An Edit Friendly DDPM Noise Space: Inversion and Manipulations Supplementary Material

A. Shifting the latent code

As described in Sec. 3, we can shift an input image by shifting its extracted latent code. This requires inserting new columns/rows at the boundary of the noise maps. To guarantee that the inserted columns/rows are drawn from the same distribution as the rest of the noise map, we simply copy a contiguous chunk of columns/rows from a different part of the noise map. In all our experiments, we copied into the boundary the columns/rows indexed $\{50, \ldots, 50 + d - 1\}$ for a shift of d pixels. We found this strategy to work better than randomly drawing the missing columns/rows from a white normal distribution having the same mean and variance as the rest of the noise map. Figure S1 depicts the MSE over the valid pixels that is incurred when shifting the noise maps. This analysis was done using 25 generated-images. As can be seen, shifting our edit-friendly code results in minor degradation while shifting the native latent code leads to a complete loss of the image structure.

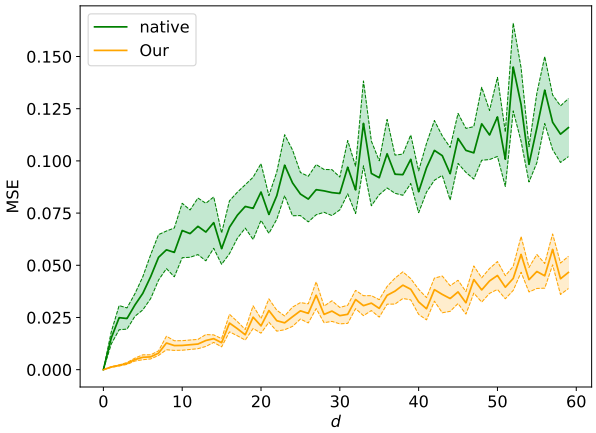

Figure S1: **Shifting the latent code**. We plot the MSE over the valid pixels after shifting the latent code and generating the image. The colored regions represent one standard error of the mean (SEM) in each direction.

B. CycleDiffusion

As mentioned in Sec. 5, CycleDiffusion [6] extracts a sequence of noise maps $\{x_T, z_T, \ldots, z_1\}$ for the DDPM scheme. However, in contrast to our method, their noise maps have statistical properties that resemble those of regular sampling. This is illustrated in Fig. S2, which depicts the per-pixel standard deviations of $\{z_t\}$ and the correlation between z_t and z_{t-1} for CycleDiffusion, for regular sampling, and for our approach. These statistics were calculated over 10 images using an unconditional diffusion model trained on Imagenet. As can be seen, the CycleDiffusion curves are almost identical to those of regular sampling, and are different from ours.

