EditSplat: Multi-View Fusion and Attention-Guided Optimization for View-Consistent 3D Scene Editing with 3D Gaussian Splatting

Dong In Lee[1] Hyeongcheol Park[1] Jiyoung Seo[1] Eunbyung Park[2] Hyunje Park[1]
Ha Dam Baek[1] Sangheon Shin[3] Sangmin Kim[3] Sangpil Kim[1]*

[1]Korea University [2]Yonsei University [3]Hanhwa Systems

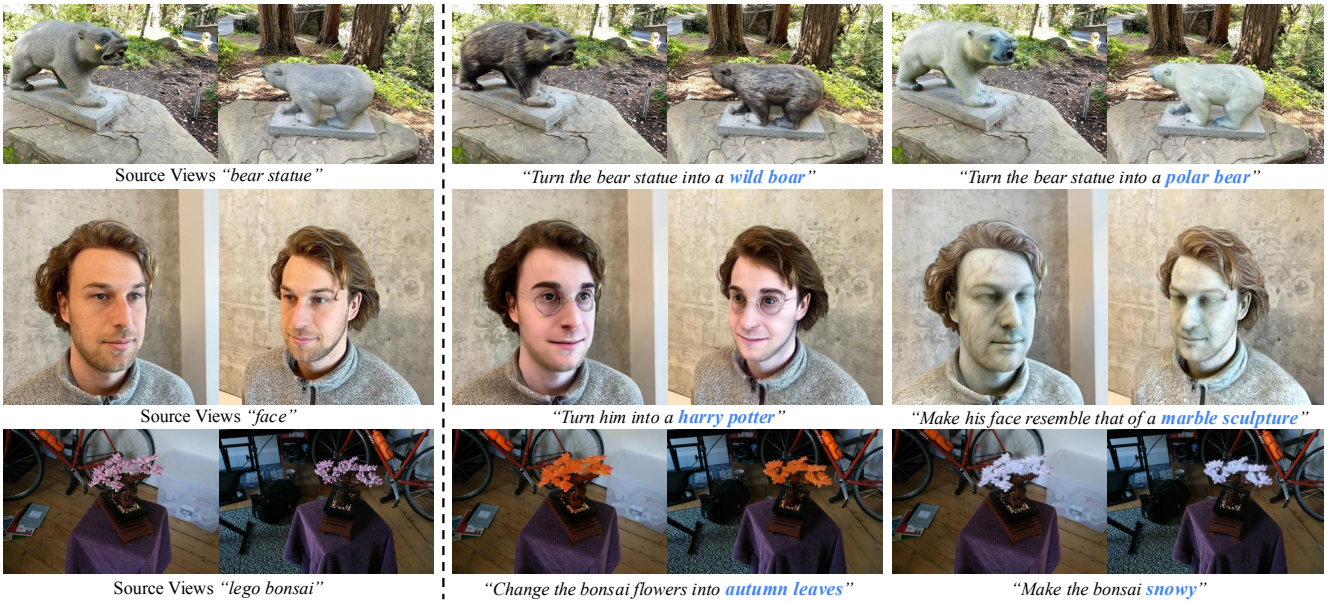

Figure 1. **Result of EditSplat**. EditSplat enables flexible and high-quality editing of pre-trained 3D Gaussian Splatting models guided solely by textual instructions. Through its design focused on multi-view consistency, efficient optimization, and precise semantic local editing, Our approach demonstrates robust performance, producing realistic and fine-grained 3D scene modifications.

Abstract

Recent advancements in 3D editing have highlighted the potential of text-driven methods in real-time, user-friendly AR/VR applications. However, current methods rely on 2D diffusion models without adequately considering multi-view information, resulting in multi-view inconsistency. While 3D Gaussian Splatting (3DGS) significantly improves rendering quality and speed, its 3D editing process encounters difficulties with inefficient optimization, as pre-trained Gaussians retain excessive source information, hindering optimization. To address these limitations, we propose ***EditSplat****, a novel text-driven 3D scene editing framework that integrates Multi-view Fusion Guidance (MFG) and Attention-Guided Trimming (AGT). Our MFG ensures multi-view consistency by incorporating essential multi-view information into the diffusion process, leveraging classifier-free guidance from the text-to-image diffusion model and the geometric structure inherent to 3DGS. Additionally, our AGT utilizes the explicit representation of 3DGS to selectively prune and optimize 3D Gaussians, enhancing optimization efficiency and enabling precise, semantically rich local editing. Through extensive qualitative and quantitative evaluations, EditSplat achieves state-of-the-art performance, establishing a new benchmark for text-driven 3D scene editing. Project website:* https://kuai-lab.github.io/editsplat2024/

1. Introduction

Text-driven 3D scene editing, which enables manipulation of 3D scenes using only text instructions, is gaining research momentum. The goal is to edit 3D representations accurately and efficiently through text prompts, facilitating real-time, user-friendly 3D editing for film and game development, digital content creation, or AR/VR applications.

InstructNeRF2NeRF (IN2N) [12] introduced the pipeline for text-driven 3D editing by leveraging 2D diffusion model [39], such as InstructPix2Pix [4]. Subsequent

*Corresponding author.

methods [9, 33, 47, 48], inspired by IN2N, iteratively edit rendered images while updating the underlying 3D representation. However, these approaches independently edit single views with 2D diffusion models, neglecting multi-view consistency, as illustrated in Fig. 2-(a). More recent methods [7, 8, 11] incorporate multi-view information, but they lack a comprehensive strategy for consistent view alignment. For example, [8, 11] use only a limited set of key views, requiring additional diffusion passes or extra-parameterized layers to integrate multi-view information. Similarly, [7] trains a diffusion model to achieve multi-view consistency, incurring high costs. Consequently, existing methods still exhibit multi-view inconsistency, resulting in noisy gradients that hinder optimization and produce suboptimal outputs, such as minimal edits or blur outputs.

3D Gaussian Splatting (3DGS) [21] has emerged as a foundational model in 3D representation, surpassing Neural Radiance Field (NeRF) [32] in rendering quality and speed. Unlike NeRF's implicit representations, 3DGS employs explicit anisotropic ellipsoids, enabling faster training and high-quality reconstruction. However, editing a pre-trained 3DGS model introduces inefficiency, as pre-trained Gaussians tend to retain source visual and geometric details excessively, impeding efficient convergence as illustrated in Fig. 2-(b). This underscores the need to manage redundant Gaussians for more effective optimization.

