MotionEditGS: Editing Motion and Appearance of 4D Scenes from Monocular Video via Semantically Anchored Gaussians

Daehyun We, Youngho Yoon, Jiyong Boo, and Kuk-Jin Yoon

Visual Intelligence Lab., KAIST, Korea
{greathyun, dudgh1732, boojiyong, kjyoon}@kaist.ac.kr

Abstract. Editing dynamic 3D scenes reconstructed from monocular videos is important for applications such as content creation and data augmentation, but remains challenging. Prior works adapt image diffusion models across frames while enforcing spatial–temporal consistency, yet they often produce uniform, appearance-only edits. We instead leverage a video diffusion model to enable photorealistic 4D edits and controllable motion changes. In monocular settings, however, fitting dynamic 3D Gaussians with pure photometric loss is under-constrained, leading to shallow geometric reconstructions and temporal artifacts in the edited scene. We address this with a novel 4D editing pipeline that uses compact latents distilled from DINO features as additional supervision during scene optimization and preserves them during editing as a semantic identity signal. Such semantic consistency regularization provides strong geometric and motion cues, enabling robust scene editing even under large or complex edits from the video diffusion model. Furthermore, to promote semantically novel content generation during editing, we apply a gradient-based feature lifting method based on accumulated feature gradients during Gaussian densification. Experiments show improved spatial-temporal realism and text alignment over image editing baselines, and demonstrate compositional motion-appearance edits with significant geometry changes that prior methods cannot achieve.

Keywords: 4D Scene Editing · Video Diffusion Models

1 Introduction

Recent advances in 3D scene representations, notably Neural Radiance Fields (NeRF) [44] and 3D Gaussian Splatting (3DGS) [30], have enabled photorealistic novel-view synthesis of real-world scenes. Building on these advances, a series of works have extended such models to dynamic 3D (4D) settings [5, 51, 71, 76, 77], allowing reconstruction of a scene with complex motion and deformation. As these dynamic representations mature, there is a growing demand for editing capabilities—to flexibly control appearance, geometry, and motion—for applications such as immersive VR/AR content creation and scalable data augmentation. However, existing approaches to dynamic scene editing mainly focus on

appearance, with limited control over geometry or motion, resulting in a lack of true 4D control. In this work, we focus on editing 4D scenes reconstructed from monocular videos, a practical yet challenging setting due to limited observations and inherent ambiguities.

Existing works [22, 27, 32, 38, 45, 59] perform text-guided 4D scene editing by leveraging 2D diffusion-based image editing model such as InstructPix2Pix (IP2P) [3]. Among these methods, a common approach is to edit each frame of a monocular or multi-view video and then adopt an Iterative Dataset Update (IDU) strategy, following the Instruct-NeRF2NeRF [21] paradigm, where the edited frames are iteratively used as supervision for 4D scene optimization. To enforce spatial–temporal consistency during this process, various techniques have been proposed. Instruct-4D-to-4D [45] propagates the edited key frames across time using depth- and optical-flow-based spatial–temporal warping, aligning them with the original motion and cross-frame correspondence. Meanwhile, CTRL-D [22] fine-tunes the IP2P model on an original–edited image pair to obtain a personalized editor, which is then applied consistently to other frames. While these methods help maintain spatial–temporal consistency, they often yield uniform edits with limited variation, resulting in poor spatial–temporal realism and changes that are largely confined to appearance rather than underlying geometry or motion.

In contrast, video diffusion models [58, 65, 81] enable controllable modifications of geometry and motion. Unlike image diffusion models [3] that require explicit enforcement of frame-wise consistency, video diffusion models are trained on real-world video datasets and inherently capture the relationships across time and viewpoints. As a result, spatial–temporal consistency is naturally preserved during the editing process. Moreover, since video diffusion models are trained on dynamic videos paired with motion-descriptive text prompts, they enable text-based motion control, opening up new possibilities for editing that induce meaningful changes in scene geometry and motion.

To this end, we propose *MotionEditGS*, a 4D scene editing pipeline built on a video diffusion model. MotionEditGS enables motion–appearance editing while maintaining accurate geometry and motion reconstruction in the monocular video setting through three key components: (1) *semantic feature guided reconstruction* to resolve geometric and motion ambiguities, (2) *semantically anchored editing* to prevent semantic drift under dynamic edits such as motion modification, and (3) a *gradient-based feature lifting* scheme that facilitates the introduction of semantically novel content during editing.

Semantic feature guided reconstruction is employed when learning the source 4D scene from the input video by jointly optimizing a photometric loss and a semantic feature–based loss, in order to alleviate geometric and motion ambiguities inherent to the monocular video setting. Since the semantic features encode part-level information in a motion-consistent manner, this strategy improves geometry and mitigates temporal flickering, yielding a semantically structured and temporally stable dynamic Gaussian representation that serves as a robust foundation for editing.

Building on this representation, *semantically anchored editing* transforms the source 4D scene into a target 4D scene using the edited video as supervision. As in the reconstruction stage, it leverages a combined RGB–feature rendering loss, but excludes the semantic features inherited from the source scene from the optimization objective and updates only the geometric and appearance attributes. This design improves temporal coherence and structural stability under extreme motion edits by constraining the optimization to realize edits through changes in motion and geometry rather than semantic reassignment. However, when semantically novel content emerges during editing, discrepancies between the semantic feature distributions of the source and target scenes can lead to poor reconstruction in these novel regions. To address this, we adopt *gradient-based feature lifting*. Analogous to accumulating screen-space gradients, it accumulates feature-only gradients for each Gaussian and, before each densification stage, selects high-gradient Gaussians and injects novel semantic features, so that semantically new regions can be faithfully represented.

Our key technical contributions are as follows:

- We present *MotionEditGS*, a video diffusion–based 4D editing framework that enables photorealistic motion–appearance edits with accurate geometry and coherent motion from monocular videos, going beyond prior image diffusion–based approaches.
- We introduce *semantically anchored editing*, which leverages semantic features to preserve part-level semantic identity during editing, thereby improving temporal coherence under challenging motion edits.
- We propose *gradient-based feature lifting*, a feature gradient–driven mechanism that injects semantic latents into densified Gaussians, enabling faithful reconstruction of novel semantic content introduced during the edit.

2 Related Works

Diffusion Model-Based 2D Editing. Diffusion models [11, 23, 63] have become a mainstream approach for image synthesis, facilitating high-fidelity Text-to-Image (T2I) generation [46, 54, 56, 58]. SDEdit [42] introduces prompt-based editing through a noise-denoise procedure in diffusion models to preserve the structure of a guide image. Subsequent research has advanced toward personalized editing, with Textual Inversion [16] and DreamBooth [57] enabling the model to learn and synthesize novel object concepts from minimal image sets through fine-tuning. For instruction-driven editing, InstructPix2Pix [3] allows for editing via explicit natural language commands, while MimicBrush [9] facilitates reference image-based synthesis. To enable more granular spatial manipulation, ControlNet [78] further provides precise spatial control by incorporating external conditions such as pose, depth, and edges.