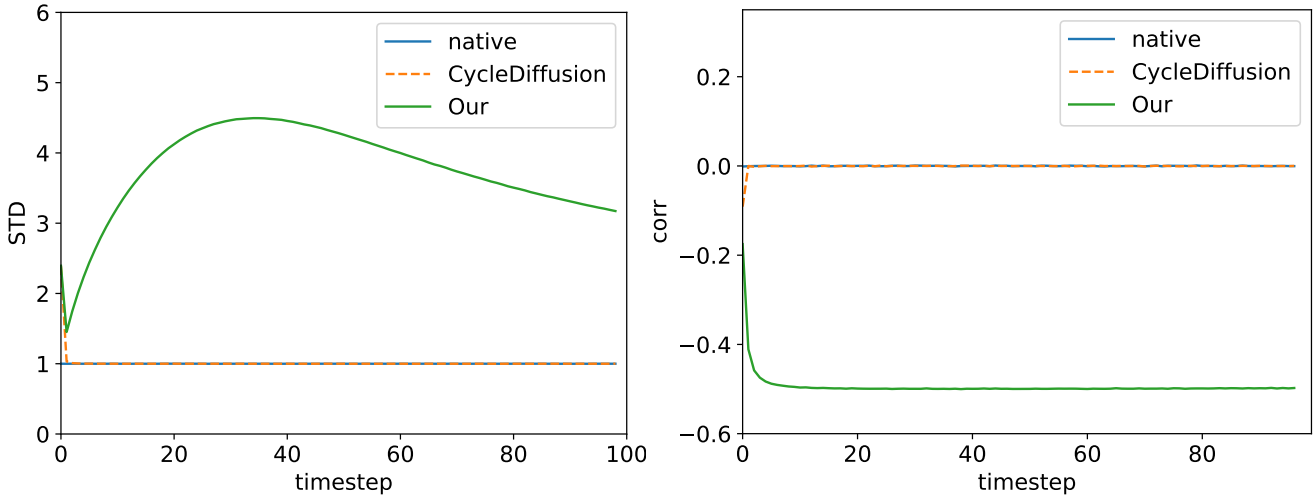

Figure S2: **CycleDiffusion noise statistics.** Here we show the per-pixel standard deviations of $\{z_t\}$ and the per-pixel correlation between them for mssodel-generated images.

The implication of this is that similarly to the native latent space, simple manipulations on CycleDiffusion's noise maps cannot be used to obtain artifact-free effects in pixel space. This is illustrated in Fig. S3 in the context of horizontal flip and horizontal shift by 30 pixels to the right. As opposed to Cycle diffusion, applying those transformations on our latent code, leads to the desired effects, while better preserving structure.

This behavior also affects the text based editing capabilities of CycleDiffusion. Particularly, the CLIP similarity and LPIPS distance achieved by CycleDiffusion on the modified ImageNet-R-TI2I dataset are plotted in Fig. S5. As can be seen, when tuned to achieve a high CLIP-similarity (*i.e.* to better conform with the text), CycleDiffusion's LPIPS loss increases significantly, indicating that the output images become less similar to the input images. For the same level of CLIP similarity, our approach achieves a substantially lower LPIPS distance.

