# CUESPLAT: GUIDING GAUSSIAN SCENE UPDATES WITH PAIRED 3D EDIT INFERENCE

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Existing text-driven Gaussian editing methods commonly edit rendered views and fit the source Gaussians to the resulting images. Due to geometric changes and multi-view inconsistencies generated during the editing process, image fitting alone is not enough to specify which source surfaces to retain or reconstruct. We present CUESplat, which uses paired 3D edit inference to guide these scene updates. During denoising, a geometry predictor updates edited depths in the source scene frame and returns them to subsequent steps for cross-view coupling. After candidate generation, paired source and edited geometry provides displacement evidence, checked against source geometry and across views, to guide local Gaussian updates. To preserve usable source content, CUESplat reuses existing surfaces before introducing geometry for missing content. This couples structural revision with image fitting for appearance modification, object addition, and object removal. Experiments demonstrate that our method outperforms existing state-of-the-art Gaussian editing methods on three tasks.

## 1 INTRODUCTION

Text-driven Gaussian editing aims to modify a 3D scene with a text instruction while preserving content outside the requested change. The explicit primitives and efficient rendering of 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) make it a practical representation for this task (Yao et al., 2026; Zhao et al., 2026). By applying text-driven image editors to rendered views, Gaussian editing methods can use the resulting images to guide appearance changes and object addition or removal (Brooks et al., 2023; Xu et al., 2023; Vachha & Haque, 2024; Chen et al., 2024b).

As shown in Fig. 1, existing Gaussian editing methods commonly follow this *render–edit–fit* pipeline, optimizing the source Gaussians to reproduce the edited views (Vachha & Haque, 2024; Chen et al., 2024b). Recent work has progressively improved the multi-view consistency of edited images through geometric correspondences, multi-view fusion, or video priors (Wu et al., 2024; Chen et al., 2024a; Lee et al., 2025; Zhu et al., 2026; Chen et al., 2026). EditSplat (Lee et al., 2025) further uses attention maps from the edit diffusion model to localize Gaussian pruning and optimization. However, better multi-view agreement does not eliminate inconsistencies introduced during editing. When edited images or intermediate attention maps guide Gaussian updates without checking the geometric support for their proposed changes, these residual inconsistencies can propagate into Gaussian appearance optimization. Our key insight is to verify whether a proposed edit is geometrically supported before using it to update the scene. The evolving edited geometry guides subsequent denoising through paired source geometry, while the final geometric changes are checked against the source scene and across views to provide evidence for Gaussian adaptation and to downweight unsupported supervision.

To address these challenges, we propose **CUESplat**, which involves three modules: **Multi-view Edit Coupling (MEC)**, **Structural Edit Inference (SEI)** and **Selective Gaussian Adaptation (SGA)**. **Multi-view Edit Coupling (MEC)** exchanges editing increments along geometric correspondences that follow the evolving candidate scene and provides consistent edited views.

**Structural Edit Inference (SEI)** uses a feed-forward geometry predictor on paired source and edited views in a common scene frame. The source prediction provides a reference for measuring edit displacement, while source-rendered depth ties that displacement to the Gaussians being

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

updated. SEI retains changes whose directions agree under both comparisons and checks their support across views. Its geometry estimates return to MEC for subsequent denoising steps, while its final edit evidence guides structural updates and weights image fitting.

Given the resulting evidence, **Selective Gaussian Adaptation (SGA)** revises the scene while preserving usable source surfaces. It associates supported removals with the Gaussians responsible for the original foreground, then checks what their removal reveals. Evidence-weighted fitting then refines the revised scene with different update permissions for retained and new primitives. As shown in Fig. 1, this makes the inferred surface changes guide both the scene structure and its subsequent optimization. Our contributions are:

- We propose **Structural Edit Inference** to compare geometry before and after editing, using the source scene and other views to check each change. This evidence guides structural updates and reduces the influence of inconsistent edited regions.
- We propose **Selective Gaussian Adaptation** to turn this evidence into local scene updates. It removes selected Gaussians, reuses existing surfaces, and adds missing geometry, then refines the scene while limiting changes to retained content.
- We propose **Multi-view Edit Coupling** to coordinate edits across camera views by sharing changes at matching scene locations. It updates these matches as the edited geometry evolves, improving agreement across views while preserving their original context.

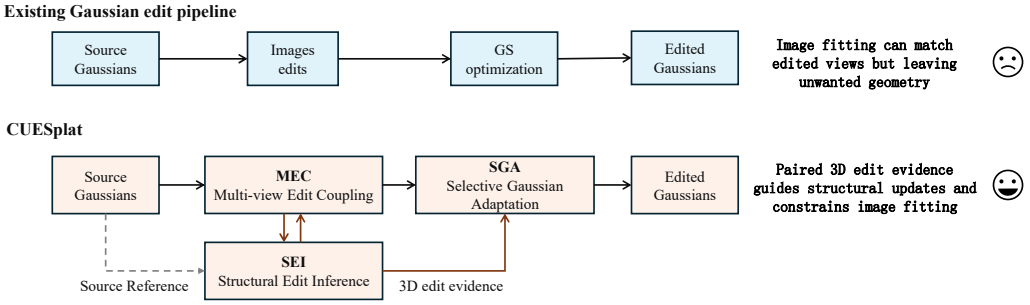


Figure 1: **Gaussian editing pipelines.** Top: the common render–edit–fit workflow optimizes source Gaussians against edited views, which can leave unwanted geometry despite matching the images. Bottom: CUESplat couples multi-view edits through MEC and uses SEI to infer 3D edit evidence relative to the source scene. SGA uses this evidence to guide structural updates and constrain subsequent image fitting. The dashed path supplies the source reference for SEI.

## 2 Related Work

**Text-driven 3D editing.** Diffusion models (Ho et al., 2020; Song et al., 2020; Meng et al., 2021; Mokady et al., 2023; Brooks et al., 2023; Xu et al., 2023; Rombach et al., 2022) provide generative priors for image editing. Neural radiance fields (Mildenhall et al., 2021) and CLIP (Radford et al., 2021) underpin several text-driven 3D methods. CLIP-NeRF (Wang et al., 2022) manipulates shape and appearance, while NeRF-Art (Wang et al., 2023) stylizes pretrained scenes. Text2Mesh (Michel et al., 2022) optimizes mesh appearance and geometry from text, and DreamFusion (Poole et al., 2022) distills a diffusion prior for text-to-3D generation. For editing existing scenes, iterative dataset updates (Haque et al., 2023; Vachha & Haque, 2024) transfer edited images into a shared representation. GaussianEditor (Chen et al., 2024b) uses semantic tracing and hierarchical Gaussian constraints for local control. GaussCtrl (Wu et al., 2024) and DGE (Chen et al., 2024a) coordinate views through depth and feature correspondences. TrAME (Luo et al., 2025), SyncNoise (Li et al., 2025), and CoreEditor (Zhu et al., 2026) further constrain cross-view exchange, while DreamCatalyst (Kim et al., 2024) studies diffusion-guided editing dynamics. ViP3DE (Chen et al., 2026) transfers video consistency to scene editing, and Omni-3DEdit (Liyi et al., 2026) learns multi-view editing across several tasks. RL3DEdit (Wang et al., 2026) uses geometric feedback as a training reward for multi-view editing. Our geometry branch supplies inference-time depth feedback to subsequent denoising steps; paired source and final edited predictions then determine local evidence for revising an existing Gaussian scene.

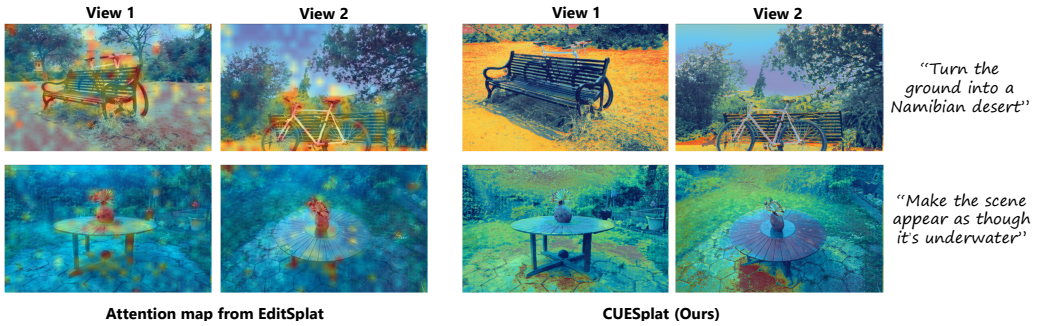


Figure 2: **Spatial guidance across views.** Left: attention maps from EditSplat (Lee et al., 2025). Right: CUESplat visualizations for the same scenes and instructions. Each example shows two camera views. The top row requests turning the ground into a Namibian desert; the bottom row requests an underwater appearance. The overlays illustrate how the spatial responses differ across the scene and between views. EditSplat's diffusion attention is concentrated on central source objects in these examples. CUESplat's SEI-guided spatial response extends to regions affected by the scene-level instructions.

**Selective scene updates.** Scene-maintenance methods GS-LTS (Fu et al., 2025) and SCaR-3D (Zhou et al., 2026) detect changes from new observations and selectively update a scene. FocusGS (Pan & Shi, 2026) studies local repair and deterministic asset editing. For generated appearance supervision, support-aware distillation (Cao et al., 2026) suppresses unsupported details while updating fixed-geometry appearance. Geometry-aware removal methods additionally address newly exposed regions (Ni et al., 2026). EditSplat (Lee et al., 2025) combines multi-view fusion with attention-guided trimming, and Look Before You Edit (Park et al., 2026) studies camera selection and multi-view alignment. We study generated multi-view candidates, where surface changes and uncertain supervision occur together. The same local evidence coordinates structural operations, reuse of source surfaces, and the allocation of optimization freedom.

