SyncNoise: Geometrically Consistent Noise Prediction for Instruction-based 3D Editing

Ruihuang Li[1,2], **Liyi Chen**[1], **Zhengqiang Zhang**[1,2], **Varun Jampani**[3], **Vishal M. Patel**[4], **Lei Zhang**[1,2*]
[1]Hong Kong Polytechnic University
[2]OPPO Research Institute
[3]Stability AI
[4]Johns Hopkins University
{csrhli, cslzhang}@comp.polyu.edu.hk

Abstract

Text-based 2D diffusion models have demonstrated impressive capabilities in image generation and editing. Meanwhile, the 2D diffusion models also exhibit substantial potentials for 3D editing tasks. However, how to achieve consistent edits across multiple viewpoints remains a challenge. While the iterative dataset update method is capable of achieving global consistency, it suffers from slow convergence and over-smoothed textures. We propose SyncNoise, a novel geometry-guided multi-view consistent noise editing approach for high-fidelity 3D scene editing. SyncNoise synchronously edits multiple views with 2D diffusion models while enforcing multi-view noise predictions to be geometrically consistent, which ensures global consistency in both semantic structure and low-frequency appearance. To further enhance local consistency in high-frequency details, we set a group of anchor views and propagate them to their neighboring frames through cross-view reprojection. To improve the reliability of multi-view correspondences, we introduce depth supervision during training to enhance the reconstruction of precise geometries. Our method achieves high-quality 3D editing results respecting the textual instructions, especially in scenes with complex textures, by enhancing geometric consistency at the noise and pixel levels.

Project Page — https://lslrh.github.io/syncnoise.github.io/

Introduction

Text-based 3D scene editing is an emerging field that focuses on creating and manipulating 3D scenes using natural language instructions. Given an original 3D representation, one can achieve a wide variety of edits using abundant and flexible textual instructions, such as modifying the geometry, appearance, lighting, textures, and other attributes of the scene to achieve desired effects or fulfill design objectives. Despite the advancements in 3D generative diffusion models (Hong et al. 2023; Wang et al. 2024), it still requires a significant amount of paired 3D scene data to adapt these models for 3D editing tasks. Given the limited availability of such data, an alternative approach is to distill prior knowledge from 2D diffusion models to improve 3D representations.

*Corresponding author.

Diffusion-based image editing approaches, including text-driven image synthesis and editing (Rombach et al. 2022; Hertz et al. 2022), stroke-based editing (Meng et al. 2021), exemplar-based methods (Yang et al. 2023), and point-based editing (Shi et al. 2023) have achieved considerable success and facilitated artistic creation. Despite the increasing maturity and accessibility of 3D reconstruction techniques (Mildenhall et al. 2021; Kerbl et al. 2023), applying 2D editing strategies to 3D scenes has not been extensively studied in the literature. One straightforward solution is to utilize a 2D diffusion model to edit each view separately, and then use the edited multi-view images to update the 3D representations to obtain the desired shapes and textures. However, due to the inherent randomness of diffusion process and the lack of 3D priors, it is challenging for a 2D model to generate multi-view consistent editing results in terms of geometry, lighting, and textures simultaneously.

To alleviate this issue, Instruct-Nerf2Nerf (Haque et al. 2023) (IN2N) presents an iterative dataset update framework to alternatively edit one randomly selected view with InstructPix2Pix (Brooks, Holynski, and Efros 2023) and optimize the 3D scenes based on the edited image. Although IN2N can achieve globally consistent editing, it suffers from *longer optimization duration* to obtain a satisfactory edited scene. Besides, it *eliminates fine-grained details* that are not consistent across views, leading to over-smoothed results.

To improve the editing efficiency, Efficient-Nerf2Nerf (Song et al. 2023) (EN2N) incorporates multi-view consistency regularization into the diffusion process and achieves consistent outputs in a single pass. However, this approach suffers from blurry results because it imposes a consistency constraint on the latent codes of diffusion models, which tends to collapse the rich and nuanced latent representation into a more averaged form, consequently leading to a loss of high-frequency details and subtle variations that are critical for realistic editing.

In order to avoid the blurred editing results and generate finer-grained textures, in this paper we propose **SyncNoise**, a geometry-aware multi-view synchronized noise prediction method for 3D scene editing. Firstly, we leverage geometric information of 3D scenes to achieve precise and dense multi-view matching, which paves the way for applying multi-view consistency constraints at the noise and pixel levels.

