figure 15). Since the foreground object is conditioned on the text and sketch, denoising with different seeds mainly varies the background generation, and our method can perform stable to make sure the foreground object is always faithful to the sketch given diverse seeds.

3.2. Comparisons with SOTAs

In the main text, we adapted DB [RLJ*23] and TI [GAA*22] with a pre-trained sketch encoder [MWX*23] to fit the task of sketch concept extraction, refer to DB-E and TI-E. Since DB learned a novel concept by binding a unique identifier (e.g., *"sks"*) with a specific subject in a text prompt, we provided a text prompt like *"a photo of a sks toy"* for the toy category for training and testing. Note that the weights of the sketch encoder in DB-E and TI-E were frozen to keep the two methods intact mostly. In the Supp, we further adapted DB and TI with two learnable sketch encoders fed with the dual-sketch representation as ours did, respectively referring to DB-FE and TI-FE. Considering vanilla DB might have enough capacity to learn a concept without sketch condition, we also separately compared our method with vanilla DB (denoted as DB/E), i.e., training vanilla DB for one concept and testing it with a pre-trained T2I-adapter (without fine-tuning). Fig. 16 shows two evaluation results on the sketch with only S_C (DB/E (S_C)) and the sketch with both types (DB/E (S)). It can be easily found that DB/E fails to correctly reconstruct the concept without sufficient sketch constraint and edit the concept using detail strokes due to the domain gap existing in the pre-trained sketch encoder. The above tuning-based methods (DB/E, DB-FE, DB-E, TI-FE, TI-E) had the same training parameters and augmentation tricks as ours.

For tuning-free methods, we compared our method with MS-E [CWQ*23] in the main text, but we found it often drifted the original style of the reference images due to the gap between the generated images and real images. A follow-up work, RIVAL [ZXLJ23], was proposed to alleviate such a gap. RIVAL employed a pre-trained ControlNet [ZRA23] to enable sketch-based editing for real images. We also compared our method with the sketch-based version of RIVAL (denoted as RIVAL-E) by directly using their released code. The tuning-based methods consist of an inversion step and an inference step. For the inversion step, we provided a reference image with the traced sketch and a text prompt (e.g., *"a photo of a toy"* for the toy category), while for the inference step, we provided an edited sketch with a target prompt (e.g., *"a photo of a toy in the snow"*). Note that for the tuning-free methods, we used the single-sketch representation for the sketch input to make the method compatible with the prior of the pre-trained sketch en-

Table 1: *Quantitative comparisons for diverse methods.*

Method	Prompt ↑	Identity ↑	Perceptual ↓
DB/E (S_C)	**0.647**	0.868	0.202
DB/E (S)	**0.647**	0.870	0.196
DB-FE	0.642	0.879	0.192
DB-E	0.641	0.889	0.182
TI-FE	0.634	0.906	0.165
TI-E	0.642	0.867	0.214
RIVAL-E	0.627	0.899	0.151
MS-E	0.633	0.884	0.16
Single-encoder	0.623	0.910	0.142
Single-sketch	0.622	0.908	0.146
w/o $\mathcal{L}_{shape}$	0.639	0.906	0.150
w/o $\mathcal{L}_{reg}$	0.618	0.909	0.142
w/o Masked $\mathcal{F}$	0.620	0.911	0.141
w/o Stage I	0.632	0.904	0.164
Ours	0.632	**0.912**	**0.134**

coder. We used the same random seed (seed=42) for our method and all the above baselines during inference.

Figure 23 shows more qualitative comparisons. It demonstrates that our method performs better in sketch- and text-based editing while preserving the annotated object's original identity compared to all the baselines. DB-FE, TI-FE, and RIVAL-E can improve the reconstruction quality a little in appearance and geometry, respectively compared to DB-E, TI-E, and MS-E. However, the three methods still could not achieve satisfactory editing results. The quantitative results could also reflect such a tendency (see Table 1).