To tackle the limitations in both the multi-view inconsistency and the optimization inefficiency of pre-trained Gaussians in editing, we propose a novel 3D scene editing framework, **EditSplat**. To ensure multi-view consistency between edited images, we propose a *Multi-view Fusion Guidance (MFG)* method that ensures multi-view consistent editing utilizing a 2D diffusion model and the geometric structure inherent to 3DGS. Inspired by [7, 11], our method projects initially edited multi-view images onto a target view using 3DGS depth maps, enabling smooth blending based on depth values across views to integrate multi-view information. Leveraging classifier-free guidance in the text-image diffusion process [4, 16], we carefully edit source images based on text prompt by incorporating multi-view fused images into the diffusion process, ensuring robust alignment of edited images with multi-view details. To further preserve source fidelity, we utilize the source images as auxiliary guidance. By balancing guidance scores between multi-view fused images, source images, and text prompts, our method enables the diffusion model to edit source images to the multi-view-consistent outputs, aligning with essential multi-view information. Unlike iterative editing approaches that require expensive optimization processes during rendered image editing, our proposed EditSplat edits source images directly, allowing for immediate 3DGS model optimization.

Additionally, we propose an *Attention-Guided Trimming (AGT)* method to improve the optimization efficiency of pre-trained Gaussians and enable semantically rich local editing. Utilizing the explicit representation of 3DGS, we assign attention weight to each Gaussian based on the attention map from the diffusion model. High attention weights highlight semantically meaningful regions, indicating substantial visual and geometric changes required for effective edits. Thus, pre-trained Gaussians with high attention weight generally need significant changes with the optimization process and become redundant during the densification process as the number of Gaussians retaining excessive source information increases. To address this problem, we prune a suitable proportion of pre-trained Gaussians with high attention weight before editing, allowing the remaining Gaussians to achieve efficient densification and optimization. In addition, we selectively optimize Gaussians with high attention weight, enabling semantic local editing. Overall, the proposed AGT simplifies optimization and improves semantic precision, enabling faster and more efficient 3DGS editing.

Figure 2. Challenging cases. (a) Conventional 2D editing without Multi-view Fusion Guidance (MFG) results in inconsistent textures across different views (e.g., bear fur), whereas applying MFG achieves consistent multi-view edits. (b) Editing without Attention-Guided Trimming (AGT) leads to inefficient optimization, resulting in less edited regions (e.g., clown's nose). AGT effectively enhances optimization quality, producing richer colors.

We conduct extensive qualitative and quantitative experiments to validate the high quality and effectiveness of EditSplat. Our key contributions are as follows:

- We propose a Multi-view Fusion Guidance (MFG) method that integrates multi-view information into the diffusion process to achieve robust alignment with multi-view details, ensuring consistent multi-view editing.
- We introduce an Attention-Guided Trimming (AGT) technique that prunes 3D Gaussians with high attention weight and selectively optimizes them, enhancing optimization efficiency and enabling semantic local editing.
- Through extensive qualitative and quantitative evaluations, we demonstrate that EditSplat outperforms existing methods in both performance and effectiveness.

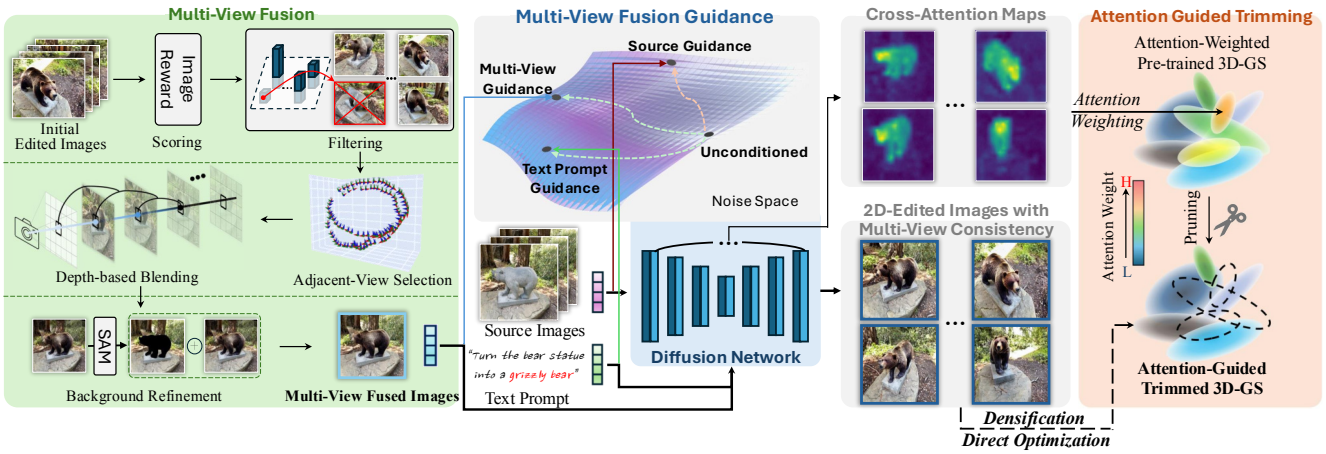

Figure 3. **EditSplat Overview.** EditSplat consists of two main methods: (1) Multi-view Fusion Guidance (MFG, Sec. 3.2), which aligns multi-view information with text prompts and source images to ensure multi-view consistency; (2) Attention-Guided Trimming (AGT, Sec. 3.3), which prunes pre-trained Gaussians for optimization efficiency and selectively optimizes Gaussians for semantic local editing.

2. Related Work

2D Image Editing. Despite advancements in 3D generative models [2, 6, 10, 20, 27, 30, 44, 51], their application in 3D editing is constrained by limited 3D data availability. To overcome this limitation, recent studies have shifted towards leveraging large-scale 2D generative models [17, 18, 35, 38, 42, 46], such as Text-to-Image (T2I) diffusion models [39, 45, 50, 53], to enable more flexible 3D editing. DreamBooth [41] proposes subject-specific editing by fine-tuning the T2I diffusion model Imagen [43]. Delta Denoising Score (DDS) [15] utilizes Score Distillation Sampling (SDS) [36] for precise image modifications. Prompt-to-Prompt (P2P) [14] harnesses cross-attention maps in the diffusion process for detailed edits. InstructPix2Pix (IP2P) [4] enables instruction-driven editing by fine-tuning Stable Diffusion [39] on a synthetic dataset of image-text pairs generated using GPT-3 [5] and P2P [14]. Our EditSplat utilizes IP2P as a 2D image editor. Utilizing its image guidance capability through classifier-free guidance, EditSplat effectively integrates multi-view information into the diffusion process, facilitating multi-view consistent editing for enhancing 3D editing.

