Instruct-GS2GS: Editing 3D Gaussian Splats with Instructions

Cyrus Vachha and Ayaan Haque

UC Berkeley

Figure 1: **Editing Gaussian Splats with Instructions.** We propose Instruct-GS2GS, a method for consistent 3D editing of a 3D gaussian splatting scene using text-based instructions. Our method can perform the same edits as Instruct-NeRF2NeRF [6] while training and rendering faster.

Abstract

We propose a method for editing 3D Gaussian Splatting scenes with text-instructions. Our work is based largely off Instruct-NeRF2NeRF which proposes an iterative dataset update method to make consistent 3D edits to Neural Radiance Fields given a text instruction. We propose a modified technique to adapt the editing scheme for 3D gaussian splatting scenes. We demonstrate comparable results to Instruct-NeRF2NeRF and show that our method can perform realistic global text edits on large real-world scenes and individual subjects. Results videos can be found on our project page: https://instruct-gs2gs.github.io/

1. Introduction

Recent advances in photo-realistic novel 3D representations such as Neural Radiance Fields (NeRF) [11] and 3D Gaussian Splatting (3DGS) [9] have given way for a multitude of works exploring 3D generation, neural 3D reconstruction, and practical applications for these representations. Editing novel 3D representations like NeRF or 3DGS remains a challenge, and traditional 3D tools are generally incompatible with these representations. Instruct-NeRF2NeRF [6] describes a method to semantically edit NeRFs with text instructions. Instruct-NeRF2NeRF uses a 2D diffusion model, InstructPix2Pix [1], to iteratively edit the training dataset and update the NeRF simultaneously. Recently, 3DGS has gained popularity as a representation, but the Instruct-NeRF2NeRF algorithm cannot be naively applied to gaussian splatting. While NeRFs offer detailed

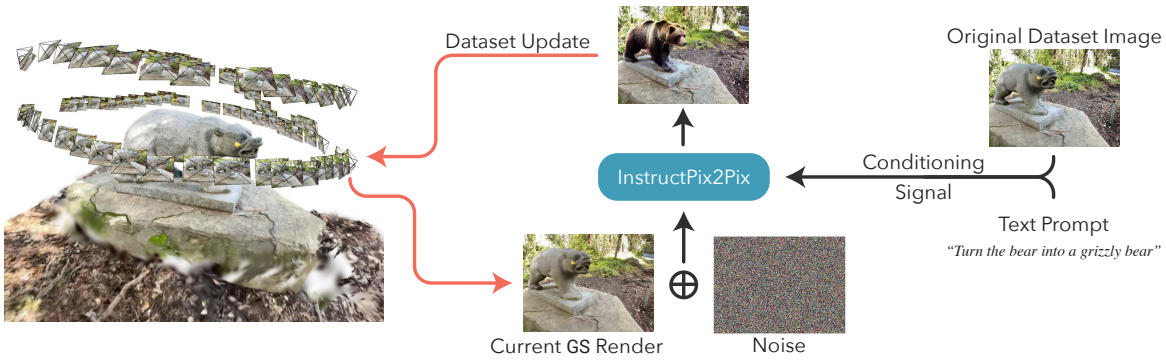

Figure 2: **Overview**: Our method gradually updates a reconstructed GS scene by updating the full dataset images while training the GS: (1) an image is rendered from the scene at a training viewpoint, (2) it is edited by InstructPix2Pix given a global text instruction, (3) the training dataset image is replaced with the edited image, and (4) this process is repeated for all images every 2.5k iterations and the GS continues training as usual. (Figure adapted from Instruct-NeRF2NeRF)

3D reconstructions, 3DGS has the primary advantage of real-time rendering speeds, making it a more suitable choice for integration with game engines, web compatibility, and virtual reality.