3.3. Ablation Study

Figure 24 shows more results for comparisons between our method and the ablated ones mentioned in the main text. We show two more ablated variants here: 1) adopting a single encoder in Stage II with the dual-sketch representation, i.e., merging S_C and S_D into one sketch map as input; 2) w/o Stage I, i.e., only jointly optimizing a newly added token and the two sketch encoders. As shown in Figure 24 and Table 1, the single-encoder setting could not effectively differentiate shape and details, thus causing worse sketch faithfulness and identity preservation than ours. Removing Stage I results in unsatisfactory reconstruction since the setting would mislead the optimization in disentangling the global semantics into $[v]$ and local features into $\mathcal{F}$. Table 1 also confirms such a conclusion (see the identity similarity and perceptual distance).

4. Applications

We implemented four applications enabled by our *CustomSketching*. Below, we show more results and the implementation details.

Local Editing. Incorporating [AFL23], our method can be applied to local image editing, which allows users to edit a local region of a given real image via sketching while keeping the unedited region intact. Figure 17 presents additional local editing results for human portrait manipulation (Top) and virtual try-on/clothing design (Bottom).

Concept Transfer. Our method can transfer the learned concepts locally or globally to a target object with similar semantics, as shown in Figure 18. Similar to the pipeline of local editing, users may provide an editing input to indicate local shape or structure to transfer a target concept $[S]$.

Multi-concept Generation. Given a set of the extracted sketch concepts $\{S_i\}=\{[v_i],\mathcal{F}_i\}$, our method can directly combine them without extra optimization. Figure 19 shows the pipeline of multi-concept generation implemented by our method. Given an input sketch annotated with diverse concepts, our method divides it into separate sketches fed into their corresponding dual-encoder F_i. Then, the extracted features are masked respectively using M_i and then aggregated together by summation, finally injected into the pre-trained T2I diffusion model. The given prompt is in the format of "$[v_1]$ and $[v_2]$ ... and $[v_i]$" to cover multiple concepts. Figure 20 shows more results of multi-concept generation.

Text-based Style Variation. Our method decouples global semantics and local features of a reference image to a textual token $[v]$ and a sketch encoder $\mathcal{F}$. Thus, our method can be used to produce diverse style variations of the target object while preserving its geometry (shape and details), as shown in Figure 21. To this end, our method first extracts a concept $[S]=\{[v],\mathcal{F}\}$ from sketch-image pair(s). Then, it takes as input the sketch (regarded as an intermediate representation of object geometry) fed to $\mathcal{F}$ and a style prompt without the learned $[v]$ from the original image (e.g., "a crayon drawing") to control the target style. We compared our method with PnP [TGBD23], a text-based image-to-image translation method in two cases, i.e., inputting the image without and with masking out the background, given a style prompt. PnP consists of an inversion step and an inference step. For comparison, we provided an initial prompt (e.g., "a photo of a toy" for the toy category) for inversion and a style prompt (e.g., "a crayon drawing of a toy") for inference to change the object style. Thanks to the extraction of a novel sketch concept, our method can better disentangle the geometry (depicted by a sketch) and style (depicted by a text), thus offering more user controllability and flexibility via sketching.

References

[AAF*23] Avrahami O., Aberman K., Fried O., Cohen-Or D., Lischinski D.: Break-a-scene: Extracting multiple concepts from a single image. *arXiv preprint arXiv:2305.16311* (2023). 1

[AFL23] Avrahami O., Fried O., Lischinski D.: Blended latent diffusion. *ACM Transactions on Graphics (TOG) 42*, 4 (2023), 1–11. 3

[CWQ*23] Cao M., Wang X., Qi Z., Shan Y., Qie X., Zheng Y.: Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (October 2023), pp. 22560–22570. 2

[GAA*22] Gal R., Alaluf Y., Atzmon Y., Patashnik O., Bermano A. H., Chechik G., Cohen-or D.: An image is worth one word: Personalizing text-to-image generation using textual inversion. In *International Conference on Learning Representations* (2022). 1, 2

[HMT*22] Hertz A., Mokady R., Tenenbaum J., Aberman K., Pritch Y., Cohen-or D.: Prompt-to-prompt image editing with cross-attention control. In *International Conference on Learning Representations* (2022). 1