Text-driven 3D Scene Editing. Text-driven 3D scene editing aims to modify 3D representations precisely and efficiently using textual instructions. Despite recent progress, key challenges such as multi-view consistency, semantic localization, and optimization efficiency remain.

Recent advancements [25, 26, 34, 55] employ Score Distillation Sampling (SDS) [36], while iterative methods such as InstructNerf2Nerf [12] and its extensions [9, 31, 33, 47, 48] repeatedly edit rendered images while optimizing the underlying 3D representation. However, those works often overlook multi-view information, leading to inconsistency.

Recent methods explicitly addressing multi-view consistency, VICA-NeRF [11] and ConsistDreamer [7], employ view projection strategies but suffer from noisy projection outputs and computational overhead due to additional diffusion steps [11] or extensive U-Net [40] training [7]. Other approaches [8, 19] utilize the epipolar lines, though they still struggle to preserve fine-grained details. Moreover, 3DEgo [23] employs classifier-free guidance with IP2P [4], but its requirement for guidance from all frames in single-view edits inevitably results in inconsistencies due to misalignment viewpoint.

Furthermore, most existing methods [8, 9, 11, 23, 48] rely on binary masks from the source image to localize the editing area, which constrains edits and overlooks semantic information when the target object is not clearly defined.

For multi-view consistency, we introduce a Multi-view Fusion Guidance (MFG) involving sequential projection and blending, avoiding noise issues and the additional training [7] or diffusion refinement [11]. Regarding semantic localization, inspired by DreamEditor [55], EditSplat explicitly assigns attention maps to 3D Gaussians, achieving higher semantic precision without binary mask constraints. Additionally, as far as we know, we are the first to address the optimization inefficiency inherent in pre-trained 3DGS models by strategically pruning 3D Gaussians, enhancing both densification and optimization efficiency.

3. Method

We propose a novel 3D editing framework, EditSplat, carefully designed to achieve three key objectives: (i) multi-view consistent editing, (ii) efficient optimization, and (iii) semantically rich local editing. To meet these goals, we introduce Multi-view Fusion Guidance (MFG) and Attention-Guided Trimming (AGT). MFG incorporates essential multi-view information into the diffusion model to ensure multi-view consistency. In AGT, attention weights from the diffusion model are explicitly assigned to each Gaussian in 3D Gaussian Splatting (3DGS). This enables efficient convergence by pruning Gaussians with high attention weight and selectively optimizing them for semantic local editing. The overall pipeline is shown in Fig. 3.

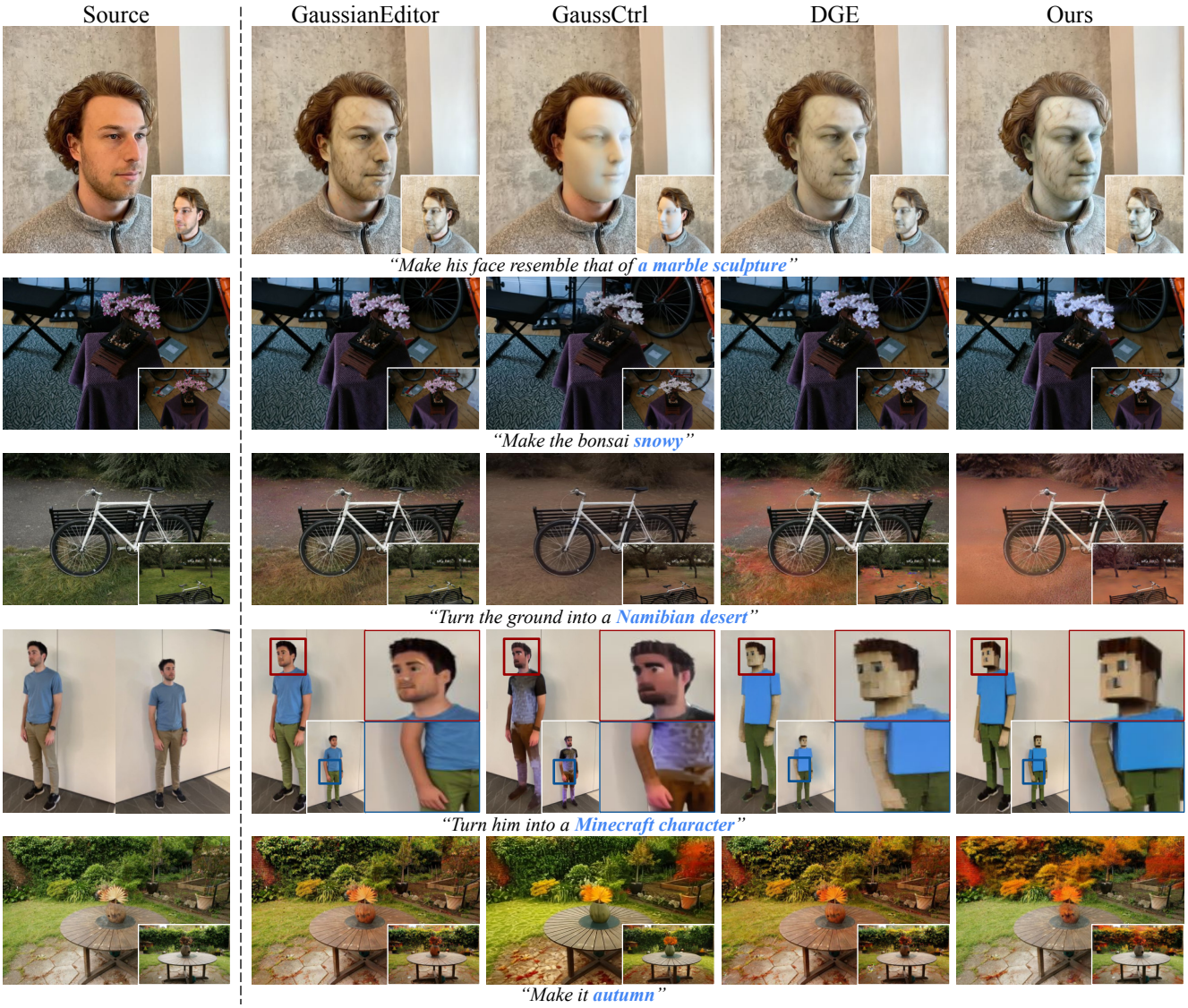

Figure 4. **Qualitative Comparison**. EditSplat provides more intense and precise editing compared to other baselines. The leftmost column shows source images, while the right columns show rendering images from edited 3DGS. In each corner of the images, we include different views of the corresponding image to compare multi-view consistency. Note that our EditSplat outperforms both local and global editing.