In this paper, we propose Instruct-GS2GS, a method to edit 3D Gaussian Splatting scenes and objects with global text instructions. Our method performs edits on a pre-captured 3DGS scene in a 3D consistent manner, similar to Instruct-NeRF2NeRF, while also having a much faster training and inference speed. Our method adapts the iterative dataset update approach from Instruct-NeRF2NeRF to work effectively for 3DGS. Our method is implemented in Nerfstudio [17], allowing users to perform edits quickly and view them in real-time.

2. Related Work

Artistic Stylization and Editing of NeRFs and GS Image stylization techniques have been extended into 3D consistent artistic stylization of NeRFs as demonstrated in various works [4, 8]. Other global stylization methods leverage language through CLIP and latent representations as demonstrated in NeRF-Art [20] and ClipNeRF [19]. While these works perform consistent edits, they are unable to perform localized edits in specified regions. Instruct-NeRF2NeRF performs edits to a NeRF from natural language instructions and has the ability to perform localized edits without masks. Concurrent works like GaussianEditor [5, 3] and Gaussian Grouping [22] demonstrate natural language edits applied on GS, in a similar method to Instruct-NeRF2NeRF, and explore other methods for localizing edits through specifying regions of interest or segmentation. These works also demonstrate methods for in-painting and object removal through segmentation and grouping.

Generating 3D Content Recent work demonstrates 2D text-conditioned diffusion models being used to perform 3D NeRF generation as shown in DreamFusion [14], as well as GS generation in DreamGaussians [18]. DreamFusion proposed the SDS loss, which uses a 2D diffusion model as a loss function on rendered images from a NeRF. Other works focus on generating a NeRF from sparse images [21, 25, 10] by synthesizing unseen views. Instead of entirely generating the scene, our work limits the inconsistency of 2D diffusion models by using a diffusion model which is conditioned on the original training images, enabling more realistic edits and generations on a given GS scene.

Instruction as an Editing Interface Recent LLMs [2, 12] offer simple text-based user interfaces for providing instructions. Other image diffusion models, including Instruct-Pix2Pix use text interfaces for performing edit instructions based on natural language. Instruct-NeRF2NeRF proposes a text interface for editing NeRFs in a similar manner. Our method also exhibits this simple interface to lower the barrier of entry to 3D content creation and to improve usability.

3. Method

Our method takes in a dataset of camera poses and training images, a trained 3DGS scene, and a user specified text-prompt instruction, e.g. "make him a marble statue". Instruct-GS2GS constructs the edited GS scene guided by the text-prompt by applying a 2D text and image conditioned diffusion model, in this case Instruct-Pix2Pix [1], to all training images over the course of training. Our iterative dataset update algorithm edits the entire dataset at once, trains the GS for 2.5k iterations, and repeats this process until it converges on the target edit. This process allows the GS to have a holistic edit and maintain 3D consistency.

