

ReFlow: Self-correction Motion Learning for Dynamic Scene Reconstruction

Yanzhe Liang¹ Ruijie Zhu¹ Hanzhi Chang¹ Zhuoyuan Li¹ Jiahao Lu¹ Tianzhu Zhang^{1,2*}

¹University of Science and Technology of China

²National Key Laboratory of Deep Space Exploration, Deep Space Exploration Laboratory

Abstract

We present ReFlow, a unified framework for monocular dynamic scene reconstruction that learns 3D motion in a novel self-correction manner from raw video. Existing methods often suffer from incomplete scene initialization for dynamic regions, leading to unstable reconstruction and motion estimation, which often resorts to external dense motion guidance and constrain the reconstruction of dynamic components. However, this introduces additional complexity and potential error propagation. To address these issues, ReFlow integrates a Complete Canonical Space Construction module for enhanced initialization of both static and dynamic regions, and a Separation-Based Dynamic Scene Modeling module that decouples static and dynamic components for targeted motion supervision. The core of ReFlow is a novel self-correction flow matching mechanism, consisting of Full Flow Matching to align 3D scene flow with time-varying 2D observations, and Camera Flow Matching to enforce multi-view consistency for static objects. Together, these modules enable robust and accurate dynamic scene reconstruction. Extensive experiments across diverse scenarios demonstrate that ReFlow achieves superior reconstruction quality and robustness, establishing a novel self-correction paradigm for monocular 4D reconstruction. Project page: [this https URL](https://github.com/yangzhe1999/ReFlow)

1. Introduction

Monocular dynamic scene reconstruction is a key task in 3D computer vision and graphics, aiming to recover time-varying 3D structure from 2D video sequences. Despite its broad applications in spatial computing [11–14, 33], autonomous driving [9, 10, 61], and AR/VR [24], accurately reconstructing dynamic scenes remains a significant challenge due to the complexity of spatiotemporal modeling and limited observations.

Recent advances in neural rendering [1–4, 28, 43, 44, 59]

*Corresponding author.

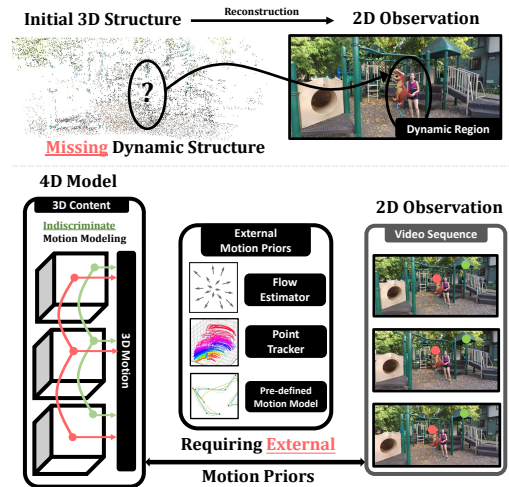


Figure 1. Typical Challenges in monocular dynamic scene reconstruction. **Top:** Incomplete initialization for dynamic regions: the initial 3D structure from SfM often misses dynamic components and initializes Gaussians without separating **static points (green)** from **dynamic points (red)**, leading to an entangled and incomplete representation. **Bottom:** To compensate, existing methods frequently resort to external dense motion guidance to constrain and stabilize the reconstruction of dynamic regions.

have significantly improved static scene reconstruction, particularly through 3D Gaussian Splatting (3DGS) [28]. Building on this success, several works have extended these techniques to dynamic scenes [36, 42, 55, 62, 63], typically representing scenes with time-varying 3D gaussians to capture temporal changes. Despite promising results, these approaches still struggle with complex scene structures and motion patterns. As shown in Fig. 1, current approaches typically initialize 3DGS primitives using classical Structure-from-Motion (SfM) methods like COLMAP, which are designed for static scene reconstruction but not suitable for dynamic scenes. The initial point cloud produced by COLMAP often lacks dynamic regions and is used to initialize Gaussians without distinguishing static from dynamic content. This incomplete and entangled initialization makes subsequent reconstruction and motion estimation difficult and