Extending these principles to video editing, however, introduces the complex challenge of maintaining temporal consistency. Early efforts adapt T2I architectures with fine-tuning based approaches: Tune-A-Video [72] utilizes one-shot

tuning with sparse-causal attention, while AnimateDiff [20] adds a separate temporal module to animate existing T2I models. A parallel trajectory focuses on tuning-free methodologies to enhance efficiency. Video-P2P [39] and Vid2Vid-Zero [68] expand cross-attention control mechanisms to the temporal dimension, while others enforce smoothness by leveraging motion cues [10,19] or latent-code propagation, as seen in Text2Video-Zero [31] and Pix2Video [7]. To address the lack of structural coherence in long-range sequences, some approaches incorporate explicit geometric conditions such as depth [14, 69] or pose maps [41, 75]. More recently, unified frameworks like VACE [28] integrate these disparate editing strategies into a single, cohesive generative model.

Diffusion Model-Based Scene Editing. The success of Neural Radiance Fields (NeRF) [44] representations has inspired various scene-level editing methods that leverage 2D diffusion models, transferring the rich generative priors of T2I models such as Stable Diffusion [56] to 3D domains. Specifically, Instruct-NeRF2NeRF [21] introduced an Iterative Dataset Update (IDU) scheme, in which training-view images rendered from a NeRF are edited using Instruct-Pix2Pix (IP2P) [3] and iteratively incorporated back into NeRF optimization to ensure consistency. In contrast, Instruct 3D-to-3D [29] applies Score Distillation Sampling (SDS) [50] to inject 2D diffusion priors directly into 3D scenes via gradient-based supervision. Subsequent works further extend these methods toward improved multi-view consistency [12] and text-driven localized edits [84]. With the emergence of 3D Gaussian Splatting (3DGS) [30], follow-up approaches [8, 67, 74, 79, 83] adapt these ideas to photorealistic, efficiently rendered static scene editing, taking advantage of the explicit nature of Gaussians. More recently, dynamic 3D (4D) scene editing has been explored [25,27,32,34,38, 45,59,61], with Instruct 4D-to-4D [45] extending the IDU paradigm to 4D using optical-flow and depth-based warping for temporal and multi-view consistency, and CTRL-D [22] fine-tuning IP2P on original–edited pairs to achieve consistent multi-frame edits. However, most of these works perform editing based on 2D diffusion priors, which inherently restricts their scope to appearance-based modifications and limits their ability to perform meaningful motion edits.

Dynamic Scene Representation. Early efforts in dynamic scene representation sought to capture temporal evolution by extending Neural Radiance Fields (NeRF) [44] through two dominant paradigms. Deformation-based methods learn a continuous canonical-to-observation mapping, warping a static canonical representation into a target live frame via a learned displacement field, as exemplified by D-NeRF [51], Nerfies [47], and HyperNeRF [48]. In parallel, spatiotemporal representations directly model the scene as a unified 4D (x, y, z, t) volume. To maintain computational tractability, these approaches often leverage efficient tensor decompositions or specialized feature grids to factorize the high-dimensional space [5,15,60]. Meanwhile, several works specifically address the ill-posed monocular setting by introducing motion-aware or geometry-based priors to effectively reduce reconstruction ambiguity from single-camera views [17, 18, 73]. This dual-paradigm structure persists in the dynamic 3D Gaussian Splatting (3DGS) [30] literature, where the explicit nature of Gaussians allows for more

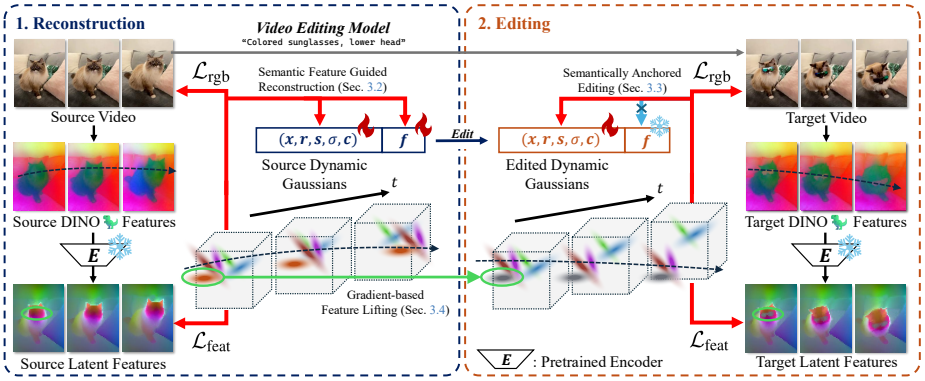

Fig. 1: Overall pipeline of our MotionEditGS. We first reconstruct a dynamic 4D scene from a monocular video using semantic-feature-augmented Gaussians. To improve geometric fidelity and temporal consistency, we jointly optimize them with both RGB ($\mathcal{L}_{\text{rgb}}$) and DINO-based latent feature rendering loss ($\mathcal{L}_{\text{feat}}$) (Sec. 3.2). Next, we edit the source video using a video diffusion model, then optimize the target 4D scene with a semantically anchored strategy that preserves the original latent identity, promoting coherent geometric and motion changes (Sec. 3.3). Finally, we apply gradient-based feature lifting to introduce new latent features and enhance the Gaussians' representational capacity for newly emerging semantic regions (highlighted by green ellipses) during densification (Sec. 3.4).

direct motion modeling. Deformation-based 3DGS approaches focus on tracking or deforming canonical Gaussians over time to preserve physical identity [36, 40, 77], while spatiotemporal methods model motion through neural fields or attributes parameterized by time to capture complex trajectories [13,35,71]. Recent monocular 3DGS methods further mitigate single-view ambiguity by introducing deformation-based regularization or hierarchical motion constraints, ensuring that the reconstructed motion remains plausible across long sequences [64, 77].

3 Method

We present MotionEditGS, a video diffusion model–based pipeline that enables text-driven editing of both motion and appearance in 4D scenes. Dynamic scene editing is the task of modifying a reconstructed 4D scene under user control, leveraging both the learned 4D representation and the original training videos [22]. Our approach establishes an end-to-end paradigm where 4D reconstruction and editing are tightly coupled—the editing stage operates on scenes whose semantics are learned during reconstruction. Given input videos, we first reconstruct the 4D scene with latents distilled from DINO features (Sec. 3.2). We then introduce *semantically anchored editing* that preserves the latent features throughout optimization (Sec. 3.3). Finally, to enable novel semantic content during editing, we propose a *gradient-based feature lifting* mechanism (Sec. 3.4). The overall pipeline is illustrated in Fig. 1.

