GENN2N: GENERATIVE NERF2NERF TRANSLATION (SUPPLEMENTARY MATERIALS)

Anonymous authors
Paper under double-blind review

1 2D IMAGE-TO-IMAGE TRANSLATOR

In our proposed GenN2N, we use a universal image-to-image translator to perform editing on the 2D domain and optimize the translated NeRF to lift these 2D edits into the 3D NeRF space. For different NeRF editing tasks, we choose different 2D translators to achieve corresponding editing tasks.

- **Text-driven Editing.** To achieve NeRF editing under text instructions, we use InstructPix2Pix (Brooks et al., 2022) as the 2D image-to-image translator in our framework. InstructPix2Pix is a diffusion-based method designed for image editing according to user-provided instructions. Specifically, InstructPix2Pix learns a U-Net to perform denoise diffusion to generate the target edited image based on the given image and the text embedding. While InstructPix2Pix can produce high-quality editing results that highly align with the input instructions, given different initial noise or input image, different content may be generated during the editing process of InstructPix2Pix, which makes it difficult to ensure the 3D consistency in the text-driven NeRF Editing process.
- **Super-resolution.** For the NeRF super-resolution task, we choose ResShift (Yue et al., 2023) as the 2D image-to-image translator in GenN2N. ResShift is the current state-of-the-art image super-resolution method designed based on diffusion model. With dedicated designs for image super-resolution, such as the residual shifting mechanism and the flexible noise schedule, ResShift can produce super-resolution images with high-quality. Thus, given a set of multi-view images of a NeRF scene, we directly use ResShift to increase the resolution of all these images by a factor of 4.
- **Object Removal.** For the task of object removal in NeRF, we use Blended Latent Diffusion (Avrahami et al., 2023) as our 2D image-to-image translator. The input of Blended Latent Diffusion is an image, a guiding text prompt and a binary mask that indicates the region of the object to be recovered. We use the segment anything model (Kirillov et al., 2023)to generate the mask of the object at the specified location. The text prompt can be either manually specified or described by the visual instruction model, e.g. (Gao et al., 2023). By utilizing the diffusion process, Blended Latent Diffusion can successfully generate contents in the desired region that are consistent with the text description, while the complementary are remains close to the input image.
- **Zoom Out.** Similar to Object Removal, we also use Blended Latent Diffusion as the 2D image-to-image translator in our GenN2N to solve the NeRF zoom out problem. Given source multi-view images of a 3D NeRF scene, we set the zoom out ratio as 1.25 for image width and height to enlarge the source images. We first automatically generate binary masks for the zoom out region and then employ Blended Latent Diffusion to recover those zoom out regions. Since different content may be generated for zoom out regions in different 2D images from different viewpoint, it is difficult to ensure the 3D consistency in those zoom out regions.
- **Inpainting.** For the 3D NeRF inpainting task, we also use Blended Latent Diffusion as the 2D image-to-image translator in our GenN2N. We support various ways to get a mask, but note that multi-view masks must correspond to the same location in the 3D scene. For example, by artificially calculating the position of the part to be inpaint in the 3D scene corresponding to the 2D image, or using the segment anything model (Kirillov et al., 2023) to get the bounding box of the same object. Moreover, we found that if the text prompt is not provided as a guide, the diffusion model tends to inpaint unreasonable content or

monotonous colors close to the surroundings, instead of drawing meaningful objects. So we artificially set up the required text prompt or used the visual question answering model (Gao et al., 2023) to get answers to the "imagine what the white area might be" question.

- **Colorization.** To achieve 3D NeRF colorization, we use DDColor (Kang et al., 2022) as our 2D image-to-image translator. Specifically, given a set of gray-scale multi-view images of a NeRF scene, we use DDColor to produce RGB color of each image. While high-quality colorization results can be obtained for each image using DDColor, different image may be assigned with different color style, which makes it difficult to generate consistent 3D colorization results.

2 NETWORK ARCHITECTURE

To make our GenN2N self-contained, we provide the detailed structure of the translated NeRF in Fig 1. Our main purpose of this design is to make the translated NeRF render 3D scenes conditioned on the edit code z. Given a pre-trained original NeRF, we discard its two layers used for density and color estimation. The edit code is concatenated with the intermediate features of the original NeRF and and then fed into two additional MLP networks to obtain the density σ and RGB color for volume rendering. During the optimization process of our GenN2N, the original NeRF parameters except the discared parameters are updated, as well as the newly added MLP networks.