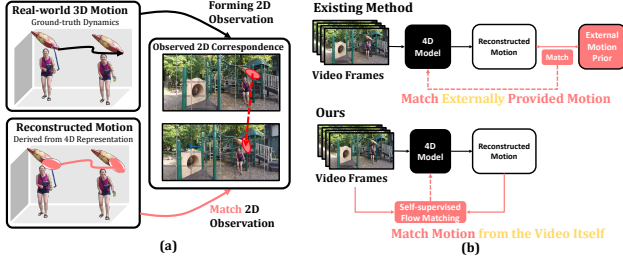


Figure 2. **Motivation of Self-correction Flow Matching.** (a) We start with a simple observation: 2D observations, such as the shifting balloon, are caused by 3D motion. Accurate reconstructed 3D Motion should naturally align with these visible changes. (b) Unlike previous methods that use external motion priors to supervise 3D motion, we instead use raw **video as motion supervision** through a self-correction flow matching mechanism to directly align predicted 3D motion projections with 2D frame differences.

unstable. As a result, many recent works resort to external dense motion supervision (optical flow [7, 19, 51, 54, 67, 68], tracking [31, 52]) to ensure the accurate reconstruction of scene dynamics. These motion cues are typically generated by separate pipelines [15, 22, 25, 26, 57, 58] and are treated as pseudo ground-truth, enforcing hard constraints that force the reconstructed motion to align with potentially flawed external estimates. This introduces additional complexity and potential error propagation into the reconstruction process making the reconstruction highly sensitive to the accuracy of external estimators. For instance, MotionGS[68] decouples optical flow into motion and camera flow, effectively constraining the motion of 3DGS. But it still depends on externally extracted motion flow labels, making its performance tightly coupled with the quality of the flow estimator and prone to failure when the estimator is inaccurate. This makes us wonder: *is it possible to unlock 4D dynamic scenes purely from 2D observations, without external motion guidance?*

To answer this question, we revisit the 4D dynamic scenes from both structural and motion perspectives. Structurally, it is crucial to initialize both static and dynamic regions and to model them with separate representations. Such initialization and disentanglement reduce early-stage ambiguity and yield more stable optimization, while also preserving the potential to apply different motion constraints on different regions. In terms of motion, we identify a natural 3D-2D motion consistency that can be exploited for self-correction motion constraints, without the need for aligning with external motion priors. As shown in Fig. 2(a), differences between consecutive video frames originate from real-world 3D motion. *If 3D motion is accurately reconstructed, it should align with the pixel-wise differences across frames.* This insight leads to our core idea: we directly supervise 3D motion by measuring how well it explains frame-to-frame variations. Specifically, we project 3D motion into a 2D flow

field to warp one frame toward the next. The misalignment between the warped frame and the real image provides a natural supervision signal to correct the reconstructed motion, creating a self-correction loop in Fig. 2(b) where motion is learned and refined by a direct and natural objective of re-rendering future or past frames, rather than imitating a potentially flawed external motion estimations.

We instantiate these considerations in **ReFlow**, a unified dynamic scene reconstruction framework that integrates three key components. Firstly, we integrate a **Complete Canonical Space Construction** module that leverages geometry foundation models [32, 41, 53, 65] to initialize both static and dynamic regions in a coarse-to-fine manner. Secondly, a **Separation-Based Dynamic Scene Modeling** module independently models static and dynamic parts of the scene, aggregating spatiotemporal information for static regions and enabling flexible motion for dynamic elements. This decoupling also allows us to apply region-specific motion constraints in the next motion learning stage. On top of these, we propose a natural **Self-correction Flow Matching** mechanism that learns 3D motion directly from raw video by enforcing consistency between the projected 3D motion and observed 2D frame-to-frame differences. It incorporates two complementary constraints: *Full Flow Matching*, which aligns the joint camera-scene motion with entire image differences, and *Camera Flow Matching*, which enforces consistency between camera-induced motion and the observed changes in static regions separated by the separation module. By coupling these constraints within a self-corrective loop, ReFlow derives region-appropriate motion supervision from the video. Through the integration of initialization, static-dynamic decoupling, and self-correction motion learning, ReFlow presents a unified framework for high-quality monocular dynamic scene reconstruction without dense motion guidance. Our main contributions are summarized as follows:

1. We propose ReFlow, a unified framework that addresses incomplete initialization, entanglement, and the overreliance on external motion guidance by combining complete canonical-space initialization, static-dynamic separation, and self-corrective motion learning into a coherent pipeline for high-quality monocular 4D reconstruction.
2. We formulate and implement a simple motion constraint mechanism for dynamic scene reconstruction in a entirely self-correction way. Instead of relying on external motion priors, our Self-correction Flow Matching directly aligns 3D motion with video frame differences, making video itself as the direct motion supervision.
3. Through extensive experiments, we demonstrate ReFlow’s superior performance in diverse dynamic scenes, significantly improving reconstruction accuracy and quality, while effectively reducing reliance on external dense motion guidance.