3.1 Preliminary

Dynamic 3D Gaussian Splatting. Dynamic 3D Gaussians extend the 3DGS framework [30] to model temporally varying scenes. A representative method, Deformable 3D Gaussians (Deformable 3DGS) [77], tackles the challenge of representing motion by decoupling scene geometry and motion. Instead of assigning time-dependent parameters to each Gaussian, Deformable 3DGS defines a canonical set of 3D Gaussians and introduces a separate implicit deformation field $\mathcal{F}_\theta$ to capture temporal variations in position $\mathbf{x}$, rotation $\mathbf{r}$, and scale $\mathbf{s}$. The deformation network takes as input the canonical Gaussian center and a time coordinate t, and outputs offsets that describe its deformation over time:

$$(\delta\mathbf{x}, \delta\mathbf{r}, \delta\mathbf{s}) = \mathcal{F}_\theta(\gamma(\text{sg}(\mathbf{x})), \gamma(t)), \tag{1}$$

where $\text{sg}(\cdot)$ indicates a stop-gradient operation, and $\gamma(\cdot)$ denotes a positional encoding function. This decoupled formulation enables joint optimization of both the canonical Gaussians and the deformation network, allowing the model to reconstruct accurate dynamic geometries and achieve real-time, high-quality rendering from even monocular video inputs. In our work, we adopt the Deformable 3DGS framework as the underlying representation to model dynamic scenes from monocular observations.

VACE. VACE [28] is a unified multimodal-to-video generation and editing framework built on Diffusion Transformers (DiT) [49], integrating tasks such as Reference-to-Video generation (R2V), Video-to-Video editing (V2V), and Masked Video-to-Video editing (MV2V). It employs a *Video Condition Unit (VCU)*, defined as $V = [T; F; M]$, which combines a text prompt (T), context video frames (F), and a mask sequence (M). VACE processes these inputs via *Concept Decoupling*, which uses the mask M to separate the frame sequence F into two streams: *Reactive Frames* ($F_c = F \times M$) for pixels to be modified ((e.g., control signals such as depth, pose, or layout) and *Inactive Frames* ($F_k = F \times (1 - M)$) for pixels to be preserved (like reference image, static regions). These decoupled streams (F_c, F_k, M) are then encoded by a *Context Latent Encoder* and transformed into unified context tokens by a *Context Embedder*. During video generation, the noisy video token X is processed by the frozen main DiT backbone, while the context tokens are simultaneously processed through a separate parallel branch composed of *Context Blocks*—copied from several DiT layers—with the text tokens T provided to both branches. The output from this adapter is injected back into the main DiT layers as an additive signal, guiding the denoising trajectory of X with multimodal conditioning. This *Context Adapter Tuning* strategy enables parameter-efficient training and pluggable conditioning. In our work, we compose the R2V, and MV2V tasks of VACE to perform text- and reference-guided localized appearance and motion editing, and utilize the results for 4D scene editing.

3.2 Semantic Feature Guided Reconstruction

To enable 4D scene editing, we first reconstruct a source dynamic Gaussian representation from a monocular video sequence $\{I_t\}_{t=1}^N \in \mathbb{R}^{N\times H\times W\times 3}$, where N

denotes the number of frames in the video, and H and W represent the height and width of each frame, respectively. Recent works have shown that jointly leveraging semantic and photometric supervision during reconstruction can improve geometric fidelity and reduce temporal flickering [33, 82]. Motivated by this, we train our deformable 3D Gaussians with both photometric and semantic supervision, where we specifically adopt DINO [6] features as the semantic representation. However, these features have high dimensionality, which makes their direct use infeasible due to excessive memory consumption and degraded rendering efficiency. Following LangSplat [52], we train an autoencoder to compress the high-dimensional DINO features $\mathbf{f}_{\text{DINO}}$ into a compact latent space. The autoencoder is trained to reconstruct DINO feature patches extracted from the source video frames using a reconstruction loss:

$$\mathcal{L}_{\text{ae}} = d_{\text{ae}}(\mathbf{f}_{\text{DINO}}, \hat{\mathbf{f}}_{\text{DINO}}), \tag{2}$$

where $d_{\text{ae}}(\cdot)$ denotes a distance function combining the $\mathcal{L}_2$ and cosine similarity terms, and $\hat{\mathbf{f}}_{\text{DINO}}$ represents the reconstructed DINO feature. After training, we freeze the encoder and apply it to the DINO features extracted from each input video frame, obtaining low-dimensional latent feature maps $\{F_t\}_{t=1}^{N} \in \mathbb{R}^{N\times h\times w\times c}$, where c is the latent feature dimension, and $h = H/p$, $w = W/p$, with p denoting the DINO feature patch size.

Let $\mathcal{G}$ denote the set of Gaussians, each parameterized by standard attributes $\theta_i = (\mathbf{x}_i, \mathbf{r}_i, \mathbf{s}_i, \sigma_i, \mathbf{c}_i)$: position, rotation, scale, opacity, and color, respectively. We further augment each Gaussian with a latent feature f_i, which enables differentiable semantic rendering analogous to color rendering. During training, we render both RGB and latent features from the Gaussians and jointly optimize these attributes using a combined RGB–latent rendering loss:

$$\min_{\{\theta_i, f_i\}} \ \lambda_{\text{rgb}}\, \mathcal{L}_{\text{rgb}}(\mathcal{G}) + \lambda_{\text{feat}}\, \mathcal{L}_{\text{feat}}(\mathcal{G}) \tag{3}$$

$$\mathcal{L}_{\text{rgb}}(\mathcal{G}) = d_{\text{rgb}}(I_t, \hat{I}_t(\mathcal{G})) \tag{4}$$

$$\mathcal{L}_{\text{feat}}(\mathcal{G}) = d_{\text{feat}}(F_t, \hat{F}_t(\mathcal{G})), \tag{5}$$

where $\hat{I}_t(\mathcal{G})$ and $\hat{F}_t(\mathcal{G})$ denote the RGB image and latent feature map rendered from the training camera view at timestep t, respectively. The function $d_{\text{rgb}}(\cdot)$ combines $\mathcal{L}_1$ and $\mathcal{L}_{\text{SSIM}}$ terms, while $d_{\text{feat}}(\cdot)$ is an $\mathcal{L}_1$ loss. λ_{rgb} and λ_{feat} are weighting coefficients that balance the two objectives.