In the following sections, we provide a review of 3DGS and classifier-free guidance in Sec. 3.1, followed by our primary methods: MFG in Sec. 3.2 and AGT in Sec. 3.3.

3.1. Preliminaries

3D Gaussian Splatting. 3D Gaussian Splatting [22] is an explicit 3D scene representation using anisotropic Gaussians to model complex structures. A set of 3D Gaussians G represents the scene, where each 3D Gaussian $g \in G$ is defined by a center $\mu \in \mathbb{R}^3$, a covariance matrix $\Sigma \in \mathbb{R}^{3\times3}$, spherical harmonic coefficients $f \in \mathbb{R}^k$ (with k indicating the degrees of freedom), and opacity $\sigma \in \mathbb{R}$. The covariance matrix Σ is composed of a rotation matrix R and a scaling matrix S as $\Sigma = RSS^TR^T$. For rendering, a 3D Gaussian function is defined as

$$g(x;\mu,\Sigma) = e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)}. \quad (1)$$

Each 3D Gaussian g is projected into a 2D image space using a world-to-image projective transform, with its Jacobian evaluated at μ. The final color of each pixel is calculated by blending contributions from Gaussians along a ray:

$$C = \sum_{i\in N} c_i\alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \quad (2)$$

where blending weight $\alpha_i \in \mathbb{R}$ is defined as the density value of g multiplied by σ, and $c \in \mathbb{R}^3$ is color of the Gaussian. This tile-based approach supports real-time rendering, with Gaussian parameters μ, Σ, f, and σ optimized to represent scenes accurately through photometric loss.

Classifier-free Guidance. Classifier-free diffusion guidance [16] improves the alignment between generated outputs and conditioning prompts in tasks, such as image- or

text-conditional image generation by guiding sampling toward regions where the implicit classifier $p_\theta(h|z_t)$ assigns high probability to the target condition h. In models with multiple guidance inputs, such as InstructPix2Pix [4] (IP2P) used in our approach, the score network $\epsilon_\theta(z_t, h_I, h_T)$ incorporates both an input image h_I and a text instruction h_T for guide generation. Here, z_t denotes the latent representation of the input image at timestep t with added noise, and ϵ_θ represents the noise prediction network (e.g., U-net [40]).

To control the influence of each conditioning, guidance scales s_I and s_T are introduced, adjusting the extent to which the generated outputs align with the input image and text instruction. The guided prediction $\tilde{\epsilon}_\theta$ is defined as:

$$\begin{aligned}\tilde{\epsilon}_\theta(z_t, h_I, h_T) = \epsilon_\theta(z_t, \varnothing, \varnothing) \\ + s_I(\epsilon_\theta(z_t, h_I, \varnothing) - \epsilon_\theta(z_t, \varnothing, \varnothing)) \\ + s_T(\epsilon_\theta(z_t, h_I, h_T) - \epsilon_\theta(z_t, h_I, \varnothing)),\end{aligned} \quad (3)$$

where s_I and s_T modulate the influence of the image condition h_I and text instruction h_T. This balances conditioned and unconditioned score estimates to guide generation, prioritizing one or both types of conditioning.

3.2. Multi-View Fusion Guidance (MFG)

Recent studies [8, 9, 11, 12, 48] struggle with consistent editing across views due to limited consideration of multi-view information, resulting in multi-view inconsistency that causes minimal edits or blurriness. To overcome this challenge, we focus on integrating essential multi-view details into the 2D diffusion model, enabling consistent editing.

Multi-View Fusion. We first edit the whole source images using the diffusion model. Inspired by [7, 11], we incorporate multi-view information by projecting and blending the initially edited multi-view images onto each target view using depth maps from 3DGS. To enhance output quality, we rank these images using ImageReward [49], which scores fidelity and text alignment based on human feedback. Since some initial edits are generally misaligned, we exclude the bottom 15% of low-scoring images from projection.

For the multi-view projection process, we select the top 5 adjacent views, prioritizing them based on their proximity to the target view, determined by camera position and orientation in world coordinates. To manage overlapping pixels across multiple views, we implement an iterative alpha blending strategy guided by depth values. Starting from the farthest pixel (i.e., highest depth value), pixel pairs are blended iteratively in decreasing depth order. The final pixel color is determined by aligning and unprojecting the multi-view images into 3D space using depth maps and camera parameters, followed by reprojecting onto the target view to resolve overlapping pixel contributions. This approach ensures that each target view captures comprehensive details shared across multiple views. To handle sparse background regions and achieve seamless integration, we refine these areas using SAM [24] to fill them with source content. This sequential multi-view fusion strategy enriches each target view with consistent and detailed information, reinforcing alignment and coherence across all views. The resulting multi-view fusion image h_M for each target view serves as an essential input for subsequent editing processes, integrating comprehensive visual and geometric details shared across the views. Note that further details are provided in the supplementary material.

Alignment with Multi-View Information. To effectively integrate multi-view fusion information into the diffusion process and edit source images without additional training [7], extra-parameterized layers [8], or repeated diffusion passes [11], we leverage classifier-free guidance. This allows the direct incorporation of multi-view fusion details into the diffusion process. According to Liu et al. [29], a conditional diffusion model can combine score estimates from multiple conditioning values. Building on this, we adapt this approach, facilitating the implicit classifier p_θ to assign high probabilities to multi-view fused visual features, color consistency, and structural properties. This ensures the edited images maintain a strong alignment with multi-view fused images throughout the editing process.