2. Related Work

2.1. Dynamic Scene Reconstruction

With rapid advances in neural rendering, dynamic scene reconstruction has witnessed significant progress in recent years. Early research primarily extended static NeRF to dynamic scenes along two technical routes: one approach [6, 16, 47] constructs a 3D canonical space to represent scene content, complemented by independent motion modules for temporal modeling; the other [35, 56] directly builds complete 4D spatiotemporal representations, modeling time-varying spatial structures in a unified framework. Despite establishing foundations, NeRF-based methods faced limitations from their implicit nature, including computational complexity and slow inference. 3D Gaussian Splatting’s success introduced a more efficient paradigm for dynamic scenes. Recent approaches adopt 3DGS representations for canonical space with temporal deformation, modeled either implicitly through neural networks [21, 23, 38, 62] and feature planes [55], or explicitly via velocity fields [8] and various trajectory models [5, 20, 27, 36, 40]. In parallel, alternative approaches [63] directly model time-varying 4D Gaussian distributions, offering a different perspective on the problem. Building on recent advances in dynamic Gaussian Splatting, we take a closer look at how the different roles of dynamic and static components, as well as complete canonical space construction strategies, can provide a more reliable foundation for learning motion constraints in a self-correction manner.

2.2. Motion Regularization in 4D Scene

Motion constraint mechanisms play a decisive role in dynamic scene reconstruction, directly affecting the temporal consistency and physical plausibility of reconstruction results. Existing methods primarily constrain scene dynamics by introducing external 2D priors, broadly categorized into optical flow-based and trajectory-based approaches. Optical flow-based methods [19, 35, 54, 67, 68] derive inherent optical flow from scene representations and establish correspondences with pre-computed 2D optical flow priors [22, 48, 57], constraining 3D motion through photometric consistency loss. While capturing fine-grained motion, they depend heavily on flow estimation quality and demonstrate sensitivity to rapid movements and occlusions. Trajectory-based methods [31, 52] lift 2D point tracking [25, 26] to 3D space to guide Gaussian trajectories. While these methods show advantages in handling complex deformations, they typically require stable estimated point trajectories and perform poorly in textureless or rapidly moving regions. Unlike these approaches, our method employs a self-correction strategy extracting motion information directly from raw inputs, ensuring reconstruction quality while simplifying the process.

3. Methodology

3.1. Overview

Given a monocular video sequence $\mathcal{V} = \{I_i\}_{i=1}^N$ with extrinsics $\mathbf{T} = \{\mathbf{T}_i \in \mathbb{R}^{3 \times 4}\}_{i=1}^N$ and intrinsics $\mathbf{K}$, our goal is to reconstruct a dynamic 3D scene and learn its motion in a fully self-correcting manner. We particularly target two practical challenges in monocular 4D reconstruction: (i) the lack of reliable initialization for dynamic regions and the need for decoupled representations of static and dynamic content, and (ii) the strong dependence on external dense motion guidance. To tackle these issues, we propose ReFlow, as illustrated in Fig. 3. It consists of three key modules that together enable proper scene initialization and decoupling, along with a novel self-correction of motion directly from video. The details of each module are presented in the following sections.

3.2. Dynamic Scene Initialization and Modeling

3.2.1. Complete Canonical Space Construction

To address the challenge of obtaining a proper initialization for dynamic scenes, we first build a complete canonical space that consistently captures both static and dynamic regions across the entire video. We leverage a geometry foundation model [65] to regress per-pixel 3D coordinates from image pairs and enforce geometric consistency over a global connectivity graph to obtain a unified point cloud. Directly applying this to videos faces two challenges: (1) memory cost grows quadratically ($O(N^2)$) with frame number; (2) limited co-visibility across distant views hinders alignment. We address these via a hierarchical coarse-to-fine strategy.

