INSTRUCTMIX2MIX: CONSISTENT SPARSE-VIEW EDITING THROUGH MULTI-VIEW MODEL PERSONALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We address the task of multi-view image editing from sparse input views, where the inputs can be seen as a *mix of images* capturing the scene from different viewpoints. The goal is to modify the scene according to a textual instruction while preserving consistency across all views. Existing methods, based on per-scene neural fields or temporal attention mechanisms, struggle in this setting, often producing artifacts and incoherent edits. We propose InstructMix2Mix (I-Mix2Mix), a framework that distills the editing capabilities of a 2D diffusion model into a pretrained multi-view diffusion model, leveraging its data-driven 3D prior for cross-view consistency. A key contribution is replacing the conventional neural field consolidator in Score Distillation Sampling (SDS) with a multi-view diffusion student, which requires novel adaptations: incremental student updates across timesteps, a specialized teacher noise scheduler to prevent degeneration, and an attention modification that enhances cross-view coherence without additional cost. Experiments demonstrate that I-Mix2Mix significantly improves multi-view consistency while maintaining high per-frame edit quality.

1 INTRODUCTION

Multi-view image editing seeks to modify a scene captured from multiple viewpoints while preserving consistency across views. Typical edits include texture and color changes, semantic manipulations, or geometric transformations, with applications in product imagery, real-estate and interior design visualization, AR/VR, and cinematic post-production.

However, multi-view editing is a difficult task that traditionally requires skilled artists, making automation highly desirable. In recent years, a variety of methods have been proposed for editing 3D scenes. Due to the difficulty of producing supervision in the form of high-quality paired scenes before and after editing, most of these approaches avoid direct multi-view editing, instead leveraging monocular editors such as InstructPix2Pix Brooks et al. (2023), either iteratively (Haque et al., 2023; Vachha & Haque, 2024) or through distillation (Li et al., 2024; Kamata et al., 2023). These approaches achieve 3D-consistent edits by operating on dense scene representations like NeRFs (Mildenhall et al., 2021) or 3D Gaussian Splats (Kerbl et al., 2023). However, in practice users often have only a sparse set of images, which provide limited scene coverage and cause existing methods to produce artifacts and inconsistent edits.

A complementary line of work, adapted from video editing, modifies the self-attention mechanism of 2D editors to encourage cross-frame coherence. While effective for semantic consistency, it struggles to preserve fine details under large viewpoint changes. Together, these challenges highlight the need for methods that enable robust, high-quality multi-view editing from sparse inputs.

In this work, we tackle the challenging problem of multi-view editing from sparse input images (a *mix* of images). Given a few source views and a textual editing instruction, our aim is to generate edits that faithfully follow the prompt while remaining consistent across all viewpoints. Following prior work, we leverage powerful 2D editors and lift their capabilities to 3D using Score Distillation Sampling (SDS) (Poole et al., 2022). However, we propose a new approach to this paradigm. We observe that current approaches face inherent limitations. Neural field representations are trained per scene – they do not hold 3D prior in their network weights. Instead, they achieve 3D consistency

by incorporating a physical prior through the rendering equations. A dense set of overlapping input images is required, however, to transform this prior into an effective consolidator. To achieve consistency with *sparse* views, we instead propose to incorporate a consolidator that embeds a strong, data-driven 3D prior directly in its weights: a multi-view synthesis diffusion model. While such models are trained to generate view-consistent scenes (e.g., from text or images), they lack editing capabilities. We bridge this gap by combining the strengths of both paradigms—distilling edits from a 2D editor (the teacher) into the multi-view model (the student). Concretely, we use InstructPix2Pix as the teacher and Stable Virtual Camera (SEVA) (Zhou et al., 2025) as the student. By leveraging a student model with an inherent 3D prior, our method — **I-Mix2Mix** — produces robust, geometrically coherent, and visually consistent edits even from extremely sparse inputs.

Replacing the neural field with a multi-view diffusion model within the SDS framework is not straightforward, and requires careful adaptation of several key steps. Instead of rendering from a scene representation, we sample from the diffusion model; to avoid costly full trajectories, we distill incrementally across student timesteps. We also introduce a specialized teacher noise scheduler to prevent collapse to poor local minima and an attention modification that strengthens multi-view consistency without extra cost. Together, these components yield a framework for consistent multi-view editing, producing high-quality results even with very sparse inputs.

To summarize, our contributions are:

1. We present I-Mix2Mix, a novel framework for distilling the knowledge of a powerful monocular editor into a pretrained multi-view diffusion model, leveraging its data-driven 3D consistency.
2. Through careful consideration of the SDS key steps, we introduce novel adaptations to support personalization of our multi-view student.
3. We demonstrate that this approach produces high-quality, consistent multi-view edits, effectively extending the SDS framework to scenarios with limited viewpoints.

We evaluate I-Mix2Mix against popular multi-view editing methods, demonstrating significant improvements in cross-view consistency both qualitatively and quantitatively in the sparse-view setting. At the same time, our method maintains competitive per-frame editing performance, highlighting the practical benefits of leveraging a data-driven multi-view prior within the SDS framework.

2 Related Works

Neural field editing. Editing 3D scenes or objects typically assumes a pre-optimized model such as a NeRF (Mildenhall et al., 2021) or 3D Gaussian Splatting (Kerbl et al., 2023). Early works explored *direct NeRF manipulation* via scribbles (Liu et al., 2021), sketches (Mikaeili et al., 2023), reference images (Bao et al., 2023), meshes (Yuan et al., 2022), point clouds (Chen et al., 2023a), and other cues (Weder et al., 2023; Mirzaei et al., 2023; Yang et al., 2021), while *NeRF stylization* transfers reference appearances to 3D scenes (Wang et al., 2022; 2023a; Nguyen-Phuoc et al., 2022; Huang et al., 2022; Chiang et al., 2022). *Instruction-based* approaches leverage 2D diffusion editors like InstructPix2Pix (Brooks et al., 2023), applying SDS-like guidance (Li et al., 2024; Sella et al., 2023; Zhuang et al., 2023; Kamata et al., 2023) or Iterative Dataset Update (Haque et al., 2023; Wang et al., 2024a; Vachha & Haque, 2024; Wang et al., 2024b; Chen et al., 2024c;a) for improved consistency. Distinctively, SHAP-Editor (Chen et al., 2023b) operates in latent space for feed-forward edits. While effective for 3D editing, the reliance of these approaches on dense input views makes them less suitable in sparse-view scenarios.

Sparse multi-view editing. In the absence of a full 3D representation, several works have explored editing a set of input images directly. A prominent direction adapts pre-trained diffusion-based monocular editors by modifying self-attention layers: as first shown in Wu et al. (2023), extending queries to attend across frames promotes consistency between the outputs. Building on this idea, a number of methods generate *edited videos* (Geyer et al., 2023; Shin et al., 2024; Khachatryan et al., 2023; Ceylan et al., 2023; Qi et al., 2023; Liu et al., 2024). While effective for temporally smooth sequences with small viewpoint changes, these approaches struggle in the sparse-view setting, where edits must remain consistent under significant viewpoint differences. DGE (Chen et al., 2024b) combines extended attention with 3D Gaussian Splatting (3DGS) lifting: attention-based edits provide rough multi-view consistency, while 3DGS is used to consolidate outputs and resolve residual

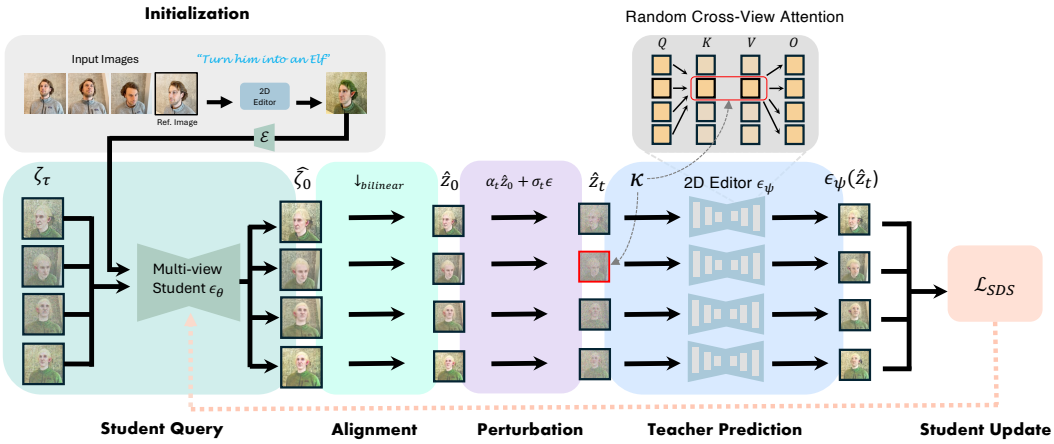


Figure 1: **I-Mix2Mix overview**. Given a set of input images, a randomly chosen reference image is edited by the frozen teacher and encoded to serve as the personalized multi-view student's input latent (*Initialization*). At each distillation iteration, noisy multi-view latents ζ_τ are denoised by the student (*Student Query*), aligned to the teacher's latent space (*Alignment*), and perturbed with our forward schedule (*Perturbation*). The teacher predicts edits with Random Cross-View Attention (*Teacher Prediction*), where all frames attend to the κ's frame, and the resulting supervision is distilled back into the student (*Student Update*). After distillation, the student outputs a set of multi-view consistent edited frames.

artifacts. However, in the sparse-view regime, 3DGS tends to overfit the limited input views rather than serve as a true cross-view aggregator, leading to persistent inconsistencies. As a result, DGE effectively reduces to an extended-attention approach, inheriting the same limitations as prior video editing methods. Most recently, Bar-On et al. (2025) proposed a feed-forward approach that propagates a user-provided 2D edit to multiple views, but their method remains limited to object-level edits. Contemporaneously with our work, Zhao et al. (2025) fine-tune FLUX Kontext (Labs et al., 2025) to enable consistent edits across image pairs, while Chi et al. (2025) distill 3D consistency priors into a 2D editor through a VSD-based framework (Wang et al., 2023b).

3 Preliminaries

Stable Virtual Camera. SEVA (Zhou et al., 2025) is a diffusion-based Novel View Synthesis (NVS) model that predicts N target images given M input images with their camera poses. Built on Stable Diffusion 2.1 Rombach et al. (2022) with architectural adaptations for NVS and trained on diverse object and scene datasets, it achieves state-of-the-art results, making it an ideal student model with a strong 3D prior.

Instruct-Pix2Pix. A monocular, instruction-based image editing diffusion model widely used in 3D and multi-view editing, Instruct-Pix2Pix (Brooks et al., 2023) is fine-tuned from a pre-trained Stable Diffusion model on a large-scale synthetic editing dataset. Given a source image and a textual instruction, it produces versatile edits by sampling the fine-tuned model. The model employs classifier-free guidance (CFG) (Ho & Salimans, 2022) with two scales: a *text CFG scale* s_T controlling adherence to the instruction, and an *image CFG scale* s_I controlling fidelity to the source image, jointly balancing edit strength and overall image quality.

