# Supplementary Material for DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation

## Background

**Text-to-Image Diffusion Models** Diffusion models are probabilistic generative models that are trained to learn a data distribution by the gradual denoising of a variable sampled from a Gaussian distribution. Specifically, this corresponds to learning the reverse process of a fixed-length Markovian forward process. In simple terms, a conditional diffusion model $\hat{\mathbf{x}}_\theta$ is trained using a squared error loss to denoise a variably-noised image $\mathbf{z}_t := \alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}$ as follows:

$$\mathbb{E}_{\mathbf{x},\mathbf{c},\boldsymbol{\epsilon},t}\big[w_t \|\hat{\mathbf{x}}_\theta(\alpha_t \mathbf{x} + \sigma_t \boldsymbol{\epsilon}, \mathbf{c}) - \mathbf{x}\|_2^2\big] \tag{1}$$

where $\mathbf{x}$ is the ground-truth image, $\mathbf{c}$ is a conditioning vector (e.g., obtained from a text prompt), $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a noise term and $\alpha_t, \sigma_t, w_t$ are terms that control the noise schedule and sample quality, and are functions of the diffusion process time $t \sim \mathcal{U}([0,1])$. At inference time, the diffusion model is sampled by iteratively denoising $\mathbf{z}_{t_1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ using either the deterministic DDIM [10] or the stochastic ancestral sampler [3]. Intermediate points $\mathbf{z}_{t_1}, \dots, \mathbf{z}_{t_T}$, where $1 = t_1 > \cdots > t_T = 0$, are generated, with decreasing noise levels. These points, $\hat{\mathbf{x}}_0^t := \hat{\mathbf{x}}_\theta(\mathbf{z}_t, \mathbf{c})$, are functions of the $\mathbf{x}$-predictions.

Recent state-of-the-art text-to-image diffusion models use cascaded diffusion models in order to generate high-resolution images from text [7, 9]. Specifically, [9] uses a base text-to-image model with 64x64 output resolution, and two text-conditional super-resolution (SR) models $64 \times 64 \rightarrow 256 \times 256$ and $256 \times 256 \rightarrow 1024 \times 1024$. Ramesh et al. [7] use a similar configuration, with unconditional SR models. A key component of high-quality sample generations from [9] is the use of noise conditioning augmentation [4] for the two SR modules. This consists in corrupting the intermediate image using noise with specific strength, and then conditioning the SR model on the level of corruption. Saharia et al. [9] select Gaussian noise as the form of augmentation.

Other recent state-of-the-art text-to-image diffusion models, such as Stable Diffusion [8], use a single diffusion model to generate high-resolution images. Specifically, the forward and backward diffusion processes occur in a lower-dimensional latent space and an encoder-decoder architecture is trained on a large image dataset to translate images into latent codes. At inference time, a random noise latent code goes through the backward diffusion process and the pre-trained decoder is used to generate the final image. Our method can be naturally applied to this scenario, where the U-Net (and possibly the text encoder) are trained, and the decoder is fixed.

**Vocabulary Encoding** The details of text-conditioning in text-to-image diffusion models are of high importance for visual quality and semantic fidelity. Ramesh et al. [7] use CLIP text embeddings that are translated into image embeddings using a learned prior, while Saharia et al. [9] use a pre-trained T5-XXL language model [6]. In our work, we use the latter. Language models like T5-XXL generate embeddings of a tokenized text prompt, and vocabulary encoding is an important pre-processing step for prompt embedding. In order to transform a text prompt $\mathbf{P}$ into a conditioning embedding $\mathbf{c}$, the text is first tokenized using a tokenizer $f$ using a learned vocabulary. Following [9], we use the SentencePiece tokenizer [5]. After tokenizing a prompt $\mathbf{P}$ using tokenizer $f$ we obtain a fixed-length vector $f(\mathbf{P})$. The language model $\Gamma$ is conditioned on this token identifier vector to produce an embedding $\mathbf{c} := \Gamma(f(\mathbf{P}))$. Finally, the text-to-image diffusion model is directly conditioned on $\mathbf{c}$.

## Dataset

Our dataset includes 30 subjects. We separate each subject into two categories: objects and live subjects/pets. 21 of the 30 subjects are objects, and 9 are live subjects/pets. We provide one sample image for each of the subjects in Figure 1. Images for this dataset were collected by the authors or sourced from Unsplash [1].

We also collected 25 prompts: 20 recontextualization prompts and 5 property modification prompts for objects. 10 recontextualization, 10 accessorization, and 5 property modification prompts for live subjects/pets. Prompts are shown in Figure 2

For the evaluation suite we generate four images per subject and per prompt, totaling 3,000 images. This allows us to robustly measure performances and generalization capabilities of a method. We make our dataset and evaluation protocol publicly available on the project webpage for future use in evaluating subject-driven generation.

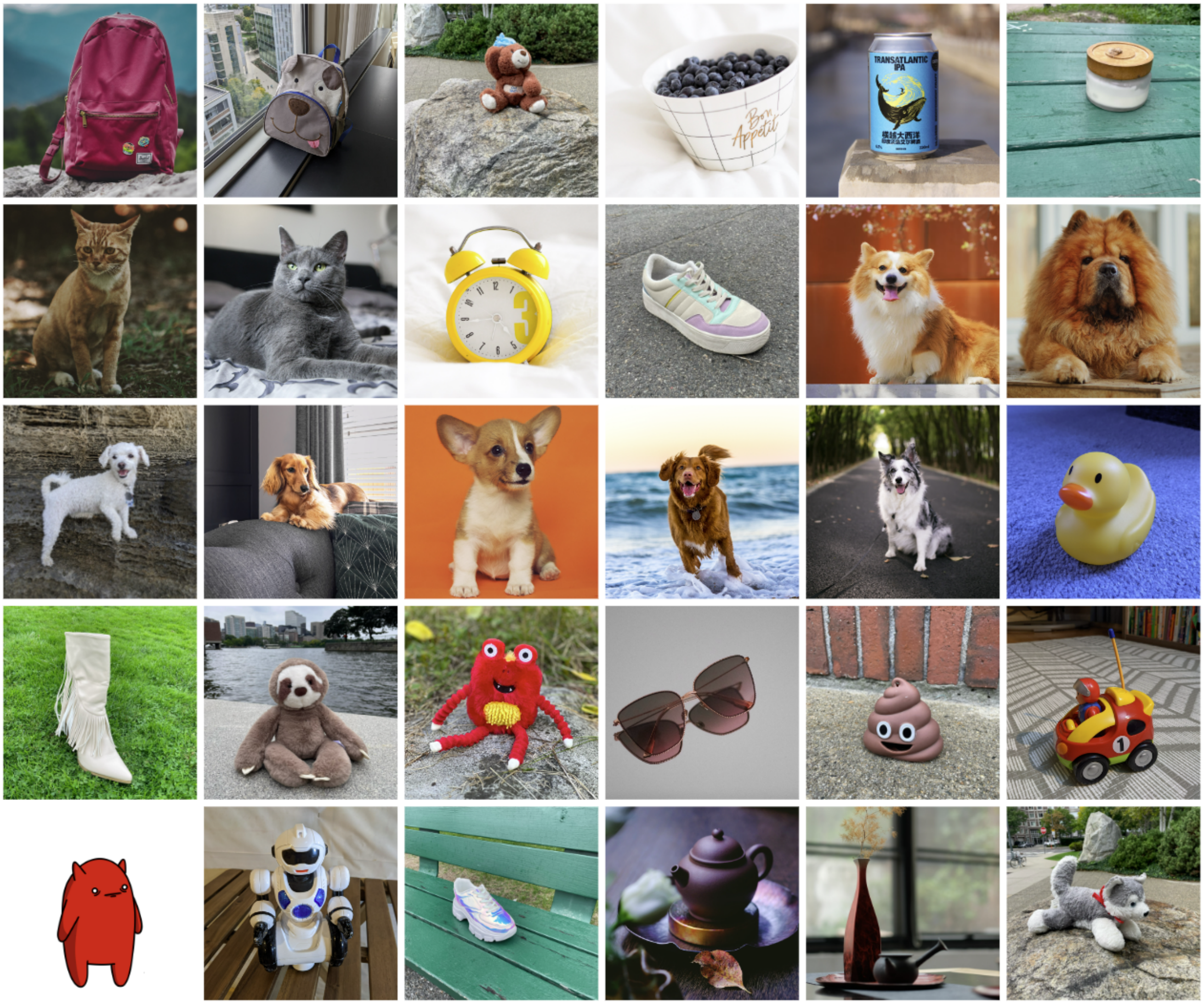
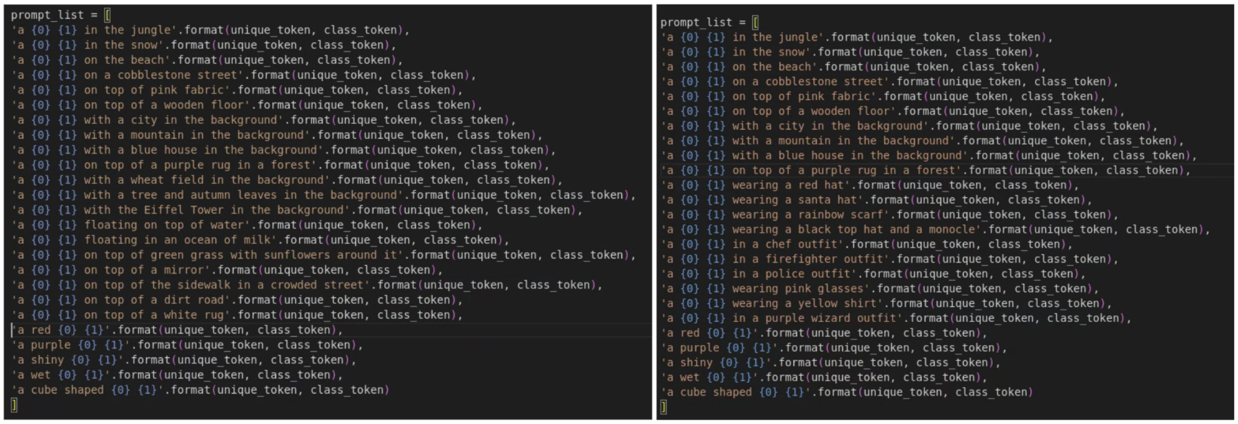