3 DATASETS

We train and evaluate our proposed pipeline using multiple datasets, including portrait datasets, open-world scene datasets, and indoor scene datasets. The Face dataset (Haque et al., 2023) comprises 65 images capturing different views of a single person captured by a smartphone. The camera poses were extracted by using the PolyCam app. The Fangzhou self-portrait dataset (Wang et al., 2023) is collected from users utilizing a front-facing camera, resulting in a total of 100 frames. The Farm and Campsite dataset (Haque et al., 2023) consists of outdoor 360-degree scenes captured by a camera, containing 250 frames in total, and we only use the former 100 frames for data efficiency. SPIn-NeRF (Mirzaei et al., 2023) and OR-NeRF (Yin et al., 2023) conducted comparative experiments on the statue dataset (Mirzaei et al., 2023), and we chose the same dataset of 29 high-resolution images of outdoor scenes. The LLFF dataset (Mildenhall et al., 2021) consists of three large-scale outdoor scenes and four indoor scenes. We also selected part of the BlendedMVS dataset (Yao et al., 2020), which covers a variety of scenarios, including cities, buildings, sculptures, and small objects.

4 IMPLEMENTATION DETAILS

We first utilize NeRFStudio (Tancik et al., 2023) to train the original NeRF. Then we leverage InstructPix2Pix (Brooks et al., 2022) to generate per-frame edited images corresponding to the unified text prompt. During this step, we randomly select the image similarity degree S_I from $\{0.5, 2.1\}$, and text similarity degree S_T from $\{6.0, 8.5\}$, producing edited images with significant diversities under the same text prompt. Finally, 28 stylized scene images are generated from the original multi-view images for two portrait datasets, and a farm scene dataset. We implement GenEdit based on PyTorch. During the training phase, we efficiently sample one image per iteration and extract 16,384 rays with 48 points per ray in a batch. The model is trained using Adam optimizer with a learning rate of 1e-2, running for 20,000 iterations per scene. The total training phase takes about 8 hours on one NVIDIA V100 GPU. During the inference phase, we randomly sample $\mathbf{z}$ from a standard Gaussian distribution and render the generated edited NeRF from arbitrary viewpoints. Our code will be released for research.

5 INTERPOLATION RESULTS

As illustrated in Fig. 2, we employ an interpolation weight $alpha$ to blend two edit codes, utilizing the resulting NeRF to generate two views based on the interpolated edit code. Evidently, the

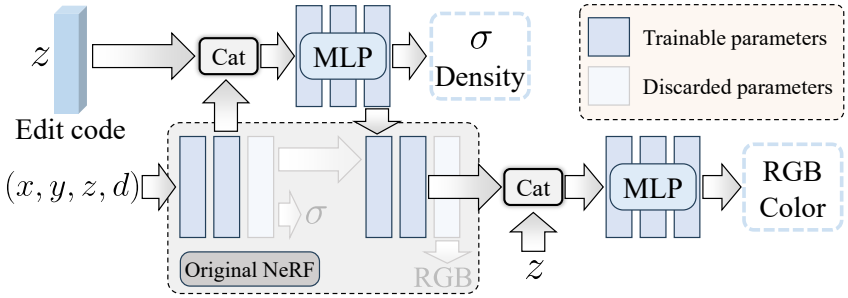

Figure 1: **Detailed structure of the translated NeRF.** Given a pre-trained NeRF model, we concatenate our edit code z with its intermediate features and produce the density σ and RGB color with two additional MLP networks.

Figure 2: **Interpolation experiment of the edit code.** Each column shows two rendered view of the translated NeRF using interpolated edit code, where α is the interpolation weight of two edit codes.

rendered images illustrate a smooth transition as the interpolation weight α increments. This observation validates that our GenN2N successfully constraints the edit code into the Gaussian distribution. Furthermore, it is worth noting that the 3D consistency is consistently maintained throughout the transition of the edit code, which also demonstrates the effectiveness of our suitable design of GenN2Nframework.