4 Method

Our goal is sparse multi-view consistent image editing. We build on the SDS framework (Poole et al., 2022), using a pre-trained image editing network as a *teacher* to distill knowledge into a neural scene representation *student*. Unlike typical settings that assume abundant input views, we work with only a few images. To address this challenge, we replace the conventional neural field with a multi-view diffusion model pre-trained for consistent view generation. We personalize this

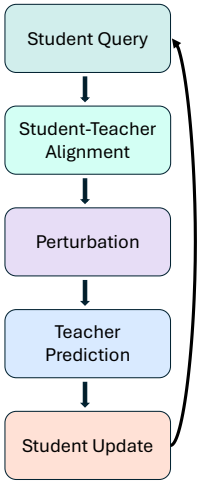
student to the target scene and edit instruction by distilling the teacher's predictions, enabling faithful and consistent edits from limited inputs.

4.1 Problem Formulation

We are given N images $\{I_i\}_{i=1}^N$, $I_i \in \mathbb{R}^{3\times H\times W}$ of a static 3D scene with camera poses $\{\pi_i\}_{i=1}^N$, $\pi_i \in \mathbb{R}^{4\times 4}$, and an editing prompt $y \in \mathcal{Y}$. We assume access to a multi-view diffusion model ϵ_θ which we refer to as the *student*, and a monocular instruction-based editing diffusion model (*teacher*) ϵ_ψ. The goal is to produce edited views $\{E_i\}_{i=1}^N$ such that (i) each E_i is a faithful edit of I_i according to y, and (ii) $\{E_i\}$ are multi-view consistent, i.e. there exists a underlying 3D scene representation $\mathcal{S}$ whose renderings under poses $\{\pi_i\}$ yield $\{E_i\}$.

4.2 Score Distillation Sampling

Originally introduced in *DreamFusion* (Poole et al., 2022) for 3D generation using 2D diffusion models, **Score Distillation Sampling (SDS)** is an iterative technique for utilizing the generative prior of a pre-trained diffusion model ϵ_ψ (*teacher*) to tune the parameters θ of a differentiable neural scene representation Φ_θ (*student*). At each iteration, the student is queried (rendered) through a differentiable operator g, yielding $\hat{\chi}_0 = g(\Phi_\theta)$. This prediction is then critiqued by the teacher, and the process repeats iteratively, updating θ until the student encodes a scene representation Φ_θ that yields plausible renderings. The overall SDS framework, can be summarized as a five-stage pipeline, schematically shown in the inset figure:



1. Student Query. The student Φ_θ produces an image or latent $\hat{\chi}_0 = g(\Phi_\theta)$ to be critiqued. Commonly this is a differentiable rendering from a NeRF.

2. Student-Teacher Alignment. The student output $\hat{\chi}_0$ is mapped to the teacher's input space as $\hat{x}_0$, e.g., through an image encoder.

3. Perturbation. The aligned prediction $\hat{x}_0$ is perturbed according to the teacher's forward diffusion process: $\hat{x}_t = \alpha_t \hat{x}_0 + \sigma_t \epsilon, \quad \epsilon \sim \mathcal{N}(0, \boldsymbol{I})$.

4. Teacher Prediction. The teacher model ϵ_ψ processes $\hat{x}_t$ (conditioned on an embedding y and the timestep t), and predicts the corresponding noise $\epsilon_\psi(\hat{x}_t; y, t)$.

5. Student Update. The residual between the sampled noise ϵ and the teacher's prediction $\epsilon_\psi(\hat{x}_t; y, t)$ defines the SDS gradient: $\nabla_\theta \mathcal{L}_{\text{SDS}} = \mathbb{E}_{t,\epsilon}\left[w(t)\left(\epsilon_\psi(\hat{x}_t; y, t) - \epsilon\right)\frac{\partial \hat{x}_0}{\partial \theta}\right]$, where $w(t)$ is a time-dependent weighting term. The gradient is backpropagated to update the student parameters θ.

SDS variants differ primarily in the specific design choice at each stage.

4.3 Consistent Sparse-View Editing Through Student Personalization

In our framework, where the student is a diffusion-based multi view synthesis model, key SDS stages require specialized adaptations, which we detail in this subsection. Our proposed approach is illustrated in Figure 1.

Step 1: Student Query. In traditional SDS with NeRFs or 3DGS, the student prediction $\hat{\chi}_0 = g(\Phi_\theta)$ is obtained via differentiable rendering. In our setting, the student is a multi-view diffusion model ϵ_θ, so the analogue is generating a sample via its denoising trajectory (Ho et al., 2020; Song et al., 2020). Running a full sampling trajectory at each SDS iteration is however slow and computationally expensive, requiring backpropagation through many denoiser evaluations. Instead, we distill incrementally at each student timestep τ, starting from $\tau = \mathrm{T}$ with latents sampled from the Gaussian distribution with student-scheduler specified variance $\{\zeta_{\mathrm{T}}^i\}_{i=1}^N \sim \mathcal{N}(0, \sigma_{\mathrm{S}}^2 \boldsymbol{I})$. We compute *single-step predictions* of the clean latents via the Tweedie formula (Efron, 2011). These estimates $\{\hat{\zeta}_0^i(\tau)\}$ serve as intermediate student predictions to be critiqued by the teacher, shaping the student's backward trajectory step by step.

Step 2: Student-Teacher Alignment. Although both student and teacher are latent diffusion models, they operate in different latent spaces and dimensions. A naive approach would decode the student's predictions $\{\hat{\zeta}_0^i\}$ with the its decoder $\mathcal{D}_\text{S}$ and encode them with the teacher encoder $\mathcal{E}_\text{T}$ before adding noise via the teacher's forward process. However, backpropagating through both $\mathcal{D}_\text{S}$ and $\mathcal{E}_\text{T}$ would be prohibitively expensive. Inspired by prior work on convergent representations (Asperti & Tonelli, 2023; Lenc & Vedaldi, 2015; Huh et al., 2024; Li et al., 2015), which suggests that simple mappings can often bridge the representation spaces of different networks, we instead resize the student's latents to the teacher's expected dimensions (H_T, W_T) via bilinear interpolation: $\hat{z}_0^i = \mathcal{I}_{bilinear}(\hat{\zeta}_0^i; H_\text{T}, W_\text{T})$. For our chosen models, this lightweight approach suffices, suggesting that the student latents implicitly align with the teacher's latent space during fine-tuning.

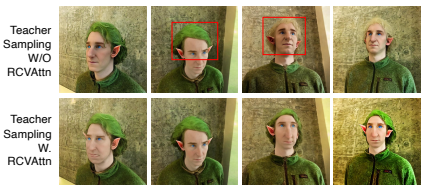
Step 3: Perturbation. The mapped latents are perturbed using the teacher's forward process $\hat{z}_t^i = \alpha_t \hat{z}_0^i + \sigma_t \epsilon_i,\ \epsilon_i \sim \mathcal{N}(0, \boldsymbol{I})$, yielding noisy latents $\{\hat{z}_t^i\}$. A key design choice is the teacher timestep t. In standard SDS (Poole et al., 2022), t is drawn uniformly from $[0.02, 0.98]$, avoiding extreme noise levels for numerical stability. This is ill-suited to our setting: early student outputs (large τ) lie off the natural image manifold, so at low t values their diffused versions fall outside the teacher's distribution, causing unstable guidance.

Annealed t schedules have also been explored (Huang et al., 2023; Lukoianov et al., 2024), and a natural variant is to match t with the student timestep τ. Yet this is too restrictive—when τ is small, forcing $t \approx \tau$ limits the teacher's ability to provide corrective gradients. We instead use a stochastic schedule: $t \sim \text{TruncNorm}\Big(\mu = b,\ \sigma = \frac{b-\tau}{f},\ a = \tau,\ b = 0.95\Big)$, where f controls skewness. Larger f concentrates probability near b, making it more likely for the teacher to operate at higher noise levels. The randomness ensures that the teacher provides strong gradients every few iterations, which we find highly effective for avoiding collapse to poor local minima. See Appendix A.1 for further details and visualizations.

Step 4: Teacher Prediction. A straightforward application of our framework would pass the perturbed latents $\{\hat{z}_t^i\}_{i=1}^N$ as a batch to the monocular teacher U-Net ϵ_ψ, which would then produce independent noise estimates for each latent. Backpropagating such conflicting signals into the student can weaken its multi-view prior, yielding inconsistent final edits. To address this, we introduce a lightweight Random Cross-View Attention (RCVAttn) mechanism that encourages the teacher to generate more consistent edits within each batch. Inspired by attention-based alignment work (Khachatryan et al., 2023), we randomly select a *key frame* index $\kappa \sim U\{1, \ldots, N\}$ at each iteration. Each frame i attends to the tokens of the key frame:

$$\text{RCVAttn}(Q, K, V, i) = \text{softmax}\left(\frac{Q_i K_\kappa^\top}{\sqrt{d}}\right) V_\kappa, \tag{1}$$

where d is the query/key dimensionality. Aligning all frames to query the key frame improves consistency substantially, aiding in retaining the student's multi-view prior. Unlike expensive extended-attention methods (Wu et al., 2023; Chen et al., 2024b; Geyer et al., 2023), RCVAttn adds no computational overhead. While non-key frames may experience reduced quality, randomly selecting κ ensures all frames occasionally serve as the key, preventing noticeable degradation. The effect of RCVAttn, when applied to full teacher sampling process, is shown in the inset figure.



Step 5: Student Update. Finally, the difference between the sampled noise and the teacher's prediction defines the guidance direction in the SDS objective:

$$\nabla_\theta \mathcal{L}_\text{SDS} = \frac{1}{N} \sum_{i=1}^{N} \big(\epsilon_\psi(\hat{z}_t^i; y, I_i, t) - \epsilon_i\big) \frac{\partial \hat{z}_0^i}{\partial \theta}, \tag{2}$$

which is backpropagated to update the student weights.

This completes a single distillation iteration at student timestep τ. We start at $\tau = \text{T}$ and repeat this process for k iterations, personalizing the student model at this timestep to the indented edit. The student then performs a sampling step with its scheduler, producing latents $\{\zeta_{\tau-\Delta\tau}^i\}$, where $\Delta\tau$ is the step size. Distillation resumes at $\tau - \Delta\tau$, repeating k updates before the next sampling step. This nested procedure continues until $\tau = 0$, yielding final edited views that are instruction-faithful and multi-view consistent.

Initialization. Our student model, SEVA, is an "M in, N out" model with $M \geq 1$, meaning that the denoiser ϵ_θ requires at least one clean input latent in addition to the N noisy latents. As a preprocessing step, we randomly select one of the input frames $I_{\text{ref}} \in \{I_i\}$ and pass it through the 2D teacher editor with prompt y to generate a valid reference edit E_{ref}. This edit is then encoded using SEVA's frozen encoder to obtain a reference latent $z_{\text{ref}} = \mathcal{E}_{\text{S}}(E_{\text{ref}})$, which serves as the input frame to the denoiser in all distillation iterations. This can be considered "Step 0" of the framework. The full framework is summarized in the inset algorithm.