## 3 METHODOLOGY

Given a source Gaussian scene $\mathcal{G}_s$, calibrated cameras $\{C_i\}$, and an instruction $y$, our goal is an edited scene $\mathcal{G}_e$ that realizes the requested change while retaining source content elsewhere. A Gaussian $g$ has center $\boldsymbol{\mu}_g$, covariance $\boldsymbol{\Sigma}_g$, opacity $o_g$, and appearance coefficients $\mathbf{c}_g$. The source images $I_i^s$ are the original training RGB views; rendering $\mathcal{G}_s$ at the same cameras supplies depths $D_i^{\mathrm{GS}}$ and opacity maps. MEC coordinates candidate edits, and SEI returns evolving geometry to its subsequent denoising steps. SEI then evaluates the final candidates to guide SGA (Fig. 1). Throughout, depth denotes positive camera-$z$ depth, and all reconstructed points are expressed in the source scene frame.

### 3.1 MULTI-VIEW EDIT COUPLING (MEC)

MEC couples editing across $N$ key views through relative edit-increment exchange along geometric correspondences. Dynamic correspondence refresh updates these matches as the edited geometry evolves. Within each view, source query/key mixing preserves spatial context, while source key/value contrast control strengthens contextual features.

Let $t = 0, \ldots, L-1$ index denoising iterations in execution order, and let $T$ and $S$ denote target and source-proxy streams. For token location $p$, the cross-view increment is

$$\Delta V_{i,t}(p) = \mathrm{Attn}\big(Q_{i,t}^T(p), \{K_{j,t}^T(q)\}, \{V_{j,t}^T(q) - V_{j,t}^S(q)\}\big), \quad q \in \mathcal{N}_{ij,t}(p),\ j \neq i. \tag{1}$$

Here $Q, K, V$ are query, key, and value tokens, Attn is scaled dot-product attention, and $\mathcal{N}_{ij,t}(p)$ contains valid corresponding tokens in view $j$. The increment corrects the local target attention output:

$$\widetilde{V}_{i,t}^T = V_{i,t}^S + \Delta V_{i,t}, \qquad O_{i,t}^T = O_{i,t}^{\mathrm{local}} + \lambda_t(\widetilde{V}_{i,t}^T - V_{i,t}^T), \tag{2}$$

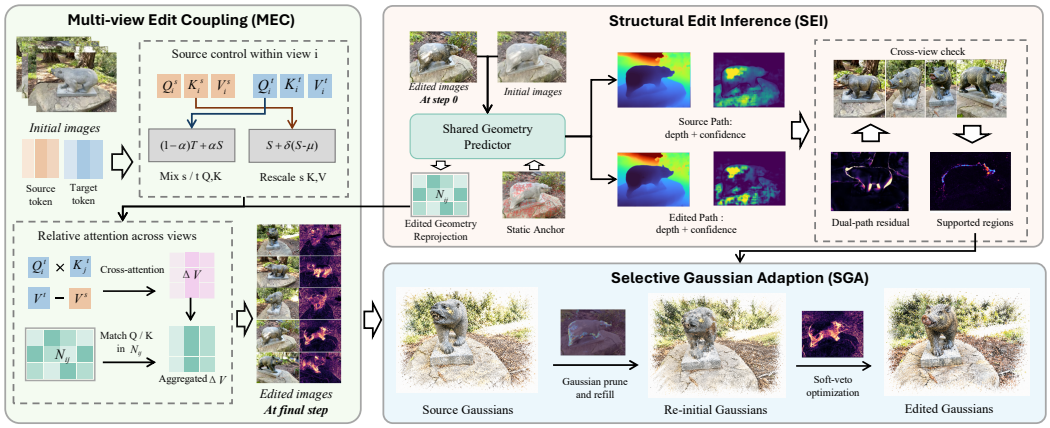


Figure 3: **Overview of CUESplat.** Left: MEC combines within-view source-feature control with correspondence-aligned editing increments across views; the edited depths returned by SEI refresh the correspondences for the next denoising step. Top right: SEI predicts source and edited geometry under shared conditions and source-depth anchors, then combines source-relative depth residuals with cross-view support to infer local edit evidence. Bottom right: SGA uses this evidence to prune source Gaussians and initialize missing content, followed by refinement that downweights unsupported edited pixels.

where $\lambda_t$ controls the correction strength. The recursive source proxy uses an RGB-change mask after iterations $t = 0$ and $t = 1$ to update edited source-proxy tokens from the current clean-sample latent, carrying forward current edits during subsequent source control. The original $I_i^s$ and $\mathcal{G}_s$ remain fixed references for SEI.

### 3.2 Structural Edit Inference (SEI)

SEI supplies geometry during denoising and structural evidence after denoising. Initially, $D^e_{i,-1} = D_i^{\mathrm{GS}}$ defines MEC's correspondences. After iteration $t$, the predicted clean sample $\hat{x}_{0,i,t}$ is decoded to $\hat{I}^e_{i,t}$, and SEI's geometry branch predicts its depth $D^e_{i,t}$ under the source cameras, scene scale, and sparse source-depth anchors. This depth returns to MEC to construct $\mathcal{N}_{ij,t+1}$ for the next attention correction in Eq. (1):

$$\mathbf{X}^e_{i,t}(p) = R_i\big(D^e_{i,t}(p)A_i^{-1}\bar{p}\big) + \mathbf{b}_i, \qquad p_j = \pi_j\big(\mathbf{X}^e_{i,t}(p)\big)\,, \tag{3}$$

where $\bar{p}$ is a homogeneous pixel coordinate, $A_i$ is the intrinsic matrix, $R_i$ and $\mathbf{b}_i$ define the camera-to-world pose, and $\pi_j$ projects into camera $j$. In-frame, positive-depth matches that agree with $D^e_{j,t}(p_j)$ within relative tolerance $\tau_d$ are mapped to tokens. Thus each subsequent MEC step uses correspondences updated from the evolving edited geometry.

After denoising, the final candidates $I^e_i = \hat{I}^e_{i,L-1}$ enter paired geometry prediction with $I^s_i$. The same predictor, cameras, scale, and source-depth anchors yield $D^s_i, D^e_i$. SEI compares edited depth against both the paired source prediction and the existing Gaussian scene:

$$r_i^{\mathrm{pair}}(p) = \log\frac{D^e_i(p)}{D^s_i(p)}, \qquad r_i^{\mathrm{src}}(p) = \log\frac{D^e_i(p)}{D_i^{\mathrm{GS}}(p)}. \tag{4}$$

Here $r_i^{\mathrm{pair}}$ measures displacement relative to the paired source prediction, while $r_i^{\mathrm{src}}$ is the source-GS residual that anchors the edited prediction to the geometry represented by the source Gaussians. On their joint valid-depth domain, structural evidence requires agreement in change direction and takes the smaller magnitude:

$$r_i(p) = \begin{cases} \mathrm{sign}(r_i^{\mathrm{pair}}(p))\,\min\big(|r_i^{\mathrm{pair}}(p)|, |r_i^{\mathrm{src}}(p)|\big), & r_i^{\mathrm{pair}}(p)r_i^{\mathrm{src}}(p) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

A receding surface ($r_i > 0$) suggests removal, while a nearer surface ($r_i < 0$) suggests addition. The corresponding evidence channels are

$$e_i^-(p) = [r_i(p)]_+ c_i^s(p) w_i^{\mathrm{rgb}}(p), \qquad e_i^+(p) = [-r_i(p)]_+ c_i^e(p) w_i^{\mathrm{rgb}}(p), \tag{6}$$

where $[\cdot]_+$ takes the positive part, $w_i^{\mathrm{rgb}}$ gates image change, and $c_i^s, c_i^e \in [0,1]$ provide per-view normalized geometric confidence for the source and edited predictions, respectively.

To verify cross-view support, local candidates with $\max(e_i^-, e_i^+) \geq \tau_e$ undergo the same reprojection-depth test on the final edited geometry. Agreement in at least $n_{\mathrm{sup}}$ remote views defines the supported-change mask $M_i$, followed by morphological closing. Unsupported candidates form $U_i$, from which a continuous fitting weight $W_i$ reduces their supervision. SEI passes three outputs to SGA:

$$\mathcal{E}_i = \big(M_i e_i^-,\ M_i e_i^+,\ W_i\big). \tag{7}$$

The first two channels guide structural updates; $W_i$ weights image fitting. These final outputs are distinct from the depth feedback used during MEC denoising.

### 3.3 Selective Gaussian Adaptation (SGA)

SGA maps SEI's evidence onto Gaussian primitives, revises the scene, and refines it against the edited views. It uses contribution-based attribution to identify affected primitives, performs source-surface reuse before initializing missing geometry, and applies evidence-weighted fitting with selective update permissions. For Gaussian $g$, its compositing contribution $a_{ig}(p) = T_{ig}(p)\alpha_{ig}(p)$ combines transmittance $T_{ig}$ and projected opacity $\alpha_{ig}$. Supported removal evidence is attributed by

$$S_g^- = \sum_{i,p} a_{ig}(p)\, M_i(p) e_i^-(p). \tag{8}$$

Per-view contributions, normalized by contributing pixel count and checked for visibility, determine the removal set $\mathcal{G}^-$.