6 More Qualitative Results

- **Text-driven editing.** We show our text-driven editing results in Fig 5, where we show two views for each scene. The input scene, four of our edited NeRF with different edit code, and the input text prompt are shown in Fig 5. As we can see, our method successfully maintains the 3D multi-view consistency, while producing high-quality editing results that are consistently aligned with the input text prompt. Moreover, by changing the edit code, diverse edited scene can be rendered, which demonstrates the diverse editing ability of our method.
- **Super-resolution.** We show our super-resolution results in Fig 3, where we show two views for each scene. The low-resolution image of the input NeRF scene, four of our edited NeRF with different edit codes, and the Ground-truth images are shown in Fig 3. Zoom in results

of the red boxes are also highlighted. As we can see, our method successfully maintains the 3D multi-view consistency, while enhance the original low-resolution scene by rendering high-quality, clear and sharp images. More super-resollution results are shown in Fig. 4.

- **Zoom out.** We show our zoom out results in Fig. 6, where we employ Blended Latent Diffusion (Avrahami et al., 2023) as the 2D image-to-image translator to enlarge the input 3D NeRF scene. As can be seen, our GenN2N can generate realistic content in the zoom out region. The newly generated contents are seamlessly aligned with the original NeRF scene and are consistent across different viewpoints, which demonstrates the effectiveness of GenN2N in tackling the zoom out translation task. Moreover, when changing the edit code, the translated scene can be rendered in different styles, which also demonstrates the diversity of our GenN2N.
- **Inpainting.** Fig. 7 shows NeRF-to-NeRF inpainting results of our GenN2N, by using Blended Latent Diffusion (Avrahami et al., 2023) as the 2d image to image translator. Given the binary mask that indicates the inpainting area together with a prompt to provide language guidance for the inpainting area, our method can produce live, realistic, and multi-view consistent 3D content for the inpainting area. Meanwhile, given different edit codes, diverse 3D content can be created to recover the inpainting areas. These results all demonstrate the effectiveness of our design.
- **Colorization.** We show our colorization results in Fig 8, where we show two views for each scene. The input gray-scale scene, five of our edited NeRF with different edit codes, and the original scenes are shown in Fig 8. As we can see, our method successfully maintains the 3D multi-view consistency, while producing high-quality colorization results. Moreover, by changing the edit code, diverse edited scene can be rendered, which demonstrates the diverse editing ability of our method.
- **Object removal.** We show our object removal results in Fig 9, where we show two views for each scene. The input original scene, four of our edited NeRF with different edit codes are shown in Fig 9. As can be seen, our method can successfully remove the target object from the scene, while the 3D consistency of the scene is maintained.
- **Video visualization.** We also provide a video of our inference edited 3D NeRF scenes for better visualization of the ability of our method in editing NeRF with multi-view consistency.