Specifically, we provide the multi-view fusion image h_M as a guidance conditioning during source image editing. This enhances alignment with multi-view details, ensuring multi-view consistent editing. To maintain source fidelity, we include the original source image h_S as auxiliary guidance, preserving its content. Our guided score prediction $\tilde{\epsilon_\theta}$ is defined as follows:

$$\begin{aligned}\tilde{\epsilon_\theta}\left(z_t, h_S, h_T, h_M\right) = \ & \epsilon_\theta\left(z_t, \varnothing, \varnothing\right) \\ & + s_T\left(\epsilon_\theta\left(z_t, h_M, h_T\right) - \epsilon_\theta\left(z_t, h_M, \varnothing\right)\right) \\ & + s_M\left(\epsilon_\theta\left(z_t, h_M, \varnothing\right) - \epsilon_\theta\left(z_t, \varnothing, \varnothing\right)\right) \\ & + s_S\left(\epsilon_\theta\left(z_t, h_S, \varnothing\right) - \epsilon_\theta\left(z_t, \varnothing, \varnothing\right)\right),\end{aligned} \quad (4)$$

where h_M, h_S, and h_T correspond to the multi-view fusion image, source image, and text prompt, respectively. Each guidance input is modulated by its respective scale factor: s_M for Multi-view Fusion Guidance, s_S for the source image, and s_T for the text prompt.

Our method efficiently directs the diffusion process toward generating outputs that achieve seamless multi-view consistency, including multi-view fusion images' texture, color, geometric, and source images. By balancing guidance scores between s_M, source images s_S, and text prompts s_T, our approach enables the diffusion model to align edits with essential multi-view information and generate consistent outputs. Unlike iterative editing approaches [9, 12, 47, 48] that iteratively edit rendered images while updating the underlying 3D representation, our approach directly edits the source image, ensuring precise alignment with multi-view information and the text prompt.

3.3. Attention-Guided Trimming (AGT)

When editing a 3DGS, optimization inefficiency occurs as pre-trained Gaussians tend to excessively retain source attributes, obstructing effective optimization. Additionally, previous methods [8, 9, 11, 48] using 2D binary masks from SAM [24] for local editing often overlook semantic regions, as these masks are extracted solely from the source image rather than the edited output. This local editing issue can lead to suboptimal results when the target object is not clearly defined. These underlines the need to manage Gaussians in editing. To improve efficiency and semantic local editing, we trim Gaussians by pruning redundant Gaussians and selectively optimizing those relevant, guided by attention maps from the diffusion model.

Attention Map. Attention maps from the text-to-image diffusion U-Net's cross-attention layers highlight regions that require intensified focus for generation [14, 28], revealing the correlation between individual words and generated image regions, computed as Softmax $\left(\frac{QK^T}{\sqrt{q}}\right)$, where Q is the query matrix projected from the spatial features of the noisy image with dimension q, and K is the key matrix projected from the textual embedding. In image editing diffusion (IP2P), high attention weights similarly indicate areas where edits should be concentrated, typically aligning with regions of significant visual change.

Attention Weighting 3D Gaussians. To assign attention weights to each pre-trained Gaussian, we first unproject multi-view-consistent cross-attention maps onto the 3D Gaussians through inverse rendering [9]. These maps, resized to rendering size using bilinear interpolation, correspond to specific word tokens for the editing target in the MFG editing process. Given the j-th Gaussian in the 3DGS, we compute its cumulative attention weight w_j by aggregating the attention map from each view v of all views V as:

$$w_j = \frac{1}{\sum_{v \in V} |S_{v,j}|} \sum_{v \in V} \sum_{s \in S_{v,j}} \text{Softmax}\left(\frac{Q_v K^T}{\sqrt{q}}\right)_s, \quad (5)$$

where $S_{v,j}$ denotes a set of indices of all attention weights affected by the j-th Gaussian from the attention map for the v-th view. Softmax $\left(\frac{Q_v K^T}{\sqrt{q}}\right)_s \in [0, 1]$ denotes an individual attention weight at s from an attention map for the v-th view. By aggregating attention weight across V, this weighting mechanism ensures that each Gaussian reflects the overall importance of multi-view attention maps, emphasizing regions that require pronounced geometrical or visual changes during editing. Details on preparing the attention map are in the supplementary material.

Trimming 3D Gaussians. After weighting attention map to each Gaussian, we apply thresholding to obtain $w'_j = w_j \mathbb{1}\{w_j \geq w_{\text{thres}}\}$ to exclude Gaussians with low attention from the optimization by setting those gradients to zero, directing updates solely toward Gaussians with high attention. This selective optimization enhances precision by concentrating on semantically rich regions and aligning edits with meaningful word-image correlations.

Pre-trained Gaussians with high attention weights hinder optimization for editing, as these regions indicate areas requiring substantial visual and geometric modifications. This inefficiency is intensified as the number of Gaussians retaining excessive source information increases during the densification process.

To improve densification and optimization efficiency, we prune an optimal proportion of Gaussians with high attention before training, focusing on essential areas for more effective editing.

$$G_{\text{pruning}} = \{g_j \mid g_j \in G, w'_j \geq \text{Top-}k\%(G)\}, \quad (6)$$

where Top-$k\%(G)$ returns the threshold weight for top $k\%$ Gaussian and then G_{pruning} consists of Gaussians selected for pruning, the top $k\%$ of Gaussians in G based on attention weight w'_j. By removing G_{pruning}, this pruning step eliminates redundant pre-trained Gaussians with high attention weight, allowing efficient densification and optimization to concentrate on the remaining necessary Gaussians and thereby enhancing overall efficiency.

Optimization for 3D Editing. Given the source Gaussians trimmed by AGT, $G_{\text{agt}} = G \setminus G_{\text{pruning}}$, and its rendering function $\hat{I}_v$, as well as the multi-view consistent edited image I_v^{mfg} by MFG editing process, the final optimized Gaussians G_{edit} is obtained by minimizing the editing loss across all views V:

$$G_{\text{edit}} = \arg\min_{G_{\text{agt}}} \sum_{v \in V} \mathcal{L}_{\text{edit}}(\hat{I}_v(G_{\text{agt}}), I_v^{\text{mfg}}), \quad (7)$$

where $\mathcal{L}_{\text{edit}}$ measures the discrepancy between rendered image $\hat{I}_v(G_{\text{agt}})$ and $\hat{I}_v^{mfg}$ for each view v. We use combination of LPIPS [54] and L_1 loss, weighted at a 1:1 ratio, as commonly empolyed in 3D scene editing task [12].