We render $\mathcal{G}_s \setminus \mathcal{G}^-$ and reuse revealed surfaces with sufficient opacity and agreement with edited depth. Within the deleted footprint, measured by the opacity drop, remaining unexplained target surfaces initialize completion Gaussians $\mathcal{G}^{\mathrm{fill}}$. Supported addition evidence initializes $\mathcal{G}^+$, yielding

$$\mathcal{G}_0 = (\mathcal{G}_s \setminus \mathcal{G}^-) \cup \mathcal{G}^{\mathrm{fill}} \cup \mathcal{G}^+. \tag{9}$$

For refinement, RGB change within $H_i = 1 - \mathrm{Dilate}(U_i)$ is attributed through the same compositing weights to select editable source primitives, including appearance-only changes. Retained editable primitives update appearance and opacity with geometry fixed; new primitives optimize all attributes. Cloning, splitting, and pruning are restricted to the editable set. For $\hat{I}_i = \mathcal{R}(\mathcal{G}, C_i)$, the evidence-weighted RGB loss is

$$\mathcal{L}_{\mathrm{rgb}} = \frac{1}{N}\sum_{i=1}^{N} \frac{\sum_p w_i(p)\|\hat{I}_i(p) - I_i^e(p)\|_1}{3\sum_p w_i(p)}, \qquad w_i(p) = \epsilon + (1-\epsilon)W_i(p), \tag{10}$$

where $0 < \epsilon < 1$ is a supervision floor. This evidence-weighted fitting is combined with an LPIPS loss. After key-view optimization, the intermediate scene supplies renders for an expanded-view editing and refinement pass that produces $\mathcal{G}_e$.

## 4 Experiments

### 4.1 Experimental Setup

**Implementation Details.** **CUESplat** represents each scene using 3DGS (Kerbl et al., 2023) and is initialized from a pretrained source Gaussian model. FLUX.2 [klein] 4B (Black Forest Labs, 2026) and MapAnything (Keetha et al., 2026) are used as the image editor and feed-forward geometry predictor, respectively. For each task, we uniformly select $N = 12$ key views from the training set. FLUX performs four denoising steps at a resolution with a 512-pixel longer side. After each step, we decode the predicted clean sample $\hat{x}_{0,i,t}$, estimate its depth using MapAnything, and refresh the geometric correspondences for the next step. In SEI, we use a local evidence threshold of $\tau_e = 0.01$, a relative reprojection-depth tolerance of $\tau_d = 0.06$, and require support from at least $n_{\mathrm{sup}} = 1$ additional view. We first optimize an intermediate Gaussian scene using the edited key views and then refine it using edited renders from all training views. Both optimization stages run for 1,600 iterations, with densification and pruning performed at iterations 400, 800, and 1,200. We use a combination of L1 and LPIPS losses and apply the same configuration to all scenes.




Figure 4: **Qualitative comparison on appearance editing.** We compare CUESplat with representative 3D Gaussian editing baselines across diverse appearance-editing tasks. Each row corresponds to a different editing instruction, and the inset shows a second viewpoint of the same edited scene.

**Scenes and Editing Tasks.** We evaluate **CUESplat** on eight scenes, including the *bear*, *face* and *person* scenes from Instruct-NeRF2NeRF (Haque et al., 2023), the *bicycle*, *bonsai*, and *garden* scenes from Mip-NeRF 360 (Barron et al., 2022), the *stone horse* scene from BlendedMVS (Yao et al., 2020), and the *fangzhou* scene from NeRF-Art (Wang et al., 2023). We evaluate 28 editing tasks in total, comprising 16 appearance-editing tasks, six object-addition tasks, and six object-removal tasks. Additional experimental details are organized in the Appendix B.

**Baselines.** For appearance editing, we compare **CUESplat** with nine state-of-the-art Gaussian-based 3D editing methods: Instruct-GS2GS (Vachha & Haque, 2024), GaussianEditor (Chen et al., 2024b), GaussCtrl (Wu et al., 2024), DGE (Chen et al., 2024a), DreamCatalyst (Kim et al., 2024), EditSplat (Lee et al., 2025), TrAME (Luo et al., 2025), ViP3DE (Chen et al., 2026), and CoreEditor (Zhu et al., 2026). We additionally compare with Omni-3DEdit (Liyi et al., 2026), a recent pre-trained multi-view image editing model. For object addition and removal, we compare our method with GaussianEditor and Omni-3DEdit. Since Omni-3DEdit produces edited multi-view images rather than an optimized Gaussian scene, we use its outputs to supervise the optimization of the original 3DGS, enabling evaluation under the same 3D representation and rendering protocol.

Table 1: **Quantitative comparison on appearance editing and add/remove editing.** We report CLIP similarity (CLIP-S), CLIP directional similarity (CLIP-D), DINO feature similarity, Gemini-based editing quality, and MEt3R multi-view inconsistency. Higher CLIP-S, CLIP-D, DINO, and Gemini scores are better, while lower MEt3R is better. **Panel (a)** reports results averaged over the 16 appearance-editing tasks, while **Panel (b)** reports results on add/remove editing.

| Method | CLIP-S ↑ | CLIP-D ↑ | DINO ↑ | Gemini ↑ | MEt3R ↓ |
|---|---|---|---|---|---|
| *(a) Appearance Editing* | | | | | |
| Instruct-GS2GS (Vachha & Haque, 2024) | 0.2270 | <u>0.1327</u> | 0.7245 | 1.50 | 0.3463 |
| GaussianEditor (Chen et al., 2024b) | 0.2346 | 0.0887 | 0.7571 | 1.63 | 0.2889 |
| GaussCtrl (Wu et al., 2024) | <u>0.2453</u> | 0.0959 | 0.7688 | <u>2.69</u> | 0.3185 |
| DGE (Chen et al., 2024a) | 0.2375 | 0.1049 | 0.7658 | <u>2.69</u> | 0.4043 |
| DreamCatalyst (Kim et al., 2024) | 0.2401 | 0.0828 | <u>0.7707</u> | 1.88 | 0.2967 |
| EditSplat (Lee et al., 2025) | 0.2445 | 0.1133 | 0.7661 | 2.25 | 0.4779 |
| TrAME (Luo et al., 2025) | 0.2430 | 0.0864 | 0.7693 | 2.31 | 0.3121 |
| ViP3DE (Chen et al., 2026) | 0.2349 | 0.0631 | 0.7530 | 1.31 | 0.2723 |
| CoreEditor (Zhu et al., 2026) | 0.2317 | 0.0622 | 0.7611 | 2.13 | 0.4005 |
| Omni-3DEdit (Liyi et al., 2026) | 0.2190 | 0.0484 | 0.7670 | 1.88 | <u>0.2476</u> |
| **CUESplat (Ours)** | **0.2608** | **0.1765** | **0.7745** | **3.13** | **0.2448** |
| *(b) Add/Remove Editing* | | | | | |
| GaussianEditor (Chen et al., 2024b) | <u>0.2321</u> | <u>0.0822</u> | <u>0.6849</u> | 1.33 | <u>0.2745</u> |
| Omni-3DEdit (Liyi et al., 2026) | 0.2253 | 0.0708 | 0.6647 | <u>1.50</u> | 0.2869 |
| **CUESplat (Ours)** | **0.2420** | **0.1113** | **0.7050** | **1.83** | **0.2530** |

**Evaluation Metric.** Following common practice in 3D editing, we evaluate semantic alignment using CLIP similarity (CLIP-S) (Radford et al., 2021) and CLIP directional similarity (CLIP-D) (Gal et al., 2022). For each task, we first average the scores over all evaluation views and then report an equal-weighted macro-average across tasks. We additionally use DINO feature similarity (DINO) (Caron et al., 2021) to measure perceptual agreement across rendered views. We employ Gemini 2.5 Pro (Gemini) (Google) as an MLLM-based evaluator, using the evaluation prompt from Omni-3DEdit (Liyi et al., 2026) to jointly assess instruction faithfulness, multi-view consistency, and visual quality. For each task, we uniformly select four evaluation views and jointly provide their source images, edited images, and the editing instruction to Gemini, using a temperature of zero. Finally, we use MEt3R (Asim et al., 2025) to measure multi-view geometric inconsistency.

### 4.2 Qualitative Results

We present the qualitative comparison in Fig. 4, showcasing editing results from two viewpoints. Compared with the baselines (Chen et al., 2024a; Kim et al., 2024; Lee et al., 2025) shown in Fig. 4, CUESplat more consistently follows the illustrated instructions across the two viewpoints. The quantitative comparisons and ablations provide broader support.

Additional qualitative comparisons on object addition and removal are provided in Appendix D.

### 4.3 Quantitative Results

The quantitative comparison with baseline methods is summarized in Tab. 1. As shown in **Panel (a)**, **CUESplat** achieves the best results across all five metrics on the 16 appearance-editing tasks. Compared with the strongest baseline for each metric, **CUESplat** improves CLIP-S from 0.2453 to 0.2608, CLIP-D from 0.1327 to 0.1765, and the Gemini score from 2.69 to 3.13. It also obtains the highest DINO similarity (0.7745) and the lowest MEt3R score (0.2448). These results indicate that **CUESplat** improves instruction alignment while producing more consistent edits across viewpoints. In particular, MEC coordinates the candidate edits during generation, while SEI and SGA reduce the influence of image-space changes that are unsupported by cross-view geometric evidence.

**Panel (b)** reports the aggregate results over six object-addition and six object-removal tasks. **CUESplat** again outperforms GaussianEditor and Omni-3DEdit across all five metrics. Relative to the strongest baseline for each metric, **CUESplat** improves CLIP-S, CLIP-D, DINO, and Gemini by 0.0099, 0.0291, 0.0201, and 0.33, respectively, while reducing MEt3R by 0.0215. Unlike appear-

Table 2: **Generalization across candidate editors.** We integrate **SEI** and **SGA** into **DGE** (Chen et al., 2024a) and **EditSplat** (Lee et al., 2025) while retaining their original 2D candidate editors. Values in parentheses denote the improvement over the corresponding original method. Higher is better for all metrics except **MEt3R**.

| Method | CLIP-S ↑ | CLIP-D ↑ | DINO ↑ | Gemini ↑ | MEt3R ↓ |
|---|---|---|---|---|---|
| DGE | 0.2375 | 0.1049 | 0.7658 | 2.69 | 0.4043 |
| DGE + SEI + SGA | 0.2488 (+0.0113) | 0.1205 (+0.0156) | 0.7691 (+0.0033) | 2.81 (+0.12) | 0.3027 (−0.1016) |
| EditSplat | 0.2445 | 0.1133 | 0.7661 | 2.25 | 0.4779 |
| EditSplat + SEI + SGA | 0.2547 (+0.0102) | 0.1361 (+0.0228) | 0.7702 (+0.0041) | 2.44 (+0.19) | 0.3225 (−0.1554) |
| CUESplat (Ours) | **0.2608** | **0.1765** | **0.7745** | **3.13** | **0.2448** |

ance editing, object addition and removal require changes to the visible surfaces and the Gaussian representation. The results therefore support the effectiveness of paired 3D edit inference for translating edited image candidates into structurally meaningful and view-consistent scene updates.