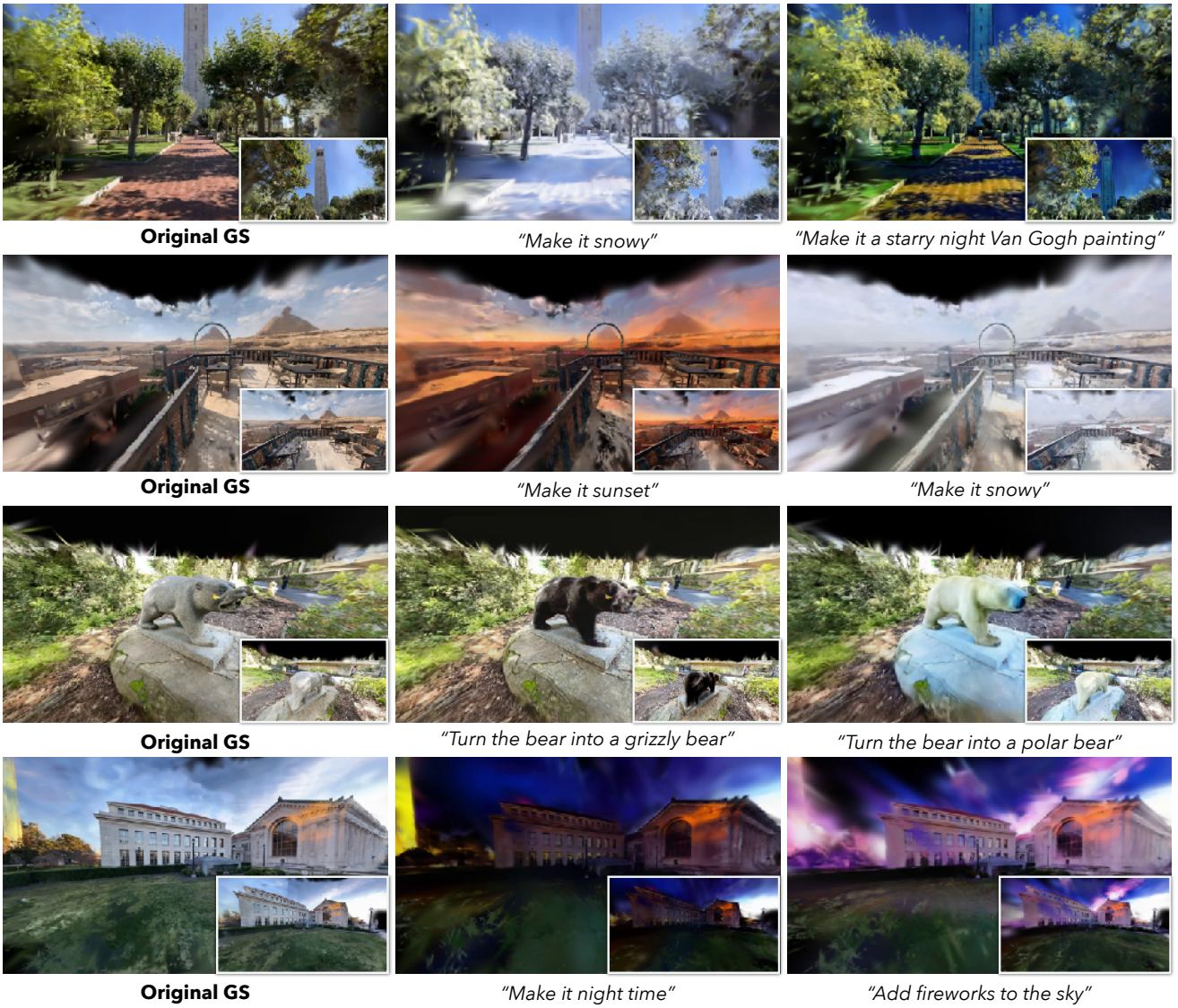

Figure 3: **Qualitative Results:** Our method is able to perform a variety of diverse contextual edits on real scenes, including stylistic or environmental changes, like adjusting the time of day, and even more localized changes that modify only a specific object in the scene.

3.1. Background

Gaussian Splatting 3D Gaussian Splatting [9] is a recent 3D representation visually similar to NeRF, which optimizes a scene represented by small 3D gaussians. These gaussian points represent an RGB color, opacity value, position, and a covariance matrix representing its shape. 3DGS trains and renders faster than NeRF with the help of differentiable rasterization, allowing for high fidelity real-time visualizations. A 3DGS scene, like NeRF, can be generated given a set of images with corresponding camera poses.

InstructPix2Pix Diffusion models are a class of generative models that transform a noisy image signal into a target data distribution by iteratively learning to denoise images [16, 7]. Instruct-Pix2Pix [1] is a text and image conditioned latent diffusion model that given an input conditioned image and a text prompt, generates a new edited image which respects both the original content in the image as well as the edit strength. The edit intensity and variance of the predicted image can be controlled by the classifier-free guidance scales.