This process yields a source dynamic Gaussian representation enriched with semantic latents distilled from DINO. Since DINO features encode part-level semantics and motion consistency, their supervision helps resolve geometric and motion ambiguities inherent to monocular video reconstruction. The resulting semantically embedded Gaussians exhibit improved geometry and temporal consistency, yielding a stable representation for the subsequent editing stage.

3.3 Semantically Anchored Editing

After obtaining the source dynamic Gaussian representation enriched with latent features distilled from DINO, we leverage our *semantically anchored editing* to generate diverse target 4D scenes, including motion-edited variants, in an efficient and stable manner. We first edit the source monocular video using an off-the-shelf video editing model, guided by textual instructions and auxiliary controls such as 2D masks or reference images. In our implementation, we adopt VACE [28] to produce the edited video sequences. The edited video serves as a supervision signal to transform the source scene into the target scene.

In the editing stage, the latent features f_i^* inherited from the source scene are excluded from the rendering-loss optimization, and only the geometric and appearance parameters θ_i are updated to fit the edited video. The target scene parameters are thus optimized with a combined RGB—latent rendering loss of the form

$$\min_{\{\theta_i\}} \lambda'_{\text{rgb}} \mathcal{L}'_{\text{rgb}}(\mathcal{G}) + \lambda'_{\text{feat}} \mathcal{L}'_{\text{latent}}(\mathcal{G}; f_i^*). \tag{6}$$

where $\mathcal{L}'_{\text{rgb}}$ and $\mathcal{L}'_{\text{latent}}$ are computed using the edited video frames and their DINO-derived latent feature maps. We refer to this training strategy—where the edited video is fitted by adjusting only θ_i while using the source latent f_i^* as reference embeddings—as *semantically anchored editing*.

Without such a constraint, directly fine-tuning all Gaussian parameters on the edited video often leads to *semantic drift*: Gaussians that originally explain a particular object or part in the source scene can be reassigned to different regions or even different objects in the edited scene. This drift breaks the correspondence between the source and target scenes and manifests as temporal inconsistencies, especially in motion edits where trajectories must remain consistent over time. The semantically anchored editing strategy biases the optimization to achieve the motion edits primarily through coherent changes in motion and geometry, rather than unstable semantic reassignment. Moreover, by constraining the semantic identity of each Gaussian, the optimization focuses on a compact set of attributes, making editing computationally efficient. In practice, this semantic anchoring induces dense, part-level correspondences between the source and edited sequences, improving temporal coherence and structural stability during 4D motion editing, while still allowing diverse appearance and motion modifications through the deformable Gaussian parameters.

3.4 Gradient-based Feature Lifting

When the autoencoder trained on the input video is used to extract features from the edited video, semantically novel content introduced during editing can lead to out-of-distribution latent features. If we directly optimize the semantically anchored Gaussians using these new latents, discrepancies between the latent feature distributions of the source and target scenes often hinder faithful reconstruction of the novel regions. This calls for selectively injecting new latent features corresponding to the novel content during editing. To this end, we propose a *gradient-based feature lifting* strategy.

Table 1: Quantitative comparison with baselines on DyCheck-iPhone and DAVIS. We report *CLIP similarity*, *VBench subject consistency*, and *LOVE* (perceptual/correspondence) on *train* and *novel* views. "VACE" is a non-reconstruction reference (novel-view metrics N/A).

Dataset	Method	LOVE-Perceptual ↑		LOVE-Corresp ↑		CLIP ↑		Consistency ↑	
		Train	Novel	Train	Novel	Train	Novel	Train	Novel
DyCheck [18]	IN4D [45]	0.420	0.324	0.521	0.428	29.29	26.60	0.941	0.838
	CTRL-D [22]	0.434	0.365	0.532	0.467	29.86	26.27	0.946	0.890
	Ours	**0.456**	**0.393**	**0.553**	**0.530**	**31.26**	**27.85**	**0.950**	**0.907**
	VACE [28]	0.504	–	0.569	–	31.49	–	0.936	–
DAVIS [4]	IN4D [45]	0.508	0.294	0.572	0.306	26.60	21.06	0.950	0.598
	CTRL-D [22]	0.501	0.467	0.536	0.536	27.61	25.47	**0.962**	**0.899**
	Ours	**0.629**	**0.557**	**0.590**	**0.580**	**28.16**	**26.76**	0.929	0.882
	VACE [28]	0.639	–	0.590	–	28.70	–	0.926	–

Similar to how screen-space gradients are accumulated for Gaussian densification, we accumulate feature gradients—specifically, the gradients of the feature loss with respect to the latent features for each Gaussian, denoted as $\nabla_{f_i} d_{\text{feat}}(F_t, \hat{F}_t)$. These gradients, being derived solely from the feature loss term, provide feature-driven cues that help identify regions where novel latent features should be created. Before each Gaussian densification stage, we project Gaussians whose accumulated feature gradient exceeds a threshold τ_{lift} onto the visible camera image planes. For each projected Gaussian, we compute the average of the corresponding pixel-wise latent features from the edited video. Subsequently, each Gaussian is cloned and assigned with the computed latent features. This process effectively *lifts* the latent representation.

Since this lifting occurs at each densification iteration, the subsequent cloning and splitting of Gaussians naturally inherit the newly assigned latent features. This facilitates the overall feature lifting process while enhancing the representational capacity for novel semantic regions. The proposed gradient-based feature lifting thus enables robust 4D scene editing, particularly when semantically new objects or structures emerge during the editing stage.

4 Experiments

4.1 Experimental Setup

Datasets. For evaluation, we use DyCheck-iPhone [18] and DAVIS [4], which provide monocular videos of object-centric dynamic scenes in indoor and outdoor environments. In particular, DyCheck-iPhone provides a challenging strict monocular setting where the scene motion is significantly faster than the camera's movement, thereby presenting a demanding benchmark.

Baselines. To validate the effectiveness of our method, we conduct comparative experiments against three baselines. Two of them are previous works, Instruct-4D-to-4D (IN4D) [45] and CTRL-D [22], while the third is a self-defined baseline, referred to as RGB-only. In this baseline, reconstruction and editing are performed using only photometric supervision from the input video and the