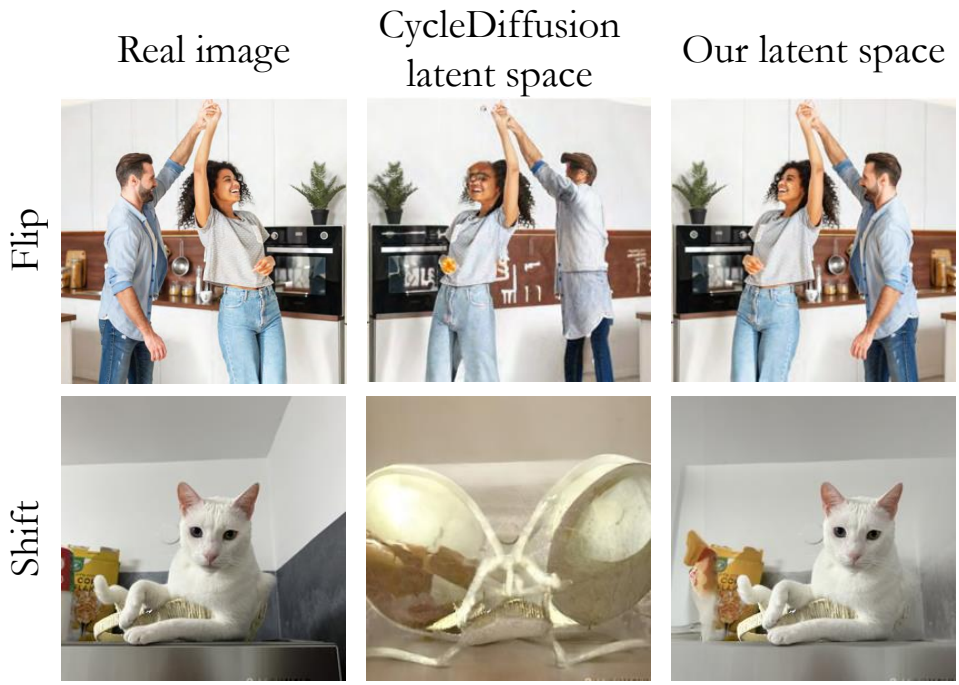

Figure S3: **Flip and shift with CycleDiffusion and with our inversion.**

C. The effects of the skip and the strength parameters

Recall from Sec. 3 that to perform text-guided image editing using our inversion, we start by extracting the latent noise maps while injecting the source text into the model, and then generate an image by fixing the noise maps and injecting a target text prompt. Two important parameters in this process are T_{skip}, which controls the timestep $(T - T_{\text{skip}})$ from which we start the generation process, and the strength parameter of the classifier-free scale [2]. Figure S4 shows the effects of these parameters. When T_{skip} is large, we start the process with a less noisy image and thus the output image remains close to the input image. On the other hand, the strength parameter controls the compliance of the output image with the target prompt.

Figure S4: **The effects of the skip and the strength parameters**.

D. Additional details on experiments and further numerical evaluation

For all our text-based editing experiments, we used Stable Diffusion as our pre-trained text-to-image model. We specifically used the StableDiffusion-v-1-4 checkpoint. We ran all experiments on an RTX A6000 GPU. We now provide additional details about the evaluations reported in the main text. All datasets and prompts will be published.

D.1. Experiments on the modified ImageNet-R-TI2I

Our modified ImageNet-R-TI2I dataset contains 44 images: 30 taken from PnP [5], and 14 from the Internet and from the code bases of other existing text-based editing methods. We verified that there is a reasonable source and target prompt for each image we added. For P2P [1] (with and without our inversion), we used the first 30 images with the "replace" option, since they were created with rendering and class changes. That is, the text prompts were of the form "a ≪rendering≫ of a ≪class≫" (*e.g.* "a sketch of a cat" to "a sculpture of a cat"). The last 14 images include prompts with additional tokens and different prompt lengths (*e.g.* changing "A photo of an old church" to "A photo of an old church with a rainbow"). Therefore for those images we used the "refine" option in P2P. We configured all methods to use 100 forward and backward steps, except for PnP whose supplied code does not work when changing this parameter.

Table S1 summarizes the hyper-parameters we used for all methods. For our inversion and for P2P with our inversion, we arrived at those parameters by experimenting with various sets of the parameters and choosing the configuration that led to the best CLIP loss under the constraint that the LPIPS distance does not exceed 0.3. For DDIM inversion and for P2P (who did not illustrate their method on real images), such a requirement could not be satisfied. Therefore for those methods we chose the configuration that led to the best CLIP loss under the constraint that the LPIPS distance does not exceed 0.62. For PnP, we used the default parameters supplied by the authors. In Fig. S5 we show the CLIP-LPIPS losses graph for all methods reported in the paper, as well as for CycleDiffusion. In this graph, we show three different parameter configurations for our inversion, for P2P with our inversion, and for CycleDiffusion. As can be seen, our method (by itself or with P2P) achieves the best LPIPS distance for any given level of CLIP similarity.

Method	#inv. steps	#edit steps	strength	T_{skip}	$\tau_{\text{x}}/\tau_{\text{a}}$
PnP	1000	50	10	0	40/25
DDIM inv.	100	100	9	0	–
Our inv.	100	100	15	36	–
P2P	100	100	9	0	80/40
P2P + Our inv.	100	100	9	12	60/20

Table S1: **Hyper-parameters used in experiments on the modified ImageNet-R-TI2I dataset**. The parameter 'strength' refers to the classifier-free scale of the generation process. As for the strength used in the inversion stage, we set it to 3.5 for all methods except for PnP, which uses 1. The timestep at which we start the generation is $T - T_{\text{skip}}$ and, in case of injecting attentions, we also report the timestep at which the cross- and self-attentions start to be injected, τ_{x} and τ_{a} respectively.

D.2. Experiments on the modified zero-shot I2IT dataset

The second dataset we used is the modified Zero-Shot I2IT dataset, which contains 4 categories (cat, dog, horse, zebra). Ten images from each category were taken from Parmar *et al.* [4], and we added 5 more images from the Internet to each category. Zero-Shot I2I [4] does not use source-target pair prompts, but rather pre-defined source-target classes (*e.g.* cat↔dog). For their optimized DDIM-inversion part, they use a source prompt automatically generated with BLIP [3]. When combining our inversion with their generative method, we use $T_{\text{skip}} = 0$ and an empty source prompt. Table S2 summarizes the hyper-parameters used in every method.