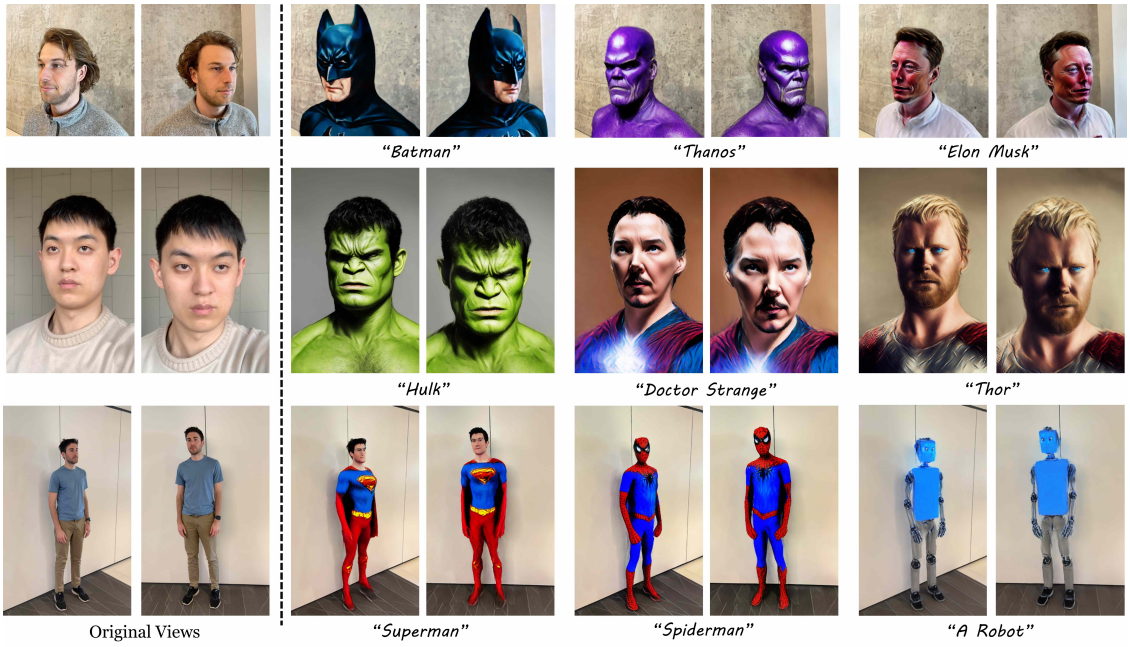

Figure 1: **SyncNoise** achieves high-quality and controllable editing that closely adheres to the instructions with minimal changes to irrelevant regions. It attains geometrically consistent editing without compromising fine-grained textures.

Since implicit 3D representations, such as Neural Radiance Field (NeRF) models, often suffer from unreliable geometry fitting, we introduce additional depth supervision produced by running Structure-from-Motion (SfM) (Schönberger and Frahm 2016; Schönberger et al. 2016) to improve the geometric reconstruction, avoiding aligning non-matched regions of different views.

Secondly, motivated by the observation that intermediate features of the noise predictor (U-Net) not only involve semantic information but also exhibit the structure-to-appearance controllability (Zhang et al. 2024; Liu et al. 2024a; Voynov et al. 2023), we enforce multi-view consistency on the U-Net features for predicting noise maps, rather than on the latent map. This not only effectively mitigates the smoothed results by performing average operations on the latent map, but also achieves multi-view consistent edits in semantic structure and low-frequency appearance. Since solely manipulating the noise predictions cannot ensure consistent high-frequency details across adjacent views, we further employ a cross-view projection strategy to propagate the anchor views to others for improving the pixel-level consistency. Fig. 1 shows some editing results on different 3D scenes. We can observe that by leveraging the geometric information to synchronously predict multi-view noise maps, and propagating well-edited view to its neighboring views, our proposed SyncNoise can achieve consistent and efficient 3D edits respecting the textual instructions and retain more details in edited scenes.

Related Work

Image Generation and Editing. Recently, diffusion models (Ho, Jain, and Abbeel 2020) have demonstrated excellent semantic understanding capability for image generation. Conditioned on a given textual prompt, DALL-E-2 (Ramesh et al. 2022) and Stable Diffusion (Rombach et al. 2022; Zhang, Li, and Zhang 2024) achieve impressive generation performance using classifier-free guidance. Most image editing methods inherit the prior knowledge of pre-trained generation models to modify the appearance and shapes of reference images while preserving their original structure. Prompt2Prompt (Hertz et al. 2022) (P2P) aligns the source attention maps of source and edited images with the given text prompts to achieve localized editing. IP2P (Brooks, Holynski, and Efros 2023) extends P2P to support instruction-based efficient editing. The following work Plug-and-Play (Tumanyan et al. 2023) injects the reference feature and self-attention layer to control the editing process. Delta denoising score (DDS) (Hertz, Aberman, and Cohen-Or 2023) extends score distillation sampling to avoid background changes. Personalized generation shares similarity with image editing. Textural inversion (Gal et al. 2022) expands the language-vision dictionary to inject the subject content into a word embedding to achieve subject-driven generation. Similarly, DreamBooth (Ruiz et al. 2023) inverses the subject by finetuning the Stable Diffusion to achieve better fidelity. The following studies (Mokady et al. 2023; Li et al. 2024a) improve the editing quality by preserving the context from the inversion process. These 2D editing studies provide a good starting point for 3D editing.

3D Scene Editing. Many studies have explored editing neural fields in different manners. Driven by the development of LLMs and multi-modality models (Radford et al. 2021; Rombach et al. 2022), instruction-based editing (Chen et al. 2024; Rojas et al. 2024) has attracted much