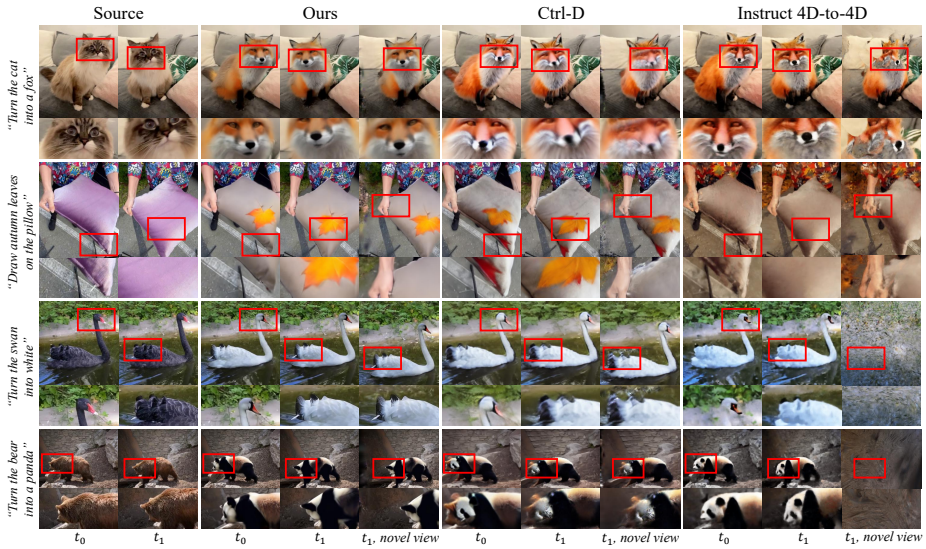

Fig. 2: Qualitative comparison with baselines. Examples include renderings from two training views and one novel view at the same time step. MotionEditGS yields sharper and more stable edits compared to the baselines. In particular, it shows markedly improved consistency in novel-view renderings.

edited video produced by VACE. This comparison allows us to evaluate how our proposed semantically anchored Gaussian-based editing pipeline improves the geometry and motion coherence of the edited scenes. Other potential baselines [27, 38, 59, 61] are not included due to the lack of public implementations, or because they specifically target multi-view settings [32].

Implementation details. We represent 4D scenes with Deformable 3D Gaussian Splatting [77]. We empirically found the camera poses provided by DyCheck [18] to be inaccurate, so we optimize the Gaussians using camera poses and a point cloud estimated from the raw video with Vipe [24]. For a fair comparison, all baselines also use the same Vipe poses and point clouds throughout reconstruction and editing. For comparison experiments, we follow CTRL-D [22] to define editing regions using masks from SAM2 [55], edit a single frame with IP2P and BlendedDiffusion [2] and mark as VACE inactive frame for the subsequent editing process. This setup facilitates a fair comparison with image-based baselines by employing the video diffusion model primarily for the temporal propagation of initial IP2P edits, thereby minimizing the influence of the model's inherent generative priors on the final results. For reference-image–based editing, we obtain an edited frame using MimicBrush [9].

We extract DINO features using DINOv3 [62] with a ViT-L/16 backbone. Features are bilinearly upsampled to the rendering resolution and used for the RGB–latent loss as well as for computing feature gradients. The autoencoder is implemented as an MLP with latent dimension $c = 16$. The source Gaussians

Table 2: Quantitative comparison with RGB-only baseline. We evaluate the effectiveness of our method by comparing our full model against the RGB-only baseline across three diverse editing scenarios. The metrics follow the same protocol as Tab. 1.

Scene / Instruction	Method	LOVE-Perceptual ↑		LOVE-Corresp ↑		CLIP ↑		Consistency ↑	
		Train	Novel	Train	Novel	Train	Novel	Train	Novel
mochi-high-five *"Turn the cat into a fox"*	RGB-only	0.461	0.410	0.606	0.547	**33.56**	**32.67**	0.955	0.927
	Ours	**0.488**	**0.477**	**0.625**	**0.609**	32.78	32.17	**0.961**	**0.947**
	VACE [28]	0.523	–	0.633	–	32.85	–	0.952	–
paper-windmill *"Turn the paper windmill into a yellow flower"*	RGB-only	0.482	0.408	0.625	0.566	**31.95**	28.74	0.959	**0.899**
	Ours	**0.490**	**0.422**	**0.633**	**0.590**	31.92	**28.92**	**0.960**	0.898
	VACE [28]	0.547	–	0.648	–	31.40	–	0.948	–
pillow *"Draw autumn leaves on the pillow"*	RGB-only	0.414	0.318	0.426	0.396	29.07	22.17	0.926	0.865
	Ours	**0.432**	**0.332**	**0.428**	**0.416**	**29.64**	**23.65**	**0.927**	**0.875**
	VACE [28]	0.441	–	0.426	–	30.22	–	0.907	–

Table 3: Quantitative comparison on computational efficiency.

Method	Compute resources	Editing time
IN4D	2 GPUs	1.5 hours
CTRL-D	1 GPU	2 hours
RGB-only	1 GPU	40 mins
Ours	1 GPU	45 mins

are optimized for 30,000 iterations; during editing we optimize for 20,000 iterations. Details of all other hyperparameters are included in the supplement. All experiments are conducted on an NVIDIA RTX 3090 GPU except for evaluation.
Metrics. While prior works often measure text alignment using CLIP similarity [53] and temporal consistency using VBench subject consistency [26], we additionally employ a more comprehensive protocol based on LOVE [66], an LMM-based metric known to align well with human annotations. We use the LOVE perception score to evaluate visual fidelity and temporal realism. To measure text-video correspondence, we compute the LOVE correspondence score between the rendered video and a synthesized target description. This target description is generated by gpt-oss [1], which is prompted with the editing instruction, demo examples, and a source description of the input video extracted by Video-LLAVA [37]. We evaluate all four metrics on videos rendered from both training views and novel views, using the standard spiral trajectory from LLFF [43] for the latter. Further details on this target description generation and spiral trajectory are provided in the supplement.

4.2 Results

Comparison with baselines. Tab. 1 reports the quantitative results for editing performance across both datasets. Our method generally achieves superior CLIP similarity and temporal consistency compared to existing baselines, while also outperforming them on the LOVE metric. Notably, our method exhibits minimal performance decay in the LOVE-Correspondence metric when transitioning to novel views, empirically validating its superior geometric stability. These improvements are qualitatively reflected in Fig. 2, where our method exhibits no-