### 4.4 Generalization across Candidate Editors

To separate the contribution from the image editing backbone, we integrate SEI and SGA into **DGE** (Chen et al., 2024a) and **EditSplat** (Lee et al., 2025), while retaining the original 2D candidate edits produced by each method. This setting therefore evaluates whether our structural edit inference (SEI) and selective Gaussian adaptation (SGA) can benefit different render–edit–fit pipelines, rather than relying on a particular editing backbone.

Adding SEI and SGA to DGE improves CLIP-S, CLIP-D, DINO, and Gemini by 0.0113, 0.0156, 0.0033, and 0.12, respectively, while reducing MEt3R from 0.4043 to 0.3027. The same trend is observed for EditSplat, suggesting that the benefit is not specific to DGE or to a particular form of 2D candidate generation. Across both backbones, the consistent improvements indicate that SEI and SGA improve both edit alignment and cross-view consistency: structural evidence helps distinguish reliable edits from unsupported changes, while selective adaptation concentrates optimization on Gaussians that are actually associated with the edited regions. Full CUESplat still achieves the strongest overall performance, suggesting that MEC and SEI/SGA play complementary roles: MEC produces more coordinated multi-view candidates, whereas SEI and SGA determine how these candidates should be interpreted and transferred back to the 3D representation.

### 4.5 Ablation Studies

Tab. 3 reports ablations on the 16 appearance-editing tasks, using the same five metrics as Tab. 1(a). Each row changes the full model independently; indented rows are component-level variants rather than cumulative removals. Appendix C specifies the corresponding operations.

**Major components.** Removing Multi-view Edit Coupling (MEC), final Structural Edit Inference (SEI), or Selective Gaussian Adaptation (SGA) degrades performance on all five metrics relative to the full model. Without MEC, CLIP-D falls from 0.1765 to 0.1433 and MEt3R rises from 0.2448 to 0.2656. The SEI ablation removes final evidence inference while retaining the stepwise depth feedback required by MEC; CLIP-S drops to 0.2492 and Gemini to 2.69. The SGA ablation retains evidence-weighted fitting but replaces structural revision and selective updates with direct Gaussian fitting; CLIP-D falls to 0.1522 and MEt3R rises to 0.2668. These comparisons support using both final edit evidence and its associated scene updates.

**Multi-view Edit Coupling.** Replacing relative edit-increment exchange in Eq. (1) with absolute target-feature aggregation raises MEt3R to 0.2528. Disabling dynamic correspondence refresh fixes matches to source-GS depth throughout denoising and reduces DINO to 0.7602. Disabling the recursive source proxy keeps the source latent fixed and raises MEt3R to 0.2650. Together, these results support source-relative feature exchange, evolving geometric matches, and an updated source context.

Table 3: **Ablation study on 16 appearance-editing tasks.** Each variant independently modifies full CUESplat. Indentation groups internal mechanisms by module. The SEI row removes final evidence inference while retaining MEC's stepwise depth feedback. Higher is better except for MEt3R. Operational definitions are given in Appendix C.

| Method | CLIP-S ↑ | CLIP-D ↑ | DINO ↑ | Gemini ↑ | MEt3R ↓ |
|---|---|---|---|---|---|
| **CUESplat (Ours)** | **0.2608** | **0.1765** | **0.7745** | **3.13** | **0.2448** |
| ↪ w/o Multi-view Edit Coupling (MEC) | 0.2527 | 0.1433 | 0.7605 | 2.75 | 0.2656 |
| ↪ w/o Relative Edit-Increment Exchange | 0.2562 | 0.1701 | 0.7704 | 3.06 | 0.2528 |
| ↪ w/o Dynamic Correspondence Refresh | 0.2601 | 0.1650 | 0.7602 | 2.81 | 0.2655 |
| ↪ w/o Recursive Source Proxy | 0.2559 | 0.1686 | 0.7617 | 2.94 | 0.2650 |
| ↪ w/o SEI Structural Evidence | 0.2492 | 0.1538 | 0.7599 | 2.69 | 0.2602 |
| ↪ w/o Source-GS Residual | 0.2513 | 0.1702 | 0.7722 | 2.88 | 0.2586 |
| ↪ w/o Cross-view Support | 0.2503 | 0.1681 | 0.7716 | 2.81 | 0.2553 |
| ↪ w/o Geometric Confidence | 0.2557 | 0.1666 | 0.7712 | 2.94 | 0.2527 |
| ↪ w/o Selective Gaussian Adaptation (SGA) | 0.2533 | 0.1522 | 0.7607 | 2.89 | 0.2668 |
| ↪ w/o Contribution-based Attribution | 0.2579 | 0.1705 | 0.7683 | 2.94 | 0.2581 |
| ↪ w/o Source-surface Reuse | 0.2573 | 0.1699 | 0.7677 | 3.06 | 0.2579 |
| ↪ w/o Evidence-weighted Fitting | 0.2594 | 0.1711 | 0.7711 | 3.00 | 0.2572 |
| ↪ w/o Expanded-view Editing and Refinement | 0.2423 | 0.1316 | 0.7609 | 2.56 | 0.2687 |

**Structural Edit Inference.** Removing the source-GS residual from Eq. (5) uses only $r_i^{\text{pair}}$ and raises MEt3R to 0.2586. Removing cross-view support admits local evidence without remote votes, reducing CLIP-S to 0.2503 and Gemini to 2.81. Removing geometric confidence sets $c_i^s = c_i^e = 1$ in Eq. (6) and also degrades all metrics. These comparisons support the source-GS reference, remote geometric support, and predictor confidence as complementary filters on final edit evidence.

**Selective Gaussian Adaptation.** Replacing contribution-based attribution with projected-center sampling raises MEt3R to 0.2581. Disabling source-surface reuse initializes completion Gaussians throughout the valid deleted footprint, including regions already explained by revealed surfaces, and raises MEt3R to 0.2579. Removing evidence-weighted fitting sets $W_i = 1$ in Eq. (10) and raises MEt3R to 0.2572. These results favor contribution-aware updates and suppression of unsupported image supervision. Because this table evaluates appearance edits, the reuse comparison concerns local surface changes inferred within those tasks; it does not isolate background completion on object-removal tasks.

**Expanded-view Editing and Refinement.** Stopping after key-view optimization reduces CLIP-S, CLIP-D, and Gemini to 0.2423, 0.1316, and 2.56, respectively, and raises MEt3R to 0.2687. This is the largest reduction in these three semantic/quality scores and the largest increase in MEt3R among the tested variants, although other ablations yield lower DINO scores. The result supports the additional supervision obtained by editing intermediate-scene renders at all training cameras.

## 5 Conclusion

We presented CUESplat, a text-driven Gaussian editing framework that connects image editing to scene updates through paired 3D edit inference. MEC shares source-relative editing increments using correspondences refreshed from evolving edited depths. SEI checks final surface changes against source geometry and across views, and SGA uses the resulting evidence to remove selected primitives, reuse available surfaces, initialize missing content, and constrain image fitting. Experiments on appearance editing, object addition, and object removal show improved instruction alignment and multi-view consistency, while transferring SEI and SGA to other candidate editors supports their usefulness beyond our editing backbone. The method remains dependent on source reconstruction quality and geometric predictions; improving their reliability and reducing the cost of the multi-stage pipeline are directions for future work.

## REPRODUCIBILITY STATEMENT

Section 3 defines the method, and Appendix A details its implementation and optimization schedule. Section 4.1 and Appendix B describe the datasets, tasks, and evaluation protocol; Appendix C specifies the ablation variants. Code will be released upon publication.

## AI USE STATEMENT

A large language model assisted with literature organization, language editing, and LaTeX drafting. The authors conducted the research and experiments and are responsible for verifying all derivations, citations, empirical claims, and final content.

## ETHICS AND BROADER IMPACT

Text-driven Gaussian editing supports content creation but can also produce misleading depictions or unauthorized alterations of captured people and private environments. Responsible use should respect consent, privacy, and dataset and model licenses, and disclose synthetic edits where provenance matters.

## REFERENCES

Mohammad Asim, Christopher Wewer, Thomas Wimmer, Bernt Schiele, and Jan Eric Lenssen. Met3r: Measuring multi-view consistency in generated images. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6034–6044. IEEE, 2025.

Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5460–5469. IEEE, 2022.

Black Forest Labs. FLUX.2 [klein] 4B. Hugging Face model card, 2026. URL `https://huggingface.co/black-forest-labs/FLUX.2-klein-4B`. Accessed September 22, 2026.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18392–18402. IEEE, 2023.

Long Cao, Zhongquan Wang, Jie Li, Yuhan Chen, Kefei Qian, Xiangfei Huang, and Guofa Li. From sparse and imperfect 2d anchors to consistent 3d gaussian street scenes: Support-aware appearance. *arXiv preprint arXiv:2606.26007*, 2026.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF international conference on computer vision (ICCV)*, pp. 9630–9640. IEEE, 2021.

Liyi Chen, Ruihuang Li, Guowen Zhang, Pengfei Wang, and Lei Zhang. Fast multi-view consistent 3d editing with video priors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 2948–2956, 2026.

Minghao Chen, Iro Laina, and Andrea Vedaldi. Dge: Direct gaussian 3d editing by consistent multi-view editing. In *European conference on computer vision*, pp. 74–92. Springer, 2024a.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21476–21485. IEEE, 2024b.