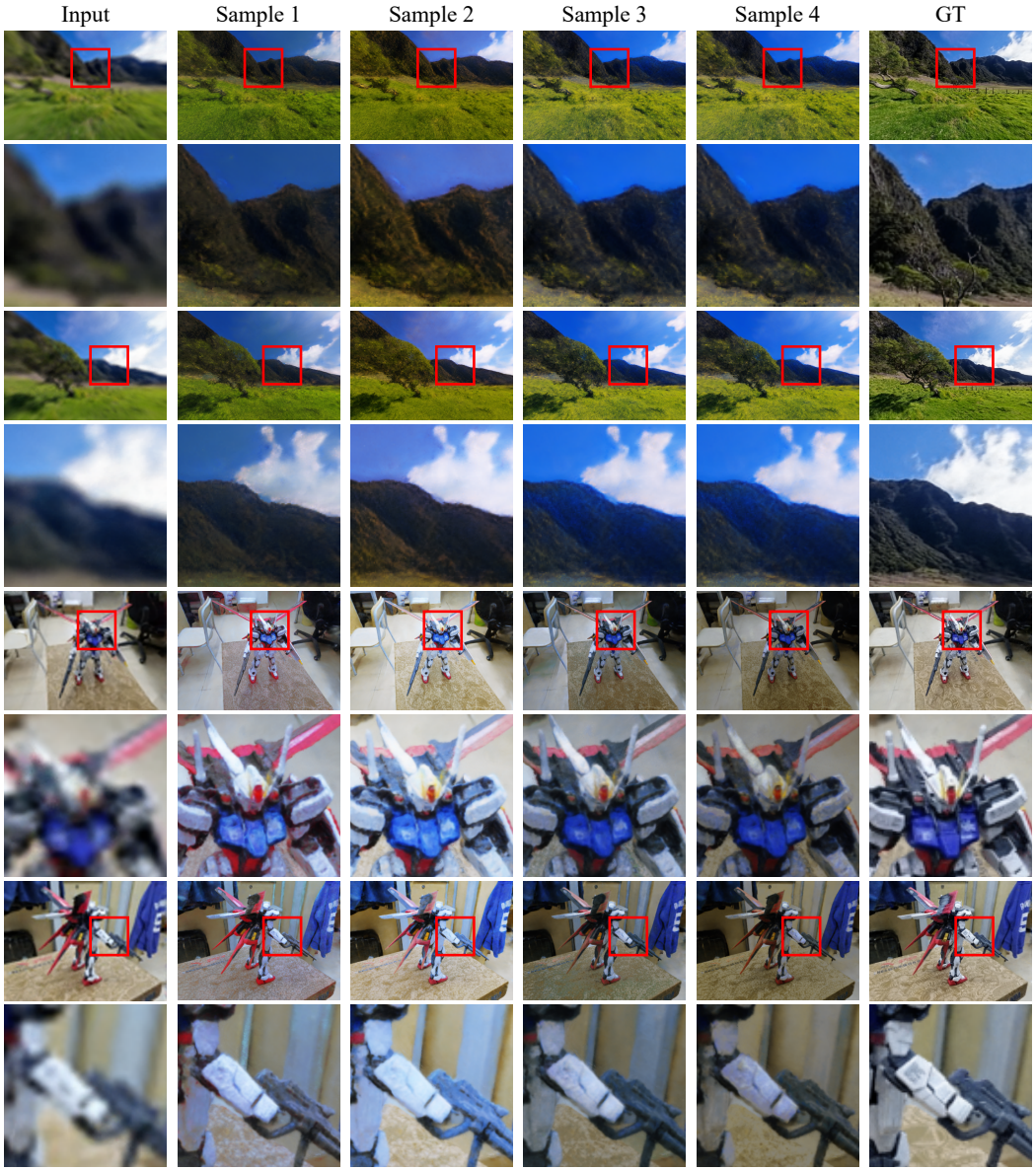

Figure 3: **Ours NeRF translation results for Super-resolution.** For each scene, we show two views of the input low-resolution scene, four of our edited NeRF rendering results from different edit code, and the Ground-truth scene. Zoom in results of the red boxes are also provided for better visualization.

REFERENCES

Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM Transactions on Graphics (TOG)*, 42(4):1–11, 2023.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022.

Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, et al. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023.

Figure 4: **Ours NeRF translation results for Super-resolution.** For each scene, we show two views of the input low-resolution scene, four of our edited NeRF rendering results from different edit code, and the Ground-truth scene.

Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. *arXiv preprint arXiv:2303.12789*, 2023.

Xiaoyang Kang, Tao Yang, Wenqi Ouyang, Peiran Ren, Lingzhi Li, and Xuansong Xie. Ddcolor: Towards photo-realistic and semantic-aware image colorization via dual decoders. *arXiv preprint arXiv:2212.11613*, 2022.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G Derpanis, Jonathan Kelly, Marcus A Brubaker, Igor Gilitschenski, and Alex Levinshtein. Spin-nerf: Multiview segmentation and perceptual inpainting with neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20669–20679, 2023.