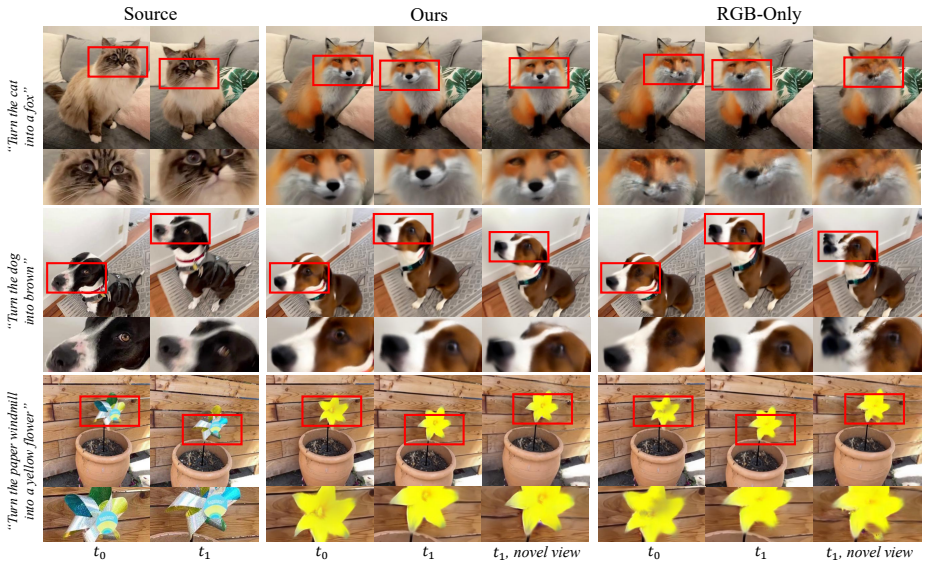

Fig. 3: Qualitative comparison with RGB-only baseline. We evaluate our framework against an RGB-only baseline using three distinct editing cases. Unlike the photometric-only approach, which suffers from significant geometric degradation and blurring in novel-view renderings, MotionEditGS maintains robust structural integrity and high visual fidelity across all viewpoints.

ticeably better structural integrity during complex edits. Tab. 3 summarizes the time and compute required for editing. The proposed pipeline requires substantially less editing time than prior works, as it eliminates the need for expensive model fine-tuning or iterative edit propagation.

Comparison with RGB-only baseline. Fig. 3 compares our proposed method against the RGB-only baseline that relies solely on photometric supervision for both reconstruction and editing. Our approach demonstrates superior visual quality; notably, while the RGB-only baseline exhibits significant geometric degradation during novel-view rendering, our model maintains a consistent level of quality across both training and novel views. This observation underscores the effectiveness of our editing methodology in enhancing geometric stability. These results are further corroborated by the quantitative results in Tab. 2, where our method shows a noticeably smaller performance gap between training and novel views across all three editing scenarios compared to the RGB-only baseline. Furthermore, Tab. 3 indicates that the proposed pipeline incurs only a marginal 12.5% increase in computational overhead relative to the RGB-only baseline.

Qualitative results of motion edits. Fig. 4 illustrates various motion editing results achieved by our method. In the *blackswan* scene, our approach demonstrates composite editing capabilities, significantly transforming the original motion via text prompts while simultaneously modifying the appearance. The

Fig. 4: Qualitative results of MotionEditGS. For each scene, we show motion and appearance edits using text prompts. The rendered frames across three time steps demonstrate that MotionEditGS achieves photorealistic edits with strong spatio-temporal consistency while preserving unrelated regions.

mochi-high-five scene further highlights the versatility of our framework, showcasing how multiple distinct motion edits can be successfully applied to the same underlying scene. All these edits yield high-fidelity outcomes, as our proposed semantically anchored editing pipeline ensures consistent geometric stability, even in the presence of rapid motion and motion blur within the edited sequences. Notably, these motion-centric transformations represent "true 4D editing" capabilities that remain unattainable for existing image-diffusion-based models and are uniquely enabled by our video diffusion-based MotionEditGS.

Ablation Studies. To validate the contribution of each component, we conduct an ablation study on each module: the Semantically Anchored Editing (SAE) and the gradient-based feature lifting (Lift) module. While the main baseline comparison evaluates the overall quality of the final edits, this ablation study measures reconstruction performance: given a fixed VACE-edited video, we assess how well each model variant can reconstruct it via the edited 4D scene. Following [45], we compute PSNR, SSIM [70], and LPIPS [80] between the VACE-edited video and the corresponding renders from the edited 4D scene. Experiments are conducted under two editing setups: a motion-editing prompt and a motion–appearance prompt that introduces novel content.

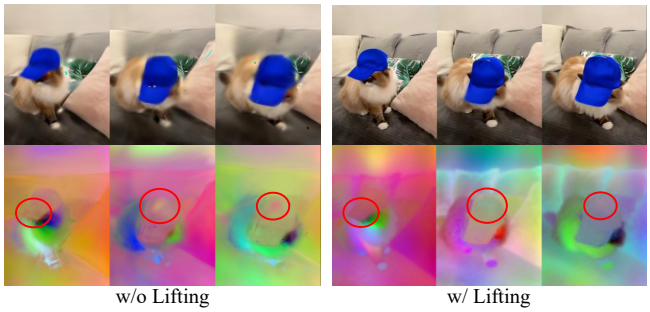

Fig. 5: Qualitative impact of gradient-based feature lifting. While SAE alone fails to form new features for semantically novel regions like the "blue cap", incorporating Lifting ensures structural integrity and prevents hollow artifacts.

Table 4: Ablation study of core modules. We evaluate reconstruction performance under motion-only ("look left") and novel-content ("blue cap") scenarios, showing the synergy between SAE and Feature Lifting.

Instruction	Method	PSNR ↑	SSIM ↑	LPIPS ↓
look left	RGB-only	30.73	0.878	0.164
	SAE	38.15	0.973	0.084
	SAE + Lift	**38.36**	**0.976**	**0.075**

Instruction	Method	PSNR ↑	SSIM ↑	LPIPS ↓
blue cap	RGB-only	34.88	0.961	0.147
	SAE	17.32	0.724	0.423
	SAE + Lift	**37.80**	**0.974**	**0.074**

Ablation results in Tab. 4 demonstrate the complementary roles of our components. In editing scenarios without novel content (e.g., *look left*), SAE contributes to significant performance gains, whereas the lifting module provides minimal impact. In contrast, when semantically novel content is introduced (e.g., *blue cap*), SAE alone struggles to adapt, causing optimization failures. This limitation is effectively addressed by the lifting module, which enables integration of novel features for superior performance. This behavior is qualitatively corroborated in Fig. 5. Without the lifting module, the model fails to form new semantic features in the *blue cap* region, resulting in hollow or incomplete artifacts. Conversely, incorporating lifting enables the robust generation of these features, ensuring structural integrity even under significant semantic changes.

5 Conclusion

We present MotionEditGS, a video diffusion-based 4D scene editing pipeline that preserves accurate scene geometry and motion through semantically anchored Gaussians and incorporates novel semantic content via gradient-based feature lifting. Our approach enables high-fidelity motion–appearance edits, going beyond the capabilities of image diffusion-based baselines. Extensive experiments demonstrate that MotionEditGS achieves superior visual quality, stronger text alignment, and more accurate 4D scene reconstruction, establishing a solid foundation for future research in video diffusion–driven 4D scene editing.

Acknowledgments

This work was supported by the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. RS-2024-00457882, AI Research Hub Project), and by the InnoCORE program of the Ministry of Science and ICT(N10250156).