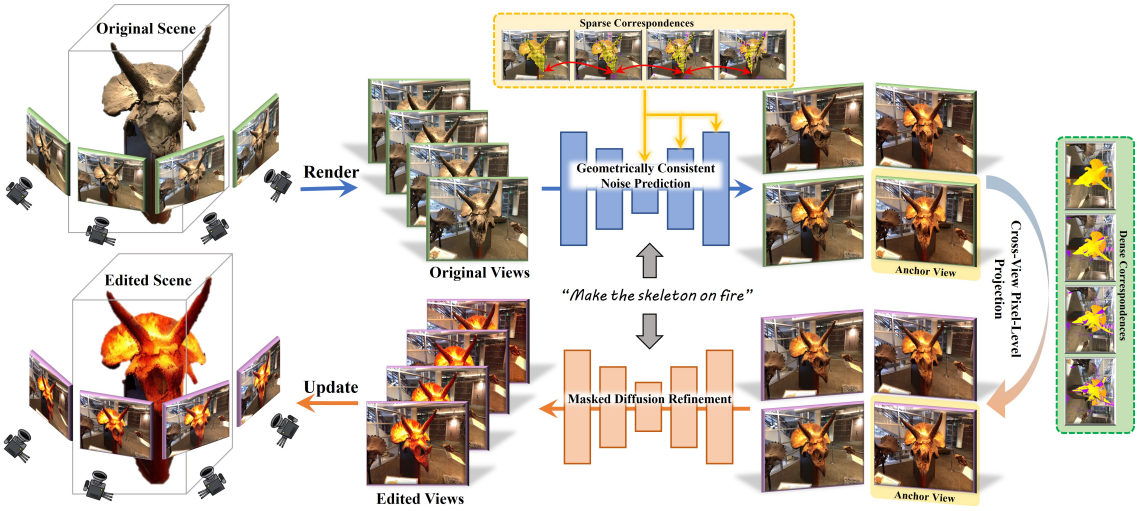

Figure 2: Overview of SyncNoise for instruction-based 3D editing. We edit rendered multi-view images while enforcing geometrical consistency at the noise and pixel levels. Based on reliable correspondences built upon 3D geometries, we first enforce coarse consistency by aligning U-Net decoder features across views. Then we use cross-view pixel-level projection to achieve fine-grained consistency by propagating the anchor view to its neighboring views. To remove artifacts led by reprojection, we refine these views with masked diffusion. Finally, we update the 3D scene based on the edited multi-view images.

attention due to its user-friendly nature. NeRF-Art (Wang et al. 2023) and ClipNeRF (Wang et al. 2022) edit global NeRF by maximizing the CLIP similarity between rendered 2D views and text prompt. FocalDreamer (Li et al. 2024b) and Instruct-3Dto3D (Kamata et al. 2023) optimize the 3D models using SDS loss (Poole et al. 2022) from pre-trained Stable Diffusion and Instruct-Pix2Pix, respectively. Similarly, Shap-Editor (Chen et al. 2023a) learns a feed-forward network to directly output the edited NeRF latent. To enable fine-grained localized editing, Distilled Feature Fields (Kobayashi, Matsumoto, and Sitzmann 2022) and Neural Feature Fusion Fields (Tschernezki et al. 2022) introduce pre-trained 2D models DINO (Caron et al. 2021) for localization. More recently, Instruct-Nerf2Nerf (Haque et al. 2023) iteratively updates 3D model and edits rendered images. DreamEditor (Zhuang et al. 2023) leverages DreamBooth (Ruiz et al. 2023) for subject-driven editing under the text prompt without sacrificing the fidelity to original object. GenN2N (Liu et al. 2024b) distills the priors from off-the-shelf 2D models in latent space to achieve 3D editing. GaussianEditor (Chen et al. 2023b) maintains a dynamic mask (Li et al. 2023) for localized editing based on 3D Gaussians.

Method

In this work, we focus on text-based 3D scene editing by resorting to 2D diffusion models. Given an original 3D representation (NeRF or Gaussian Splatting), multi-view images and their camera poses, we aim to produce an edited scene under the guidance of natural-language instructions.

As shown in Fig. 2, we leverage instruction-based 2D diffusion models to edit multi-view images, followed by optimizing the original 3D representations using the edited views as supervision. Ensuring multi-view consistent editing is crucial, as any inconsistencies in textures between views can lead to undesirable smoothing effects. To this end, we first leverage 3D geometry to establish precise multi-view correspondences. Secondly, we impose multi-view consistency constraint on the noise predictions throughout the denoising (editing) process, for enhancing semantic and appearance coherence across the views. Furthermore, to preserve more high-frequency details, we employ cross-view projection to propagate the editing effects from anchor views to their neighboring views, so as to achieve multi-view consistent edits at the pixel level.

Reliable Geometry-guided Correspondence

To establish reliable correspondences among multiple views, we incorporate depth supervision to enhance the reconstructed geometry. Furthermore, we leverage the reprojected depth and cycle consistency constraints to filter out unreliable matching points, ensuring the matching accuracy.

Depth Supervision. The implicit 3D representation, such as NeRF, exhibits limited capability in fitting geometry, particularly in scenarios with sparse views. Consequently, the predicted depth by NeRF tends to be unreliable. As shown in Fig. 3(a), there are significant offsets when reprojecting points from reference view to others. To address this limitation, we follow (Deng et al. 2022) to introduce depth supervision into the training process of NeRF. Specifically, we derive the depth supervision from 3D keypoints obtained by running Structure-form-Motion (SfM) solver (Schönberger and Frahm 2016), and add a depth loss to enforce the estimated depth to match the depth of keypoints. As shown in Fig. 3(b), by adding the depth supervision, we are able to estimate more precise depth, which in turn enables us to build dense and accurate correspondences among different views.

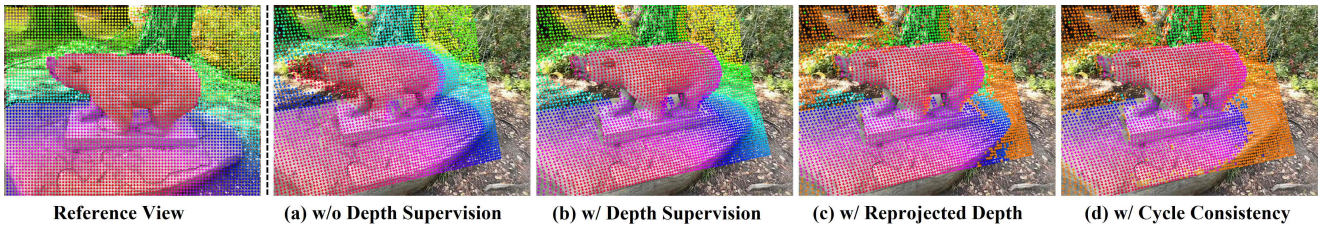

Figure 3: The estimated depth on reference view D_{ref} and the re-projected depth from reference view to novel view $D_{ref\to k}$. By imposing the depth supervision and two constraints, we can obtain reliable geometric correspondences across views. **Orange** denotes noisy points to be filtered.