Figure 1. **Dataset**. Example images for each subject in our proposed dataset.

## Subject Fidelity Metrics

In the main paper we comment on the superiority of our proposed DINO metric in terms of subject fidelity. We hypothesize that this is because DINO is, in essence, trained in a self-supervised manner to distinguish different images from each other modulo data augmentations. This is in contrast to the CLIP-I metric, where CLIP is trained with text-image pairs and encodes more descriptive information about images - but not necessarily fine details that are not present in the text annotations. We give an example in Figure 3, where the first column contains a reference real image, the second column a different real image, the third column a DreamBooth generated image and the last column an image generated using Textual Inversion. We compare the 2nd, 3rd and 4th image to the real reference image using the CLIP-I and DINO metrics. We observe that the 2nd real image obtains both the highest CLIP-I and DINO scores. The DreamBooth sample looks much more similar to the reference sample than the Textual Inversion sample, yet the CLIP-I score for the Textual Inversion sample is much higher than the DreamBooth sample. However, we can see that the DINO similarity is higher for the DreamBooth sample - which more closely tracks human evaluation of subject fidelity. In order to quantitatively test this, we compute correlations between DINO/CLIP-I scores and normalized human preference scores. DINO has a Pearson correlation coeff. of 0.32 with human preference (vs. 0.27 for the CLIP-I metric used in [20]), with a very low p-value of $9.44 \times 10^{-30}$.



```
prompt_list = [
'a {0} {1} in the jungle'.format(unique_token, class_token),
'a {0} {1} in the snow'.format(unique_token, class_token),
'a {0} {1} on the beach'.format(unique_token, class_token),
'a {0} {1} on a cobblestone street'.format(unique_token, class_token),
'a {0} {1} on top of pink fabric'.format(unique_token, class_token),
'a {0} {1} on top of a wooden floor'.format(unique_token, class_token),
'a {0} {1} with a city in the background'.format(unique_token, class_token),
'a {0} {1} with a mountain in the background'.format(unique_token, class_token),
'a {0} {1} with a blue house in the background'.format(unique_token, class_token),
'a {0} {1} on top of a purple rug in a forest'.format(unique_token, class_token),
'a {0} {1} with a wheat field in the background'.format(unique_token, class_token),
'a {0} {1} with a tree and autumn leaves in the background'.format(unique_token, class_token),
'a {0} {1} with the Eiffel Tower in the background'.format(unique_token, class_token),
'a {0} {1} floating on top of water'.format(unique_token, class_token),
'a {0} {1} floating in an ocean of milk'.format(unique_token, class_token),
'a {0} {1} on top of green grass with sunflowers around it'.format(unique_token, class_token),
'a {0} {1} on top of a mirror'.format(unique_token, class_token),
'a {0} {1} on top of the sidewalk in a crowded street'.format(unique_token, class_token),
'a {0} {1} on top of a dirt road'.format(unique_token, class_token),
'a {0} {1} on top of a white rug'.format(unique_token, class_token),
'a red {0} {1}'.format(unique_token, class_token),
'a purple {0} {1}'.format(unique_token, class_token),
'a shiny {0} {1}'.format(unique_token, class_token),
'a wet {0} {1}'.format(unique_token, class_token),
'a cube shaped {0} {1}'.format(unique_token, class_token)
]
```

Object Prompts

```
prompt_list = [
'a {0} {1} in the jungle'.format(unique_token, class_token),
'a {0} {1} in the snow'.format(unique_token, class_token),
'a {0} {1} on the beach'.format(unique_token, class_token),
'a {0} {1} on a cobblestone street'.format(unique_token, class_token),
'a {0} {1} on top of pink fabric'.format(unique_token, class_token),
'a {0} {1} on top of a wooden floor'.format(unique_token, class_token),
'a {0} {1} with a city in the background'.format(unique_token, class_token),
'a {0} {1} with a mountain in the background'.format(unique_token, class_token),
'a {0} {1} with a blue house in the background'.format(unique_token, class_token),
'a {0} {1} on top of a purple rug in a forest'.format(unique_token, class_token),
'a {0} {1} wearing a red hat'.format(unique_token, class_token),
'a {0} {1} wearing a santa hat'.format(unique_token, class_token),
'a {0} {1} wearing a rainbow scarf'.format(unique_token, class_token),
'a {0} {1} wearing a black top hat and a monocle'.format(unique_token, class_token),
'a {0} {1} in a chef outfit'.format(unique_token, class_token),
'a {0} {1} in a firefighter outfit'.format(unique_token, class_token),
'a {0} {1} in a police outfit'.format(unique_token, class_token),
'a {0} {1} wearing pink glasses'.format(unique_token, class_token),
'a {0} {1} wearing a yellow shirt'.format(unique_token, class_token),
'a {0} {1} in a purple wizard outfit'.format(unique_token, class_token),
'a red {0} {1}'.format(unique_token, class_token),
'a purple {0} {1}'.format(unique_token, class_token),
'a shiny {0} {1}'.format(unique_token, class_token),
'a wet {0} {1}'.format(unique_token, class_token),
'a cube shaped {0} {1}'.format(unique_token, class_token)
]
```

Live Subject Prompts

Figure 2. **Prompts**. Evaluation prompts for both objects and live subjects.