Instruct-NeRF2NeRF Leveraging Instruct-Pix2Pix to perform edits on training images, Instruct-NeRF2NeRF [6]

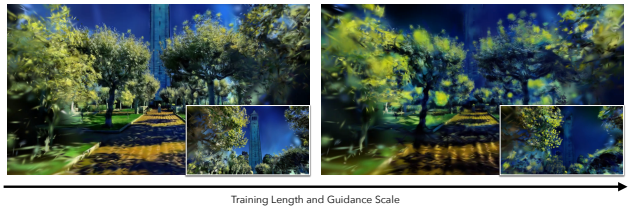

Figure 4: **Guidance Scale:** By varying the image guidance and number of full dataset updates, we can control how much the edit looks like the original scene. Note that these are renderings from the edited 3D scenes.

is able to perform 3D consistent edits over a captured NeRF. This work proposes iterative dataset update where at each iteration, a randomly chosen set of training images is replaced with an edited image from Instruct-Pix2Pix, which is conditioned on a rendering of the NeRF and the original image, ensuring that the edits are 3D consistent and can iteratively update the scene.

3.2. Instruct-GS2GS

Our method performs edits on a given gaussian splatting scene with its training dataset, based on a user-specified text-prompt. Over time, our model will converge the global scene edits closer to the prompt in a 3D consistent way as shown in Fig. 2. We perform these edits using an update scheme modified from Instruct-NeRF2NeRF in which all training dataset images are updated using a diffusion model individually, for sequential iterations spanning the size of the training images, every 2.5k training iterations. The edits from the diffusion model are consolidated when training the GS on the updated images.

Editing a rendered image In our method, we update the training images individually using a diffusion model. In particular, Instruct-Pix2Pix which receives a conditioning image, a user-specified text prompt, and a noisy image input. Our process is similar to Instruct-NeRF2NeRF where for a given training camera view, we set the original training image as the conditioning image, the noisy image input as the NeRF rendered from the camera combined with some randomly selected noise, and receive an edited image respecting the text conditioning. With this method we are able to propagate the edited changes to the GS scene. We are able to maintain grounded edits by conditioning Instruct-Pix2Pix on the original unedited training image.

Dataset Update We perform a modified version of the iterative dataset update proposed by Instruct-NeRF2NeRF where instead of iteratively updating a randomly selected training image, we update the whole dataset at a time. This is because unlike NeRF, 3DGS does not use rays for training and we need to render full images, meaning we cannot have a mixed signal of edited images and un-edited images. In our method, we edit every single image in the dataset update in a row every 2.5k training iterations. When updating the entire dataset, every following dataset update takes in the rendered image as input allowing the diffusion model to make stronger and more accurate 3D edits over time.

3.3. Implementation details

We use Nerfstudio's gsplat library [23] for our underlying gaussian splatting model. We adapt similar parameters for the diffusion model from Instruct-NeRF2NeRF. Among these are the values for $[t_{\text{min}}, t_{\text{max}}] = [0.70, 0.98]$, which define the amount of noise (and therefore the amount signal retained from the original images). We vary the classifier-free guidance scales per edit and scene, using a range of values from $s_I = (1.2, 1.5)$ and $s_T = (7.5, 12.5)$. We edit the entire dataset and then train the scene for 2.5k iterations. For GS training, we use L1 and LPIPS [24] losses. We train our method for a maximum of 30k iterations. However, in practice we stop training once the edit has converged. In many cases, the optimal training length is a subjective decision — a user may prefer more subtle or more extreme edits that are best found at different stages of training. This is shown in Figure 4.

4. Results

We conduct experiments on real scenes optimized using gsplat in Nerfstudio [17]. We edit a variety of scenes that vary in complexity: 360 scenes of environments, objects, and faces. The camera poses were extracted using either COLMAP [15] or through the PolyCam [13] app. The size of each dataset ranges from 50-300 images.