Reprojected Depth Constraint. While explicit depth supervision can enhance the quality of 3D geometry, there are still deviations in the matching points due to the noise of 3D key points. To filter out noisy correspondences, we compare the reprojected depth from the reference view I_{ref} to the k-th novel view I_k, denoted by $D_{ref\to k}$, with the estimated depth on the k-th view, denoted by D_k, and retain the matching points that satisfy the following condition:

$$|D_{ref\to k} - D_k| < \tau_d, \tag{1}$$

where τ_d denotes the depth threshold used to eliminate noisy matching points. As can be observed in Fig. 3(c), most of background points, occluded points in the novel view, and points located at the edges of objects have been filtered out. As the span between views increases, the number of reliable matching points gradually decreases.

Cycle Consistency Constraint. In addition to the reprojected depth constraint, reliable matching points should also adhere to cycle consistency constraint. The pixel distance between the point back-projected from I_k to I_{ref}, denoted by $P_{ref\to k\to ref}$, and its original starting point, denoted by P_{ref}, should satisfy the following condition:

$$|P_{ref\to k\to ref} - P_{ref}| < \tau_p, \tag{2}$$

where $|\cdot|$ calculates the pixel distance between two points, and τ_p is the threshold used to filter out noisy points that can not be back-projected to their original locations. *Please refer to **Appendix A.1** for more details.*

Geometrically Consistent Noise Prediction

Building upon the precise geometric correspondences we have constructed, in this section we aim to enforce the editing results from multiple views to be consistent throughout the whole denoising process from T to 0 steps. A simple and effective approach to achieve this goal is by averaging the corresponding latent features across multiple views (Song et al. 2023). However, this method has two major limitations. On one hand, directly manipulating the latent maps can lead to smoothed results in generated images, as shown in Fig. 4(b). On the other hand, assigning equal weights to different views is not reasonable due to varying qualities of matching points across views, making the model biased to views with poor correspondences.

Prior studies (Zhang et al. 2024; Voynov et al. 2023; Liu et al. 2024a) have demonstrated that the intermediate features of noise predictor (U-Net) not only capture semantic information but also influence the final appearance of image. This motivates us to enhance multi-view consistency on the U-Net features rather than latent maps. In Fig. 4(d), it can be observed that by enforcing consistency on the intermediate decoder features of U-Net, we can achieve multi-view consistent editing results without introducing blurred artifacts. When the constraint is applied to skip features, the impact is relatively minor, as shown in Fig 4(c). *Please refer to **Appendix A.8** for the architecture of U-Net and the effects of aligning different layers of U-Net.*

Figure 4: Multi-view editing results obtained (a) without alignment, (b) by aligning latent features of different views, by enforcing consistencies on (c) skip features and (d) decoder features of U-Net. By enforcing the decoder features of noise predictor to be consistent, we can obtain multi-view consistent edits without introducing blurs. The text prompt is "*make the man look like Tolkien Elf*".

Initial Noise Alignment. We first align the initial noise from multiple views. Specifically, given the random noise from K different views, denoted by $\{Z_T^1, \cdots, Z_T^K\}$, and the correspondences among them, we define the noise of each reference view as the weighted sum of noises from K views:

$$Z_T^{ref,i} = \sum_{k=1}^{K} \sqrt{w_k^i} \cdot \mathbf{1}_{match}^{k,i} \cdot Z_T^{k,i}, \tag{3}$$

where $Z_T^{ref,i}$ denotes the noise vector of the i-th point in the

reference view. $\mathbf{1}^{k,i}_{match}$ is an indicator function that equals 1 if the k-th view contains a matching point for the i-th point in the reference view. w^i_k represents the weight assigned to the k-th view, which is inversely proportional to the reprojection error, denoted by $\delta^i_k = |D_{ref\to k} - D_k|$. The weight is defined as follows:

$$w^i_k = e^{-\mu\cdot\delta^i_k}, \tag{4}$$

where μ controls the relative gaps between weights on different views. Each weight is normalized by the sum of weights for all matching points.

Masked Multi-View Consistent Noise Editing. In addition to initial random noise, we also align the noise predictions across all diffusion steps $t \in [0, \cdots, T]$. Given the l-th layer features of noise predictor from K different views, denoted by $\{F^1_l, \cdots, F^K_l\}, l \in \{1, \cdots, 11\}$, we aggregate multi-view noise features corresponding to the i-th point into the I_{ref} through the following formula:

$$F^{ref,i}_l = \sum_{k=1}^{K} w^i_k \cdot \mathbf{1}^{k,i}_{match} \cdot F^{k,i}_l. \tag{5}$$

Furthermore, to achieve more precise foreground editing without modifying irrelevant regions, we introduce masks to restrict the matching and editing regions. We retain only the correspondences within the mask, and filter out redundant associations from unrelated regions. In addition, during each denoising step, we apply a mask to limit the region of text guidance and modify the noise estimate equation as follows (*please refer to* ***Appendix A.2*** *for more details about this equation.*):

$$\begin{aligned}\hat{\epsilon}_\theta(z_t, c_I, c_T) = \epsilon_\theta(z_t, \varnothing, \varnothing) + g_I \cdot (\epsilon_\theta(z_t, c_I, \varnothing) - \epsilon_\theta(z_t, \varnothing, \varnothing)) \\ + g_T \cdot (\epsilon_\theta(z_t, c_I, c_T) - \epsilon_\theta(z_t, c_I, \varnothing)) \cdot M_{soft},\end{aligned} \tag{6}$$

where c_I, c_T and $\varnothing$ denote the image, text, and no conditions, respectively. ϵ_θ is the denoising U-Net. g_I and g_T are two classifier-free guidance scales for balancing the quality and diversity of samples generated by the diffusion model. It is worth noting that we employ a **soft mask**, denoted by M_{soft}, instead of a binary mask for masked noise estimation. This is because *trivially reducing the background weights to zero would also decrease the editing fidelity on foreground*. Specifically, the weights of foreground regions are set to 1, while the weights of backgrounds gradually decay from 0.5 to 0 as they move away from the center of foreground.