Bin Fu, Jialin Li, Bin Zhang, Ruiping Wang, and Xilin Chen. Gs-lts: 3d gaussian splatting-based adaptive modeling for long-term service robots. *arXiv preprint arXiv:2503.17733*, 2025.

Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (ToG)*, 41(4):1–13, 2022.

Google. Gemini API: Models. Online documentation. URL `https://ai.google.dev/gemini-api/docs/models`. Accessed September 22, 2026.

Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19683–19693. IEEE, 2023.

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

Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, et al. Mapanything: Universal feed-forward metric 3d reconstruction; map-anything. github. io. In *2026 International Conference on 3D Vision (3DV)*, pp. 499–509. IEEE, 2026.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Jiwook Kim, Seonho Lee, Jaeyo Shin, Jiho Choi, and Hyunjung Shim. Dreamcatalyst: Fast and high-quality 3d editing via controlling editability and identity preservation. *arXiv preprint arXiv:2407.11394*, 2024.

Dong In Lee, Hyeongcheol Park, Jiyoung Seo, Eunbyung Park, Hyunje Park, Ha Dam Baek, Sangheon Shin, Sangmin Kim, and Sangpil Kim. Editsplat: Multi-view fusion and attention-guided optimization for view-consistent 3d scene editing with 3d gaussian splatting. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11135–11145. IEEE, 2025.

Ruihuang Li, Liyi Chen, Zhengqiang Zhang, Varun Jampani, Vishal M Patel, and Lei Zhang. Syncnoise: Geometrically consistent noise prediction for instruction-based 3d editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4905–4913, 2025.

Chen Liyi, Wang Pengfei, Zhang Guowen, Ma Zhiyuan, and Zhang Lei. Omni-3dedit: Generalized versatile 3d editing in one-pass. *arXiv preprint arXiv:2603.17841*, 2026.

Chaofan Luo, Donglin Di, Xun Yang, Yongjia Ma, Zhou Xue, Wei Chen, Xiaofei Gou, and Yebin Liu. Trame: Trajectory-anchored multi-view editing for text-guided 3d gaussian manipulation. *IEEE Transactions on Multimedia*, 27:2886–2898, 2025.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13482–13492. IEEE, 2022.

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.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6038–6047. IEEE, 2023.

Yuanxiang Ni, Xianliang Huang, Chenhang Ma, Chen Xiao, Yuewen Ma, Ruxin Wang, and Hao Zhang. Cogeo-gs: Concept-driven and geometry-aware multi-object removal in 3d scenes. *arXiv preprint arXiv:2608.26656*, 2026.

Yiqun Pan and Yukun Shi. Focusgs: Spatial delta layers for local repair and deterministic editing of trained 3d gaussian assets. *arXiv preprint arXiv:2607.28834*, 2026.

Jaeyeon Park, Taeho Kang, and Youngki Lee. Look before you edit: Attention-guided camera placement and multi-view alignment for 3d gaussian splatting editing. *arXiv preprint arXiv:2607.19777*, 2026.

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

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*, pp. 8748–8763. PmLR, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Cyrus Vachha and Ayaan Haque. Instruct-gs2gs: Editing 3d gaussian splats with instructions. *URL https://instruct-gs2gs. github. io*, 2024.

Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3825–3834. IEEE, 2022.

Can Wang, Ruixiang Jiang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Nerf-art: Text-driven neural radiance fields stylization. *IEEE Transactions on Visualization and Computer Graphics*, 30(8):4983–4996, 2023.

Jiyuan Wang, Chunyu Lin, Lei Sun, Zhi Cao, Yuyang Yin, Lang Nie, Zhenlong Yuan, Xiangxiang Chu, Yunchao Wei, Kang Liao, et al. Geometry-guided reinforcement learning for multi-view consistent 3d scene editing. *arXiv preprint arXiv:2603.03143*, 2026.

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. In *European conference on computer vision*, pp. 55–71. Springer, 2024.

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

Nanjie Yao, Hao Wang, Chong Cheng, Zhikang Chen, Wenzhe Li, Jiafei Lyu, Li Shen, Peilin Zhao, Zongqing Lu, Gao Huang, et al. World models for embodied intelligence: From plausible to controllable to actionable. *arXiv preprint arXiv:2609.16697*, 2026.

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 *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1787–1796. IEEE, 2020.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Beizhen Zhao, Boyi Fu, Sicheng Yu, Zijian Wang, Kaiyong Zhao, Pengcheng Wu, Hao Wang, Steven Hoi, and Hui Xiong. 3d gaussian splatting rasterization: A survey. 2026.

Zirui Zhou, Junfeng Ni, Shujie Zhang, Yixin Chen, and Siyuan Huang. 3d scene change modeling with consistent multi-view aggregation. In *2026 International Conference on 3D Vision (3DV)*, pp. 576–586. IEEE, 2026.

Zhe Zhu, Honghua Chen, Peng Li, and Mingqiang Wei. Coreeditor: Correspondence-constrained diffusion for consistent 3d editing. *IEEE Transactions on Visualization and Computer Graphics*, 2026.

# A Implementation Details

This appendix follows the data flow in Section 3: candidate generation and depth feedback, final evidence extraction, Gaussian revision, and expanded-view refinement. It uses the same notation as the main text. In particular, $A_i$ denotes camera intrinsics, $N$ is the number of key views, $K_{i,t}^{S/T}$ denotes attention keys, and $H_i$ and $W_i$ are hard attribution masks and soft fitting weights, respectively.

## A.1 MEC: Candidate Generation and Depth Feedback

We use FLUX.2 [klein] 4B (Black Forest Labs, 2026) with $N = 12$ uniformly selected training views and $L = 4$ denoising iterations. Original training RGB images provide $I_i^s$; source-GS renders provide aligned depth and opacity. The image longer side is 512 pixels, with both dimensions divisible by 16. MapAnything (Keetha et al., 2026) predicts edited depth from each decoded clean estimate. The sequence is

$$D_{i,-1}^e = D_i^{\mathrm{GS}}, \qquad \hat{x}_{0,i,t} \longrightarrow \hat{I}_{i,t}^e \longrightarrow D_{i,t}^e \longrightarrow \mathcal{N}_{ij,t+1}, \quad t = 0, \ldots, L-2. \tag{11}$$

The last iteration produces the final candidates for paired SEI. Matches are formed by Eq. (3), using positive depth, valid image bounds, and the relative-depth check in Eq. (17). Only depth is returned during denoising; the final masks $M_i, U_i$ and fitting weight $W_i$ are computed after candidate generation.

**Source-feature control and recursive source proxy.** Within each view, source query/key mixing stabilizes spatial context without copying source values into the target stream. Source key/value contrast control operates around the mean over retained source tokens:

$$(K^S)' = \bar{K}^S + \delta_K(K^S - \bar{K}^S), \qquad (V^S)' = w_s\big[\bar{V}^S + \delta_V(V^S - \bar{V}^S)\big]. \tag{12}$$

The view and iteration indices are suppressed here; $\delta_K, \delta_V$ control contrast and $w_s$ controls the source-value gain. These within-view operations complement source-relative cross-view exchange in Eqs. (1) and (2).

Let $x_{i,t}^S$ be the source-proxy latent and $P_{i,t}$ its binary token-change mask, obtained from the RGB difference between the current clean estimate and the current source proxy. After iterations $t = 0$ and $t = 1$,

$$x_{i,t+1}^S = (1 - P_{i,t}) \odot x_{i,t}^S + P_{i,t} \odot \hat{x}_{0,i,t}. \tag{13}$$

The proxy is then held fixed. This update carries current edits into subsequent source control; it does not modify the original $I_i^s$ or $\mathcal{G}_s$ used by final SEI, and $P_{i,t}$ is distinct from the final supported-change mask $M_i$.

**Task-dependent schedules and addition layout.** Appearance edits use stronger source control. Addition uses weaker control and a shared layout proposal; removal delays source control by one step and reduces its strength to allow background completion. For addition, a reference-view edit is lifted into coarse 3D support, summarized by a contact-to-top capsule, and projected into the other cameras with source-depth visibility checks. This proposal specifies approximate location, extent, and visibility; final surfaces are obtained from the multi-view candidates and SEI evidence.

## A.2 SEI: Shared Geometry and Final Edit Evidence

**Common scene frame.** Source and edited predictions share calibrated cameras and sparse source-depth anchors. Let $s$ be the median depth of the selected anchors. Camera translations and anchor depths are normalized by the same scale, and predicted depths are restored afterward:

$$\mathbf{b}_i' = \mathbf{b}_i/s, \qquad D_i^{\mathrm{anchor}\prime} = D_i^{\mathrm{anchor}}/s, \qquad D_i^{s/e} = s\,\widetilde{D}_i^{s/e}. \tag{14}$$

No additional similarity transform is estimated. Anchors require source opacity at least $0.95$, low RGB change, and a weak log-depth gradient. Sampling excludes an eight-pixel border and uses stride four; inference requires at least 32 anchors.

**RGB gating and confidence.** For $d_i(p) = \|I_i^e(p) - I_i^s(p)\|_1/3$, define

$$\tau_i^{\mathrm{rgb}} = \max(\mathrm{median}(d_i) + 2.5 \cdot 1.4826\,\mathrm{MAD}(d_i), Q_{0.65}(d_i))\,, \tag{15}$$

$$G_i^{\mathrm{rgb}} = \mathrm{Dilate}\Big(\mathrm{Open}\Big(\mathbb{I}[d_i \geq \tau_i^{\mathrm{rgb}}]\Big)\Big)\,, \qquad w_i^{\mathrm{rgb}} = G_i^{\mathrm{rgb}}\,\mathrm{clip}\left(\frac{d_i}{\tau_i^{\mathrm{rgb}}}, 0, 1\right). \tag{16}$$

Here MAD is the median absolute deviation and $Q_{0.65}$ the 65th percentile. When the threshold and RGB difference are both zero, the gate weight is taken as zero. Geometric confidences $c_i^s, c_i^e$ are per-view percentile ranks of the predictor's confidence outputs, rather than calibrated probabilities. Eqs. (4)–(6) are evaluated only where all three depths are finite and positive, prediction validity holds, and source opacity exceeds $0.05$; evidence is zero elsewhere.