Our qualitative results are shown in Figure 1 and Figure 3. For each edit, we show multiple views to illustrate the 3D consistency. On the portrait capture in Figure 1, we are able to perform the same edits as Instruct-NeRF2NeRF, as well as new edits like "turn him into a Lego Man." In certain cases, the results look more 3D consistent and higher quality, and we provide a comparison in Figure 5. However, the gaussian splatting representation makes it challenging to add entirely new geometry. These edits also extend to subjects other than people, like changing a bear statue into a real polar bear, panda, and grizzly bear (Figure 3, last row). We are able to edit large-scale scenes just like Instruct-NeRF2NeRF [6], while maintaining the same level of 3D consistency. Most importantly, we find that our method outputs a reasonable result in around 13 min while Instruct-NeRF2NeRF takes approximately 50 min on the same scene.

Figure 5: **Comparison with Instruct-NeRF2NeRF:** Our results are on par with Instruct-NeRF2NeRF.

5. Conclusion

In this paper, we have introduced Instruct-GS2GS, a promising extension of Instruct-NeRF2NeRF, now enabling real-time rendering and faster training times. Our method allows for intuitive and simple 3D gaussian splatting editing using text instructions. Our method can take any precomputed 3D Gaussian Splatting scene and edit the scene and maintain 3D-consistency. We have demonstrated a wide variety of edits on environments, objects, and people.

6. Acknowledgements

We thank our instructors Alexei A. Efros and Angjoo Kanazawa for their support. This work was completed as a course project for CS180/280A. We would also like to thank the Nerfstudio and gsplat team for providing the 3D Gaussian Splatting implementation.

References

[1] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 1, 2, 3

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 2

[3] Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting, 2023. 2

[4] Pei-Ze Chiang, Meng-Shiun Tsai, Hung-Yu Tseng, Wei sheng Lai, and Wei-Chen Chiu. Stylizing 3d scene via implicit representation and hypernetwork, 2021. 2

[5] Jiemin Fang, Junjie Wang, Xiaopeng Zhang, Lingxi Xie, and Qi Tian. Gaussianeditor: Editing 3d gaussians delicately with text instructions, 2023. 2

[6] Ayaan Haque, Matthew Tancik, Alexei Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 1, 3, 4

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

[8] Yi-Hua Huang, Yue He, Yu-Jie Yuan, Yu-Kun Lai, and Lin Gao. Stylizednerf: consistent 3d scene stylization as stylized nerf via 2d-3d mutual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18342–18352, 2022. 2

[9] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. 1, 3

[10] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Realfusion: 360° reconstruction of any object from a single image. *arXiv e-prints*, pages arXiv–2302, 2023. 2

[11] 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. In *ECCV*, 2020. 1

[12] OpenAI. ChatGPT. 2

[13] Polycam. Polycam - lidar & 3d scanner for iphone & android. 4

[14] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022. 2

[15] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 4

[16] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Confer-*

ence on Machine Learning, pages 2256–2265. PMLR, 2015. 3

[17] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Justin Kerr, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, David McAllister, and Angjoo Kanazawa. Nerfstudio: A modular framework for neural radiance field development. *arXiv preprint arXiv:2302.04264*, 2023. 2, 4

[18] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation, 2023. 2

[19] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. *arXiv preprint arXiv:2112.05139*, 2021. 2

[20] Can Wang, Ruixiang Jiang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Nerf-art: Text-driven neural radiance fields stylization. *arXiv preprint arXiv:2212.08070*, 2022. 2

[21] Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P. Srinivasan, Dor Verbin, Jonathan T. Barron, Ben Poole, and Aleksander Holynski. Reconfusion: 3d reconstruction with diffusion priors, 2023. 2

[22] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3d scenes. *arXiv preprint arXiv:2312.00732*, 2023. 2

[23] Vickie Ye, Matias Turkulainen, and the Nerfstudio team. gsplat. 4

[24] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 4

[25] Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction, 2022. 2