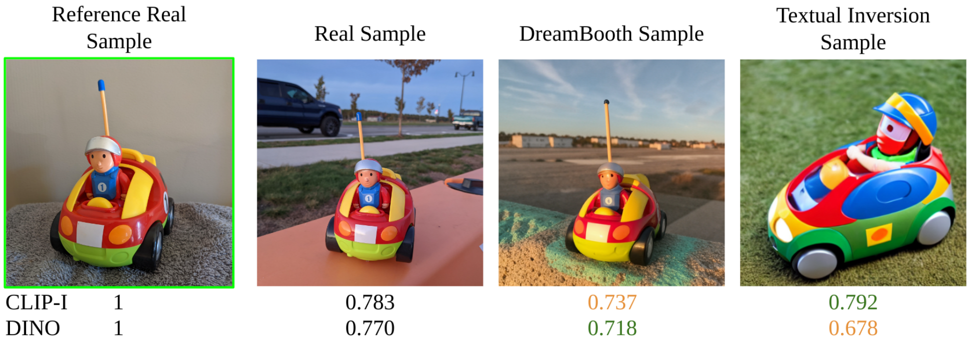


Figure 3. **CLIP-I vs. DINO Metrics.** The DreamBooth CLIP-I similarity to the reference image is lower than that of the Textual Inversion sample, even though the DreamBooth subject looks more similar to the reference subject. The DINO metric more closely tracks human evaluation of subject fidelity here.

## User Study

Below we include the full instructions used for our user study. For *subject fidelity*:

- Read the task carefully, inspect the reference items and then inspect the generated items.
- Select which of the two generated items (A or B) reproduces the identity (e.g. item type and details) of the reference item.
- The subject might be wearing accessories (e.g. hats, outfits). These should not affect your answer. Do not take them into account.
- If you're not sure, select Cannot Determine / Both Equally.

For *text fidelity*:

- Read the task carefully, inspect the reference text and then inspect the generated items.
- Select which of the two generated items (A or B) is best described by the reference text.
- If you're not sure, select Cannot Determine / Both Equally.

For each study we asked 72 users to answer questionnaires of 25 comparative questions (3 users per questionnaire), totaling 1800 answers - with 600 image pairs evaluated.

## Additional Applications and Examples

**Additional Samples** We provide a large amount of additional random samples in an annex HTML file. We compare real images, to DreamBooth generated images using Imagen and Stable Diffusion as well as images generated using Textual Inversion on Stable Diffusion.

**Recontextualization** We show additional high-quality examples of recontextualization in Figure 4.

**Art Renditions** We show additional examples of original artistic renditions of a personalized model in Figure 5.

**Expression Manipulation** Our method allows for new image generation of the subject with modified expressions that are not seen in the original set of subject images. We show examples in Figure 6. The range of expressiveness is high, ranging from negative to positive valence emotions and different levels of arousal. In all examples, the uniqueness of the subject dog is preserved - specifically, the asymmetric white streak on its face remains in all generated images.

**Novel View Synthesis** We show more viewpoints for novel view synthesis in Figure 7, along with prompts used to generate the samples.

**Accessorization** An interesting capability stemming from the strong compositional prior of the generation model is the ability to accessorize subjects. In Figure 8 we show examples of accessorization of a Chow Chow dog. We prompt the model with a sentence of the form: “a [V] [class noun] wearing [accessory]”. In this manner, we are able to fit different accessories onto this dog - with aesthetically pleasing results. Note that the identity of the dog is preserved in all frames, and subject-accessory contact and articulation are realistic.

**Property Modification** We are able to modify subject instance properties. For example we can include a color adjective in the prompt sentence “a [color adjective] [V] [class noun]”. In that way, we can generate novel instances of our subject with different colors. The generated scene can be very similar to the original scene, or it can be changed given a descriptive prompt. We show color changes of a car in the first row of Figure 9. We select similar viewpoints for effect, but we can generate different viewpoints of the car with different colors in different scenarios. This is a simple example of property modification, but more semantically complex property modifications can be achieved using our method. For example, we show crosses between a specific Chow Chow dog and different animal species in the bottom row of Figure 9. We prompt the model with sentences of the following structure: “a cross of a [V] dog and a [target species]”. In particular, we can see in this example that the identity of the dog is well preserved even when the species changes - the face of the dog has certain individual properties that are well preserved and melded with the target species. Other property modifications are possible, such as material modification (e.g. a dog made out of stone). Some are harder than others and depend on the prior of the base generation model.

| Method | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Backpack | 0.494 | 0.515 | 0.596 | **0.604** | 0.597 |
| Dog | 0.798 | 0.851 | 0.871 | **0.876** | 0.864 |

Table 1. Effect of the number of input images on subject fidelity (DINO).

**Comic Book Generation** In addition to photorealistic images, our method is able to capture the appearance of drawn media and more. In Figure 10 we present, to the best of our knowledge, the first instance of a full comic with a persistent character generated by a generative model. Each comic frame was generated using a descriptive prompt (e.g “a [V] cartoon grabbing a fork and a knife saying “time to eat””).

## Additional Experiments

### Prior Preservation Loss

Here we show qualitative examples of how our prior preservation loss (PPL) conserves variability in the prior and show sample results in Figure 11. We verify that a vanilla model is able to generate a large variety of dogs, while a naively fine-tuned model on the subject dog exhibits language drift and generates our subject dog given the prompt “a dog”. Our proposed loss preserves the variability of the prior and the model is able to generate new instances of our dog given a prompt of the style “a [V] dog” but also varied instances of dogs given a “a dog” prompt.

### Effect of Training Images

Here we run an experiment on the effects of the number of input images for model personalization. Specifically, we train models for two subjects, 5 models per subject with input images ranging from 1 to 5. We generate 4 images for 10 different recontextualization prompts for each subject. We present qualitative results in Figure 12. We can observe that for some subjects that are more common, and lie more strongly in the distribution of the diffusion model, such as the selected Corgi dog, we are able to accurately capture the appearance using only two images - and sometimes only one, given careful hyperparameter choice. For objects that are more rare, such as the selected backpack, we need more samples to accurately preserve the subject and to recontextualize it to diverse settings. Our quantitative results support these conclusions - we present the DINO subject fidelity metric in Table 1 and the CLIP-T prompt fidelity metric in Table 2. For both subjects we see that the optimal amount of input images for subject and prompt is 4. This number can vary depending on the subject and we settle on 3-5 images for model personalization.

| Method | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Backpack | 0.798 | 0.851 | 0.871 | **0.876** | 0.864 |
| Dog | 0.646 | 0.683 | 0.734 | **0.740** | 0.730 |

Table 2. Effect of the number of input images on prompt fidelity (CLIP-T).

## Personalized Instance-Specific Super-Resolution and Low-level Noise Augmentation for Imagen

While the text-to-image diffusion model controls for most visual semantics, the super-resolution (SR) models are essential to achieve photorealistic content and to preserve subject instance details. We find that if SR networks are used without fine-tuning, the generated output can contain artifacts since the SR models might not be familiar with certain details or textures of the subject instance, or the subject instance might have hallucinated incorrect features, or missing details. Figure 13 (bottom row) shows some sample output images with no fine-tuning of SR models, where the model hallucinates some high-frequency details. We find that fine-tuning the $64 \times 64 \rightarrow 256 \times 256$ SR model is essential for most subjects, and fine-tuning the $256 \times 256 \rightarrow 1024 \times 1024$ model can benefit some subject instances with high levels of fine-grained detail.

We find results to be suboptimal if the training recipes and test parameters of Saharia et al. [9] are used to fine-tune the SR models with the given few shots of a subject instance. Specifically, we find that maintaining the original level of noise augmentation used to train the SR networks leads to the blurring of high-frequency patterns of the subject and of the environment. See Figure 13 (middle row) for sample generations. In order to faithfully reproduce the subject instance, we reduce the level of noise augmentation from $10^{-3}$ to $10^{-5}$ during fine-tuning of the $256 \times 256$ SR model. With this small modification, We are able to recover fine-grained details of the subject instance. We show how using lower noise to train the super-resolution models improves fidelity. Specifically, we show in Figure 13 that if the super-resolution models are not fine-tuned, we observe hallucination of high-frequency patterns on the subject which hurts identity preservation. Further, if we use the ground-truth noise augmentation level used for training the Imagen $256 \times 256$ model ($10^{-3}$), we obtain blurred and non-crisp details. If the noise used to train the SR model is reduced to $10^{-5}$, then we conserve a large amount of detail without pattern hallucination or blurring.

## Comparisons

We include additional qualitative comparisons with Gal et al. [2] in Figure 14. For this comparison, we train our model on the training images of two objects appear in the teaser of their work (headless sculpture and cat toy) kindly provided by Gal et al. [2], and apply the prompts suggested in their paper. For prompts where they present several generated images, we handpicked their best sample (with the highest image quality and morphological similarity to the subject). We find that our work can generate the same semantic variations of these unique objects, with a high emphasis on preserving the subject identity, as can be seen, for instance, by the detailed patterns of the cat sculpture that are preserved.