Algorithm 1: I-Mix2Mix

Input:
$\{I_i\}_{i=1}^N, \{\pi_i\}_{i=1}^N, y$ ▷ Images, poses and text prompt
$\epsilon_\psi, \epsilon_\theta$ ▷ Frozen teacher and trainable student
$z_{\text{ref}} \leftarrow \mathcal{E}_S(\text{TeacherEdit}(I_{\text{ref}}, y))$
Initialize $\zeta_T^i \sim \mathcal{N}(0, \sigma_S^2 I)$
for $\tau = T, T-\Delta\tau, \ldots, 0$ **do**
  for k steps **do**
    $\{\hat{\zeta}_0^i\} \leftarrow \epsilon_\theta(\{\zeta_\tau^i\}, z_{\text{ref}}, \{\pi_i\})$
    $\hat{z}_0^i \leftarrow \mathcal{I}_{\text{bilinear}}(\hat{\zeta}_0^i)$
    $t \sim \text{TruncNorm}(\mu{=}b, \sigma{=}\frac{b-\tau}{f}, a{=}\tau, b{=}0.95)$
    $\epsilon_i \sim \mathcal{N}(0, I)$
    $\hat{z}_t^i \leftarrow \alpha_t \hat{z}_0^i + \sigma_t \epsilon_i$
    $\kappa \sim U\{1, \ldots, N\}$ ▷ Select keyframe
    $\{\tilde{\epsilon}_i\} \leftarrow \epsilon_\psi(\{\hat{z}_t^i\}; y, \{I_i\}, t)$ ▷ With RCVAttn
    $\nabla_\theta \mathcal{L}_{\text{SDS}} \leftarrow \frac{1}{N}\sum_i (\tilde{\epsilon}_i - \epsilon_i)\frac{\partial \hat{z}_0^i}{\partial \theta}$
    $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{SDS}}$
  end for
  $\{\zeta_{\tau-\Delta\tau}^i\} \leftarrow \text{StudentStep}(\{\zeta_\tau^i\}, z_{\text{ref}}, \{\pi_i\}, \Delta\tau)$
end for
Output: $E_i \leftarrow \mathcal{D}_T(\hat{z}_0^i)$

5 Experiments

Methods in comparison. We compare with four widely used, open-source methods that also employ InstructPix2Pix as the 2D editor, covering distinct paradigms for multi-view editing: *Instruct-NeRF2NeRF (I-N2N)* (Haque et al., 2023) and its 3DGS variant *Instruct-GS2GS (I-GS2GS)* (Vachha & Haque, 2024) (both following the Iterative Dataset Update paradigm), *Text2Video-Zero (T2VZ)* (Khachatryan et al., 2023) (a zero-shot image-to-video adaptation), and *DGE* (Chen et al., 2024b) (extended attention for multi-view editing with 3DGS-based consolidation). Since I-N2N requires a trained NeRF, we optimize a Nerfacto (Tancik et al., 2023) model on the N input views; similarly, because I-GS2GS and DGE require a 3DGS, we optimize a Splatfacto (Tancik et al., 2023) model. All baselines are run with default settings from the official implementations or papers.

Evaluation. We evaluate our method on scenes from several datasets: I-N2N (Haque et al., 2023), Tanks and Temples (Knapitsch et al., 2017), CO3D (Reizenstein et al., 2021), and Mip-NeRF 360 (Barron et al., 2022). Following prior protocols, for comparison with baselines we apply 20 edits to three standard test scenes from I-N2N (full edit set detailed in Appendix B); qualitative results on additional scenes appear in Appendix F. Per-frame edit quality and cross-view consistency are assessed with three CLIP-based (Radford et al., 2021) metrics commonly used in prior work (Haque et al., 2023; Chi et al., 2025; Chen et al., 2024b): (i) *CLIP Similarity*, the cosine similarity between an edited image and the prompt; (ii) *CLIP Directional Similarity* (Gal et al., 2022; Brooks et al., 2023), which measures alignment between prompt change and image change; (iii) *CLIP Directional Consistency* (Haque et al., 2023), which quantifies multi-view consistency by comparing the relative changes between pairs of original views O_i, O_j and their corresponding edited views E_i, E_j via $cos\_sim\big(\phi(O_i) - \phi(O_j),\ \phi(E_i) - \phi(E_j)\big)$, where $\phi(\cdot)$ denotes the CLIP embedding. This metric captures whether the semantic difference between two views is preserved after editing. Unlike the original formulation, which considers only consecutive frames, we average over all $\binom{N}{2}$ pairs to account for our unordered, sparse-view setting.

We use $N = 4$ frames in main experiments, with additional results for larger N in Appendix G. Full implementation details, are detailed in Appendix A.

5.1 Comparison with Prior Work

Quantitative results are reported in Table 1, and representative qualitative comparisons are shown in Figure 2. Enlarged visualizations and additional examples are included in Appendix E.

Our method achieves the highest performance in *CLIP Directional Consistency (CLIP Cons.)*, indicating that edits remain more consistent across different views. Importantly, this does not come at

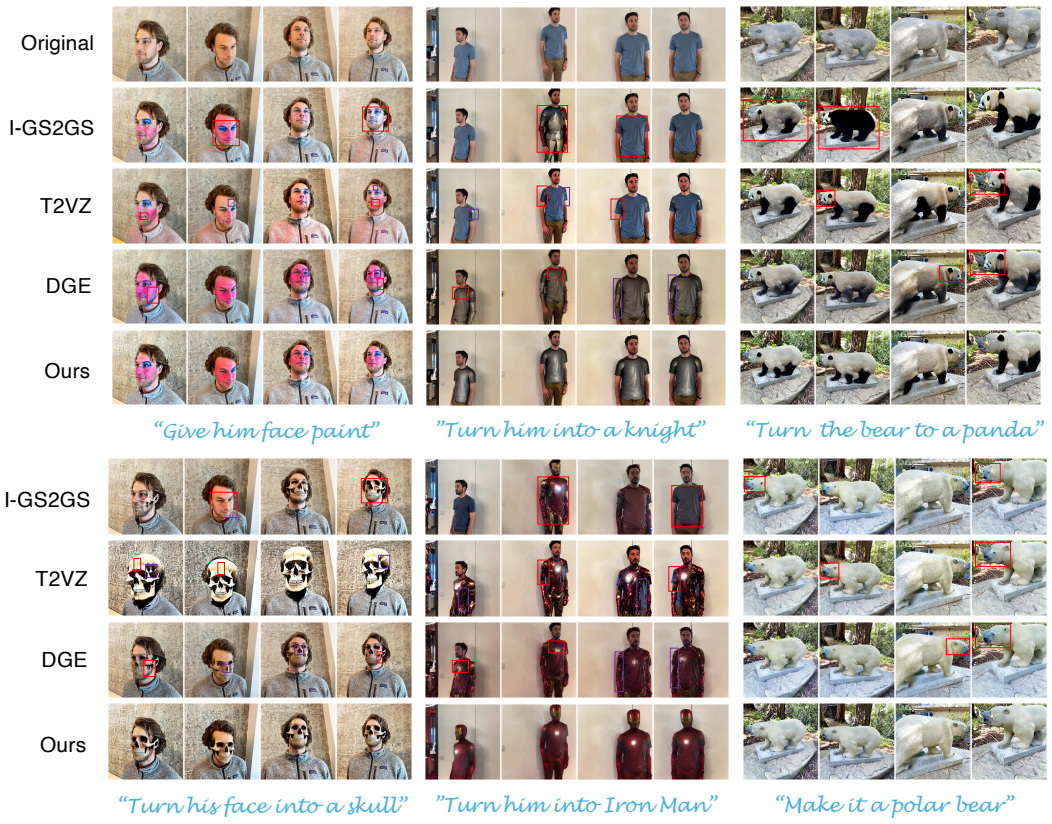


Figure 2: Qualitative comparison with prior work. The top row shows the original scenes, and the lower rows present edits from different methods. Matching red or purple rectangles indicate pairs of inconsistent regions, which frequently appear in baselines but not in our edits. Please zoom in electronically for details; enlarged views are provided in Appendix E.

Method	CLIP Cons. ↑	CLIP Sim. ↑	CLIP Dir. ↑
I-N2N	0.034	0.196	0.105
I-GS2GS	0.314	0.253	0.169
T2VZ	0.310	0.251	0.159
DGE	0.287	0.256	**0.182**
Ours	**0.342**	**0.258**	0.173

Table 1: Comparison of methods across view consistency, semantic alignment and edit performance.

the cost of per-frame edit quality, as demonstrated by CLIP Sim. and CLIP Dir. scores, where our method is either superior to or competitive with the baselines.

Notably, I-N2N fails completely in this sparse-view setting. We observed that Nerfacto struggles to fit the scene, producing severe floater artifacts even when rendering source poses. These distortions lie out of distribution for the 2D editor, leading to unusable edits as shown in Appendix C.

The advantages of our approach compared to other baselines, are most clearly demonstrated qualitatively. In the sparse-view setting, baseline methods often struggle to maintain consistent edits across viewpoints due to two factors: (i) 3DGS-based consolidation becomes unreliable with limited views, as the 3DGS tends to overfit the training images; and (ii) cross-frame attention, while improving general appearance alignment, fails to enforce fine-grained consistency. Figure 2 illustrates these issues: I-GS2GS edits remain largely view-independent (e.g., the *Face Paint* edit), while T2VZ and DGE, though producing roughly appearance-consistent edits, introduce inconsistency in the details—such as mismatched sleeve and chest textures in the *Knight* and *Iron Man* edits, varying

SDS Stage	Config	CLIP Cons. ↑	CLIP Sim. ↑	CLIP Dir. ↑
–	Student Only	0.014	0.212	0.161
–	Teacher Only	0.228	0.252	0.184
Initialization (0)	W/O Editing Ref. Frame	0.326	0.264	0.174
Alignment (2)	Learned Mapping	0.287	0.259	0.180
Perturbation (3)	Uniform t	0.363	0.260	0.146
	τ-matched t	0.435	0.231	0.107
Teacher Pred. (4)	W/O RCVAttn	0.230	0.260	0.175
	Full	0.337	0.263	0.178

Table 2: Ablation study evaluating different design choices. Weak results are highlighted in red.

face paint colors and intensities in *Face Paint*, and view-dependent differences in *Skull* details such as the nose, cheek, and forehead. The lack of robust 3D consistency is especially evident in the *Bear* edit, which exhibits Janus-like multi-face artifacts. In contrast, I-Mix2Mix produces edits that are not only faithful to the instruction but also highly consistent across all views, without sacrificing image quality. This combination of instruction alignment, visual fidelity, and strong 3D consistency represents a clear improvement over existing approaches in the sparse-view setting.