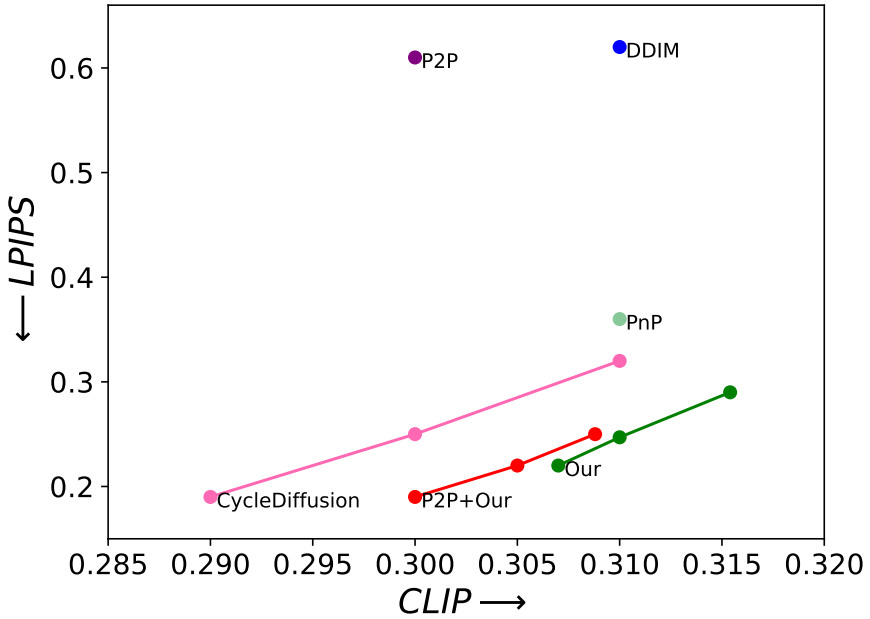

Figure S5: **Fidelity to source image vs. compliance with target text.** We show a comparison of all methods over the modified ImageNet-R-TI2I in terms of the LPIPS and CLIP losses. The parameters used for the evaluation are reported in Tab. S1. In addition, we depict three options for our inversion, for P2P with our inversion, and for CycleDiffusion. These three options correspond to different choices of the parameters (strength,T_{skip}): for our method $(15, 36), (12, 36), (9, 36)$, for P2P+Ours $(7.5, 8), (7.5, 12), (9, 20)$, and for CycleDiffusion $(3, 30), (4, 25), (4, 15)$. In CLIP loss, higher is better while in LPIPS loss, lower is better.

Method	#inv. steps	#edit steps	strength	T_{skip}	λ_{xa}
Zero-Shot	50	50	7.5	0	0.1
Zero-Shot+Our	50	50	7.5	0	0.03

Table S2: **Hyper-parameters used in experiments on the modified Zero-Shot I2IT dataset**. In this method, cross-attention guidance weight is the parameter used to control the consistency in the cross-attention maps, denoted here as λ_{xa}. We set the strength (classifier-free scale) in the inversion part to be 1 and 3.5 for "Zero-shot" and "Zero-shot+Our" respectively.

E. Additional results

Due to the stochastic manner of our method, we can generate diverse outputs, a feature that is not naturally available with methods relying on the DDIM inversion. Figures S6 and S7 show several diverse text-based editing results. Figures S8 and S9 provide further qualitative comparisons between all methods tested on the ImageNet-R-TI2I dataset.