4. Experiments

4.1. Implementation Details

Our EditSplat is built upon vanilla 3DGS [21] using PyTorch. We use InstructPix2Pix [4] (IP2P) as a 2D editor with our proposed MFG, as described in 3.2. To validate the 3D scene editing performance of EditSplat, we collect 4 scenes from IN2N [12], one scene from BlendedMVS [52], and 3 scenes from Mip-NeRF360 [3], covering datasets utilized in previous studies, including complex real-world large 360° scenes. All experiments are conducted on a single RTX A6000 GPU. The complete editing process takes approximately 6 minutes for the "Face" scene in IN2N.

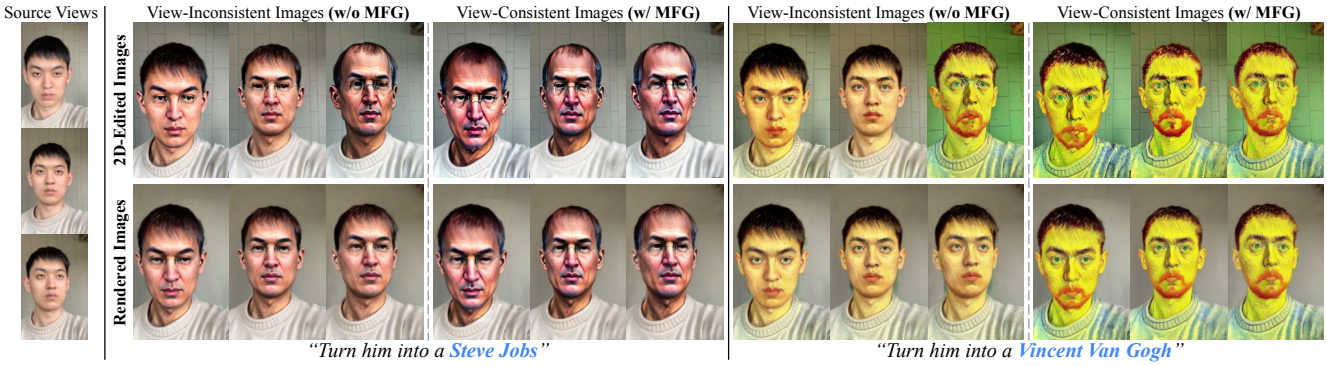

Figure 5. **MFG ablation**. The top row shows the results of 2D editing with and without MFG. With MFG, the 2D diffusion model produces multi-view-consistent results. In the bottom row, we have images rendered from edited 3DGS based on the above images. Editing 3DGS with view-consistent images results in clear outputs with high fidelity, while other cases produce inconsistent results with low fidelity.

Hyperparameters All experiments are conducted with fixed hyperparameters across all scenes and prompts, except for the number of epochs (ranging from 5 to 10). For a fair comparison with baseline methods such as GaussianEditor [9] and DGE [8], which adopt the IP2P [4] and set to $s_T = 7.5$, $s_I = 1.5$, we set the guidance scales of MFG to $s_T = 7.5$, $s_M = 1.0$, and $s_S = 0.5$. Specifically, we matched the text guidance scale s_T with the baselines and adjusted the image guidance scales such that the sum of s_M and s_S equals s_I, ensuring comparable experimental conditions. In AGT, the threshold value for localization w_{thres} is set to 0.1, and the pruning proportion k is 0.15.

4.2. Baselines

We mainly compare EditSplat with the current state-of-the-art methods that are based on 3DGS like ours: GaussianEditor [9], GaussCtrl [48], and DGE [8]. Although both GaussianEditor and DGE employ IP2P as we do, with instruction-based prompts, GaussCtrl utilizes ControlNet [53], which uses description-like prompts. To ensure a fair comparison, we generate the source scene descriptions to reproduce GaussCtrl's input conditions. Since these baselines represent the current leading approaches and provide a comprehensive benchmark in 3DGS-based text-driven 3D scene editing, we evaluate the performance of EditSplat in comparison with them.

4.3. Qualitative Comparisons

Fig. 1 illustrates EditSplat's ability to achieve multi-view consistent editing and elaborate optimization for 3D scene editing. In the first row, EditSplat modifies objects in a 360° scene according to the text prompt, maintaining consistent textures for both "wild boar" and "polar bear" across different views. The second row shows EditSplat transforming a man's face into "Harry Potter" and a "marble sculpture" with consistent results, despite the geometric complexity of facial features. In the third row, EditSplat demonstrates precise editing of intricate objects, accurately altering leaves by effectively optimizing Gaussians in targeted regions.

In Fig. 4, we compare EditSplat with recent editing baselines. Most baselines suffer from suboptimal visual changes or blurring artifacts in rendered images due to multi-view inconsistency of 2D-edited images. In comparison, it is clear that our method gives more pronounced edited results, closely aligning with the given text prompt.

Notably, in the fourth row, prompted by "Turn him into a Minecraft character," GaussianEditor yields minimal edits with artifacts in the facial region, while DGE and GaussCtrl produce blurry and suboptimal results, respectively. In contrast, our method produces clear outputs, achieving a more authentic Minecraft-like appearance. This highlights that our MFG is more effective guidance for diffusion than depth-based conditioning in GaussCtrl, epipolar constraints in DGE, or iterative dataset updating in GaussianEditor.

Additionally, EditSplat demonstrates superior performance in global transformations and localized editing for large real-world scenes. Specifically, in the last row with the prompt "Make it autumn," other methods fail to preserve the original table color, whereas our method maintains the table's color while effectively modifying the surrounding foliage. This effectiveness arises from AGT's selective optimization of Gaussians based on the semantics of the attention map corresponding to the prompt token "autumn," enabling precise and context-aware semantic localization.