Next, we show comparisons of recontextualization of a subject clock, with distinctive features using our method and prompt engineering using vanilla Imagen [9] and the public API of DALL-E 2 [7]. After multiple iterations using both models, we settle for the base prompt “retro style yellow alarm clock with a white clock face and a yellow number three on the lower right part of the clock face” to describe all of the important features of the subject clock example. We find that while DALL-E 2 and vanilla Imagen are able to generate retro-style yellow alarm clocks, they struggle to represent a number 3 on the clock face, distinct from the clock face numbers. In general, we find that it is very hard to control fine-grained details of subject appearance, even with exhaustive prompt engineering. Also, we find that context can bleed into the appearance of our subject instance. We show the results in Figure 15, and can observe that our method conserves fine-grained details of the subject instance such as the shape, the clock face font, and the large yellow number three on the clock face, among others.

## Societal Impact

This project aims to provide users with an effective tool for synthesizing personal subjects (animals, objects) in different contexts. While general text-to-image models might be biased towards specific attributes when synthesizing images from text, our approach enables the user to get a better reconstruction of their desirable subjects. On contrary, malicious parties might try to use such images to mislead viewers. This is a common issue, existing in other generative models approaches or content manipulation techniques. Future research in generative modeling, and specifically of personalized generative priors, must continue investigating and revalidating these concerns.

## Acknowledgement

We thank Rinon Gal, Adi Zicher, Ron Mokady, Bill Freeman, Dilip Krishnan, Huiwen Chang and Daniel Cohen-Or for their valuable inputs that helped improve this work, and to Mohammad Norouzi, Chitwan Saharia and William Chan for providing us with their support and the pretrained Imagen models. Finally, a special thanks to David Salesin for his feedback, advice and support for the

project.

## References

[1] Unsplash. `https://unsplash.com/`.

[2] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[4] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23:47–1, 2022.

[5] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *EMNLP (Demonstration)*, 2018.

[6] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67, 2020.

[7] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

[8] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[9] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo-Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022.

[10] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020.

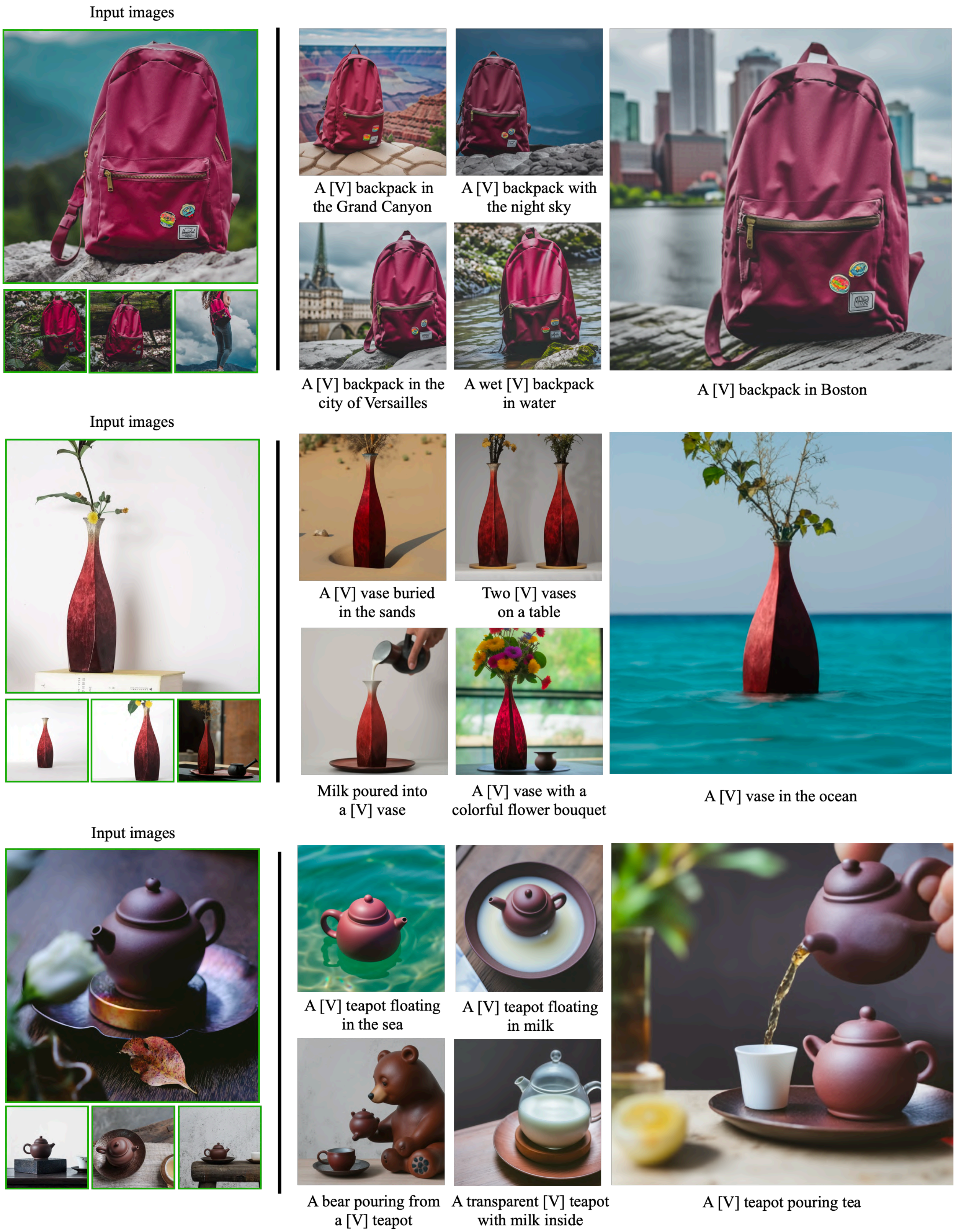

Input images
A [V] backpack in the Grand Canyon
A [V] backpack with the night sky
A [V] backpack in the city of Versailles
A wet [V] backpack in water
A [V] backpack in Boston
Input images
A [V] vase buried in the sands
Two [V] vases on a table
Milk poured into a [V] vase
A [V] vase with a colorful flower bouquet
A [V] vase in the ocean
Input images
A [V] teapot floating in the sea
A [V] teapot floating in milk
A bear pouring from a [V] teapot
A transparent [V] teapot with milk inside
A [V] teapot pouring tea

Figure 4. **Additional recontextualization samples of a backpack, vase, and teapot subject instances.** We are able to generate images of the subject instance in different environments, with high preservation of subject details and realistic interaction between the scene and the subject. We display the conditioning prompts below each image.