**Cross-view support and fitting weights.** For an edited point $\mathbf{X}_i^e(p)$, let $p_j$ and $z_j$ be its projection and camera-$z$ depth in view $j$. With nearest-pixel depth lookup,

$$s_{ij}(p) = \mathbb{I}\left[p_j \text{ is valid},\ z_j > 0,\ D_j^e(p_j) > 0,\ \frac{|z_j - D_j^e(p_j)|}{D_j^e(p_j)} \leq \tau_d\right], \qquad j \neq i. \tag{17}$$

The validity test includes finite depth and valid prediction support. Local candidates and their accepted/unsupported masks are

$$\begin{aligned} B_i &= \mathbb{I}[\max(e_i^-, e_i^+) \geq \tau_e], \\ M_i &= \mathrm{Close}\left(B_i \wedge \mathbb{I}\left[\sum_{j \neq i} s_{ij} \geq n_{\mathrm{sup}}\right]\right), \qquad U_i = B_i \wedge \neg M_i. \end{aligned} \tag{18}$$

Addition candidates are further localized by the shared spatial proposal. The hard attribution mask and continuous fitting weight are

$$H_i = 1 - \mathrm{Dilate}_2(U_i), \qquad W_i = 1 - \mathcal{K}_8 * \mathrm{Dilate}_2(U_i), \tag{19}$$

where dilation has radius two pixels and $\mathcal{K}_8$ is a normalized Gaussian kernel with standard deviation eight pixels. $H_i$ excludes unsupported regions from RGB-to-Gaussian attribution; $W_i$ softens their influence on fitting. Pixel support requires $n_{\mathrm{sup}} = 1$ remote view at $\tau_d = 0.06$, after thresholding local evidence at $\tau_e = 0.01$.

### A.3 SGA: Attribution, Surface Reuse, and Initialization

**From pixels to primitives.** For the compositing weights $a_{ig}(p)$ in Eq. (8), let $n_{ig}$ be the number of contributing pixels and $v_{ig}$ the renderer's visibility weight. The per-view removal score and removal set are

$$q_{ig}^- = \frac{\sum_p a_{ig}(p) M_i(p) e_i^-(p)}{\max(1, n_{ig})}, \qquad \mathcal{G}^- = \left\{g : \sum_i \mathbb{I}[q_{ig}^- \geq 0.01\ \wedge\ v_{ig} \geq 0.05] \geq 1\right\}. \tag{20}$$

This per-Gaussian one-view threshold follows pixel-level cross-view verification; it does not replace the remote support requirement in Eq. (18). RGB attribution uses the same compositing mechanism on $H_i d_i$ to select editable source primitives, including appearance changes with negligible depth displacement.

**Reuse before completion.** Render the source scene before and after removal, obtaining accumulated opacities $\rho_i^{\mathrm{before}}, \rho_i^{\mathrm{after}}$ and post-removal depth $D_i^{\mathrm{after}}$. On valid target-depth pixels, define

$$\begin{aligned} F_i &= \mathbb{I}[\rho_i^{\mathrm{before}} - \rho_i^{\mathrm{after}} \geq 0.01], \\ J_i &= \mathbb{I}\left[\rho_i^{\mathrm{after}} \geq 0.35,\ D_i^{\mathrm{after}} > 0,\ \frac{|D_i^{\mathrm{after}} - D_i^e|}{D_i^e} \leq 0.06\right]. \end{aligned} \tag{21}$$

$F_i$ is the deleted footprint; $J_i$ marks finite, valid revealed surfaces that already explain the target depth. Completion points are unprojected only from $F_i \wedge \neg J_i$, while addition points require $M_i = 1$ and $e_i^+ \geq 0.005$. Cross-view point fusion and local scale constraints initialize $\mathcal{G}^{\mathrm{fill}}$ and $\mathcal{G}^+$ in Eq. (9).

Points are fused with voxel size $0.15$ times the source point spacing. Eight-neighbor distances determine local scales, using radius multipliers $0.70$ for completion and $0.20$ for addition; scales are capped by the median source scale. New colors initialize the DC spherical-harmonic coefficients. Addition opacity starts at $0.25$–$0.35$, with count limits

$$|\mathcal{G}^{+}| \le 0.07|\mathcal{G}_s|, \qquad |\mathcal{G}^{+}| + |\mathcal{G}^{\mathrm{fill}}| \le 0.15|\mathcal{G}_s|. \tag{22}$$

### A.4 Selective Fitting and Expanded-View Refinement

For key-view fitting, source primitives are editable when their attributed RGB score is at least $0.03$, observation weight at least $0.05$, and hard-validity coverage at least $0.90$. The coverage threshold for newly initialized primitives is $0.30$, relaxed to $0.10$ when the configuration contains addition Gaussians. Editable retained primitives update appearance and opacity with center, scale, and rotation fixed; new primitives can update all attributes. Cloning, splitting, and pruning operate within the editable set.

The RGB loss is Eq. (10) with $\epsilon = 0.02$, combined with LPIPS (Zhang et al., 2018). After the first optimization, let $\mathcal{G}_{\mathrm{mid}}$ denote the intermediate scene. Expanded-view refinement uses

$$I_i^{s,(2)} = \mathcal{R}(\mathcal{G}_{\mathrm{mid}}, C_i) \longrightarrow I_i^{e,(2)} \longrightarrow \mathcal{G}_e, \qquad i \in \mathcal{V}_{\mathrm{train}}. \tag{23}$$

All training views are edited in batches of 12. Appearance tasks retain the original instruction; addition instructions preserve the existing inserted object and its placement, while removal instructions preserve its absence and refine exposed content. Both optimization stages use 1,600 iterations, with densification and pruning at iterations 400, 800, and 1,200, as specified in Section 4.1.

## B Experimental Protocol

### B.1 Scenes, Tasks, and Aggregation

The eight scenes and their dataset sources are listed in Section 4.1. Tab. 4 summarizes the reported evaluation subsets; all tasks use 12 key views. Scores are first averaged across evaluation views for each task and then averaged equally across tasks in the relevant subset. The addition/removal panel combines six tasks of each type. Tab. 3 uses only the 16 appearance-editing tasks.

Table 4: Task subsets used in the reported comparisons. The eight scenes are shared across the overall benchmark; per-subset scene counts are not specified.

| Task type | Tasks | Reported comparison |
|---|---|---|
| Appearance editing | 16 | Tab. 1(a), Tab. 3 |
| Object addition | 6 | Tab. 1(b), pooled with removal |
| Object removal | 6 | Tab. 1(b), pooled with addition |

### B.2 Editing Instructions

Tab. 5 lists the complete editing instructions used in our experiments. Each scene–instruction pair defines one editing task. We use 16 appearance-editing instructions, six object-addition instructions, and six object-removal instructions.

Table 5: **Complete editing instructions used in our experiments.** Only the editing instruction supplied to the image editor is reported.

| **Scene** | **Editing instruction** |
|---|---|
| *Appearance editing* | |
| Bear | Turn the bear statue into a wild boar. |

*Continued on the next page.*

*Table 5 continued from the previous page.*

| Scene | Editing instruction |
|---|---|
| Bear | Turn the bear statue into a metallic robot bear. |
| Bicycle | Turn the ground into a Namibian desert. |
| Bicycle | Make the entire scene look as if it is painted in watercolor style. |
| Bonsai | Make the bonsai snowy. |
| Bonsai | Change the bonsai to look like it is made of white paper, folded into intricate origami shapes. |
| Face | Make his face resemble that of a marble sculpture. |
| Face | Make him appear like he is made of paper with folded edges. |
| Fangzhou | Turn him into the Joker. |
| Fangzhou | Turn him into the Hulk. |
| Garden | Make it autumn. |
| Garden | Make the scene appear as though it is underwater. |
| Person | Make him wear a suit. |
| Person | Turn the man into a Minecraft character. |
| Stone horse | Turn the horse statue into a wooden carving. |
| Stone horse | Turn the stone horse into a jade carving. |
| *Object addition* | |
| Bear | Place exactly one upright orange traffic cone on top of the bear's head, centered between its ears with the base touching the head. Keep everything else unchanged. |
| Bicycle | Place exactly one upright red fire hydrant on the paved path to the left of the bicycle's handlebar-side front wheel, not on the grass. Keep everything else unchanged. |
| Bonsai | Add exactly one red rose at the center of the bonsai foliage, with its stem among the branches. Keep everything else unchanged. |
| Garden | Add exactly one small open book lying flat directly in front of the flower vase. Keep everything else unchanged. |
| Person | Add exactly one bright red baseball cap worn naturally on the man's head, with the brim facing forward. Keep everything else unchanged. |
| Stone horse | Place exactly one closed red metal toolbox with a black handle on the ground directly against the front face of the stone pedestal. Keep everything else unchanged. |
| *Object removal* | |
| Bear | Remove only the bear statue from the stone pedestal. Leave the complete empty stone pedestal, camera, lighting, and background unchanged, and reconstruct only the regions formerly occluded by the bear. |
| Bicycle | Remove only the white bicycle. Preserve the complete black bench, ground, camera, lighting, and park background, and reconstruct only the regions formerly occluded by the bicycle. |
| Bonsai | Remove only the bonsai tree and its pot from the purple table. Preserve the complete empty table, camera, lighting, and room background, and reconstruct only the formerly occluded area. |
| Garden | Remove only the vase and flowers from the round table. Preserve the complete empty tabletop, camera, lighting, and garden background, and reconstruct only the formerly occluded area. |
| Person | Remove only the man standing by the white wall. Preserve the complete wall, ground, camera, lighting, and background, and reconstruct only the regions formerly occluded by him. |
| Stone horse | Remove only the stone horse statue from the pedestal. Preserve the complete empty pedestal, camera, lighting, and museum background, and reconstruct only the regions formerly occluded by the horse. |

### B.3 Metrics and Candidate-Editor Transfer

CLIP-S (Radford et al., 2021) measures image–text alignment, while CLIP-D (Gal et al., 2022) evaluates the direction of change relative to the source. DINO (Caron et al., 2021) measures the reported cross-view feature similarity. Gemini 2.5 Pro (Google) jointly assesses instruction faithfulness, multi-view consistency, and visual quality from four uniformly selected evaluation views,

their source/edited image pairs, and the instruction, at temperature zero. MEt3R (Asim et al., 2025) measures multi-view inconsistency; lower values are better. These metrics characterize the reported edits and do not provide ground-truth labels for individual SEI decisions.