[MWX*23] Mou C., Wang X., Xie L., Zhang J., Qi Z., Shan Y., Qie X.: T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453* (2023). 1, 2

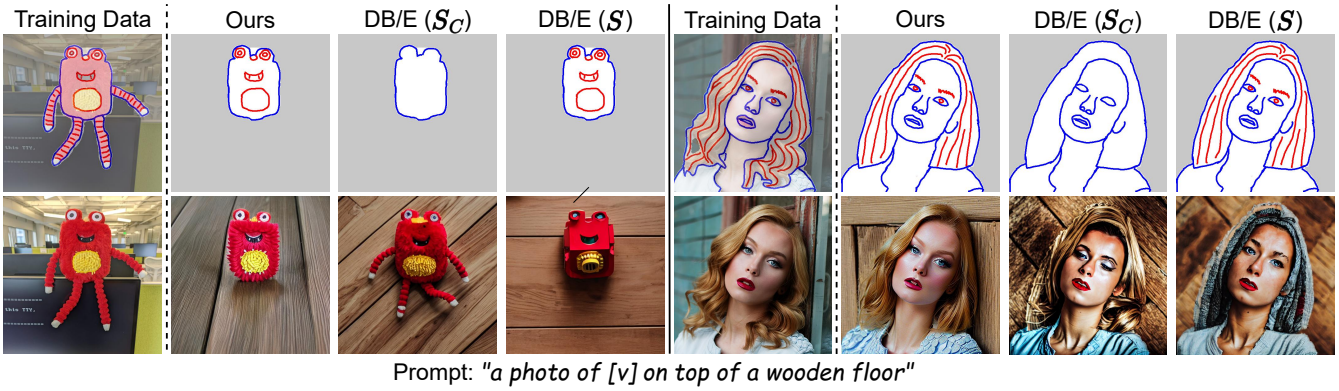

Figure 16: *Qualitative comparison between ours and vanilla DB with a pre-trained sketch T2I-adapter (DB/E). The results show DB/E fails in correctly reconstruction and editing the reference image.*

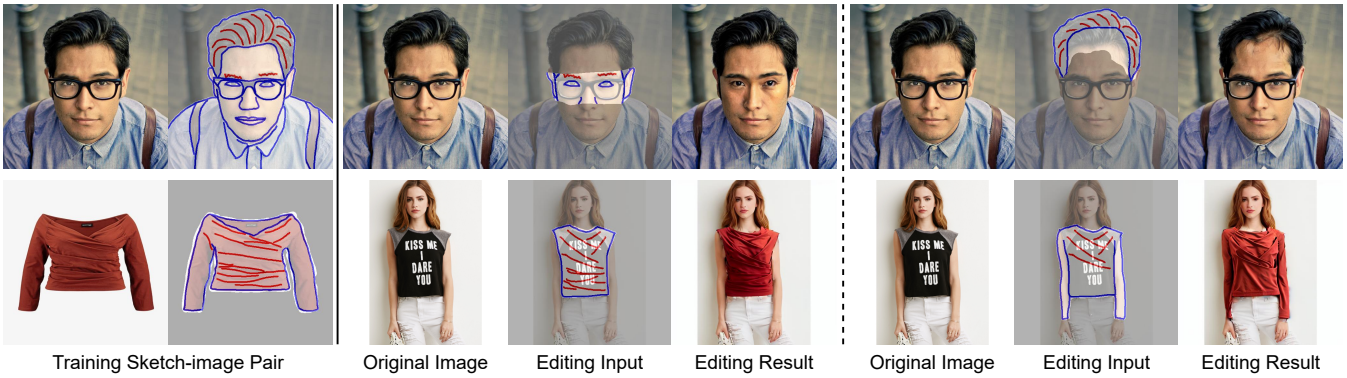

Figure 17: *Additional results of local editing enabled by our method. The top row is for human portrait manipulation (removing the glasses and changing the hair region), while the bottom row is for virtual try-on and clothing design.*

[RBL*22] ROMBACH R., BLATTMANN A., LORENZ D., ESSER P., OMMER B.: High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 10684–10695. 1