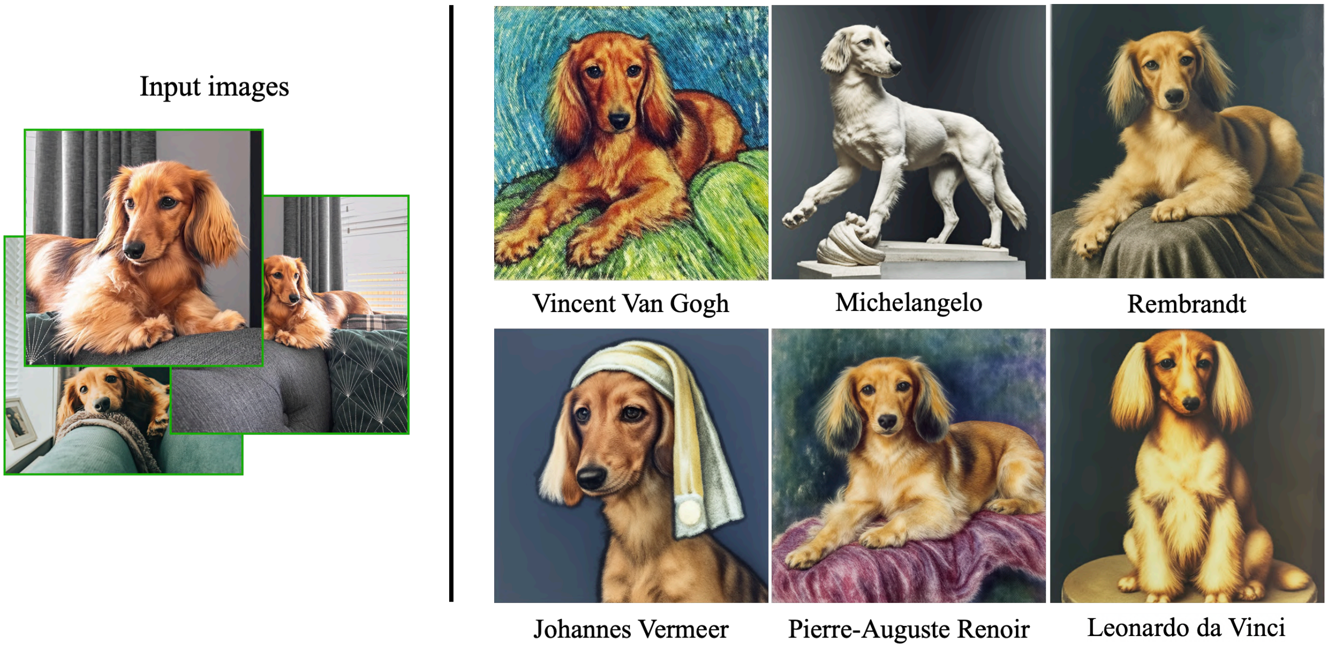


Figure 5. **Additional artistic renderings of a dog instance in the style of famous painters**. We remark that many of the generated poses, e.g., the Michelangelo renditions, were not seen in the training set. We also note that some renditions seem to have novel compositions and faithfully imitate the style of the painter.

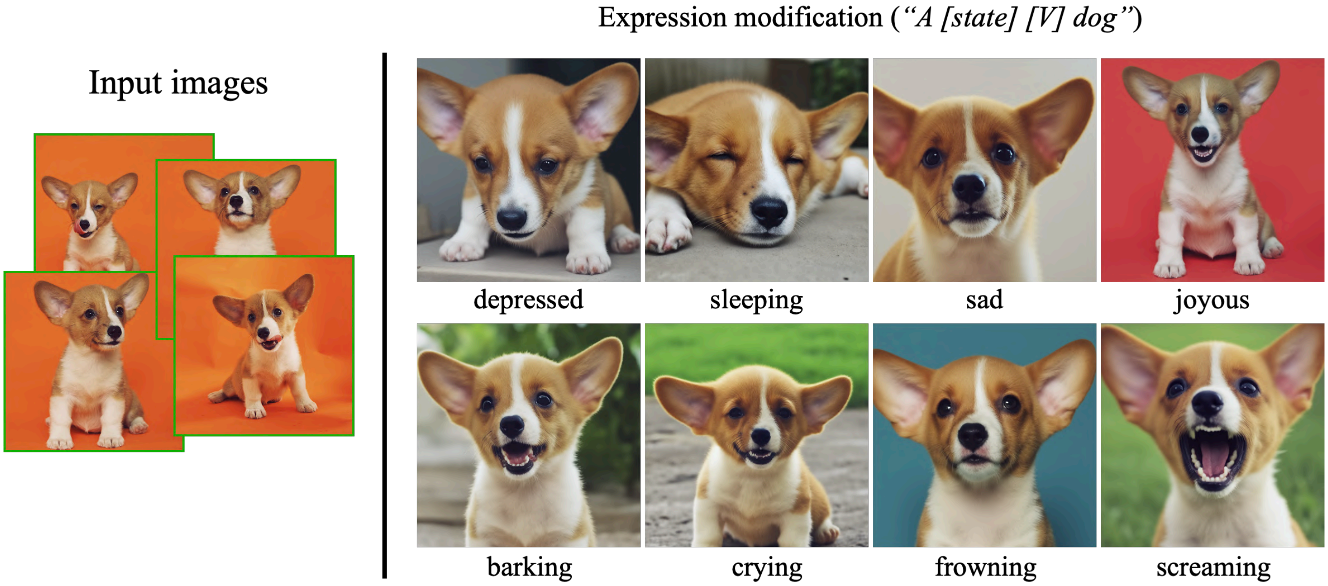


Figure 6. **Expression manipulation of a dog instance.** Our technique can synthesize various expressions that do not appear in the input images, demonstrating the extrapolation power of the model. Note the unique asymmetric white streak on the subject dog's face.

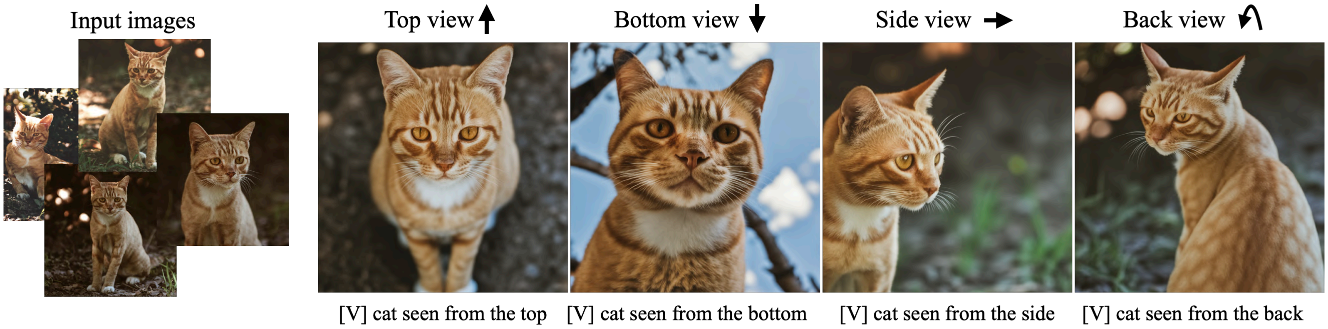

Input images
Top view
Bottom view
Side view
Back view
[V] cat seen from the top
[V] cat seen from the bottom
[V] cat seen from the side
[V] cat seen from the back

Figure 7. **Text-guided view synthesis**. Our technique can synthesize images with specified viewpoints for a subject cat (left to right: top, bottom, side, and back views). Note that the generated poses are different from the input poses, and the background changes in a realistic manner given a pose change. We also highlight the preservation of complex fur patterns on the subject cat's forehead.

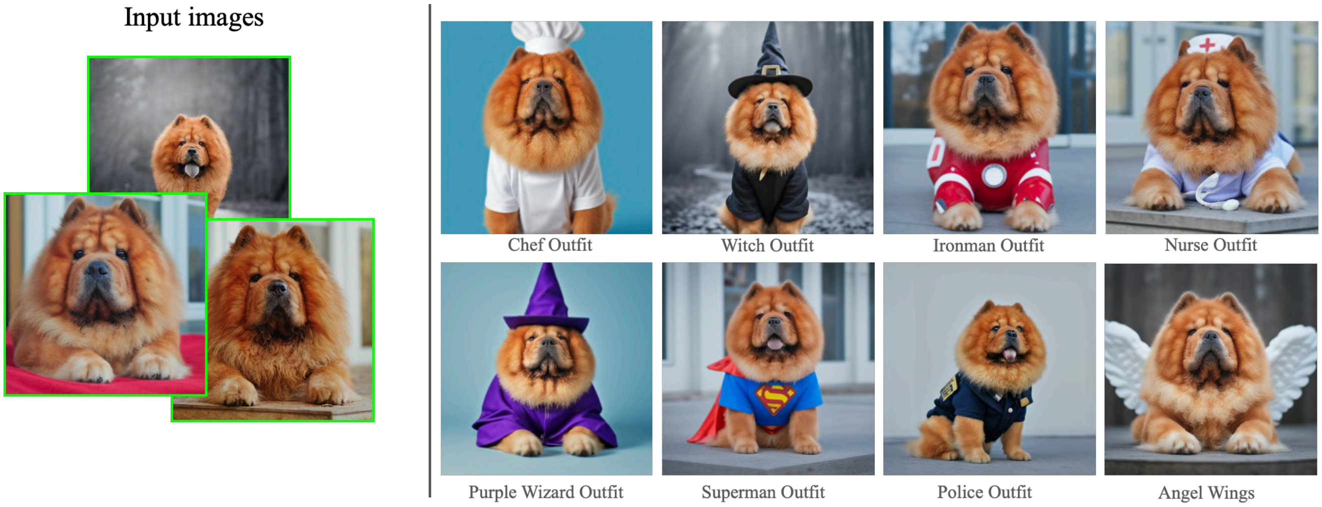


Figure 8. **Outfitting a dog with accessories**. The identity of the subject is preserved and many different outfits or accessories can be applied to the dog given a prompt of type "a [V] dog wearing a police/chef/witch outfit". We observe a realistic interaction between the subject dog and the outfits or accessories, as well as a large variety of possible options.