4.4. Quantitative Comparisons

Despite the subjective nature of 3D scene generation and editing, we adhere to established practices [8, 9, 13, 25, 34, 47, 48] by employing CLIP [37] for quantitative evaluation. Our comparison metrics include two types of CLIP scores. CLIP text-image directional similarity captures the alignment between changes in text captions and the corresponding changes between rendered and source images in CLIP space. Meanwhile, CLIP text-image similarity measures the alignment between the CLIP embeddings of the target text and the rendered images. In addition, we perform user study based on human preference. The user study was conducted with 100 participants via Amazon Mechanical Turk [1]. All of our metrics are evaluated under 8 scenes with 2 prompts each to ensure fair comparison and diverse

Method	$CLIP_{dir}$	$CLIP_{sim}$	User Study
GaussianEditor [9]	0.0923	0.2243	0.1643
GaussCtrl [48]	0.0979	0.2336	0.1987
DGE [8]	0.1222	0.2394	0.2143
EditSplat (Ours)	**0.1431**	**0.2531**	**0.4227**

Table 1. **Quantitative Comparison**. $CLIP_{dir}$: CLIP text-image direction similarity; $CLIP_{sim}$: CLIP text-image similarity. User study conducted to evaluate human preference.

feedback. As shown in Table 1, EditSplat achieves the highest scores across all categories, demonstrating superior performance. This demonstrates that EditSplat yields results that most closely align with the text prompt while achieving the highest human perceptual quality. Details are provided in the supplementary materials.

4.5. Ablation Study

To demonstrate the effectiveness of our proposed components: Multi-view Fusion Guidance (MFG) and Attention-Guided Trimming (AGT), we conducted an ablation study.

Multi-View Fusion Guidance (MFG). Fig. 2-(a) illustrates the impact of MFG on rendered results from edited 3DGS. MFG integrates multi-view information from edited outputs and provides this guidance to the diffusion process, facilitating consistent 2D edits that contribute to reliable 3D editing. Without MFG, inconsistent 2D edits lead to suboptimal outcomes, such as blurring, artifacts, and minimal visual changes in the rendered 3DGS images.

For example, as shown in Fig. 5, when prompted with "Turn him into a Steve Jobs" and "Turn him into a Vincent Van Gogh," editing without MFG results in minimal visual changes or blurriness and artifacts. Conversely, using MFG produces consistent final renderings that accurately reflect the intended 2D edits, underscoring the importance of MFG in achieving high-quality 3D scene editing.

Attention-Guided Trimming (AGT). In Fig. 2-(b), we demonstrate AGT's effectiveness in handling significant geometric transformations, such as reshaping the man's nose into a circular red clown nose. Applying AGT results in a deeper red nose, showcasing its capability for substantial visual and geometric modifications.

At the top-right of 2D edited images in Fig. 6, the cross-attention map highlights the head region (in yellow) with high attention weights, indicating areas that require considerable visual and geometric adjustments. Without AGT, pre-trained Gaussians retaining excessive source information persist in these high-attention regions, hindering convergence and leading to suboptimal 3D edits with remnants of the original face shape and artifacts, particularly around the nose. In contrast, AGT prunes these redundant Gaussians, aligning the 3D scene more closely with the editing guidance. This results in rendered images where the face undergoes precise geometric changes that align with the 2D edits, such as transforming into a "mannequin."

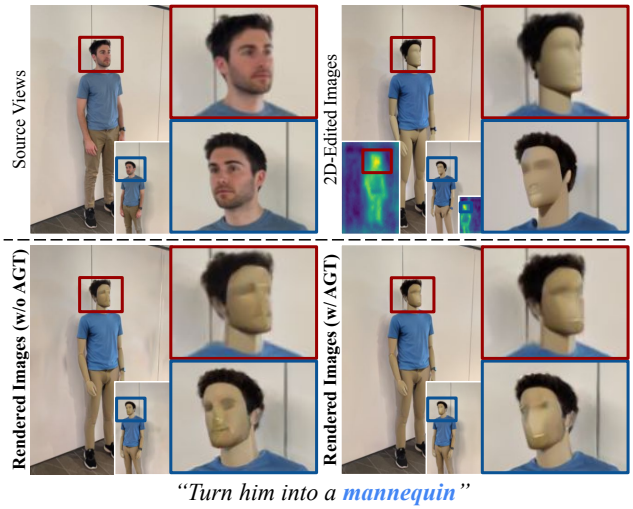

Figure 6. **AGT ablation**. With our proposed AGT, rendered images from edited 3DGS have more significant geometric changes, while others fail. As a result, the outcome aligns more closely with the desired 2D-edited images.

Moreover, without AGT, local editing lacks control, leading to unintended editing in non-target areas (e.g., the clown's jacket in Fig. 2-(b) and the background in Fig. 6). With AGT, Gaussians with high attention weights are selectively optimized, while those with low attention are excluded, ensuring precise local edits.

5. Conclusion

We introduce EditSplat, a robust 3D scene editing framework comprising Multi-view Fusion Guidance (MFG) and Attention-Guided Trimming (AGT). MFG addresses the critical challenge of achieving multi-view consistency, which often results in suboptimal outputs such as minimal edits or blurred results. By integrating critical multi-view information into the diffusion process, MFG ensures consistent and precise edits across different viewpoints, facilitating editing of source images and direct optimization of the 3D Gaussian Splatting model, unlike iterative editing approaches. AGT enhances optimization efficiency and enables precise and semantically rich local editing by pruning pre-trained Gaussians and selectively optimizing those relevant. Extensive evaluations of various complex real-world scenes demonstrate that EditSplat achieves state-of-the-art performance existing methods, setting a new benchmark for text-driven 3D scene editing.

Limitations. EditSplat relies on the quality of 2D diffusion models [4] and the depth maps rendered by 3DGS [21]. Current 2D diffusion models may struggle with complex or nuanced prompts, potentially limiting the effectiveness of 3D editing guidance. Additionally, the rendering and depth map quality of 3DGS are still developing, which can affect the projection process in MFG. We expect that advancements in both 2D diffusion models and 3DGS will enhance the robustness and precision of our framework.

Acknowledgements

This research was supported by Korea Research Institute for defense Technology planning and advancement - Grant funded by Defense Acquisition Program Administration(DAPA)((KRIT-CT-23-021), 90%), the Ministry of Science and ICT (MSIT) of Korea, under the National Research Foundation (NRF) grant (RS-2024-00337548, 5%; RS-2025-00521602, 1%), the Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024 (International Collaborative Research and Global Talent Development for the Development of Copyright Management and Protection Technologies for Generative AI, RS-2024-00345025, 3%), Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. RS-2019-II190079, Artificial Intelligence Graduate School Program(Korea University), 1%), and Artificial intelligence industrial convergence cluster development project funded by the Ministry of Science and ICT(MSIT, Korea)&Gwangju Metropolitan City.