Coarse-to-fine Alignment. Given a video sequence $\{I_i\}_{i=1}^N$, we divide it into K non-overlapping clips $\{S_k\}_{k=1}^K$ and select the first frame I_{s_k} of each clip as a keyframe. We build a smaller connectivity graph G_K among keyframes to establish a coarse globally consistent structure. For each keyframe pair (I_a, I_b) , the geometry foundation model regresses 3D coordinates $\mathbf{X}_a^a, \mathbf{X}_a^b$ in a ’s coordinate system, alongside confidence maps $\mathbf{C}_a^a, \mathbf{C}_a^b$. Global alignment is obtained by optimizing keyframe camera poses K_{pose} , intrinsics K_{intr} , and depth maps K_{depth} using the 3D geometry consistency constraint $\mathcal{L}_{\text{align}}$ from [65]:

$$\min_{K_{\text{pose}}, K_{\text{intr}}, K_{\text{depth}}} \sum_{(a,b) \in E_K} \mathcal{L}_{\text{align}}(\mathbf{X}_a^a, \mathbf{X}_a^b, \mathbf{C}_a^a, \mathbf{C}_a^b)$$

We then recover geometry for all frames via local alignment within each clip S_k , using a similar loss initialized from keyframe alignment. This hierarchical design enforces both local consistency between temporally close frames and global consistency across the entire sequence. Further details are provided in Appendix A.1.

Static-Dynamic Disentanglement. We separate static and dynamic elements using dynamic masks M_i^{dyn} from the

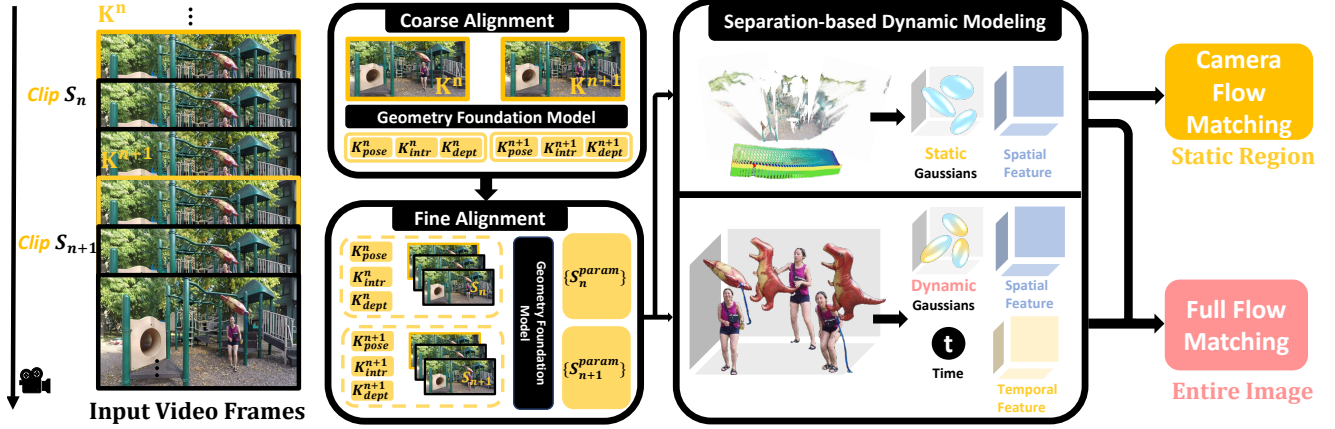


Figure 3. **Overview of ReFlow.** We start by constructing a complete canonical space (Sec. 3.2.1), which includes both static and dynamic components, ensuring a reliable 3D scene initialization. Next, we disentangle these elements using spatial and spatiotemporal feature planes (Sec. 3.2.2), providing a structured representation that separately handles static and dynamic regions. This preparation allows us to introduce targeted motion constraints (Sec. 3.3): Full Flow supervises motion across the entire scene, while Camera Flow enforces consistency in static regions, enabling the self-correction learning mechanism for accurate 3D motion reconstruction.