[RLJ*23] RUIZ N., LI Y., JAMPANI V., PRITCH Y., RUBINSTEIN M., ABERMAN K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 22500–22510. 2

[TGBD23] TUMANYAN N., GEYER M., BAGON S., DEKEL T.: Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2023), pp. 1921–1930. 3, 6

[ZRA23] ZHANG L., RAO A., AGRAWALA M.: Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), pp. 3836–3847. 2

[ZXLJ23] ZHANG Y., XING J., LO E., JIA J.: Real-world image variation by aligning diffusion inversion chain. *Advances in Neural Information Processing Systems* (2023). 2

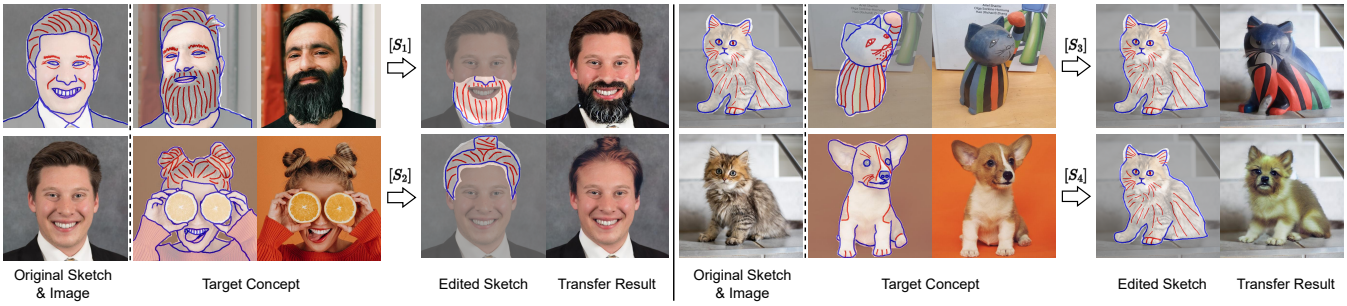

Figure 18: *Additional results of concept transfer enabled by our method. The left half shows examples of local concept transfer for adding a beard (Top) and adding a hair bun (Bottom). The right half shows examples of global concept transfer for changing the object semantics while preserving its shape and pose.*

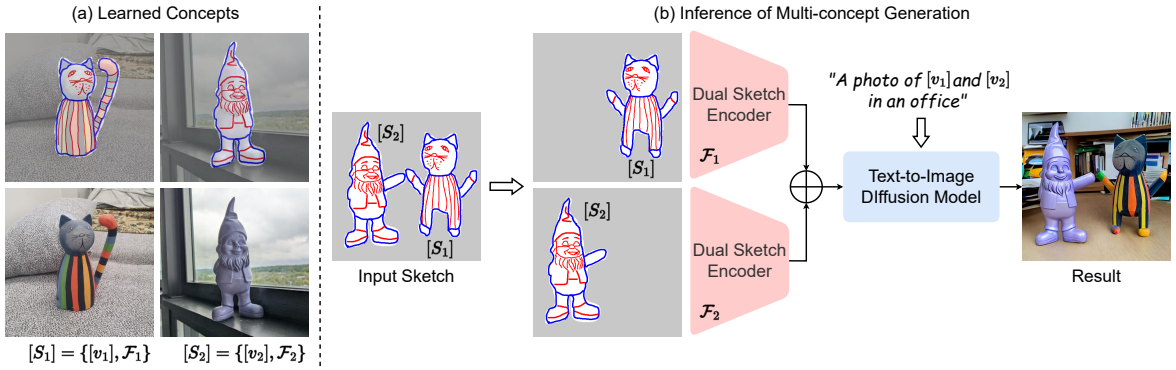

Figure 19: *The pipeline of multi-concept generation enabled by our method. Separately learning each concept (a), our method can directly combine them for multi-concept generation during inference (b) without extra fine-tuning.*