References

1. Agarwal, S., Ahmad, L., Ai, J., Altman, S., Applebaum, A., Arbus, E., Arora, R.K., Bai, Y., Baker, B., Bao, H., et al.: gpt-oss-120b & gpt-oss-20b model card. arXiv preprint arXiv:2508.10925 (2025)
2. Avrahami, O., Lischinski, D., Fried, O.: Blended diffusion for text-driven editing of natural images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18208–18218 (2022)
3. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: Learning to follow image editing instructions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18392–18402 (2023)
4. Caelles, S., Pont-Tuset, J., Perazzi, F., Montes, A., Maninis, K.K., Van Gool, L.: The 2019 davis challenge on vos: Unsupervised multi-object segmentation. arXiv preprint arXiv:1905.00737 (2019)
5. Cao, A., Johnson, J.: Hexplane: A fast representation for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 130–141 (2023)
6. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
7. Ceylan, D., Huang, C.H.P., Mitra, N.J.: Pix2video: Video editing using image diffusion. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 23206–23217 (2023)
8. Chen, M., Laina, I., Vedaldi, A.: Dge: Direct gaussian 3d editing by consistent multi-view editing. In: European Conference on Computer Vision. pp. 74–92. Springer (2024)
9. Chen, X., Feng, Y., Chen, M., Wang, Y., Zhang, S., Liu, Y., Shen, Y., Zhao, H.: Zero-shot image editing with reference imitation. Advances in Neural Information Processing Systems **37**, 84010–84032 (2024)
10. Cong, Y., Xu, M., Simon, C., Chen, S., Ren, J., Xie, Y., Perez-Rua, J.M., Rosenhahn, B., Xiang, T., He, S.: Flatten: optical flow-guided attention for consistent text-to-video editing. arXiv preprint arXiv:2310.05922 (2023)
11. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in neural information processing systems **34**, 8780–8794 (2021)
12. Dong, J., Wang, Y.X.: Vica-nerf: View-consistency-aware 3d editing of neural radiance fields. Advances in Neural Information Processing Systems **36**, 61466–61477 (2023)
13. Duan, Y., Wei, F., Dai, Q., He, Y., Chen, W., Chen, B.: 4d-rotor gaussian splatting: towards efficient novel view synthesis for dynamic scenes. In: ACM SIGGRAPH 2024 Conference Papers. pp. 1–11 (2024)
14. Esser, P., Chiu, J., Atighehchian, P., Granskog, J., Germanidis, A.: Structure and content-guided video synthesis with diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7346–7356 (2023)

15. Fridovich-Keil, S., Meanti, G., Warburg, F.R., Recht, B., Kanazawa, A.: K-planes: Explicit radiance fields in space, time, and appearance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12479–12488 (2023)
16. Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A.H., Chechik, G., Cohen-Or, D.: An image is worth one word: Personalizing text-to-image generation using textual inversion. arXiv preprint arXiv:2208.01618 (2022)
17. Gao, C., Saraf, A., Kopf, J., Huang, J.B.: Dynamic view synthesis from dynamic monocular video. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. p. 5712–5721 (2021)
18. Gao, H., Li, R., Tulsiani, S., Russell, B., Kanazawa, A.: Monocular dynamic view synthesis: A reality check. Advances in Neural Information Processing Systems **35**, 33768–33780 (2022)
19. Geyer, M., Bar-Tal, O., Bagon, S., Dekel, T.: Tokenflow: Consistent diffusion features for consistent video editing. arXiv preprint arXiv:2307.10373 (2023)
20. Guo, Y., Yang, C., Rao, A., Liang, Z., Wang, Y., Qiao, Y., Agrawala, M., Lin, D., Dai, B.: Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. arXiv preprint arXiv:2307.04725 (2023)
21. Haque, A., Tancik, M., Efros, A.A., Holynski, A., Kanazawa, A.: Instruct-nerf2nerf: Editing 3d scenes with instructions. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 19740–19750 (2023)
22. He, K., Wu, C.H., Gilitschenski, I.: Ctrl-d: Controllable dynamic 3d scene editing with personalized 2d diffusion. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26630–26640 (2025)
23. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
24. Huang, J., Zhou, Q., Rabeti, H., Korovko, A., Ling, H., Ren, X., Shen, T., Gao, J., Slepichev, D., Lin, C.H., et al.: Vipe: Video pose engine for 3d geometric perception. arXiv preprint arXiv:2508.10934 (2025)
25. Huang, Y.H., Sun, Y.T., Yang, Z., Lyu, X., Cao, Y.P., Qi, X.: Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4220–4230 (2024)
26. Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., et al.: Vbench: Comprehensive benchmark suite for video generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21807–21818 (2024)
27. Iqbal, H., Karim, N., Khalid, U., Farooq, A., Zhong, Z., Chen, C., Hua, J.: Psf-4d: A progressive sampling framework for view consistent 4d editing. arXiv preprint arXiv:2503.11044 (2025)
28. Jiang, Z., Han, Z., Mao, C., Zhang, J., Pan, Y., Liu, Y.: Vace: All-in-one video creation and editing. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17191–17202 (2025)
29. Kamata, H., Sakuma, Y., Hayakawa, A., Ishii, M., Narihira, T.: Instruct 3d-to-3d: Text instruction guided 3d-to-3d conversion. arXiv preprint arXiv:2303.15780 (2023)
30. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
31. Khachatryan, L., Movsisyan, A., Tadevosyan, V., Henschel, R., Wang, Z., Navasardyan, S., Shi, H.: 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)
32. Kwon, J., Cho, H., Kim, J.: Efficient dynamic scene editing via 4d gaussian-based static-dynamic separation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 26855–26865 (2025)
33. Labe, I., Issachar, N., Lang, I., Benaim, S.: Dgd: Dynamic 3d gaussians distillation. In: European Conference on Computer Vision. pp. 361–378. Springer (2024)
34. Lee, D.I., Doh, H., Chi, S., Duan, R., Kim, S., Ramani, K.: Dynamic-editor: Training-free text-driven 4d scene editing with multimodal diffusion transformer. arXiv preprint arXiv:2512.00677 (2025)
35. Li, Z., Chen, Z., Li, Z., Xu, Y.: Spacetime gaussian feature splatting for real-time dynamic view synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8508–8520 (2024)
36. Liang, Y., Khan, N., Li, Z., Nguyen-Phuoc, T., Lanman, D., Tompkin, J., Xiao, L.: Gaufre: Gaussian deformation fields for real-time dynamic novel view synthesis. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 2642–2652. IEEE (2025)
37. Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., Yuan, L.: Video-llava: Learning united visual representation by alignment before projection. In: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing. pp. 5971–5984 (2024)
38. Liu, L., Wang, C., Chen, Z., Xu, D.: 4dgs-craft: Consistent and interactive 4d gaussian splatting editing. arXiv preprint arXiv:2510.01991 (2025)
39. Liu, S., Zhang, Y., Li, W., Lin, Z., Jia, J.: 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)
40. Luiten, J., Kopanas, G., Leibe, B., Ramanan, D.: Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In: 2024 International Conference on 3D Vision (3DV). pp. 800–809. IEEE (2024)
41. Ma, Y., He, Y., Cun, X., Wang, X., Chen, S., Li, X., Chen, Q.: Follow your pose: Pose-guided text-to-video generation using pose-free videos. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 4117–4125 (2024)
42. Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.Y., Ermon, S.: Sdedit: Guided image synthesis and editing with stochastic differential equations. arXiv preprint arXiv:2108.01073 (2021)
43. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (ToG) **38**(4), 1–14 (2019)
44. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
45. Mou, L., Chen, J.K., Wang, Y.X.: Instruct 4d-to-4d: Editing 4d scenes as pseudo-3d scenes using 2d diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20176–20185 (2024)
46. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv preprint arXiv:2112.10741 (2021)
47. Park, K., Sinha, U., Barron, J.T., Bouaziz, S., Goldman, D.B., Seitz, S.M., Martin-Brualla, R.: Nerfies: Deformable neural radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5865–5874 (2021)