Cross-View Pixel-Level Projection

We have aligned the initial noise and noise predictions of U-Net from multiple views, which can achieve globally consistent edits in a more efficient manner than the iterative refinement strategy (Haque et al. 2023). However, as shown in Fig. 5(b), noise-level alignment can only ensure consistency in semantic structure and low-frequency textures, but cannot guarantee consistency in high-frequency details. Even a small misalignment in these details can ultimately result in smoothed textures in 3D edits.

Figure 5: Multi-view editing results. Noise alignment (b) is responsible for consistent edits in semantic structure and low-frequency appearance, while cross-view pixel reprojection (c) ensures the consistency in high-frequency details. The text prompt is "*Turn him into an Egyptian sculpture*".

Cross-View Projection. To preserve more fine-grained textures in the edited 3D scenes, we need to perform pixel-level alignment. Specifically, we propose to propagate partial anchor views to others based on the established dense correspondences. First, we utilize metrics such as CLIP directional similarity score to select well-edited views as the anchors, denoted by I^e_{anchor}. Then we reproject each I^e_{anchor} to its neighboring views I^e_k and use the reprojected pixels to replace the corresponding pixels in I^e_k:

$$I^e_k[M^{valid}_{anchor\to k}] = I^e_{anchor}[M^{valid}_{anchor}], \tag{7}$$

where $M^{valid}_{anchor\to k}$ and M^{valid}_{anchor} indicate valid correspondences satisfying the depth and cycle consistency constraints in I^e_k and I^e_{anchor}, respectively.

Masked Diffusion Refinement. As shown in Fig. 5(c), cross-view reprojection further improves the consistency in fine-grained details across adjacent views. However, it may cause artifacts in novel views. To address this issue, we further perform masked refinement by feeding the edited view I^e_k into the 2D editing model as follows:

$$\begin{aligned}\hat{\epsilon}_\theta(z_t, c_{I^e_k}, c_T) = \epsilon_\theta(z_t, \varnothing, \varnothing) + g_I \cdot (\epsilon_\theta(z_t, c_{I^e_k}, \varnothing) - \epsilon_\theta(z_t, \varnothing, \varnothing)) \\ + g_T \cdot (\epsilon_\theta(z_t, c_{I^e_k}, c_T) - \epsilon_\theta(z_t, c_{I^e_k}, \varnothing)) \cdot (M_k - M^{valid}_{anchor\to k}).\end{aligned} \tag{8}$$

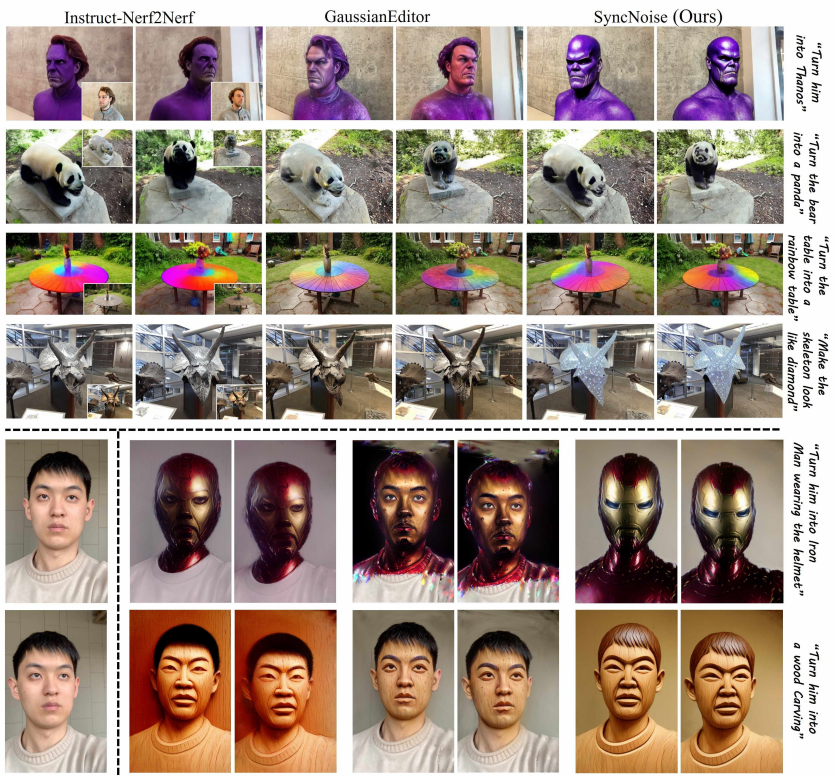

Figure 6: Qualitative comparisons. SyncNoise produces consistent (*e.g.* "rainbow table"), finer-grained (*e.g.* "wood carving"), and instruction-following 3D editing (*e.g.* "Iron Man", "Thanos") with minimal changes to irrelevant regions.