For candidate-editor transfer (Tab. 2), DGE (Chen et al., 2024a) and EditSplat (Lee et al., 2025) supply their original 2D candidates, after which SEI and SGA update the Gaussian scene. This comparison changes the downstream scene update while retaining the corresponding candidate edits. It is separate from replacing the complete pipeline with full CUESplat.

# C Ablation Definitions

The variants below correspond to Tab. 3; each starts from the full pipeline. Section 4.5 discusses their measured effects on appearance editing. Module names refer to their main-text interfaces, while the following definitions separate final SEI evidence from the depth feedback used during MEC.

## C.1 Module-Level Variants

**Without MEC.** We disable cross-view edit-increment exchange by independently editing the key views, while retaining the same within-view source-feature controls, recursive source proxy, prompts, and denoising schedule. Final SEI and SGA are then applied to the resulting candidates. This variant isolates cross-view coupling without replacing the editor with vanilla FLUX.

**Without final SEI evidence.** We retain MEC, including its stepwise edited-depth feedback, but do not consume the final SEI outputs during scene revision or fitting. Evidence-guided removal, completion, addition, validity weighting, and selective parameter permissions are disabled, and the source Gaussian scene is directly fitted to the MEC candidates. This variant evaluates whether coordinated 2D candidates alone are sufficient for scene updating.

**Without SGA.** Retain MEC and SEI, including $W_i$ in the RGB loss, but fit the original Gaussian scene directly. Disable evidence-guided removal/initialization and selective parameter permissions. Thus this variant tests the scene-update mechanism beyond using evidence to weight supervision.

## C.2 Internal Mechanisms

**Relative edit-increment exchange.** Replace source-relative aggregation by absolute target-value aggregation while retaining the same geometric matches and residual injection:

$$\widetilde{V}^{T,\mathrm{abs}}_{i,t}(p) = \mathrm{Attn}\big(Q^T_{i,t}(p), \{K^T_{j,t}(q)\}, \{V^T_{j,t}(q)\}\big), \quad q \in \mathcal{N}_{ij,t}(p),\ j \neq i. \tag{24}$$

Use this consensus in Eq. (2) without adding the local source value.

**Dynamic correspondence refresh and recursive source proxy.** The first variant keeps $\mathcal{N}_{ij,t} = \mathcal{N}_{ij,0}$ from source-GS depth at every iteration. The second disables Eq. (13), keeping $x^S_{i,t} = x^S_{i,0}$. These are separate changes: depth feedback controls geometric matches, whereas the RGB-driven proxy changes the source features used by attention.

**Source-GS reference, cross-view support, and geometric confidence.** The source-GS-reference variant sets $r_i = r_i^{\mathrm{pair}}$ in place of Eq. (5); source-depth anchors remain in geometry prediction. The cross-view-support variant drops remote voting, giving $M_i = \mathrm{Close}(B_i)$ and $U_i = B_i \wedge \neg M_i$. The confidence variant sets $c^s_i = c^e_i = 1$ in Eq. (6). All other evidence gates remain active in each variant.

**Contribution-based attribution.** Replace compositing-weighted lifting by sampling the same pixel evidence at each Gaussian's projected center, for both removal and RGB-based editable-source selection. This tests the attribution rule in Eq. (8); it does not remove SEI's pixel evidence.

Table 6: **Analysis of key hyperparameters.** We evaluate the number of key views $N$, the minimum remote-view support, and the relative reprojection-depth tolerance $\tau_d$. The symbol † denotes the default setting. The best result within each group is shown in bold. Higher is better for CLIP-S, CLIP-D, DINO, and Gemini, while lower is better for MEt3R.

| Factor | Setting | CLIP-S ↑ | CLIP-D ↑ | DINO ↑ | Gemini ↑ | MEt3R ↓ |
|---|---|---|---|---|---|---|
| Key views $N$ | 8 | 0.2568 | 0.1679 | 0.7708 | 2.75 | 0.2484 |
| | 12† | 0.2608 | 0.1765 | 0.7745 | 3.13 | 0.2448 |
| | 16 | **0.2632** | **0.1809** | **0.7747** | 3.19 | 0.2437 |
| | 20 | 0.2623 | 0.1807 | 0.7741 | **3.50** | **0.2428** |
| Remote support $n_{\text{sup}}$ | 0 | 0.2503 | 0.1681 | 0.7716 | 2.81 | 0.2553 |
| | 1† | **0.2608** | **0.1765** | **0.7745** | **3.13** | **0.2448** |
| | 2 | 0.2600 | 0.1754 | 0.7704 | 3.06 | 0.2482 |
| Tolerance $\tau_d$ | 0.03 | 0.2588 | 0.1734 | 0.7706 | 3.06 | 0.2633 |
| | 0.06† | **0.2608** | **0.1765** | **0.7745** | **3.13** | **0.2448** |
| | 0.09 | 0.2601 | 0.1721 | 0.7725 | 3.00 | 0.2459 |
| | 0.12 | 0.2592 | 0.1728 | 0.7734 | 2.81 | 0.2460 |

**Source-surface reuse and evidence-weighted fitting.** The reuse variant disables $J_i$ in Eq. (21), initializing completion candidates throughout the deleted footprint wherever target depth is valid. The fitting variant sets $W_i = 1$, hence $w_i = 1$ in Eq. (10), while retaining the structural channels and hard attribution mask $H_i$. It changes soft image supervision without disabling evidence-based primitive selection.

### C.3 Expanded-View Refinement

This variant returns $\mathcal{G}_{\text{mid}}$, omitting all-training-view editing and the second optimization in Eq. (23). It changes both camera coverage and optimization budget, so their individual effects are not isolated.

### C.4 Key Hyperparameter Analysis

We analyze three key hyperparameters controlling view coverage and cross-view evidence verification: the number of key views $N$, the minimum number of supporting remote views, and the relative reprojection-depth tolerance $\tau_d$. We vary one parameter at a time while keeping all other settings fixed. Results are averaged over the 16 appearance-editing tasks.

**Number of Key Views** Increasing $N$ from 8 to 12 improves all metrics, demonstrating the importance of sufficient camera-space coverage for edit coupling and structural inference. Increasing $N$ further yields only modest improvements: $N = 16$ achieves the highest CLIP-S, CLIP-D, and DINO scores, while $N = 20$ performs best on Gemini and MEt3R. The gains largely saturate beyond 12 views despite the additional image-editing and geometry-prediction cost. We therefore use $N = 12$ as the default quality–efficiency trade-off.

**Minimum Remote-view Support** The minimum remote-view support $n_{\text{sup}}$ specifies how many additional views must provide depth-consistent support for a candidate pixel before it enters the supported-change mask $M_i$. Setting $n_{\text{sup}} = 0$ allows single-view hypotheses to directly control the shared scene, reducing CLIP-S from 0.2608 to 0.2503 and increasing MEt3R from 0.2448 to 0.2553. Requiring one additional view ($n_{\text{sup}} = 1$) achieves the best overall performance. Increasing the requirement to $n_{\text{sup}} = 2$ slightly degrades the results, since valid changes with limited visibility may no longer receive sufficient support. We therefore use $n_{\text{sup}} = 1$ by default.

**Reprojection-depth Tolerance** The tolerance $\tau_d$ controls the relative depth agreement required between a reprojected 3D candidate and the predicted surface in a remote view. A smaller value imposes stricter geometric verification, whereas a larger value accepts more candidate correspondences. The strict setting $\tau_d = 0.03$ rejects useful support and increases MEt3R to 0.2633. The default value $\tau_d = 0.06$ provides the best overall balance. Looser settings of 0.09 and 0.12 produce

relatively similar CLIP-S, DINO, and MEt3R scores, although CLIP-D and Gemini gradually decrease as less reliable correspondences are admitted. These results indicate that CUESplat operates stably over a moderate tolerance range rather than relying on a narrowly tuned threshold.

## D Additional Qualitative Results

Fig. 5 presents additional qualitative comparisons on representative object addition and removal tasks. The first two rows show object addition, where a new object must be inserted at the instructed location while preserving the surrounding scene. The last two rows show object removal, where the specified foreground object must be removed and the newly exposed regions must be reconstructed consistently across viewpoints. We compare CUESplat with GaussianEditor (Chen et al., 2024b) and Omni-3DEdit (Liyi et al., 2026). Each comparison includes an additional viewpoint to inspect the requested change and the surrounding scene content.