References

[1] Amazon mechanical turk. https://www.mturk.com/, 2005. 7

[2] Titas Anciukevičius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J Mitra, and Paul Guerrero. Renderdiffusion: Image diffusion for 3d reconstruction, inpainting and generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12608–12618, 2023. 3

[3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022. 6

[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 1, 2, 3, 5, 6, 7, 8

[5] Tom B Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 3

[6] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3

[7] Jun-Kun Chen, Samuel Rota Bulò, Norman Müller, Lorenzo Porzi, Peter Kontschieder, and Yu-Xiong Wang. Consistdreamer: 3d-consistent 2d diffusion for high-fidelity scene editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21071–21080, 2024. 2, 3, 5

[8] Minghao Chen, Iro Laina, and Andrea Vedaldi. Dge: Direct gaussian 3d editing by consistent multi-view editing. *arXiv preprint arXiv:2404.1892*, 2024. 2, 3, 5, 6, 7, 8

[9] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21476–21485, 2024. 2, 3, 5, 6, 7, 8

[10] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 3

[11] Jiahua Dong and Yu-Xiong Wang. Vica-nerf: View-consistency-aware 3d editing of neural radiance fields. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 5, 6

[12] Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19740–19750, 2023. 1, 3, 5, 6

[13] Runze He, Shaofei Huang, Xuecheng Nie, Tianrui Hui, Luoqi Liu, Jiao Dai, Jizhong Han, Guanbin Li, and Si Liu. Customize your nerf: Adaptive source driven 3d scene editing via local-global iterative training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6966–6975, 2024. 7

[14] 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. 3, 6

[15] Amir Hertz, Kfir Aberman, and Daniel Cohen-Or. Delta denoising score. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2328–2337, 2023. 3

[16] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2, 4

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3

[18] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *Journal of Machine Learning Research*, 23(47):1–33, 2022. 3

[19] Nazmul Karim, Hasan Iqbal, Umar Khalid, Chen Chen, and Jing Hua. Free-editor: zero-shot text-driven 3d scene editing. In *European Conference on Computer Vision*, pages 436–453. Springer, 2025. 3

[20] Animesh Karnewar, Andrea Vedaldi, David Novotny, and Niloy J Mitra. Holodiffusion: Training a 3d diffusion model using 2d images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18423–18433, 2023. 3

[21] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time

radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2, 6, 8

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

[23] Umar Khalid, Hasan Iqbal, Azib Farooq, Jing Hua, and Chen Chen. 3dego: 3d editing on the go! In *European Conference on Computer Vision*, pages 73–89. Springer, 2025. 3

[24] 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 5, 6

[25] Juil Koo, Chanho Park, and Minhyuk Sung. Posterior distillation sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13352–13361, 2024. 3, 7

[26] Yuhan Li, Yishun Dou, Yue Shi, Yu Lei, Xuanhong Chen, Yi Zhang, Peng Zhou, and Bingbing Ni. Focaldreamer: Text-driven 3d editing via focal-fusion assembly. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3279–3287, 2024. 3

[27] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 3

[28] Bingyan Liu, Chengyu Wang, Tingfeng Cao, Kui Jia, and Jun Huang. Towards understanding cross and self-attention in stable diffusion for text-guided image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7817–7826, 2024. 6

[29] Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B Tenenbaum. Compositional visual generation with composable diffusion models. In *European Conference on Computer Vision*, pages 423–439. Springer, 2022. 5

[30] Qihao Liu, Yi Zhang, Song Bai, Adam Kortylewski, and Alan Yuille. Direct-3d: Learning direct text-to-3d generation on massive noisy 3d data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6881–6891, 2024. 3

[31] Chaofan Luo, Donglin Di, Xun Yang, Yongjia Ma, Zhou Xue, Chen Wei, and Yebin Liu. Trame: Trajectory-anchored multi-view editing for text-guided 3d gaussian splatting manipulation. *arXiv preprint arXiv:2407.02034*, 2024. 3

[32] 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. 2

[33] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Marcus A Brubaker, Jonathan Kelly, Alex Levinshtein, Konstantinos G Derpanis, and Igor Gilitschenski. Watch your steps: Local image and scene editing by text instructions. In *European Conference on Computer Vision*, pages 111–129. Springer, 2025. 2, 3

[34] JangHo Park, Gihyun Kwon, and Jong Chul Ye. Ed-nerf: Efficient text-guided editing of 3d scene with latent space nerf. In *The Twelfth International Conference on Learning Representations*, 2024. 3, 7

[35] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 3

[36] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 3

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 7

[38] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 3

[39] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 3

[40] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 3, 5

[41] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. 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*, pages 22500–22510, 2023. 3

[42] Chitwan Saharia, William Chan, Huiwen Chang, Chris Lee, Jonathan Ho, Tim Salimans, David Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022. 3

[43] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 3

[44] J. Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20875–20886, 2023. 3

[45] Chenyang Si, Ziqi Huang, Yuming Jiang, and Ziwei Liu. Freeu: Free lunch in diffusion u-net. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4733–4743, 2024. 3

[46] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 3

[47] Yuxuan Wang, Xuanyu Yi, Zike Wu, Na Zhao, Long Chen, and Hanwang Zhang. View-consistent 3d editing with gaussian splatting. In *European Conference on Computer Vision*, pages 404–420. Springer, 2025. 2, 3, 5, 7

[48] Jing Wu, Jia-Wang Bian, Xinghui Li, Guangrun Wang, Ian Reid, Philip Torr, and Victor Adrian Prisacariu. Gaussctrl: multi-view consistent text-driven 3d gaussian splatting editing. *arXiv preprint arXiv:2403.08733*, 2024. 2, 3, 5, 6, 7, 8

[49] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[50] Sihan Xu, Yidong Huang, Jiayi Pan, Ziqiao Ma, and Joyce Chai. Inversion-free image editing with natural language. *arXiv preprint arXiv:2312.04965*, 2023. 3

[51] Yinghao Xu, Hao Tan, Fujun Luan, Sai Bi, Peng Wang, Jiahao Li, Zifan Shi, Kalyan Sunkavalli, Gordon Wetzstein, Zexiang Xu, and Kai Zhang. DMV3d: Denoising multi-view diffusion using 3d large reconstruction model. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[52] 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*, pages 1790–1799, 2020. 6

[53] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 3, 7

[54] 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 (CVPR)*, 2018. 6

[55] Jingyu Zhuang, Chen Wang, Liang Lin, Lingjie Liu, and Guanbin Li. Dreameditor: Text-driven 3d scene editing with neural fields. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–10, 2023. 3