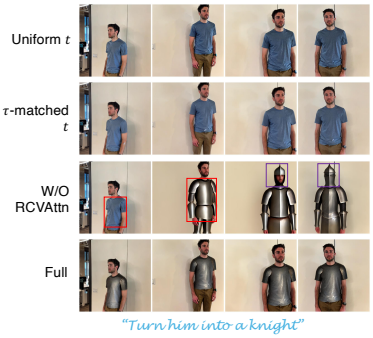
5.2 Ablation Study

We conduct an ablation study on 6 representative edits (listed in Appendix B) to assess the contributions of our design choices across the SDS pipeline. Quantitative results are summarized in Table 2, with particularly weak results highlighted in red. Our findings show that each component is essential to achieve both faithful edits and strong multi-view consistency.

Role of teacher and student. We first test the student and teacher models in isolation. In the *Student Only* setting, the student is given an edited frame as input and asked to sample additional views. This fails for several reasons: the student never sees the scene content captured by the other frames, leading to poor faithfulness; the SEVA prior struggles under single-view input; and we suspect that the edited scenes lie out-of-distribution for the model. This suggests that our approach *distills new capabilities into the student*, rather than simply searching within its existing sampling distribution. Conversely, in the *Teacher Only* setting, we rely on the teacher to edit each view independently. While individual frames adhere to the instruction edit, the lack of a 3D prior leads to severe cross-view inconsistency, as reflected by the low CLIP Consistency score. We present representative visualizations in Appendix D. Together, these results confirm the necessity of distilling from the teacher into the student, rather than using either in isolation.

Initialization stage. In the *W/O Editing Ref. Frame* setting, we input one of the original frames to the student encoder, to serve as the reference latent, without editing it first: $z_{\text{ref}} = \mathcal{E}_{\text{S}}(I_{\text{ref}})$. Skipping the reference frame edit negatively affected the distillation process, as the initial student predictions are further away from the target. This results in slightly lower multi-view consistency.

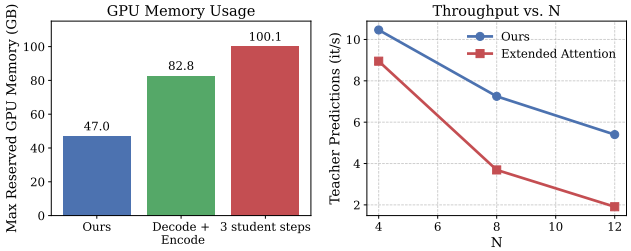
Alignment stage. Following findings in prior work on latent space alignment (Huh et al., 2024; Li et al., 2015), we replaced the bilinear interpolation with a learnable convolutional mapping (*Learned Mapping*), optimized during distillation, but found this brought no measurable benefit – the necessary transformation is likely captured during the fine-tuning of the student.



Perturbation stage. We evaluated two alternatives to our proposed forward schedule: *Uniform* t (similar to Poole et al. (2022)), where t is sampled uniformly in $[0.05, 0.95]$, and τ-*matched* t, where the teacher's timestep follows the student's. Both variants tend to collapse to near-identity reconstructions of the input scene, which explains their paradoxically high CLIP Consistency: such outputs are trivially consistent but fail to realize

the intended edit as reflected in their low CLIP Directional scores. Visual examples are provided in the first two rows of the *Knight* inset figure.

Teacher prediction stage. Finally, disabling our Random Cross-View Attention mechanism (*W/O RCVAttn*) leads the teacher to process each perturbed latent independently. Without cross-view coupling, the student receives conflicting signals across views, leading to degraded multi-view consistency and breaking its 3D prior. This is again reflected in low CLIP Consistency, and illustrated in the third row of the *Knight* inset figure.



Efficiency considerations. Several components of I-Mix2Mix were explicitly designed for memory and compute efficiency. Our choice of using a single-step prediction in the *Student Query* stage is crucial: a three-step alternative more than doubles peak memory usage for $N = 4$ views (see inset). In the *Student-Teacher Alignment* stage, interpolating latents rather than passing through the student's decoder and teacher's encoder reduces memory usage by over 50%. Finally, replacing the RCVAttn module with full extended attention significantly degraded throughput, worsening as the number of frames increased. We additionally experimented with fine-tuning the student using LoRA (Hu et al., 2022) rather than updating the full U-Net. While more parameter-efficient, this approach underperformed, and we leave the adaptation of lightweight variants to future work.

5.3 Beyond Image Editing

Our approach is not inherently limited to editing tasks. In principle, any pre-trained image-to-image diffusion model can serve as the teacher, with the multi-view student acting as a consolidator to produce a multi-view–to–multi-view solution. To explore this, we experimented with multi-view conditional generation. Specifically, we employed pre-trained ControlNets (Zhang et al., 2023) as teachers to translate multiple depth maps or Canny edge maps of a 3D scene into consistent RGB images. Qualitative examples are provided in Appendix H. While the outputs were faithful to the conditioning inputs and maintained multi-view consistency, they often exhibited excessive blurriness—a known artifact of SDS-based optimization (Poole et al., 2022).

6 Discussion: Parallel to Diffusion Guidance

In standard diffusion guidance (Bansal et al., 2023; Chung et al., 2022), the model predictions at a given timestep are often critiqued, and the resulting gradient is used to modify the sampling trajectory. In our framework, rather than applying such potentially unstable updates to the latents, we backpropagate the guidance signal to the student's weights. This approach effectively transfers the teacher's knowledge without making aggressive modifications to the latents themselves, avoiding manifold slips – divergence from the target distribution. Consequently, the student gradually learns to generate multi-view consistent edits while maintaining stable sampling dynamics.

7 Conclusion, Limitations, and Future Work

We presented I-Mix2Mix, a novel framework for multi-view image editing that achieves high multi-view consistency in sparse-view settings, where prior methods typically fail. While effective, our approach inherits the limitations of its backbones, specifically InstructPix2Pix and SEVA, which can struggle with certain edit prompts or with maintaining perfect consistency across views. Given our framework's modular nature, we anticipate that integrating stronger future backbones could mitigate these issues. Additionally, I-Mix2Mix requires multiple distillation iterations per noise level, making it more than twice as slow as our strongest competitor, DGE. We plan to explore strategies to reduce this overhead in future work. Finally, as discussed in Section 5.3, our framework is potentially general and applicable to a range of image manipulation tasks beyond editing. However, performance on these tasks currently lags behind our editing results, often producing blurry outputs. We leave the investigation of these directions to future work.

Ethics statement. Our work builds on publicly available datasets (Haque et al., 2023; Knapitsch et al., 2017; Reizenstein et al., 2021; Barron et al., 2022) that were released for research purposes. Some of these datasets include human subjects, for which consent has been obtained as described in the original publications. As our method enables multi-view image editing, it could in principle be misused for generating misleading or harmful content (e.g., deepfakes). While our focus is on advancing controllable and consistent scene editing for academic and scientific applications, we caution that such techniques should be applied responsibly, in accordance with ethical standards and legal regulations.

Reproducibility statement. We provide a detailed description of our method in Section 4, along with a pseudo-algorithm in Algorithm 1. Implementation details are given in Appendix A. The evaluation protocol, datasets, and editing prompts are described in Section 5 and Appendix B.

REFERENCES

Andrea Asperti and Valerio Tonelli. Comparing the latent space of generative models. *Neural Computing and Applications*, 35(4):3155–3172, 2023.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 843–852, 2023.

Chong Bao, Yinda Zhang, Bangbang Yang, Tianxing Fan, Zesong Yang, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Sine: Semantic-driven image-based nerf editing with prior-guided editing field. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20919–20929, 2023.

Roi Bar-On, Dana Cohen-Bar, and Daniel Cohen-Or. Editp23: 3d editing via propagation of image prompts to multi-view. *arXiv preprint arXiv:2506.20652*, 2025.

Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *CVPR*, 2022.

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*, pp. 18392–18402, 2023.

Duygu Ceylan, Chun-Hao P Huang, and Niloy J Mitra. Pix2video: Video editing using image diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23206–23217, 2023.

Hansheng Chen, Ruoxi Shi, Yulin Liu, Bokui Shen, Jiayuan Gu, Gordon Wetzstein, Hao Su, and Leonidas Guibas. Generic 3d diffusion adapter using controlled multi-view editing. *arXiv preprint arXiv:2403.12032*, 2024a.

Jun-Kun Chen, Jipeng Lyu, and Yu-Xiong Wang. Neuraleditor: Editing neural radiance fields via manipulating point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12439–12448, 2023a.

Minghao Chen, Junyu Xie, Iro Laina, and Andrea Vedaldi. Shap-editor: Instruction-guided latent 3d editing in seconds. *arXiv preprint arXiv:2312.09246*, 2023b.

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, 2024b.

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*, pp. 21476–21485, 2024c.

Yufeng Chi, Huimin Ma, Kafeng Wang, and Jianmin Li. Disco3d: Distilling multi-view consistency for 3d scene editing. *arXiv preprint arXiv:2508.01684*, 2025.

Pei-Ze Chiang, Meng-Shiun Tsai, Hung-Yu Tseng, Wei-Sheng Lai, and Wei-Chen Chiu. Stylizing 3d scene via implicit representation and hypernetwork. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 1475–1484, 2022.

Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

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.

Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arXiv:2307.10373*, 2023.

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*, pp. 19740–19750, 2023.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Yi-Hua Huang, Yue He, Yu-Jie Yuan, Yu-Kun Lai, and Lin Gao. Stylizednerf: consistent 3d scene stylization as stylized nerf via 2d-3d mutual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18342–18352, 2022.

Yukun Huang, Jianan Wang, Yukai Shi, Boshi Tang, Xianbiao Qi, and Lei Zhang. Dreamtime: An improved optimization strategy for diffusion-guided 3d generation. *arXiv preprint arXiv:2306.12422*, 2023.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. The platonic representation hypothesis. *arXiv preprint arXiv:2405.07987*, 2024.

Hiromichi Kamata, Yuiko Sakuma, Akio Hayakawa, Masato Ishii, and Takuya Narihira. Instruct 3d-to-3d: Text instruction guided 3d-to-3d conversion. *arXiv preprint arXiv:2303.15780*, 2023.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. URL `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`.

Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15954–15964, 2023.

Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. Flux. 1 kontext: Flow matching for in-context image generation and editing in latent space. *arXiv preprint arXiv:2506.15742*, 2025.

Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 991–999, 2015.

Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. Convergent learning: Do different neural networks learn the same representations? *arXiv preprint arXiv:1511.07543*, 2015.

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*, volume 38, pp. 3279–3287, 2024.

Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8599–8608, 2024.

Steven Liu, Xiuming Zhang, Zhoutong Zhang, Richard Zhang, Jun-Yan Zhu, and Bryan Russell. Editing conditional radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5773–5783, 2021.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Artem Lukoianov, Haitz Sáez de Ocáriz Borde, Kristjan Greenewald, Vitor Guizilini, Timur Bagautdinov, Vincent Sitzmann, and Justin M Solomon. Score distillation via reparametrized ddim. *Advances in Neural Information Processing Systems*, 37:26011–26044, 2024.