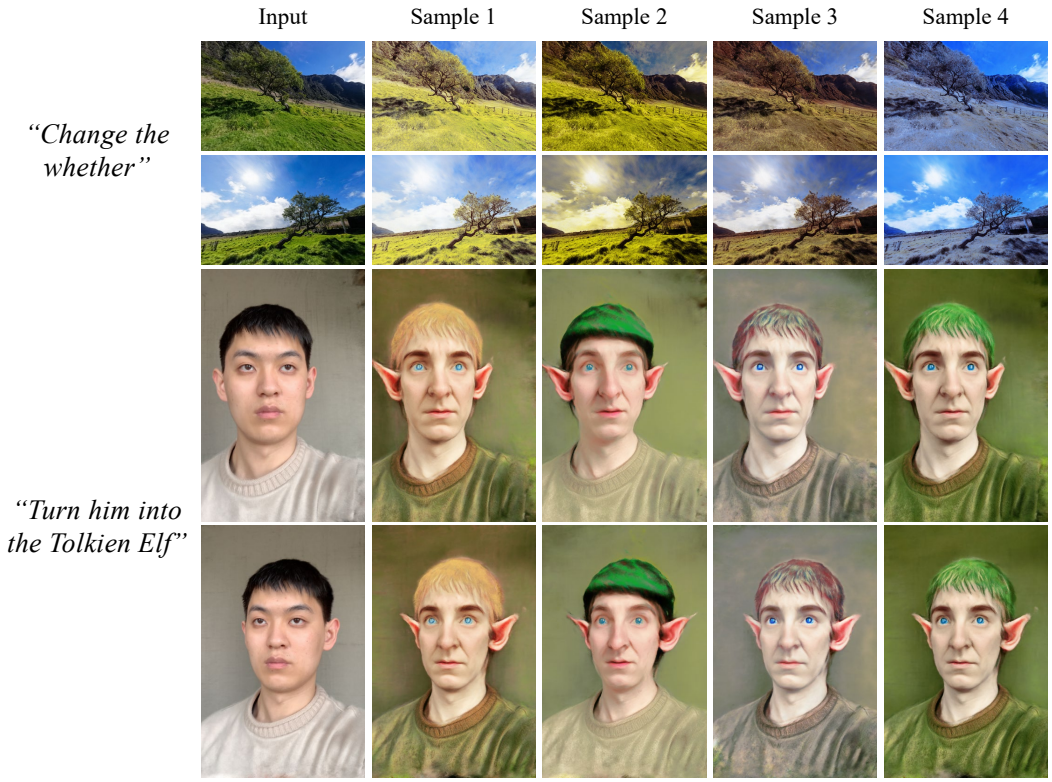

Figure 5: **Ours NeRF translation results for Text-driven editing.** For each scene, we show the input text prompt, two views of the input scene, four of our edited NeRF rendering results from different edit code.

Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, et al. Nerfstudio: A modular framework for neural radiance field development. In *ACM SIGGRAPH 2023 Conference Proceedings*, pp. 1–12, 2023.

Can Wang, Ruixiang Jiang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Nerf-art: Text-driven neural radiance fields stylization. *IEEE Transactions on Visualization and Computer Graphics*, 2023.

Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1790–1799, 2020.

Youtan Yin, Zhoujie Fu, Fan Yang, and Guosheng Lin. Or-nerf: Object removing from 3d scenes guided by multiview segmentation with neural radiance fields. *arXiv preprint arXiv:2305.10503*, 2023.

Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. *arXiv preprint arXiv:2307.12348*, 2023.