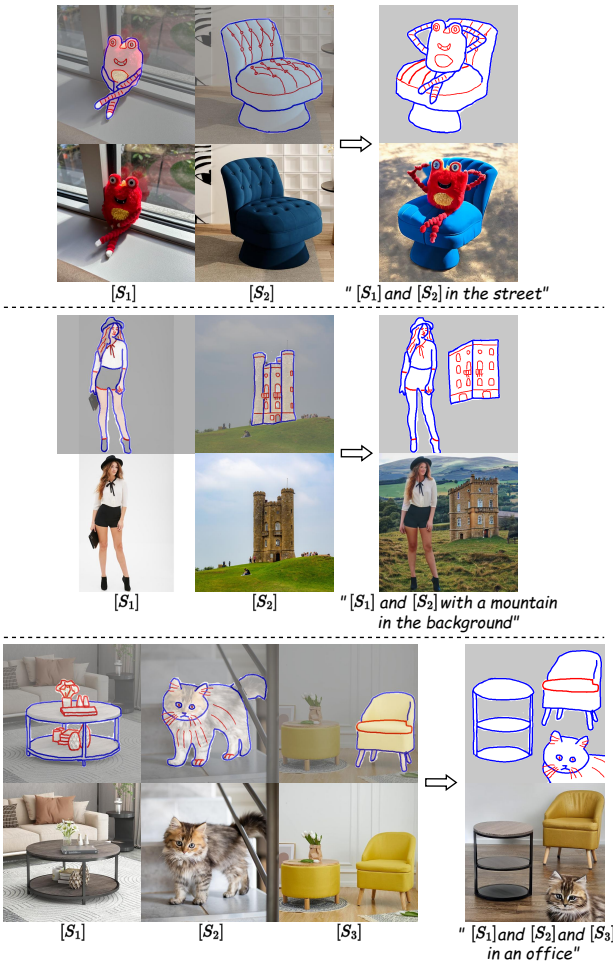

Figure 20: *Additional results of multi-concept generation enabled by our method. The prefix of the text prompt is "a photo of ...".*

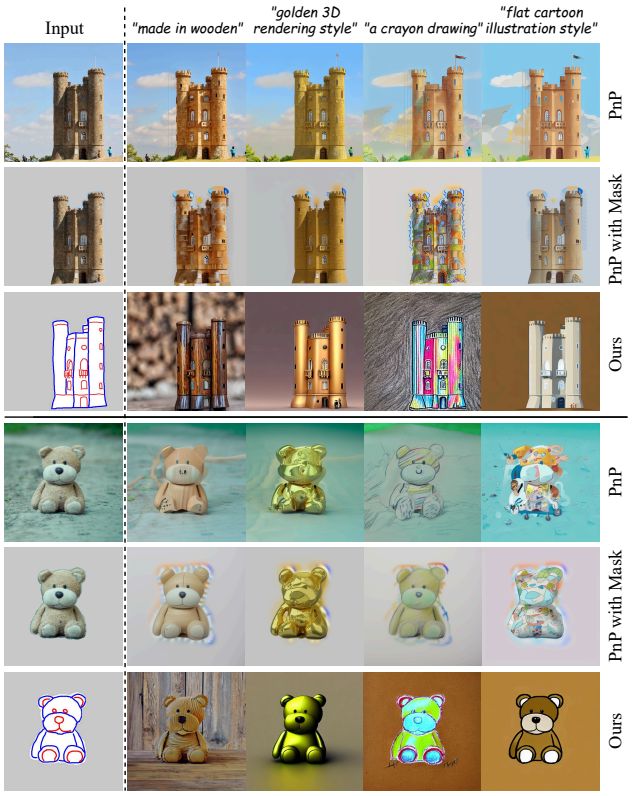

Figure 21: *Comparisons of the results by our method and PnP [TGBD23] for text-based style variation.*

Figure 22: *A thumbnail of our created dataset for training and testing. The pairs of reference images and the corresponding traced sketches are with purple borders, while the edited sketches are with yellow borders.*

Figure 23: *Comparisons of the results generated by our method and the adapted state-of-the-art methods, given the same training data (sketch-image pairs in Columns 1 & 2), edited sketch (Column 3), and text prompt (at the bottom of each group of results).*