Aryan Mikaeili, Or Perel, Mehdi Safaee, Daniel Cohen-Or, and Ali Mahdavi-Amiri. Sked: Sketch-guided text-based 3d editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14607–14619, 2023.

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

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

Thu Nguyen-Phuoc, Feng Liu, and Lei Xiao. Snerf: stylized neural implicit representations for 3d scenes. *arXiv preprint arXiv:2207.02363*, 2022.

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

Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15932–15942, 2023.

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.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10901–10911, 2021.

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*, pp. 10684–10695, 2022.

Etai Sella, Gal Fiebelman, Peter Hedman, and Hadar Averbuch-Elor. Vox-e: Text-guided voxel editing of 3d objects. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 430–440, 2023.

Chaehun Shin, Heeseung Kim, Che Hyun Lee, Sang-gil Lee, and Sungroh Yoon. Edit-a-video: Single video editing with object-aware consistency. In *Asian Conference on Machine Learning*, pp. 1215–1230. PMLR, 2024.

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

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

Cyrus Vachha and Ayaan Haque. Instruct-gs2gs: Editing 3d gaussian splats with instructions, 2024.

Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, Steven Liu, William Berman, Yiyi Xu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. URL `https://github.com/huggingface/diffusers`.

Binglun Wang, Niladri Shekhar Dutt, and Niloy J Mitra. Proteusnerf: Fast lightweight nerf editing using 3d-aware image context. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 7(1):1–17, 2024a.

Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3835–3844, 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, 2023a.

Junjie Wang, Jiemin Fang, Xiaopeng Zhang, Lingxi Xie, and Qi Tian. Gaussianeditor: Editing 3d gaussians delicately with text instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20902–20911, 2024b.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in neural information processing systems*, 36:8406–8441, 2023b.

Silvan Weder, Guillermo Garcia-Hernando, Aron Monszpart, Marc Pollefeys, Gabriel J Brostow, Michael Firman, and Sara Vicente. Removing objects from neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16528–16538, 2023.

Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7623–7633, 2023.

Bangbang Yang, Yinda Zhang, Yinghao Xu, Yijin Li, Han Zhou, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Learning object-compositional neural radiance field for editable scene rendering. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 13779–13788, 2021.

Yu-Jie Yuan, Yang-Tian Sun, Yu-Kun Lai, Yuewen Ma, Rongfei Jia, and Lin Gao. Nerf-editing: geometry editing of neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18353–18364, 2022.

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*, pp. 3836–3847, 2023.

Canyu Zhao, Xiaoman Li, Tianjian Feng, Zhiyue Zhao, Hao Chen, and Chunhua Shen. Tinker: Diffusion's gift to 3d–multi-view consistent editing from sparse inputs without per-scene optimization. *arXiv preprint arXiv:2508.14811*, 2025.

Jensen Jinghao Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. *arXiv preprint arXiv:2503.14489*, 2025.

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*, pp. 1–10, 2023.

APPENDIX

CONTENTS

A Implementation Details

We use SEVA 1.1 (Zhou et al., 2025) as the pre-trained student model and InstructPix2Pix (Brooks et al., 2023) from the Diffusers library (von Platen et al.) as the frozen teacher. Consistent with prior observations Haque et al. (2023); Chen et al. (2024b), the teacher's classifier-free guidance (CFG) scales for both prompt and input image have a significant effect on the *degree of edit intensity*—a factor that is often subjective and a matter of personal taste. For most edits we adopt the default $S_T = 7.5$ for the prompt and $S_I = 1.5$ for the input image, with adjustments detailed appendix B. We perform distillation over 40 student timesteps ($\Delta\tau = \frac{1}{40}$), with $k = 50$ updates per step. Optimization is done with AdamW (Loshchilov & Hutter, 2017), using a maximum learning rate of 1×10^{-4} after 200 iterations of linear warm-up, followed by cosine decay down to 5×10^{-5}. This yields just over 2000 distillation iterations per experiment, which take about 40 minutes on a single NVIDIA H200 GPU.

A.1 Teacher Forward Schedule

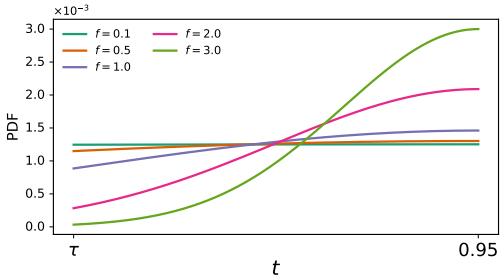


Figure 3: Teacher timestep schedule for different skewness factors f.

We employ a stochastic schedule for the teacher forward process, sampling timesteps as

$$t \sim \text{TruncNorm}\big(\mu = b,\ \sigma = (b - \tau)/f,\ a = \tau,\ b = 0.95\big),$$

where τ is the current student timestep and f controls the skewness of the distribution. Larger f concentrates probability near b, making the teacher more likely to operate at higher noise levels. The stochasticity ensures the teacher provides strong gradients every few iterations, which we find effective for avoiding collapse to poor local minima. Figure 13 illustrates how different f values shape the probability distribution. In practice, we use $f = 0.5$, yielding an approximately uniform distribution over $[\tau, 0.95]$.

B Evaluation Scenes and Edits

We detail in Table 3 the edits used in our evaluations, applied to the standard Face, Bear, and Person scenes from the Instruct-NeRF2NeRF dataset Haque et al. (2023). The *Edit Prompt* is the editing instruction provided as input to the evaluated methods, while the *Original Prompt* and *Edited Prompt* are employed for CLIP-based evaluation. For each edit, we also report the teacher's text and image CFG scales, s_T and s_I, used in quantitative evaluation. Edits with bolded prompts indicate those selected for the ablation experiments.

Scene	Original Prompt	Edit Prompt	Edited Prompt	Text CFG	Image CFG
Face	"A man with curly hair in a grey jacket"	"Give him a Venetian mask"	"A man with curly hair in a grey jacket with a Venetian mask"	7.5	1.5
		"Turn him into a vampire"	"A vampire with curly hair"	7.5	1.5
		"Turn him into Tolkien Elf"	"A Tolkien Elf with curly hair"	9.0	1.5
		"Turn him into batman"	"A batman"	7.5	1.5
		"Turn his face into a skull"	"A man with a skull head in a grey jacket"	7.5	1.5
		"Turn him into Albert Einstein"	"Albert Einstein with curly hair"	7.5	1.5
		"Turn it to a Van Gogh painting"	"A Van Gogh painting of a man with curly hair in a jacket"	7.5	1.5
		"Give him face paint"	"A man with curly hair in a grey jacket with face paint"	7.5	1.5
Bear	"A stone bear in a garden"	**"Turn the bear to a panda bear"**	"A panda bear in a garden"	6.0	1.5
		"Turn the bear to a polar bear"	"A polar bear in a garden"	6.0	1.5
		"Turn the bear to a grizzly bear"	"A grizzly bear in a garden"	5.5	1.5
		"Turn the bear to a wooden bear"	"A wooden bear in a garden"	8.5	1.5
Person	"A man standing next to a wall wearing a blue T-shirt and brown pants"	"Turn him into Iron Man"	"An Iron Man standing next to a wall"	7.5	1.5
		"Turn the man into a robot"	"A robot standing next to a wall"	5.5	1.8
		"Make him in a suit"	"A man standing next to a wall wearing a suit"	6.5	1.8
		"Turn him into a clown"	"A clown standing next to a wall"	6.0	1.8
		"Make him into a marble statue"	"A marble statue of a man next to a wall"	7.5	1.5
		"Turn him into a cowboy with a hat"	"A cowboy wearing a hat standing next to a wall"	6.0	1.5
		"Turn him into a soldier"	"A soldier standing next to a wall"	7.5	1.5
		"Turn him into a knight"	"A knight standing next to a wall"	6.0	1.5

Table 3: Prompts and CFG values for each edit used for quantitative evaluation.

C Limitations of Instruct-NeRF2NeRF in Sparse-View Settings

In the sparse-view regime, Instruct-NeRF2NeRF (I-N2N) Haque et al. (2023) fails to produce coherent results. Its underlying Nerfacto (Tancik et al., 2023) model, trained with default configurations for 30K iterations, struggles to reconstruct the scene accurately, generating severe floater artifacts even when rendering the original input poses. These distortions fall far outside the distribution expected by the 2D editor, rendering the resulting edits unusable. Figure 4 illustrates two representative examples of such failures, corresponding to the *Clown* and *Face Paint* edits.

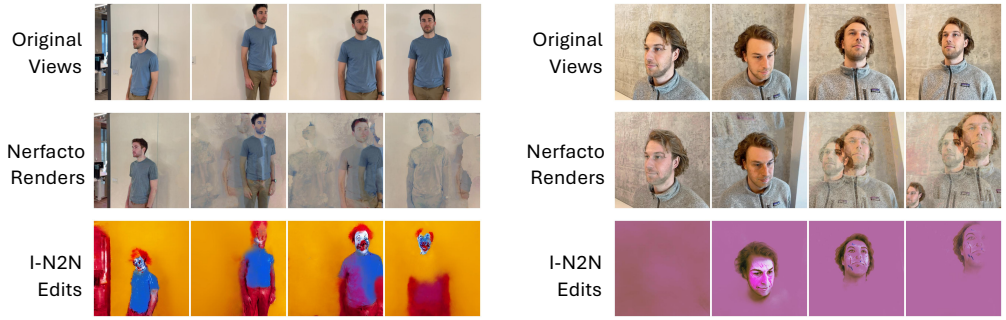


Figure 4: Examples for I-N2N failures in the sparse-view setting.

D STUDENT AND TEACHER LIMITATIONS

Figure 5 illustrates the limitations of the student (SEVA) and teacher (Instruct-Pix2Pix) when used individually. Even on the unedited scene, SEVA can struggle to produce high-quality results with only a single input frame, as shown in the second row. When used as an editing baseline—receiving a single edited frame and asked to generate the remaining views—it fails to produce coherent frames (third row). Individual predictions from the teacher (final row) are independent across views, resulting in inconsistent and sometimes implausible edits.

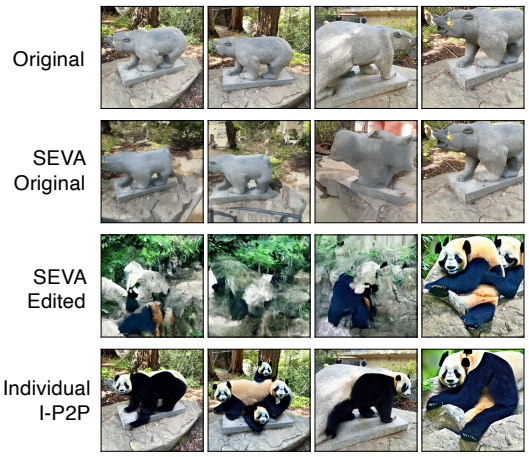


Figure 5: Student and Teacher models limitation example, on the *Bear* scene and *Panda* edit.