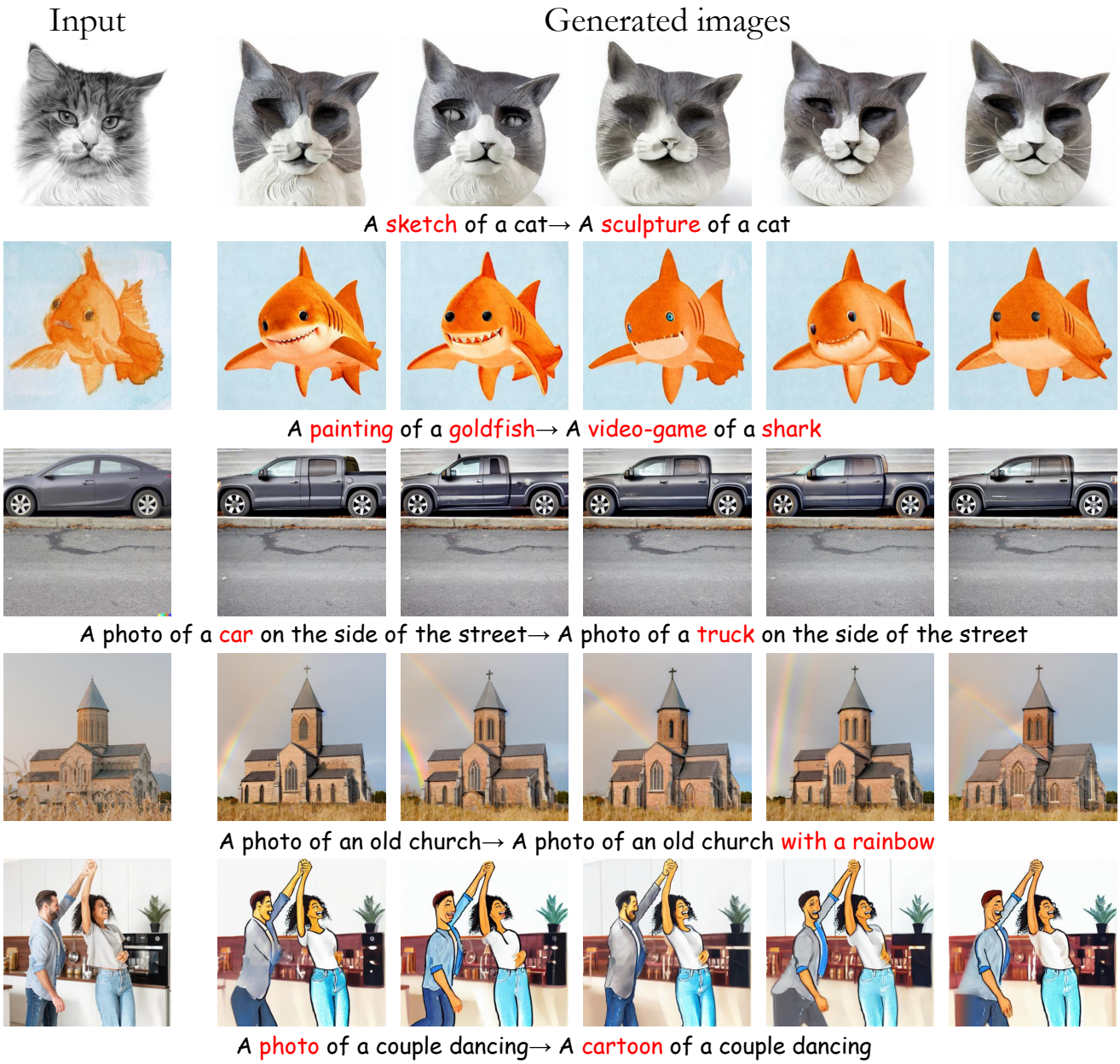

Figure S6: **Diverse text-based editing with our method**. We apply our inversion five times with the same source and target prompts (shown beneath each example). Note how the variability between the results is not negligible, while all of them conform to the structure of the input image and comply with the target text prompt. Notice *e.g.* the variability in the sculpture cat's eyes and mouth, and how the rainbow appears in different locations and angles.