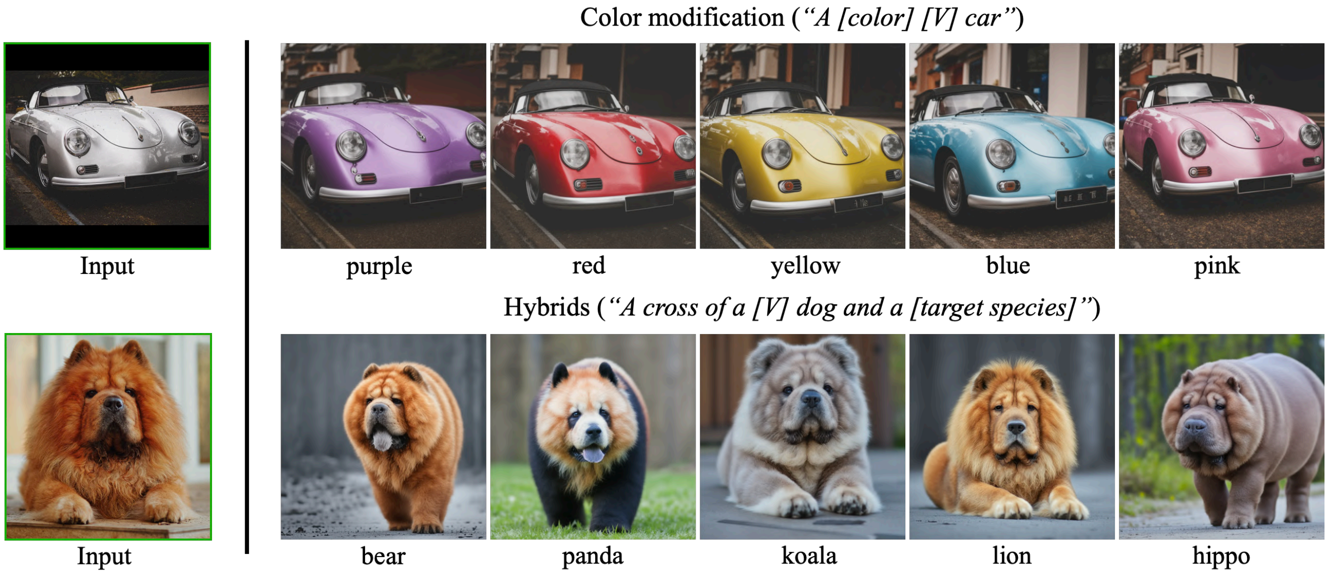

Color modification (*"A [color] [V] car"*)
Input
purple
red
yellow
blue
pink
Hybrids (*"A cross of a [V] dog and a [target species]"*)
Input
bear
panda
koala
lion
hippo

Figure 9. **Modification of subject properties while preserving their key features.** We show color modifications in the first row (using prompts "a [color] [V] car"), and crosses between a specific dog and different animals in the second row (using prompts "a cross of a [V] dog and a [target species]"). We highlight the fact that our method preserves unique visual features that give the subject its identity or essence, while performing the required property modification.

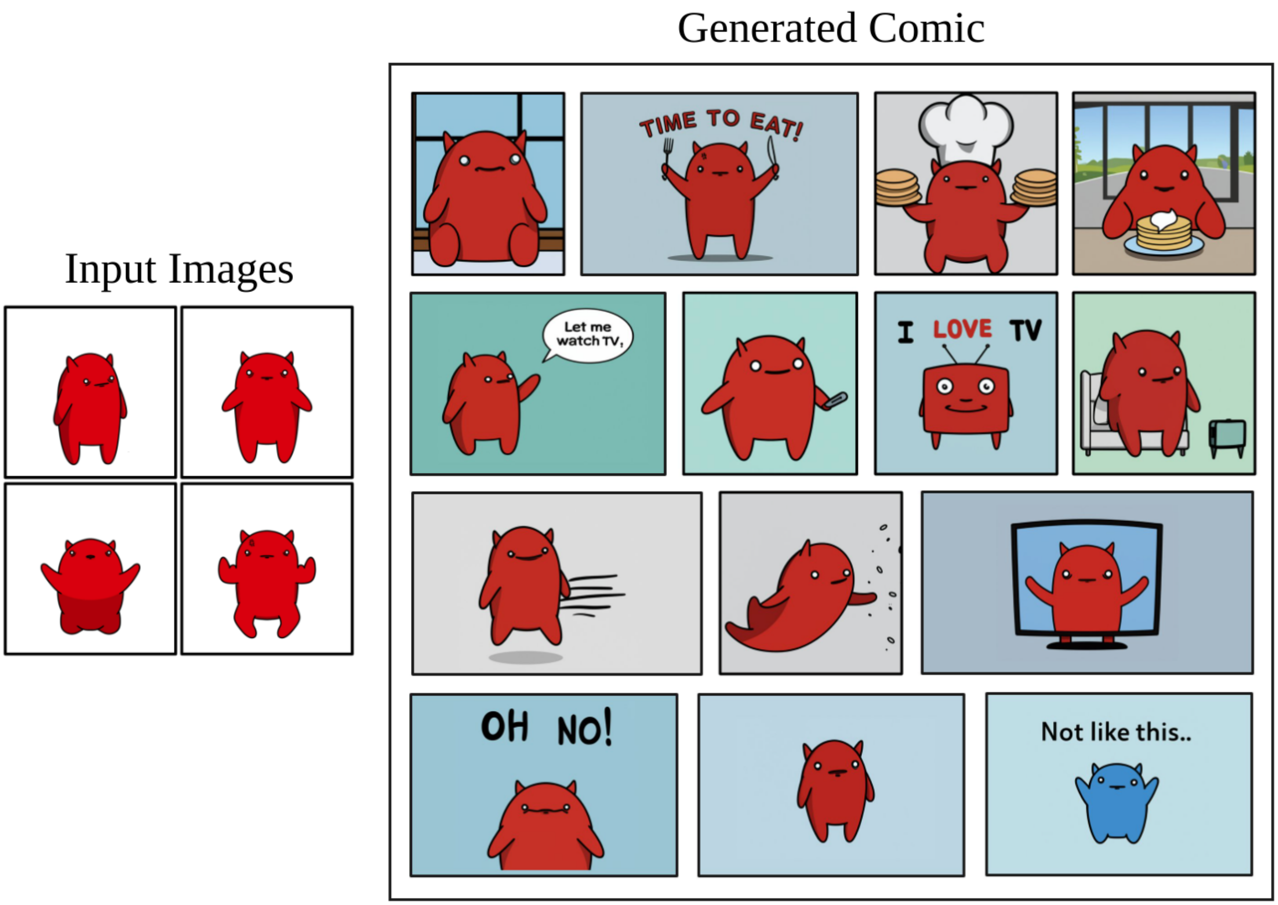


Figure 10. **Generated comic.** We present, to the best of our knowledge, the first comic comic with a persistent character generated by prompting a generative model.