E EXTENDED QUALITATIVE COMPARISONS WITH BASELINES

We present additional qualitative comparisons to prior methods, including both enlarged versions of the edits shown in Figure 2 and additional edits. Matching red or purple rectangles highlight regions with multi-view inconsistencies.

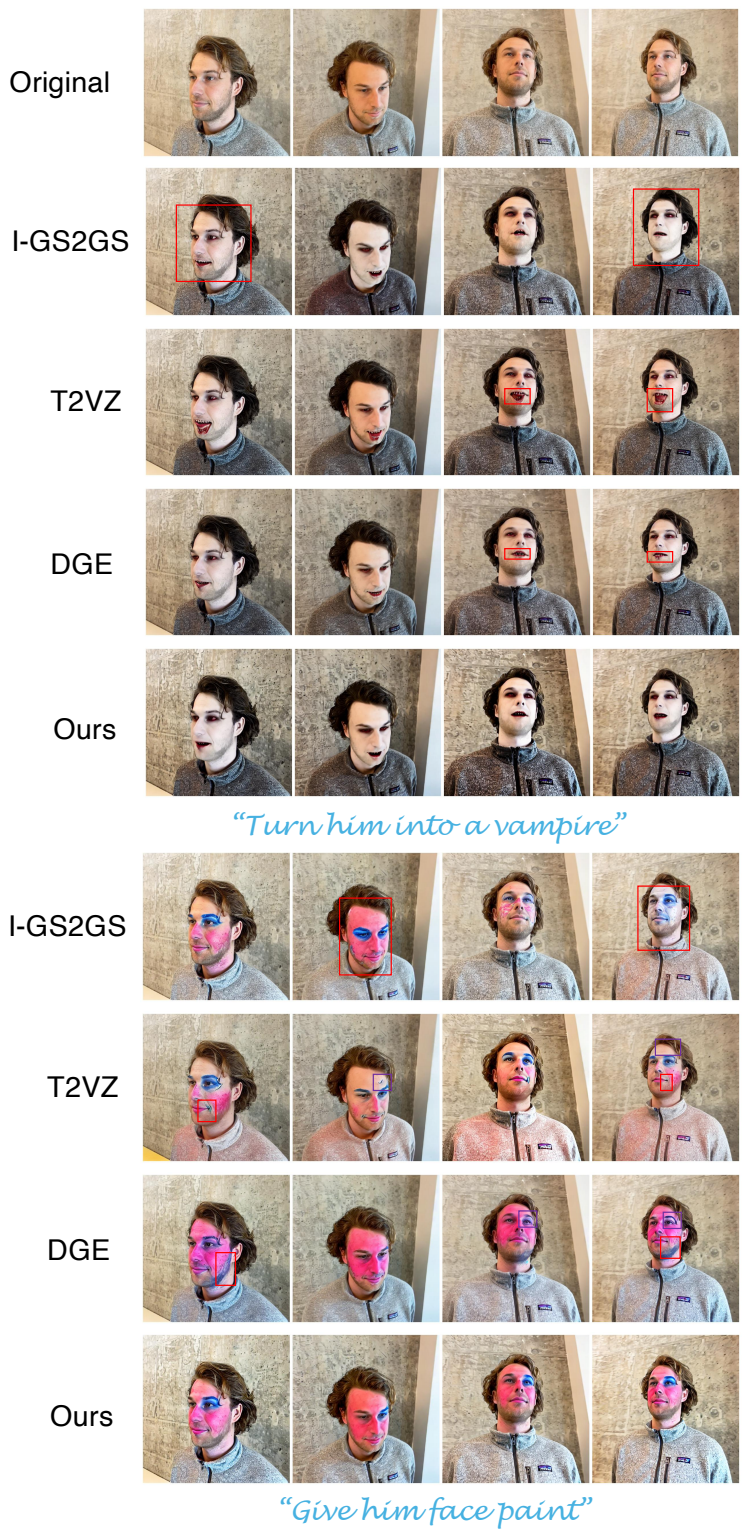


Figure 6: Comparison to baselines on Face scene edits.

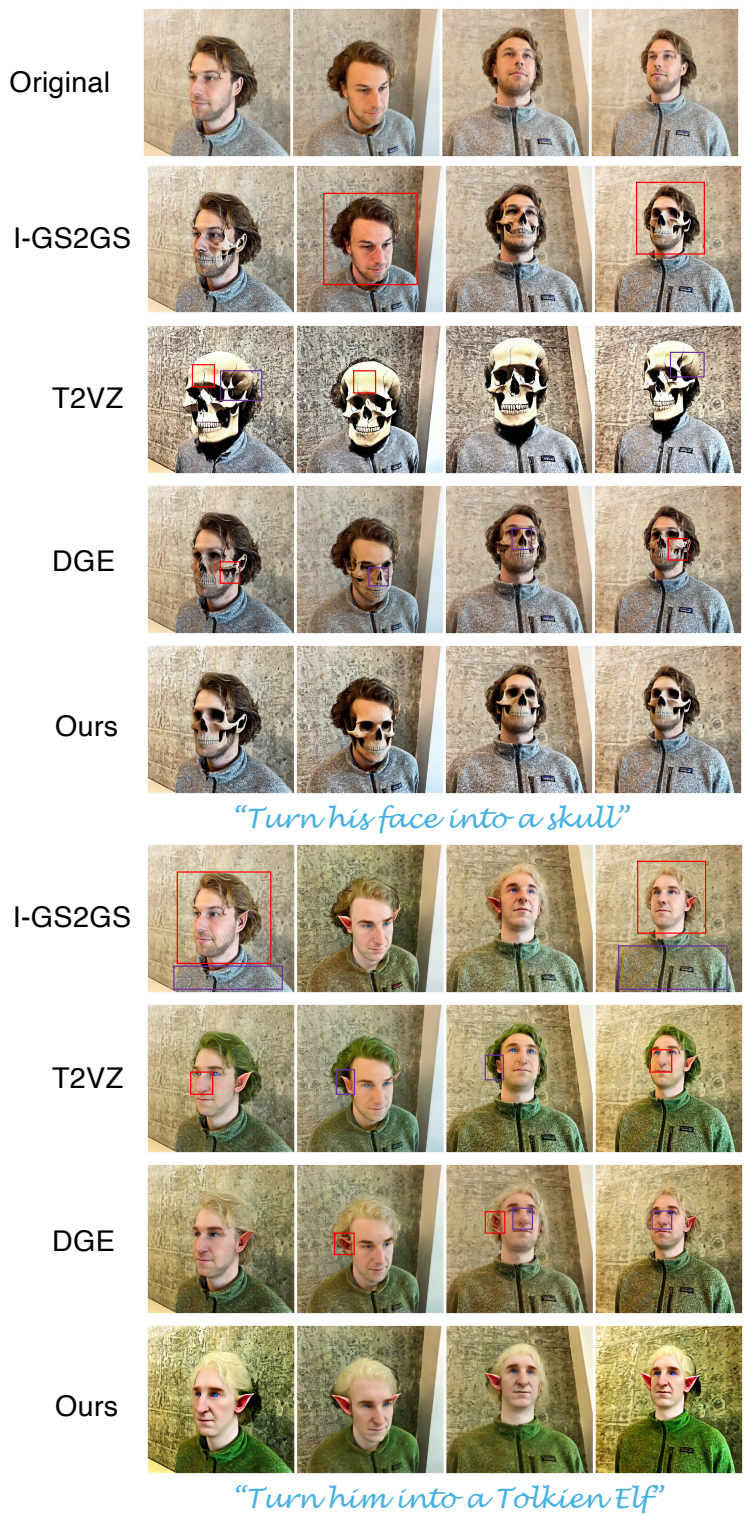


Figure 7: Comparison to baselines on Face scene edits.

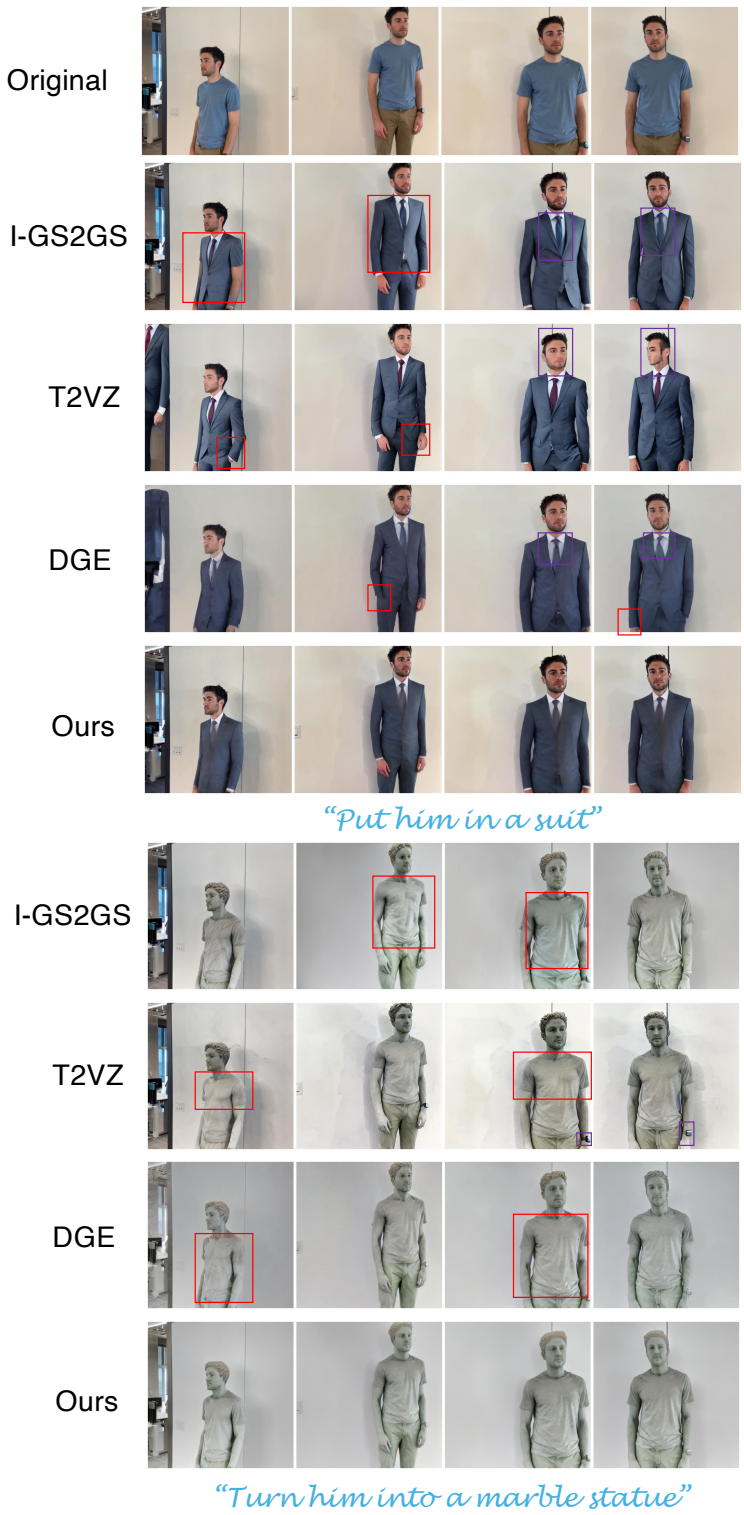


Figure 8: Comparison to baselines on Person scene edits.