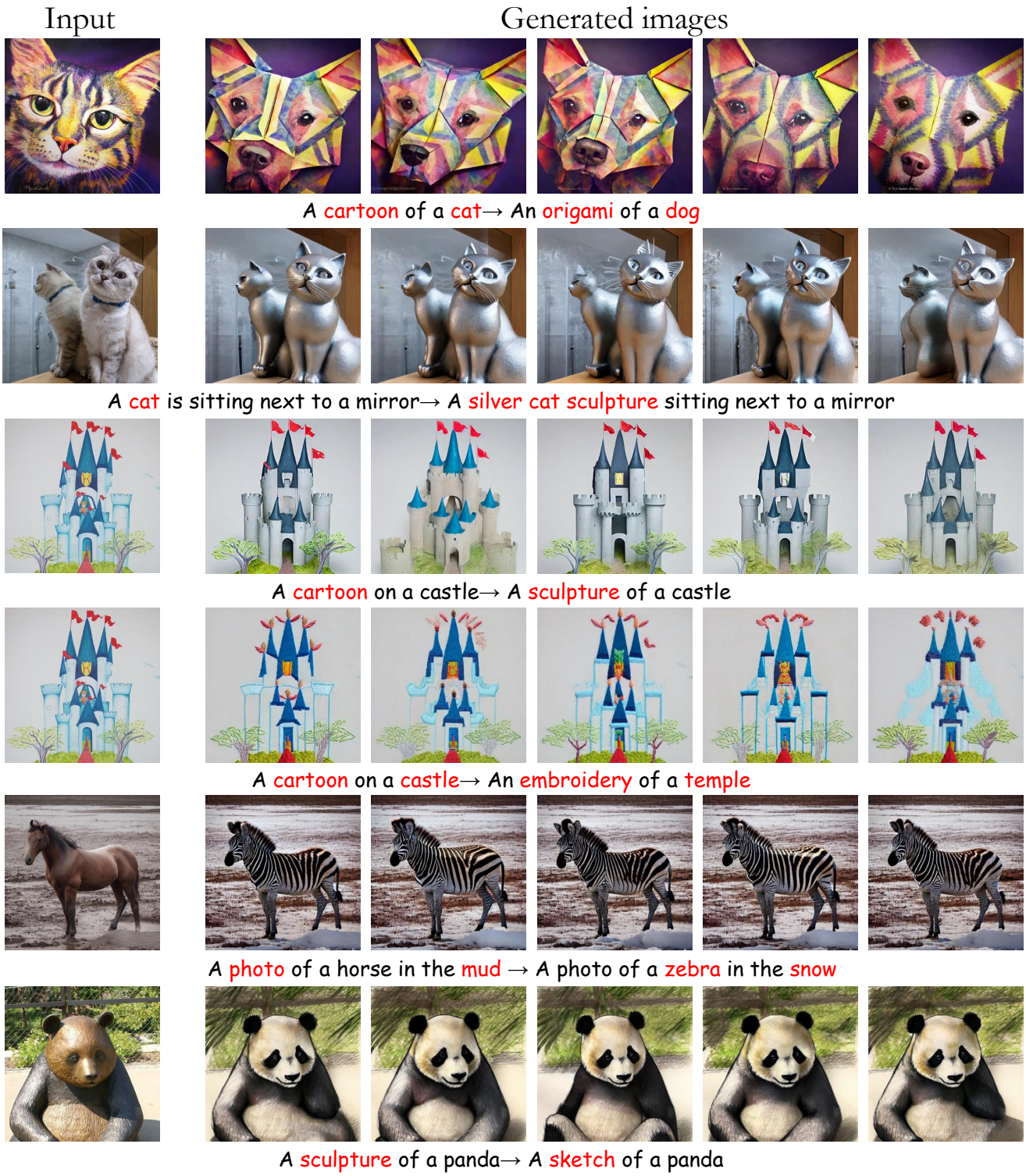

Figure S7: **Additional results for diverse text-based editing with our method**. Notice that each edited result is slightly different. For example, the eyes and nose of the origami dog change between samples, and so do the zebra's stripes.