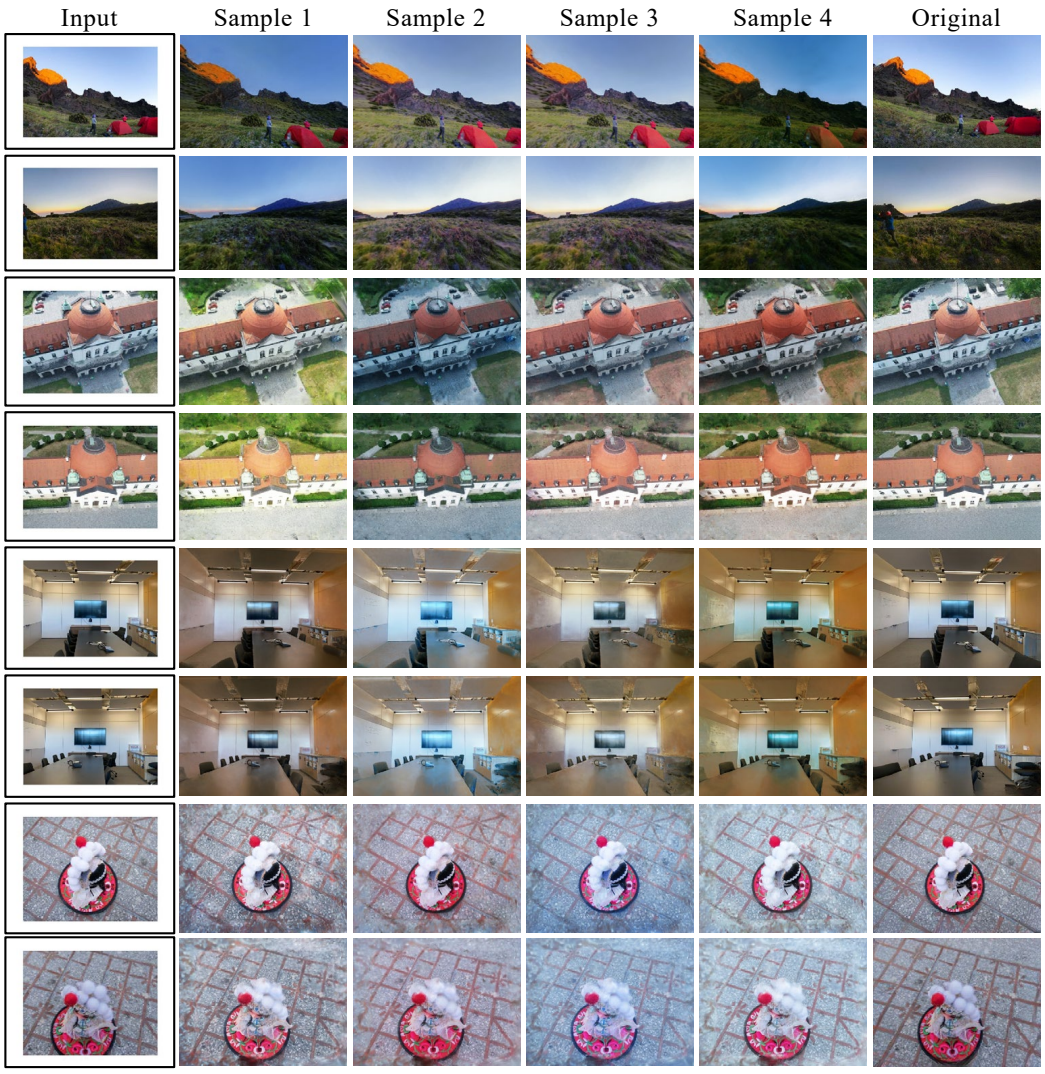

Figure 6: **Ours NeRF translation results for zoom out.** For each scene, we show the input image with the masked region, and four of our edited NeRF rendering results from different edit codes, and the original image. For each scene, we represent two rendering views for multi-view consistent check.

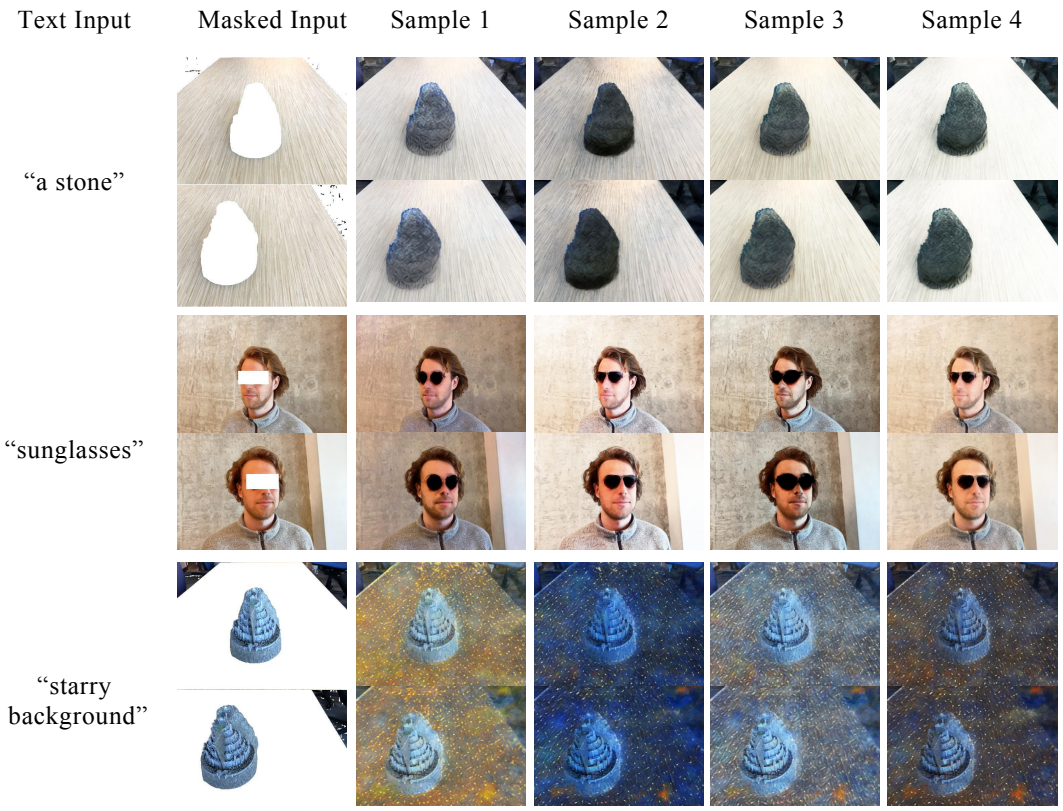

Figure 7: **Ours NeRF translation results for inpainting.** For each scene, we show the text input, the input image with the masked region, and four of our edited NeRF rendering results from different edit codes. For each scene, we represent two rendering views for multi-view consistent check.