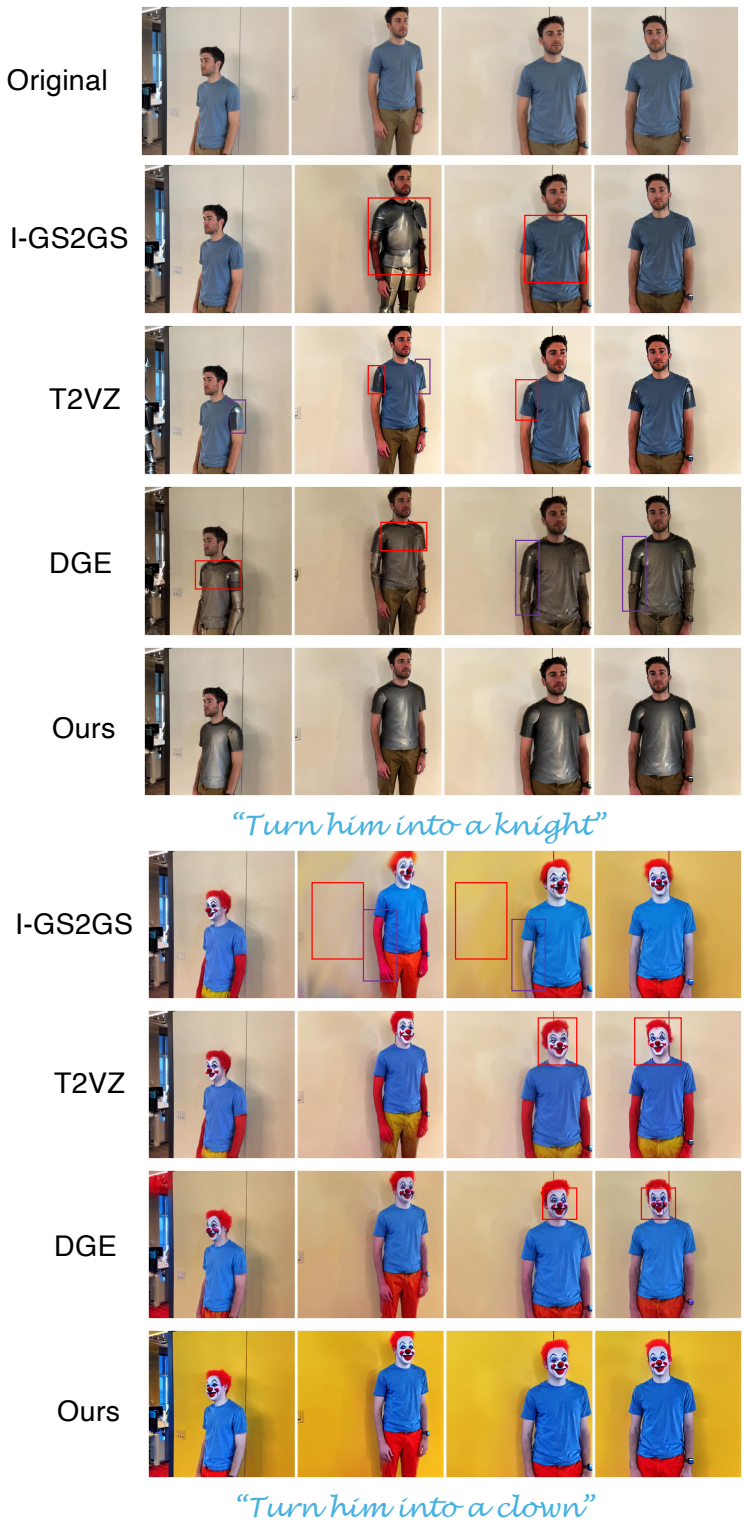


Figure 9: Comparison to baselines on Person scene edits.

Figure 10: Comparison to baselines on Person scene edits.

Figure 11: Comparison to baselines on Person scene edits.

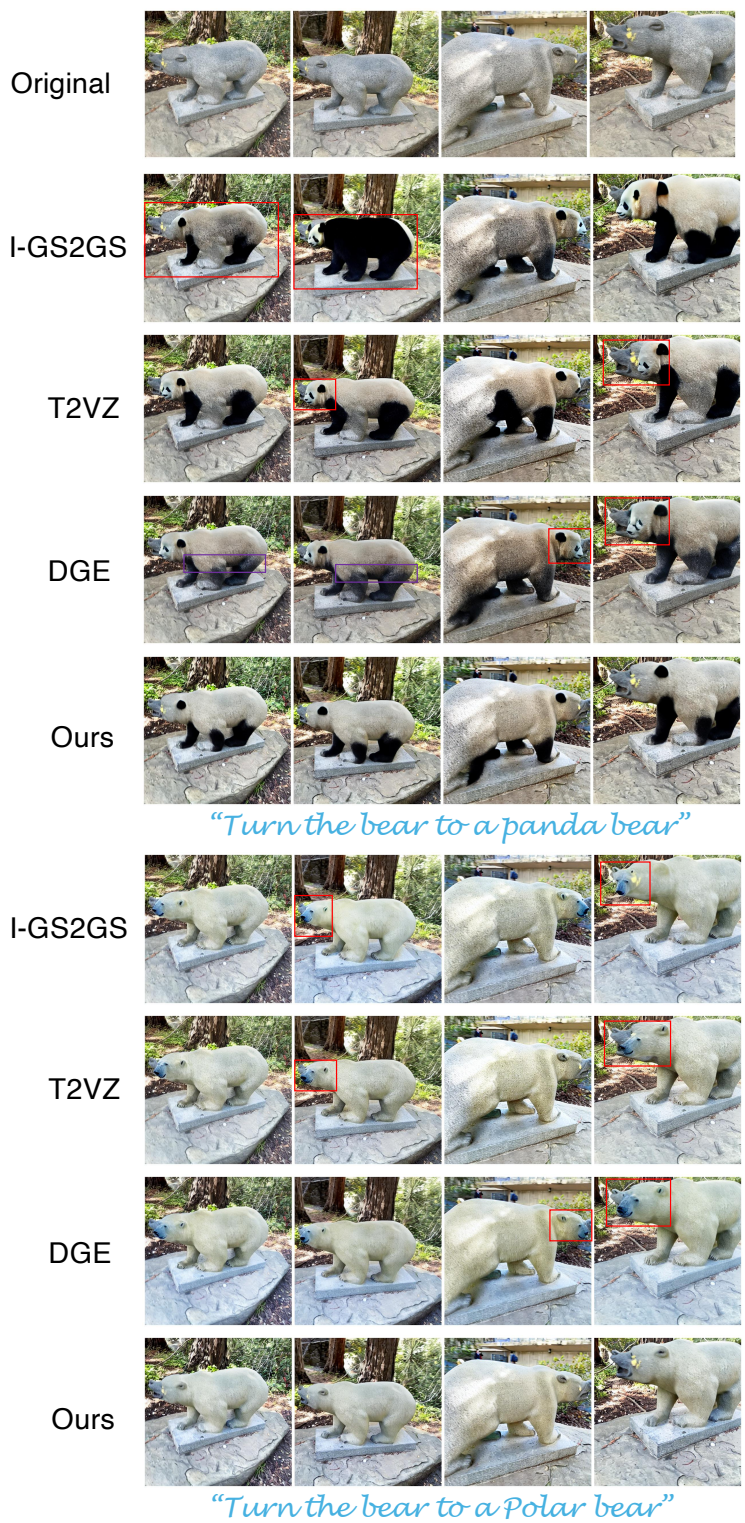


Figure 12: Comparison to baselines on Bear scene edits.

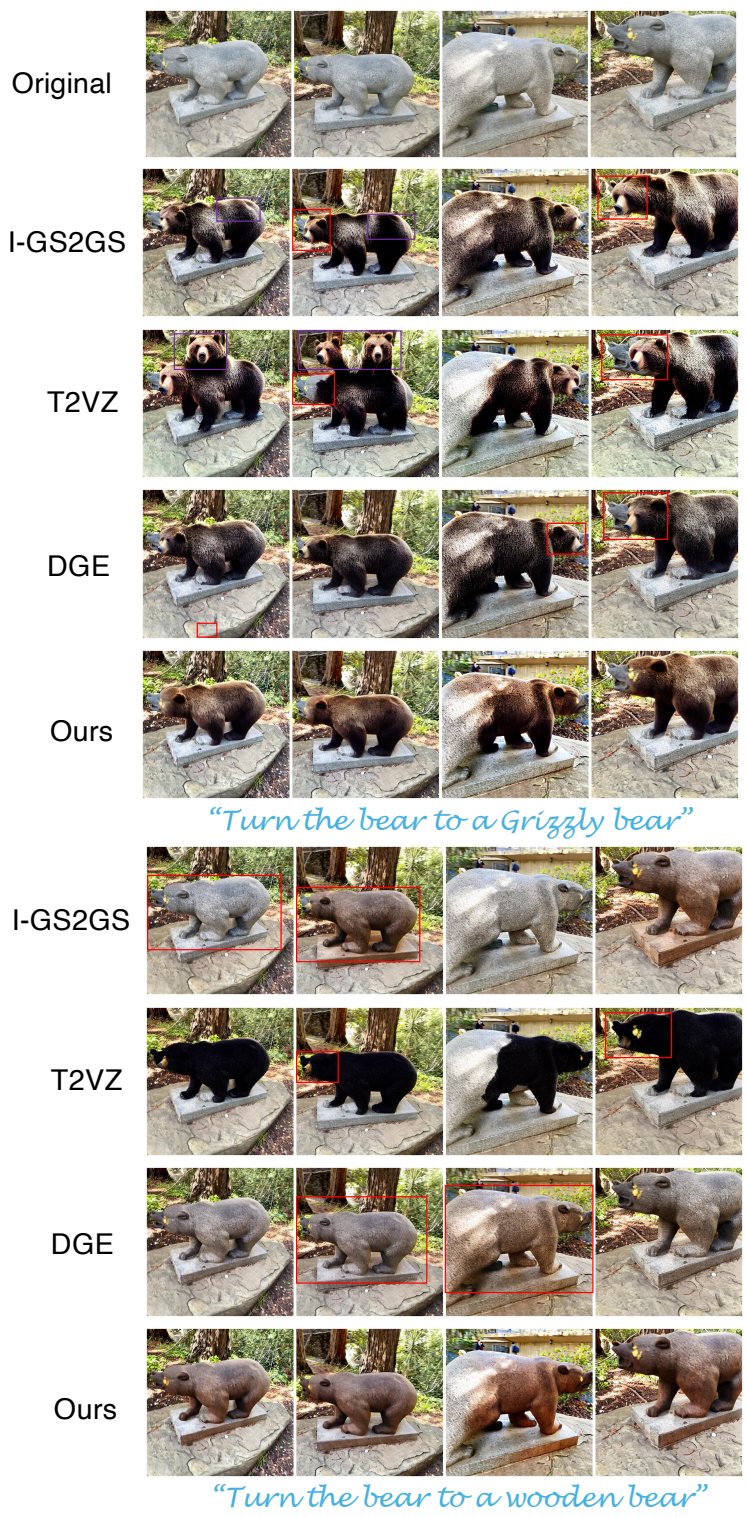


Figure 13: Comparison to baselines on Bear scene edits.