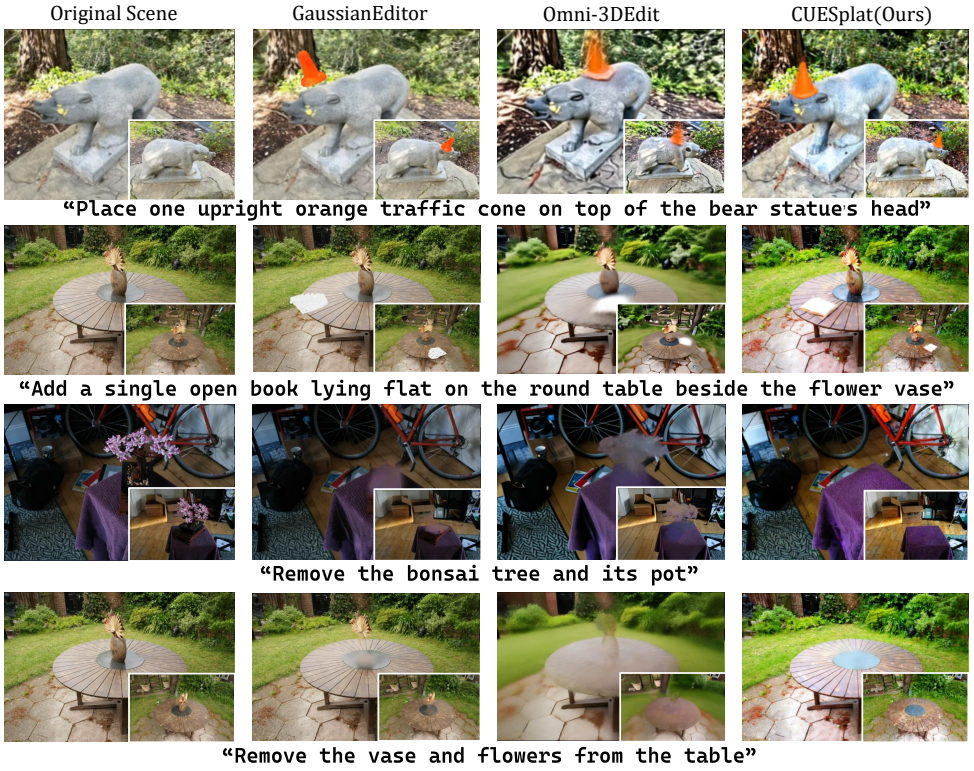


Figure 5: **Qualitative comparison on object addition and removal.** We compare CUESplat with GaussianEditor and Omni-3DEdit on representative object addition and removal tasks. Insets show an additional viewpoint of each edited scene.

Fig. 6 presents additional multi-view renderings produced by CUESplat across appearance modification, object addition, and object removal. For each task, we show five camera viewpoints with substantial viewpoint changes. The results demonstrate that CUESplat realizes diverse editing instructions while maintaining the identity, spatial location, and surrounding scene content of the edit across views.

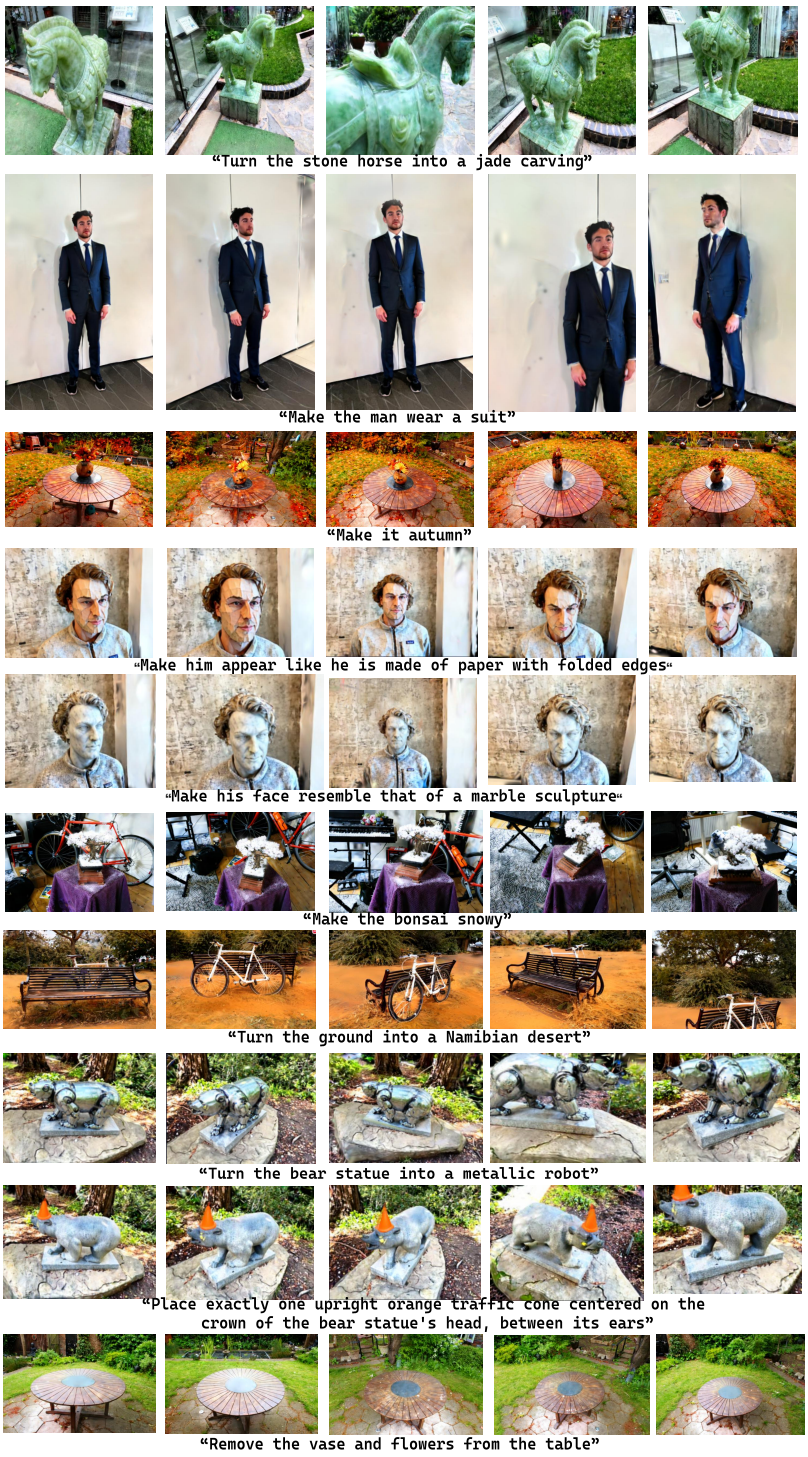


Figure 6: **Multi-view qualitative results of CUESplat.** We show diverse text-guided edits, including material and appearance changes, scene modifications, and object addition and removal. Each row presents five viewpoints of an edited scene, with the corresponding instruction shown below.

## E LIMITATIONS AND FAILURE CASES

CUESplat assumes a pretrained source 3DGS with calibrated cameras, and its performance can therefore be affected by inaccuracies in the source reconstruction and feed-forward geometry predictions. As shown in Fig. 7, these limitations are most noticeable for edits involving small objects or heavily occluded regions. In the bicycle-addition example, the inserted fire hydrant exhibits variations in scale and position across viewpoints. Small added objects may receive insufficient geometric support when they are visible in only a limited number of cameras. In the person-removal example, residual color artifacts remain on the newly exposed wall, indicating that background completion remains difficult when a large foreground object occludes most of the target surface. Finally, in the bonsai-addition example, the requested single rose affects a broader region of the foliage. This illustrates that geometrically consistent evidence cannot by itself prevent a candidate editor from producing an overly broad semantic change. The multi-stage pipeline also introduces additional computation from joint key-view editing, repeated geometry prediction, and two-stage Gaussian optimization. Adaptive view selection, improved completion of occluded surfaces, and more localized semantic control provide promising directions for future work.

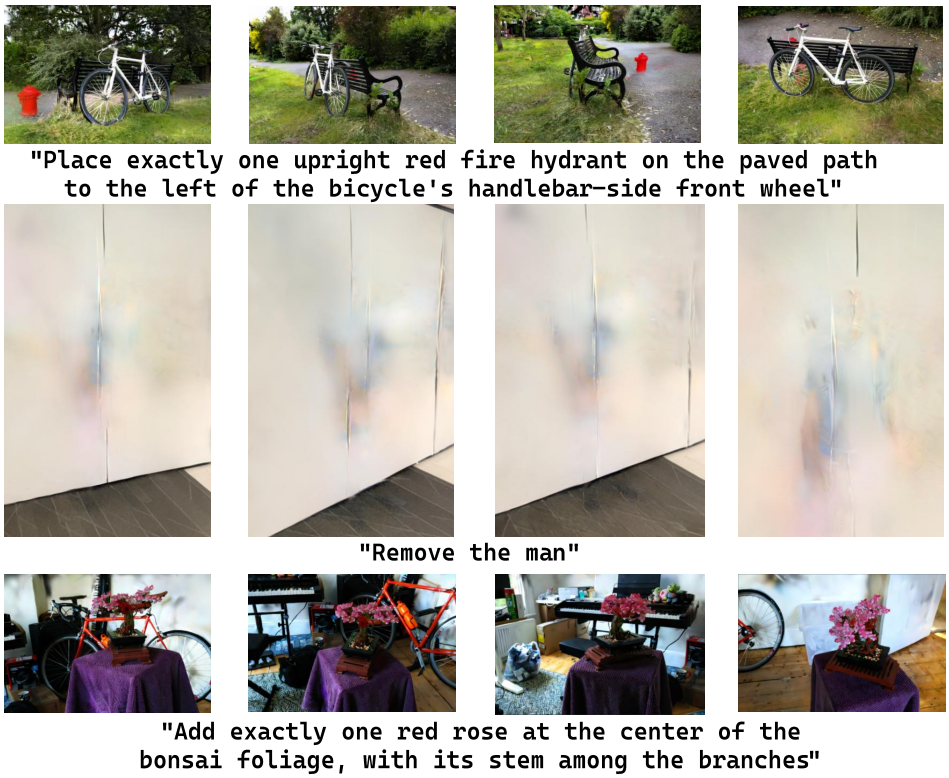


Figure 7: **Representative failure cases of CUESplat.** Each row shows four viewpoints of an edited scene. Top: inconsistent placement and scale of an added fire hydrant. Middle: residual color artifacts on the wall after person removal. Bottom: adding a single rose produces overly broad changes to the bonsai foliage.