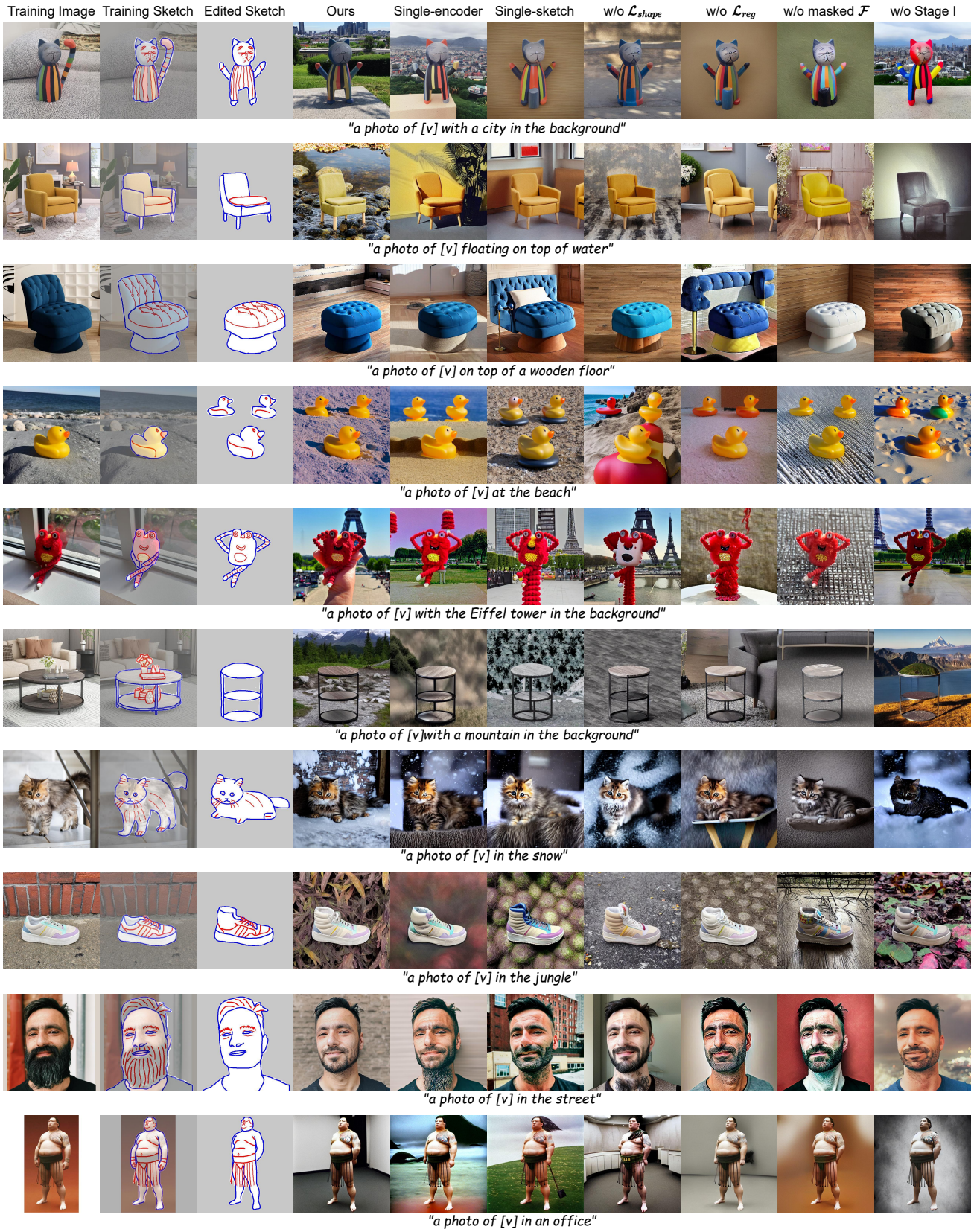

Figure 24: *Comparisons of the results generated by our method and the ablated variants, given the same training data (sketch-image pairs in Columns 1 & 2), edited sketch (Column 3), and text prompt (at the bottom of each group of results).*