F Additional Results on Diverse Scenes

In Figures 14, 15 we present further qualitative results of I-Mix2Mix applied to four different scenes: *Car* from the CO3D dataset Reizenstein et al. (2021), *Garden* from the Mip-NeRF 360 dataset Barron et al. (2022), and *Horse* and *Ignatius* from the Tanks and Temples dataset Knapitsch et al. (2017).

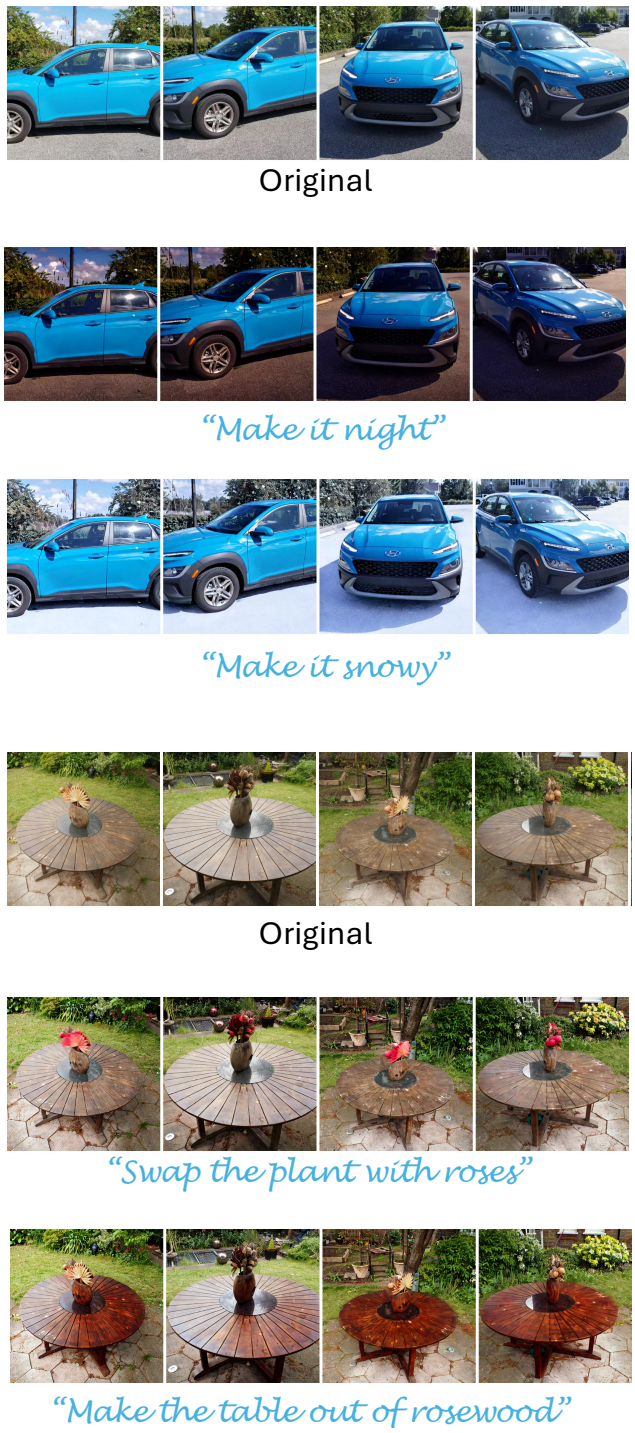


Figure 14: I-Mix2Mix edits on the Car (top three rows) and Garden (bottom rows) scenes.

Original

"Make it during sunset"

"Change the statue to gold"

Original

"Make it spring"

"Make the floor out of ice"

Figure 15: I-Mix2Mix edits on the Horse (top three rows) and Ignatius (bottom rows) scenes.

G Results with More Input Frames

Figure 16 presents outputs of I-Mix2Mix when using $N = 8$ input frames.

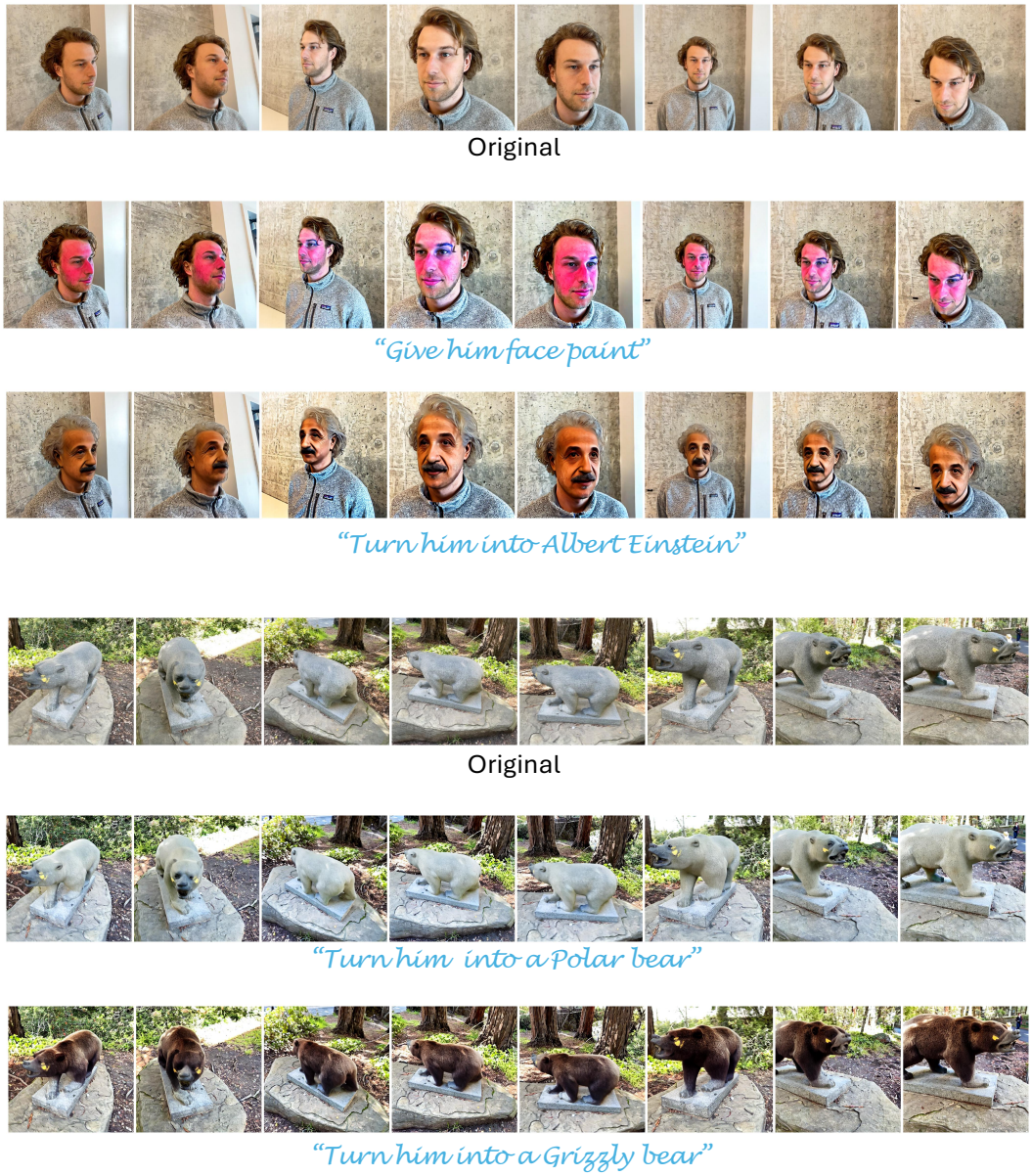


Figure 16: I-Mix2Mix edits on 8 input frames on Face and Bear scenes.

H Beyond Image Editing

I-Mix2Mix is not tied to a specific editor or to editing tasks, and can in principle generalize to other multi-view conditional generation scenarios. To illustrate this, we used pre-trained ControlNets (Zhang et al., 2023) as teachers to translate multiple depth or Canny maps of a 3D scene into consistent RGB images. Figure 17 shows examples. While outputs respect the conditioning and maintain multi-view consistency, they often appear overly blurry, highlighting limitations of SDS-based optimization (Poole et al., 2022).

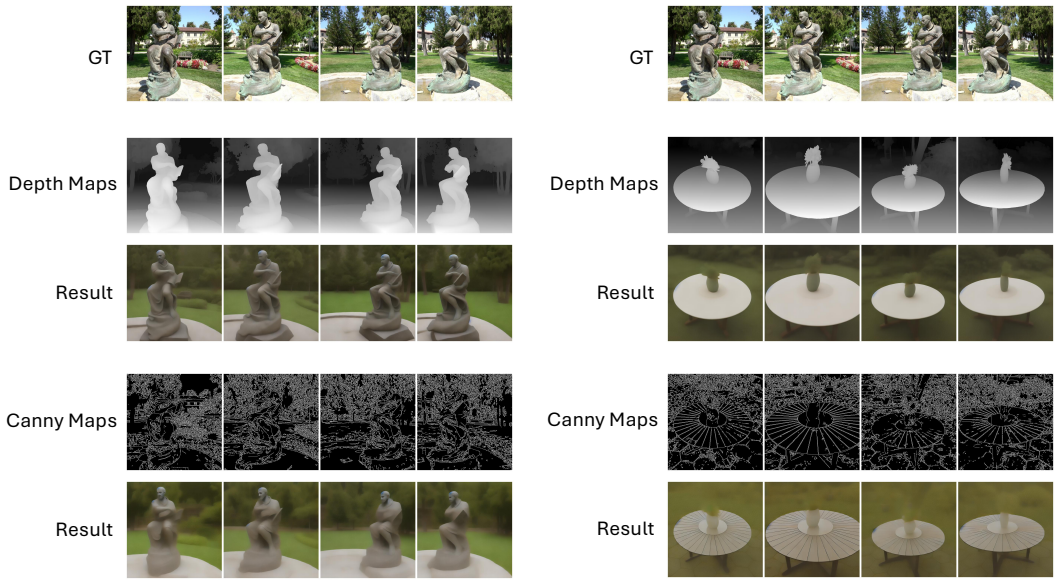


Figure 17: Example results of I-Mix2Mix with Canny edge map and Depth maps as input, with corresponding ControlNet teachers.

I USE OF LARGE LANGUAGE MODELS

Large language models were employed as general-purpose assistants for both writing and coding throughout this work.