Unlike Eq. 6, in this equation, the edited image latent rather than the original image is used as image condition. Additionally, the text guidance is only applied on the regions not replaced by anchor views, which are represented by $(M_k - M^{valid}_{anchor \to k})$, where M_k denotes the mask of target object. As can be seen in Fig. 5(d), through masked refinement, the prior knowledge from I^e_{anchor} is incorporated into the unprojected regions of I^e_k.

3D Representation Optimization

The proposed multi-view synchronized noise prediction achieves consistent edits in both structure and appearance, while the cross-view pixel-level projection further enhances the consistency among neighboring views. Based on the edited results of all views, we first train the 3D model for 1000-2000 iterations, depending on the complexity of the scenes, to inject the 2D edits into 3D representation. Subsequently, we employ an iterative refinement (Haque et al. 2023) approach to further enhance the 3D representation. Note that our approach differs from IN2N (Haque et al. 2023) in a key aspect. In IN2N, during the early optimization steps, the multi-view image edits exhibit significant inconsistencies, leading to blurry 3D edits. While our method first generates multi-view consistent 2D edits to ensure general consistency, and then employs an iterative refinement process to adjust finer details.

Experiments

Implementation Details. During the editing process, we first edit 80 multi-view images while enforcing consistency on the layer-5 and layer-8 of U-Net features. Subsequently, for the anchor view selection, we pick the view with the highest CLIP direction score in every 10 adjacent views as the anchor view, and reproject them onto neighboring views with about 80% overlap. *Please refer to* ***Appendix A.3*** *for more implementation details and evaluation metrics.*

Qualitative Results

In Fig. 1, we demonstrate some edits with different text prompts. As can be observed from the edits with prompts "Batman" and "Robot", our method still exhibits multi-view consistency even when the geometry and shape of original scenes undergo obvious changes. Additionally, we can see finer details in the hair of "Hulk". *Please refer to* ***Appendix A.6*** *for more results on non-human scenes.*

We compare our SyncNoise with two representative instruction-based methods, Instruct-Nerf2Nerf (Haque et al. 2023) and GaussianEditor (Fang et al. 2023) in Fig. 6. We reproduce the results of compared methods with their official codes and default parameters. Our SyncNoise achieves realistic and consistent edits that are faithful to the input textual instruction. In the example of "Rainbow table", our edits exhibit better multi-view consistency compared to the other two methods. IN2N exhibits color blending issues due to in-

Method	3D Model	Noise	Pixel	CLIP Score↑	CLIP Text-Image Direction Similarity↑	CLIP Temporal Direction Similarity↑	CLIP-IQA↑	MUSIQ↑	Avg. Time
IN2N (Haque et al. 2023)	NeRF			29.18%	16.49%	90.12%	0.489	64.729	57min
EN2N (Song et al. 2023)	NeRF			28.46%	15.74%	90.47%	0.496	63.853	19min
GaussianEditor (Fang et al. 2023)	GS			26.55%	17.04%	88.42%	0.511	64.425	11min
SyncNoise	NeRF	✓		30.64%	17.93%	91.42%	**0.559**	65.901	23min
			✓	29.50%	16.97%	89.90%	0.504	65.221	
		✓	✓	**30.86%**	**18.31%**	92.04%	0.540	**66.668**	
	GS	✓	✓	30.54%	18.14%	**92.54%**	0.524	65.471	7min

Table 1: Quantitative evaluation. SyncNoise achieves high fidelity to the instructions without sacrificing the visual quality.

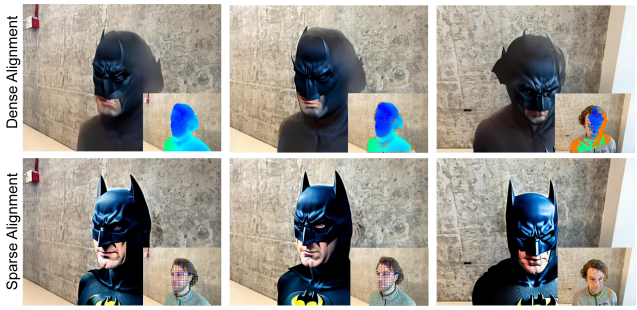

Figure 7: Comparison between dense and sparse alignment.

consistent edits in each iteration. For the example "Wood carving", our SyncNoise successfully edits even the hair and produces **fine-grained** textures. In addition, we generate highly realistic helmet for "Iron Man". However, GuassianEditor hardly changes the appearance, as it restricts the updates of old Gaussian points, hindering their editing fidelity to texts. Our method achieves superior edits by enforcing global structure and local texture to be consistent. *Please see the **Appendix A.4** and **A.5** for more qualitative comparisons with IN2N and DreamEditor (Zhuang et al. 2023).*

Quantitative Comparison

We provide the quantitative results between SyncNoise, IN2N (Haque et al. 2023), EN2N (Song et al. 2023) and GaussianEditor (Fang et al. 2023) in Tab. **??**. We evaluate all the compared methods on a total of four scenes (*i.e.*, 'bear', 'face', 'fangzhou' and 'person') and 10 different text prompts. One can see that our method achieves superior editing performance on not only editing fidelity but also visual quality. Our method achieves better instruction-following edits and better temporal consistency, compared to IN2N, while requiring only half the editing time. Besides, our method outperforms GaussianEditor by 1.27% and 2.243 in terms of CLIP text-image direction similarity score and MUSIQ, respectively. GaussianEditor limits the update of partial 3D Gaussians points of original scene so that it cannot adhere to the instructions very well. By introducing pixel-level consistency, SyncNoise further enhances the fidelity to instruction and visual quality, achieving finer-grained editing details across different views.