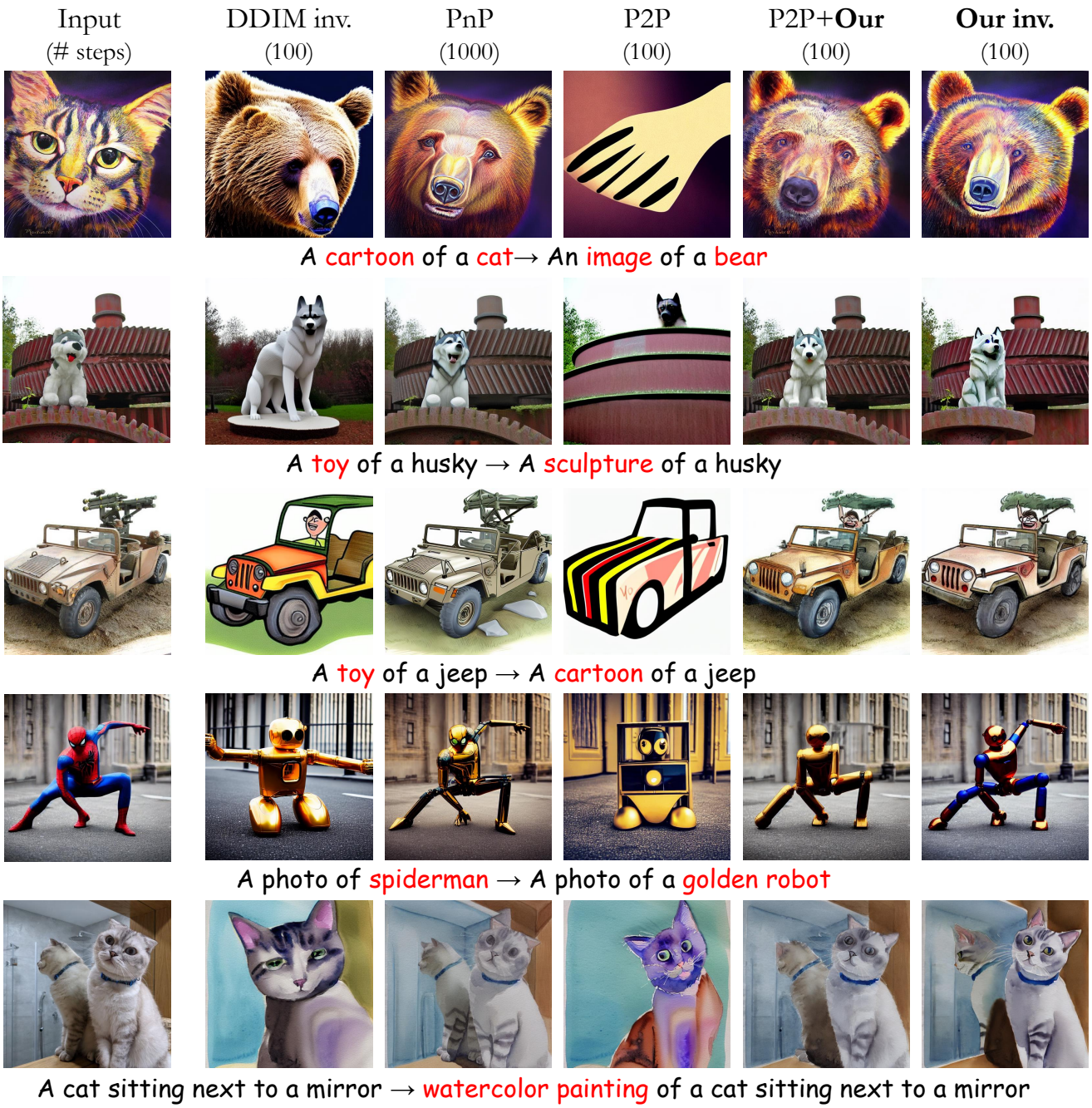

Figure S8: **Qualitative comparisons between all methods.**

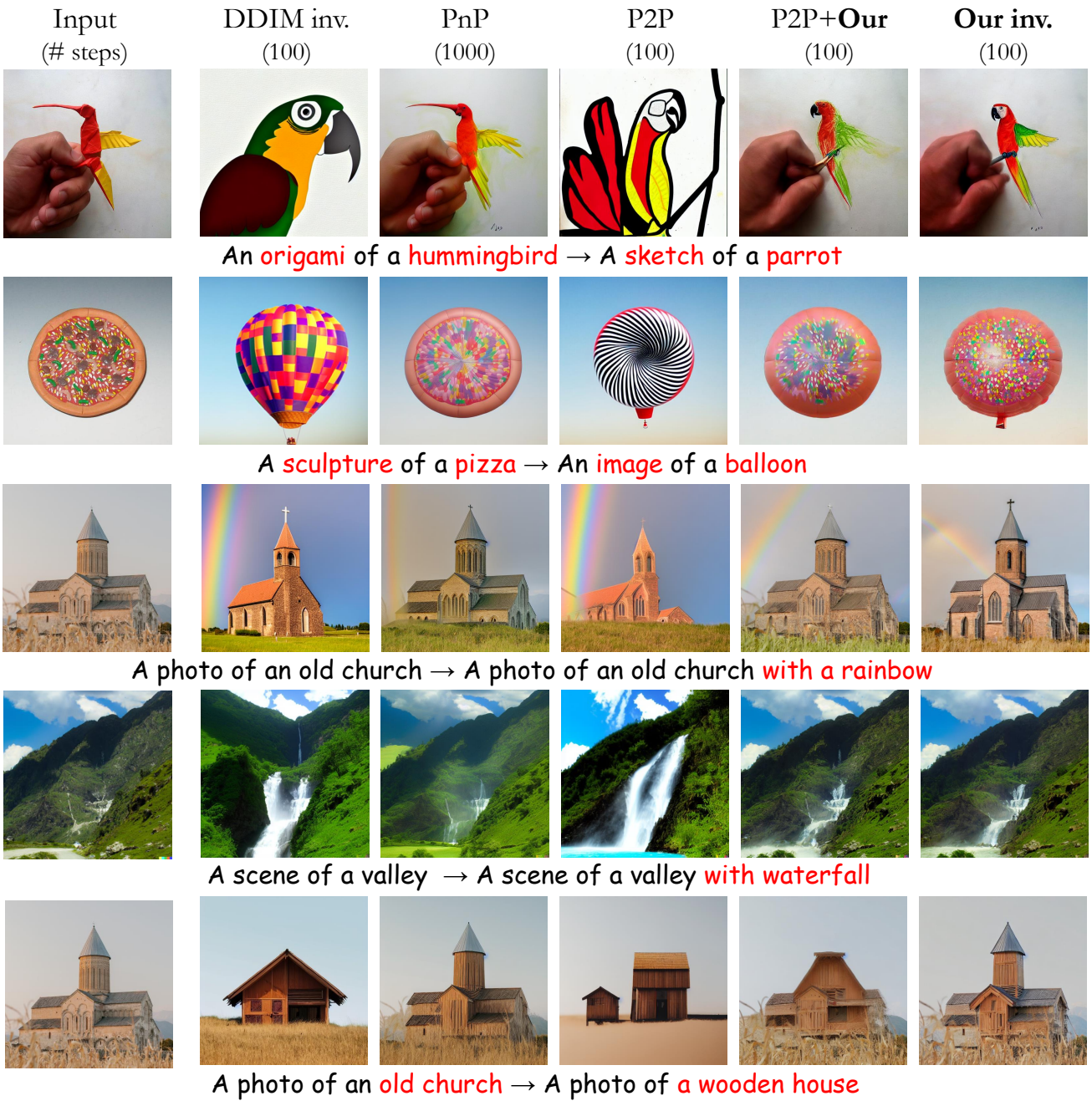

Figure S9: **Additional qualitative comparisons between all methods.**

References

[1] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 5

[2] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 4

[3] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning (ICML)*, pages 12888–12900, 2022. 5

[4] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. *arXiv preprint arXiv:2302.03027*, 2023. 5

[5] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. *arXiv preprint arXiv:2211.12572*, 2022. 5

[6] Chen Henry Wu and Fernando De la Torre. Unifying diffusion models' latent space, with applications to CycleDiffusion and guidance. *arXiv preprint arXiv:2210.05559*, 2022. 2