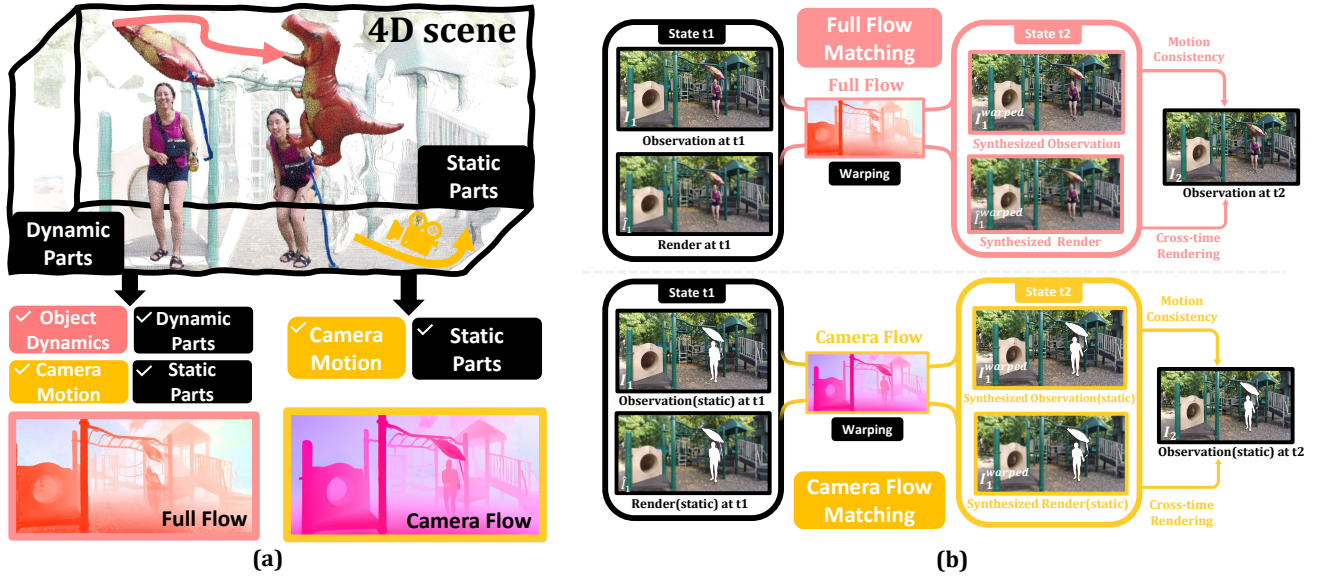


Figure 4. **Self-correction flow matching mechanism.** (a) **Different Motion and Flow in the 4D Scene.** Static areas move only due to camera motion (camera flow), while dynamic areas involve both camera and object motion (full flow). Accurate motion learning requires region-specific flow supervision. (b) **Self-correction flow matching.** We apply full flow to warp the entire image from state t_1 to state t_2 and compare with the real observation, validating overall motion. Camera flow is used similarly but only on static regions, ensuring their stability. Together, these provide a complementary self-correction signal for 3D motion learning.

geometry foundation model for each frame i in clip S_k . Each frame is back-projected to 3D using depth maps S_k^{depth} , camera intrinsics S_k^{intr} , and poses S_k^{pose} . Aggregating all clips, we obtain separate point clouds for static content $\mathbf{P}^{3D, \text{stat}} = \{\mathbf{P}_i^{3D, \text{stat}}\}_{i=1}^N$ and dynamic regions $\mathbf{P}^{3D, \text{dyn}} = \{\mathbf{P}_i^{3D, \text{dyn}}\}_{i=1}^N$, which enable subsequent region-specific modeling of static and dynamic components.

3.2.2. Separation-Based Dynamic Scene Modeling

With the aligned point clouds $\mathbf{P}^{3D, \text{stat}}$ and $\mathbf{P}^{3D, \text{dyn}}$, we adapt separate representations for static and dynamic components, enabling motion-aware modeling and preparing the representation for motion-specific constraints.

Separate Initialization. To initialize our Gaussian models, we process the point cloud sequences into unified represen-

tations. For dynamic objects, we select a reference frame with maximal mask coverage and color diversity to comprehensively represent $\mathbf{P}^{3D, \text{dyn}}$. For static regions, point clouds from current and keyframes are fused into $\mathbf{P}_{\text{fusion}}^{3D, \text{stat}}$ to increase viewpoint coverage.

Modeling Static and Dynamic Components. Static components use tri-plane spatial features $F_s = \{F^{xy}, F^{xz}, F^{yz}\}$ encoding position-dependent information. Dynamic components add temporal planes $F_t = \{F^{xt}, F^{yt}, F^{zt}\}$ for motion modeling:

$$G_s : \{\mu_s, s_s, q_s, \sigma_s, c_s\} = D_s(x, y, z; F_s)$$

$$G_d : \{\mu_d(t), s_d, q_d(t), \sigma_d, c_d\} = D_d(x, y, z, t; F_s, F_t)$$

where D_s, D_d decode static and dynamic Gaussian parameters respectively. We provide detailed explanation in Appendix A.2. This representation cleanly separates static structures from time-varying dynamic parts and provides exactly the representation needed for the motion-specific constraints introduced next in our self-correction flow matching module.