Ablation Study

Sparse Alignment. We compare multi-view editing results based on dense and sparse alignments over U-Net features in Fig. 7. Although dense alignment achieves more consistent edits across views, it cannot change the geometry of objects and leads to over-smoothed artifacts. In contrast, sparse alignment can successfully edit the geometry of objects while also achieving multi-view consistency.

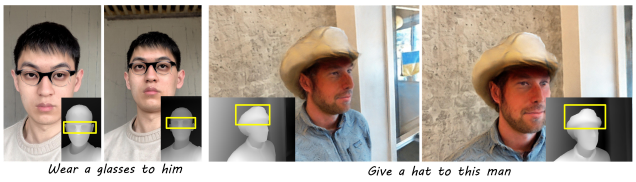

Figure 8: Geometry editing results.

Figure 9: Effect of masked diffusion refinement.

Geometry Editing. We provide the editing results and corresponding depth maps in Fig. 8. Our method can successfully edit the geometry of target object. This is because we enforce multi-view consistency on the semantically rich U-Net features, which can affect the geometries of objects.

Masked Refinement. As shown in Fig. 9, our method can successfully remove most of the artifacts caused by cross-view pixel-level projection without affecting the editing effects from the anchor view.

Conclusion

In this work, we focused on achieving multi-view consistent edits in 3D scene editing. We proposed a novel approach called SyncNoise, which leveraged geometry-guided multi-view consistency to enhance the coherence of edited scenes. By synchronously editing multiple views using a 2D diffusion model and enforcing geometric consistency on the features of noise predictor, we avoided blurred outcomes. The pixel-level reprojection between neighboring views further helped generate more fine-grained details. Our experimental results demonstrated that SyncNoise outperformed existing methods in terms of achieving high-quality 3D editing while respecting textual instructions.

References

Brooks, T.; Holynski, A.; and Efros, A. A. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18392–18402.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9650–9660.

Chen, J.-K.; Bulò, S. R.; Müller, N.; Porzi, L.; Kontschieder, P.; and Wang, Y.-X. 2024. ConsistDreamer: 3D-Consistent 2D Diffusion for High-Fidelity Scene Editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21071–21080.

Chen, J.-K.; Lyu, J.; and Wang, Y.-X. 2023. Neuraleditor: Editing neural radiance fields via manipulating point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12439–12448.

Chen, M.; Laina, I.; and Vedaldi, A. 2024. DGE: Direct Gaussian 3D Editing by Consistent Multi-view Editing. *arXiv preprint arXiv:2404.18929*.

Chen, M.; Xie, J.; Laina, I.; and Vedaldi, A. 2023a. SHAP-EDITOR: Instruction-guided Latent 3D Editing in Seconds. *arXiv preprint arXiv:2312.09246*.

Chen, Y.; Chen, Z.; Zhang, C.; Wang, F.; Yang, X.; Wang, Y.; Cai, Z.; Yang, L.; Liu, H.; and Lin, G. 2023b. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. *arXiv preprint arXiv:2311.14521*.

Deng, K.; Liu, A.; Zhu, J.-Y.; and Ramanan, D. 2022. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12882–12891.

Fang, J.; Wang, J.; Zhang, X.; Xie, L.; and Tian, Q. 2023. Gaussianeditor: Editing 3d gaussians delicately with text instructions. *arXiv preprint arXiv:2311.16037*.

Gal, R.; Alaluf, Y.; Atzmon, Y.; Patashnik, O.; Bermano, A. H.; Chechik, G.; and Cohen-Or, D. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*.

Haque, A.; Tancik, M.; Efros, A. A.; Holynski, A.; and Kanazawa, A. 2023. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 19740–19750.

Hertz, A.; Aberman, K.; and Cohen-Or, D. 2023. Delta denoising score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2328–2337.

Hertz, A.; Mokady, R.; Tenenbaum, J.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2022. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Hong, Y.; Zhang, K.; Gu, J.; Bi, S.; Zhou, Y.; Liu, D.; Liu, F.; Sunkavalli, K.; Bui, T.; and Tan, H. 2023. LRM: Large Reconstruction Model for Single Image to 3D. In *The Twelfth International Conference on Learning Representations*.

Jambon, C.; Kerbl, B.; Kopanas, G.; Diolatzis, S.; Leimkühler, T.; and Drettakis, G. 2023. Nerfshop: Interactive editing of neural radiance fields. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 6(1).

Kamata, H.; Sakuma, Y.; Hayakawa, A.; Ishii, M.; and Narihira, T. 2023. Instruct 3d-to-3d: Text instruction guided 3d-to-3d conversion. *arXiv preprint arXiv:2303.15780*.

Kania, K.; Yi, K. M.; Kowalski, M.; Trzciński, T.; and Tagliasacchi, A. 2022. Conerf: Controllable neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18623–18632.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4): 1–14.

Kobayashi, S.; Matsumoto, E.; and Sitzmann, V. 2022. Decomposing nerf for editing via feature field distillation. *Advances in Neural Information Processing Systems*, 35: 23311–23330.

Lazova, V.; Guzov, V.; Olszewski, K.; Tulyakov, S.; and Pons-Moll, G. 2023. Control-nerf: Editable feature volumes for scene rendering and manipulation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 4340–4350.

Li, R.; He, C.; Zhang, Y.; Li, S.; Chen, L.; and Zhang, L. 2023. Sim: Semantic-aware instance mask generation for box-supervised instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7193–7203.

Li, R.; Li, R.; Guo, S.; and Zhang, L. 2024a. Source Prompt Disentangled Inversion for Boosting Image Editability with Diffusion Models. *arXiv preprint arXiv:2403.11105*.