Figure 8: **Ours NeRF translation results for colorization.** For each scene, we show two views of the input gray-scale scene, five of our edited NeRF rendering results from different edit code, and the Ground-truth scene.

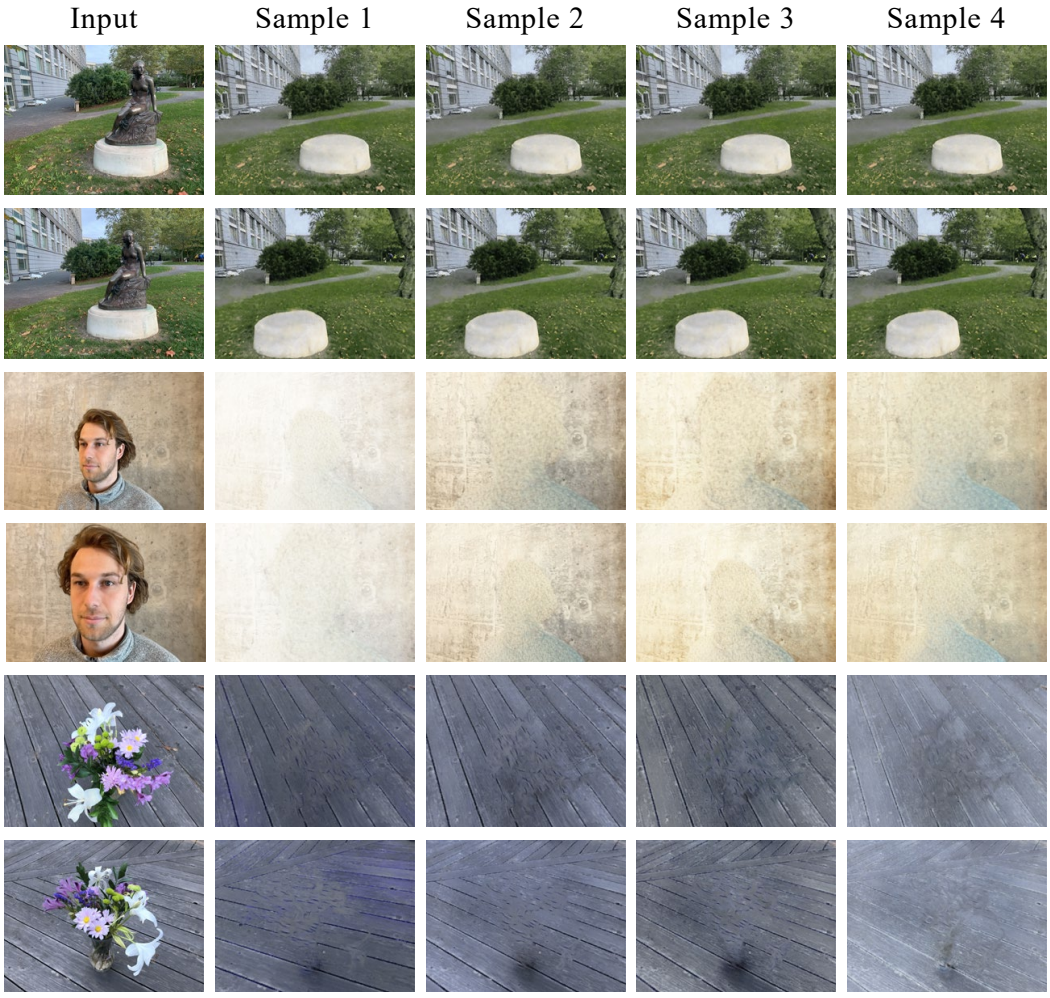

Figure 9: **Ours NeRF translation results for object removal.** For each scene, we show two views of the oribinal scene, five of our edited NeRF based on different edit code. As can be seen, our method can successfully remove the target object from the scene, while the 3D consistency of the scene is maintained.