48. Park, K., Sinha, U., Hedman, P., Barron, J.T., Bouaziz, S., Goldman, D.B., Martin-Brualla, R., Seitz, S.M.: Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. arXiv preprint arXiv:2106.13228 (2021)
49. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4195–4205 (2023)
50. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
51. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-nerf: Neural radiance fields for dynamic scenes. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10318–10327 (2021)
52. Qin, M., Li, W., Zhou, J., Wang, H., Pfister, H.: Langsplat: 3d language gaussian splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20051–20060 (2024)
53. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021)
54. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 **1**(2), 3 (2022)
55. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., et al.: Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714 (2024)
56. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: 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)
57. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22500–22510 (2023)
58. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems **35**, 36479–36494 (2022)
59. Shao, R., Sun, J., Peng, C., Zheng, Z., Zhou, B., Zhang, H., Liu, Y.: Control4d: Efficient 4d portrait editing with text. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4556–4567 (2024)
60. Shao, R., Zheng, Z., Tu, H., Liu, B., Zhang, H., Liu, Y.: Tensor4d: Efficient neural 4d decomposition for high-fidelity dynamic reconstruction and rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16632–16642 (2023)
61. Shi, J.C., Su, C., Wang, J., Shamir, A., Wang, M.: Mono4deditor: Text-driven 4d scene editing from monocular video via point-level localization of language-embedded gaussians. arXiv preprint arXiv:2510.09438 (2025)
62. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., et al.: Dinov3. arXiv preprint arXiv:2508.10104 (2025)
63. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International conference on machine learning. pp. 2256–2265. pmlr (2015)

64. Stearns, C., Harley, A., Uy, M., Dubost, F., Tombari, F., Wetzstein, G., Guibas, L.: Dynamic gaussian marbles for novel view synthesis of casual monocular videos. In: SIGGRAPH Asia 2024 Conference Papers. p. 1–11 (2024)
65. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
66. Wang, J., Duan, H., Jia, Z., Zhao, Y., Yang, W.Y., Zhang, Z., Chen, Z., Wang, J., Xing, Y., Zhai, G., et al.: Love: Benchmarking and evaluating text-to-video generation and video-to-text interpretation. arXiv preprint arXiv:2505.12098 (2025)
67. Wang, J., Fang, J., Zhang, X., Xie, L., Tian, Q.: Gaussianeditor: Editing 3d gaussians delicately with text instructions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20902–20911 (2024)
68. Wang, W., Jiang, Y., Xie, K., Liu, Z., Chen, H., Cao, Y., Wang, X., Shen, C.: Zero-shot video editing using off-the-shelf image diffusion models. arXiv preprint arXiv:2303.17599 (2023)
69. Wang, X., Yuan, H., Zhang, S., Chen, D., Wang, J., Zhang, Y., Shen, Y., Zhao, D., Zhou, J.: Videocomposer: Compositional video synthesis with motion controllability. Advances in Neural Information Processing Systems **36**, 7594–7611 (2023)
70. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
71. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Wang, X.: 4d gaussian splatting for real-time dynamic scene rendering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20310–20320 (2024)
72. Wu, J.Z., Ge, Y., Wang, X., Lei, S.W., Gu, Y., Shi, Y., Hsu, W., Shan, Y., Qie, X., Shou, M.Z.: 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)
73. Xian, W., Huang, J.B., Kopf, J., Kim, C.: Space-time neural irradiance fields for free-viewpoint video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. p. 9421–9431 (2021)
74. Xu, T., Chen, J., Chen, P., Zhang, Y., Yu, J., Yang, W.: Tiger: Text-instructed 3d gaussian retrieval and coherent editing. arXiv preprint arXiv:2405.14455 (2024)
75. Xu, Z., Zhang, J., Liew, J.H., Yan, H., Liu, J.W., Zhang, C., Feng, J., Shou, M.Z.: Magicanimate: Temporally consistent human image animation using diffusion model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1481–1490 (2024)
76. Yang, Z., Yang, H., Pan, Z., Zhang, L.: Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In: International Conference on Learning Representations (ICLR) (2024)
77. Yang, Z., Gao, X., Zhou, W., Jiao, S., Zhang, Y., Jin, X.: Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20331–20341 (2024)
78. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3836–3847 (2023)
79. Zhang, Q., Xu, Y., Wang, C., Lee, H.Y., Wetzstein, G., Zhou, B., Yang, C.: 3ditscene: Editing any scene via language-guided disentangled gaussian splatting. arXiv preprint arXiv:2405.18424 (2024)

80. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)
81. Zheng, Z., Peng, X., Yang, T., Shen, C., Li, S., Liu, H., Zhou, Y., Li, T., You, Y.: Open-sora: Democratizing efficient video production for all. arXiv preprint arXiv:2412.20404 (2024)
82. Zhou, S., Chang, H., Jiang, S., Fan, Z., Zhu, Z., Xu, D., Chari, P., You, S., Wang, Z., Kadambi, A.: Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21676–21685 (2024)
83. Zhuang, J., Kang, D., Cao, Y.P., Li, G., Lin, L., Shan, Y.: Tip-editor: An accurate 3d editor following both text-prompts and image-prompts. ACM Transactions on Graphics (TOG) **43**(4), 1–12 (2024)
84. Zhuang, J., Wang, C., Lin, L., Liu, L., Li, G.: Dreameditor: Text-driven 3d scene editing with neural fields. In: SIGGRAPH Asia 2023 Conference Papers. pp. 1–10 (2023)