Li, Y.; Dou, Y.; Shi, Y.; Lei, Y.; Chen, X.; Zhang, Y.; Zhou, P.; and Ni, B. 2024b. Focaldreamer: Text-driven 3d editing via focal-fusion assembly. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 3279–3287.

Liu, H.; Xu, C.; Yang, Y.; Zeng, L.; and He, S. 2024a. Drag Your Noise: Interactive Point-based Editing via Diffusion Semantic Propagation. *arXiv preprint arXiv:2404.01050*.

Liu, S.; Zhang, X.; Zhang, Z.; Zhang, R.; Zhu, J.-Y.; and Russell, B. 2021. Editing conditional radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, 5773–5783.

Liu, X.; Xue, H.; Luo, K.; Tan, P.; and Yi, L. 2024b. GenN2N: Generative NeRF2NeRF Translation. *arXiv preprint arXiv:2404.02788*.

Meng, C.; He, Y.; Song, Y.; Song, J.; Wu, J.; Zhu, J.-Y.; and Ermon, S. 2021. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*.

Mikaeili, A.; Perel, O.; Safaee, M.; Cohen-Or, D.; and Mahdavi-Amiri, A. 2023. Sked: Sketch-guided text-based

3d editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 14607–14619.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106.

Mokady, R.; Hertz, A.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2023. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6038–6047.

Poole, B.; Jain, A.; Barron, J. T.; and Mildenhall, B. 2022. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*.

Qi, C.; Cun, X.; Zhang, Y.; Lei, C.; Wang, X.; Shan, Y.; and Chen, Q. 2023. Fatezero: Fusing attentions for zero-shot text-based video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15932–15942.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2): 3.

Rojas, S.; Philip, J.; Zhang, K.; Bi, S.; Luan, F.; Ghanem, B.; and Sunkavall, K. 2024. DATENeRF: Depth-Aware Text-based Editing of NeRFs. *arXiv preprint arXiv:2404.04526*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Ruiz, N.; Li, Y.; Jampani, V.; Pritch, Y.; Rubinstein, M.; and Aberman, K. 2023. 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*, 22500–22510.

Schönberger, J. L.; and Frahm, J.-M. 2016. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Schönberger, J. L.; Zheng, E.; Pollefeys, M.; and Frahm, J.-M. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In *European Conference on Computer Vision (ECCV)*.

Shi, Y.; Xue, C.; Pan, J.; Zhang, W.; Tan, V. Y.; and Bai, S. 2023. Dragdiffusion: Harnessing diffusion models for interactive point-based image editing. *arXiv preprint arXiv:2306.14435*.

Song, L.; Cao, L.; Gu, J.; Jiang, Y.; Yuan, J.; and Tang, H. 2023. Efficient-NeRF2NeRF: Streamlining Text-Driven 3D Editing with Multiview Correspondence-Enhanced Diffusion Models. *arXiv preprint arXiv:2312.08563*.

Tschernezki, V.; Laina, I.; Larlus, D.; and Vedaldi, A. 2022. Neural feature fusion fields: 3d distillation of self-supervised 2d image representations. In *2022 International Conference on 3D Vision (3DV)*, 443–453. IEEE.

Tumanyan, N.; Geyer, M.; Bagon, S.; and Dekel, T. 2023. 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*, 1921–1930.

Voynov, A.; Chu, Q.; Cohen-Or, D.; and Aberman, K. 2023. $P+$: Extended Textual Conditioning in Text-to-Image Generation. *arXiv preprint arXiv:2303.09522*.

Wang, C.; Chai, M.; He, M.; Chen, D.; and Liao, J. 2022. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3835–3844.

Wang, C.; Jiang, R.; Chai, M.; He, M.; Chen, D.; and Liao, J. 2023. Nerf-art: Text-driven neural radiance fields stylization. *IEEE Transactions on Visualization and Computer Graphics*.

Wang, Z.; Wang, Y.; Chen, Y.; Xiang, C.; Chen, S.; Yu, D.; Li, C.; Su, H.; and Zhu, J. 2024. Crm: Single image to 3d textured mesh with convolutional reconstruction model. *arXiv preprint arXiv:2403.05034*.

Wu, J. Z.; Ge, Y.; Wang, X.; Lei, S. W.; Gu, Y.; Shi, Y.; Hsu, W.; Shan, Y.; Qie, X.; and Shou, M. Z. 2023. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 7623–7633.

Yang, B.; Bao, C.; Zeng, J.; Bao, H.; Zhang, Y.; Cui, Z.; and Zhang, G. 2022. Neumesh: Learning disentangled neural mesh-based implicit field for geometry and texture editing. In *European Conference on Computer Vision*, 597–614. Springer.

Yang, B.; Gu, S.; Zhang, B.; Zhang, T.; Chen, X.; Sun, X.; Chen, D.; and Wen, F. 2023. Paint by example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18381–18391.

Yuan, Y.-J.; Sun, Y.-T.; Lai, Y.-K.; Ma, Y.; Jia, R.; and Gao, L. 2022. Nerf-editing: geometry editing of neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18353–18364.

Zhang, J.; Herrmann, C.; Hur, J.; Polania Cabrera, L.; Jampani, V.; Sun, D.; and Yang, M.-H. 2024. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. *Advances in Neural Information Processing Systems*, 36.

Zhang, Z.; Li, R.; and Zhang, L. 2024. FreCaS: Efficient Higher-Resolution Image Generation via Frequency-aware Cascaded Sampling. *arXiv preprint arXiv:2410.18410*.

Zhuang, J.; Wang, C.; Lin, L.; Liu, L.; and Li, G. 2023. Dreameditor: Text-driven 3d scene editing with neural fields. In *SIGGRAPH Asia 2023 Conference Papers*, 1–10.