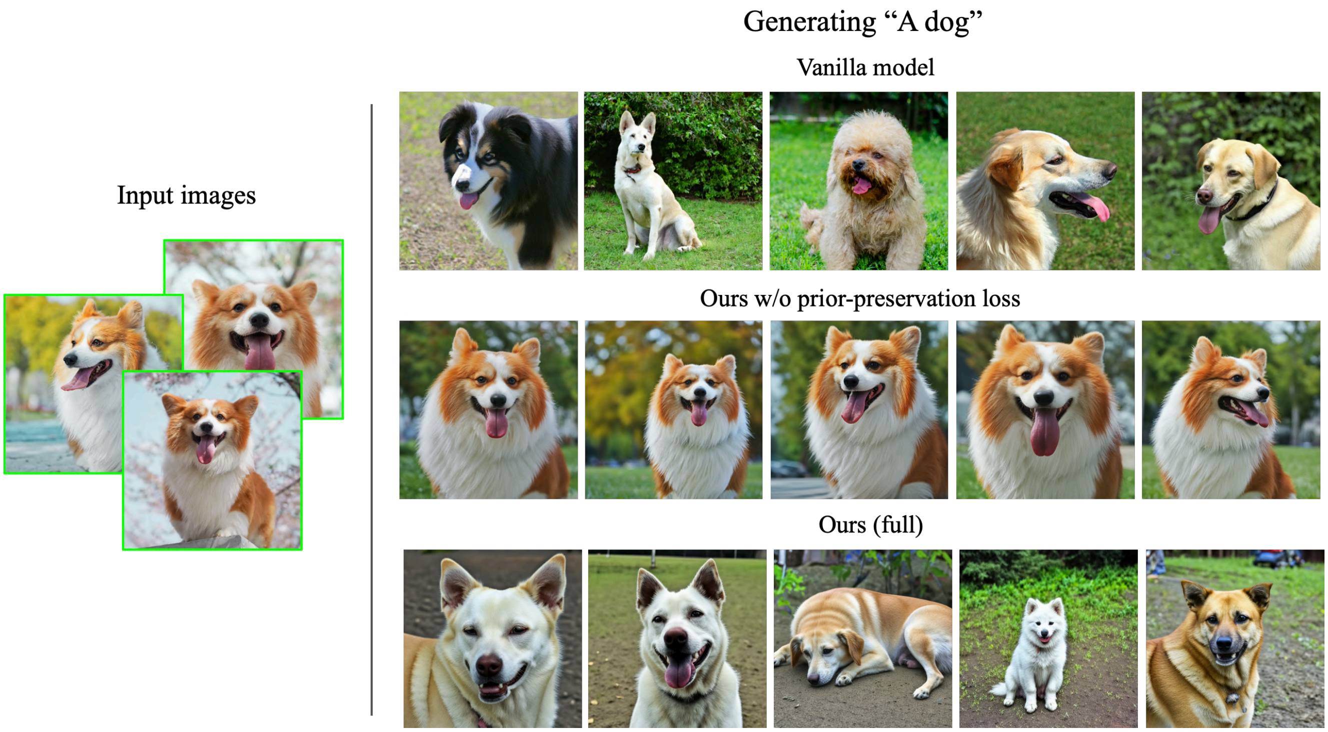


Figure 11. **Preservation of class semantic priors with prior-preservation loss.** Fine-tuning using images of our subject without prior-preservation loss results in language drift and the model loses the capability of generating other members of our subject's class. Using a prior-preservation loss term allows our model to avoid this and to preserve the subject class' prior.

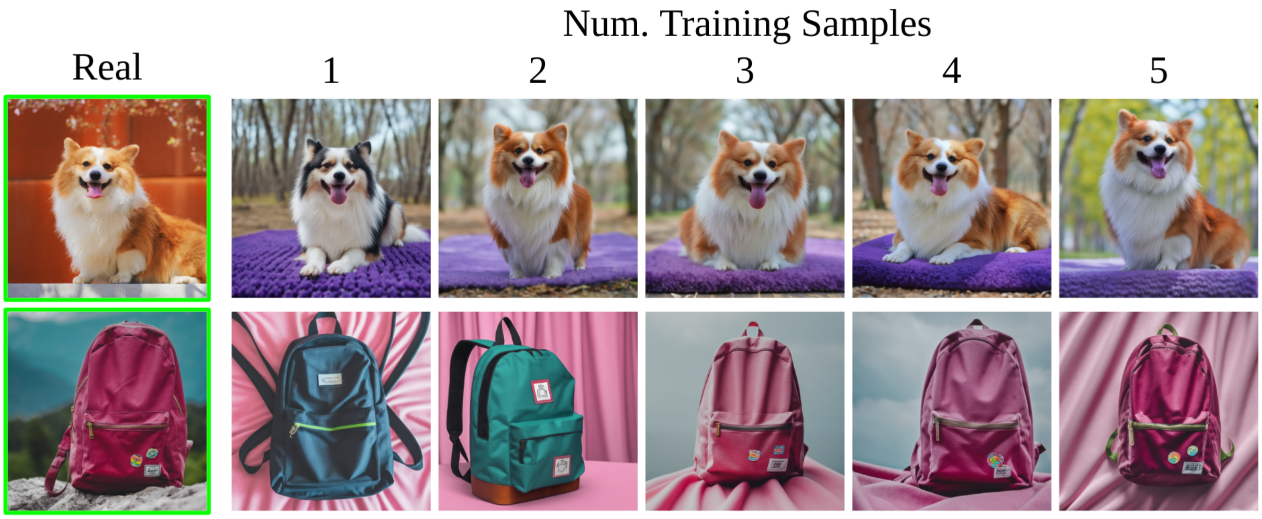


Figure 12. **Impact of number of input images.** We observe that given only one input image, we are close to capture the identity of some subjects (e.g. Corgi dog). More images are usually needed - two images are sufficient to reconstruct the Corgi dog in this example whereas at least 3 are needed for a more rare item such as the backpack.

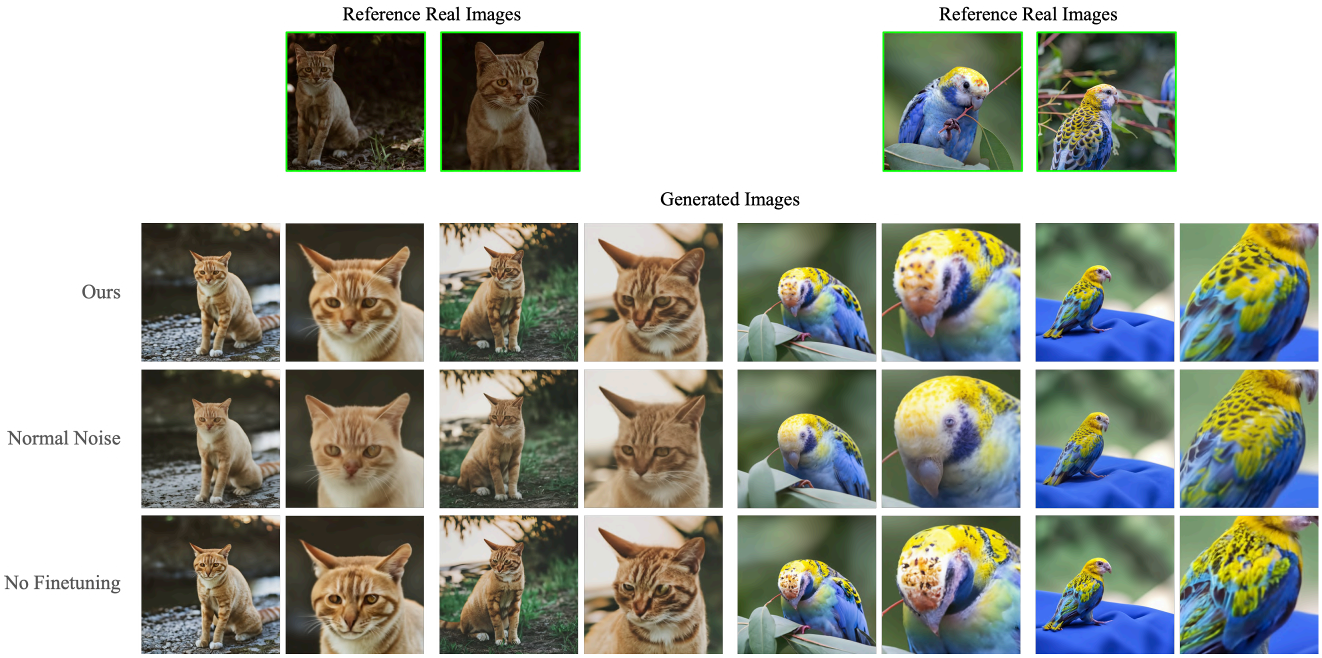


Figure 13. **Ablations with fine-tuning the super-resolution (SR) models.** Using the normal level of noise augmentation of [9] to train the SR models results in blurred high-frequency patterns, while no fine-tuning results in hallucinated high-frequency patterns. Using low-level noise augmentation for SR models improves sample quality and subject fidelity. Image credit (input images): Unsplash.

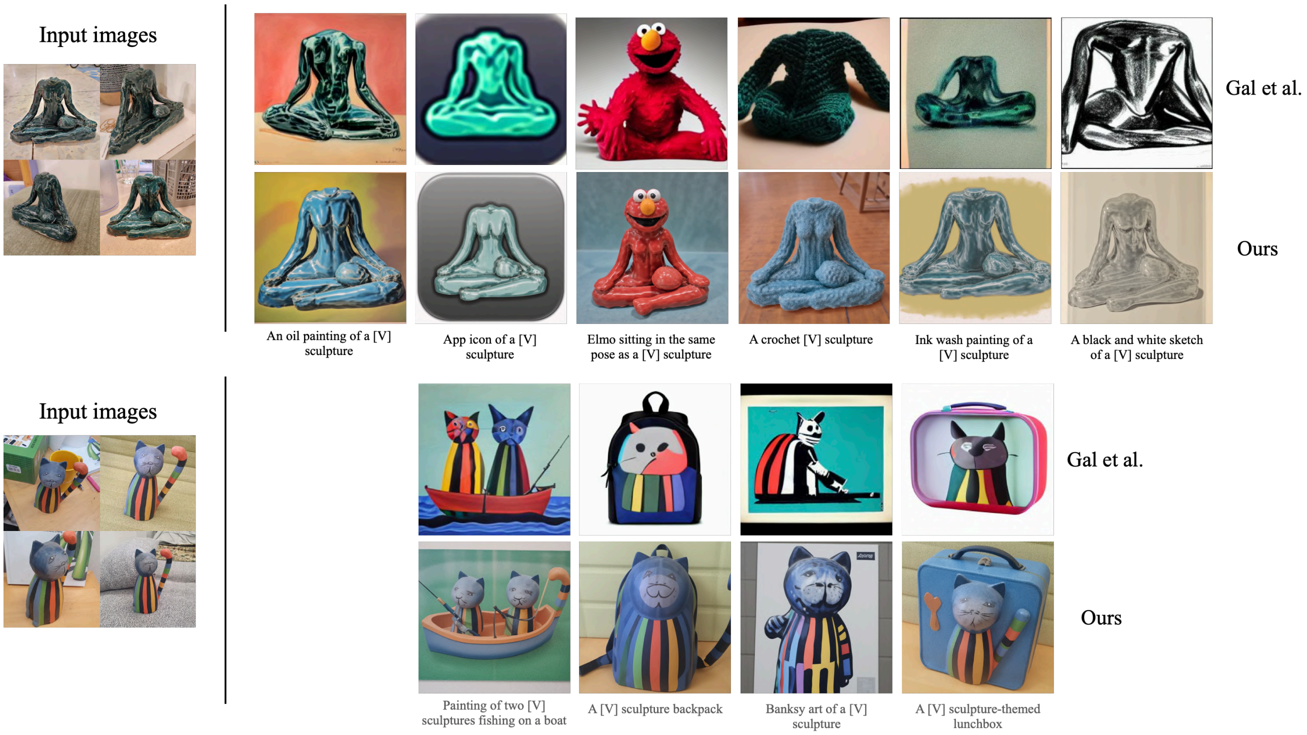

Input images
Gal et al.
Ours
An oil painting of a [V] sculpture
App icon of a [V] sculpture
Elmo sitting in the same pose as a [V] sculpture
A crochet [V] sculpture
Ink wash painting of a [V] sculpture
A black and white sketch of a [V] sculpture
Input images
Gal et al.
Ours
Painting of two [V] sculptures fishing on a boat
A [V] sculpture backpack
Banksy art of a [V] sculpture
A [V] sculpture-themed lunchbox

Figure 14. **Comparisons with Gal et al. [2]** using the subjects, images, and prompts from their work. Our approach is able to generate semantically correct variations of unique objects, exhibiting a higher degree of preservation of subject features. Input images provided by Gal et al. [2].

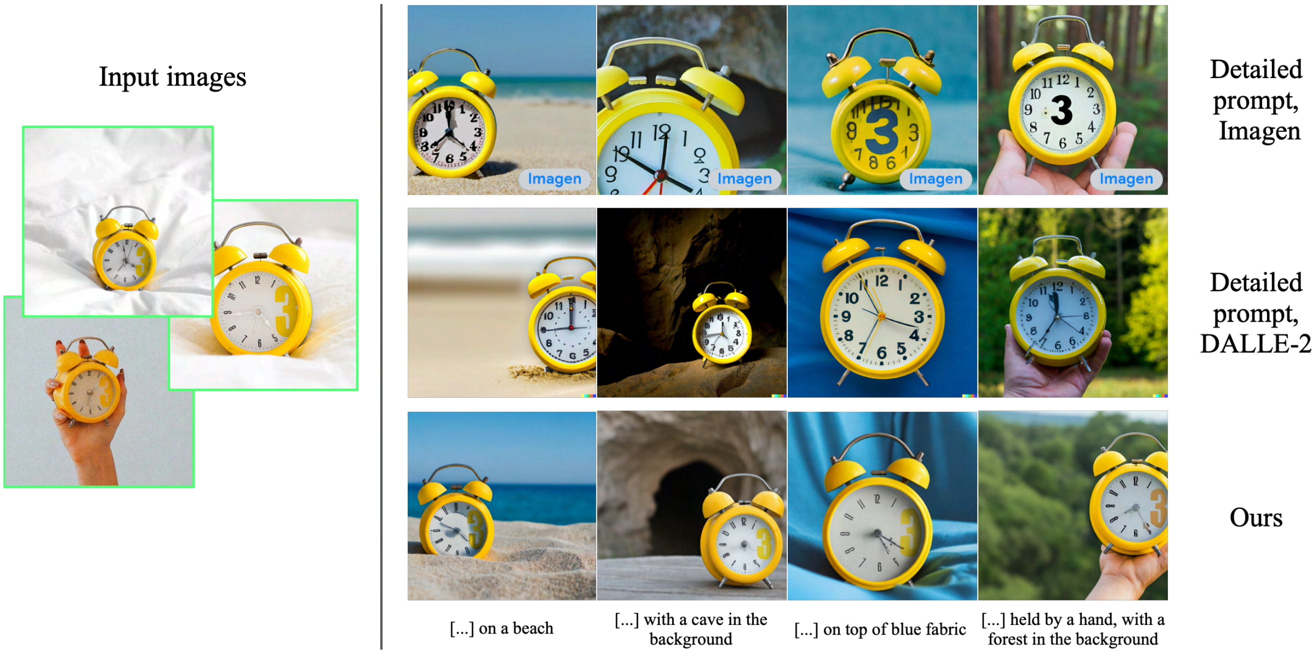

Input images
Detailed prompt, Imagen
Imagen
Imagen
Imagen
Imagen
Detailed prompt, DALLE-2
Ours
[...] on a beach
[...] with a cave in the background
[...] on top of blue fabric
[...] held by a hand, with a forest in the background

Figure 15. **Comparison with DALL-E 2 and Imagen with detailed prompt engineering.** After several trial-and-error iterations, the base prompt used to generate DALL-E 2 and Imagen results was *"retro style yellow alarm clock with a white clock face and a yellow number three on the right part of the clock face"*, which is highly descriptive of the subject clock. In general, it is hard to control fine-grained details of subject appearance using prompts, even with large amounts of prompt engineering. Also, we can observe how context cues in the prompt can bleed into subject appearance (e.g. with a blue number 3 on the clock face when the context is "on top of blue fabric"). Image credit (input images): Unsplash.