3.3. Self-correction Flow Matching

With coherent initialization and static–dynamic separation, the remaining goal is to learn 3D motion consistent with the actual temporal evolution of the video. Instead of relying on external optical flow or tracking, we aim to explore the capability of a motion learning mechanism derived purely from the raw video.

3.3.1. Formulation

Formulate 3D–2D Alignment as Warping. Given consecutive frames I_1 and I_2 , an accurate 3D motion should explain the observed image changes. If the estimated motion reflects real-world dynamics, it should induce a 2D flow that warps I_1 to closely match I_2 . This 3D–2D alignment can be realized as a simple pixel-wise warping: projecting 3D motion into 2D flow and warping one frame toward the other, then aligning the warped image with the target frame. We name this process **self-correction flow matching**, where the reconstructed 3D motion aligns with the observed visual changes between input video frames rather than external motion estimation, using the video itself as a self-correction signal for learning 3D motion.

Constructing Flows for Diverse Motion Types. As our self-correction matching relies on deriving 2D flow, and following prior work [7, 19, 68], we distinguish two types of motion—object motion and camera ego-motion and define two corresponding flow fields to separately capture their effects (see Fig. 4(a)). *Full Flow* represents the total image-space displacement caused by both object motion and camera movement, corresponding to changes across the entire frame. Given the scene at frames t_1 and t_2 , represented by 3D Gaussians G_{t_1}, G_{t_2} , and the corresponding camera parameters

$\mathbf{P}_1, \mathbf{P}_2$, we synthesize Full Flow as

$$\mathbf{F}_{\text{full}} = \text{FlowRender}(G_{t_1}, G_{t_2}, \mathbf{P}_1, \mathbf{P}_2), \quad (1)$$

where $\mathbf{P}_i = (\mathbf{K}_i, \mathbf{T}_i)$ denotes the intrinsic and extrinsic matrices $\mathbf{K}_i$ and $\mathbf{T}_i$ respectively. *Camera Flow* isolates the 2D motion induced solely by camera movement, assuming the scene is static. As static regions should remain geometrically stable, Camera Flow provides an unambiguous and clean supervision signal in static regions where observed changes arise only from viewpoint shifts, ensuring structural consistency and avoiding false motion supervision or dynamic overfitting. We compute it by warping the static regions of the scene at t_1 according to the camera’s pose change:

$$\mathbf{F}_{\text{cam}} = \text{CamFlowRender}(G_{t_1}, \mathbf{P}_1, \mathbf{P}_2), \quad (2)$$

Implementation details for both flow renderings are provided in Appendix A.3. In the following sections, we design separate matching mechanisms for each flow type to impose region-specific constraints.

3.3.2. Full Flow Matching

Specifically, we define two constraints by warping the input and rendered images, offering supervision from different perspectives.

Motion Consistency Constraint. We warp the image I_1 from time t_1 to t_2 using the synthesized flow field $\mathbf{F}$, producing I_1^{warped} . Ideally, if the motion modeling is accurate, I_1^{warped} should be consistent with the real image I_2 . Our motion consistency loss L_{mc} verifies if the derived 2D motion accurately reflects real-world visual changes:

$$\mathcal{L}_{mc} = \mathcal{L}_{\text{photo}}(I_1^{\text{warped}}, I_2) \quad (3)$$

where $\mathcal{L}_{\text{photo}}$ combines L1 distance and structural similarity metrics [28]. This warping-based supervision provides an explicit correction signal: whenever the 3D motion is inaccurate, the warped frame will deviate from the observation, pushing the model toward the correct motion field.

Cross-Time Rendering Constraint. We further warp the rendered image $\hat{I}_1$ at time t_1 through the same flow field, creating a cross-time rendering constraint L_{cr} :

$$\mathcal{L}_{cr} = \mathcal{L}_{\text{photo}}(\hat{I}_1^{\text{warped}}, I_2) \quad (4)$$

This enables predicting scene appearance at t_2 from its state at t_1 , reinforcing temporal consistency. The complete Full Flow Matching objective combines both constraints:

$$\mathcal{L}_{\text{fullflow}} = \lambda_{mc} \mathcal{L}_{mc} + \lambda_{cr} \mathcal{L}_{cr} \quad (5)$$

3.3.3. Camera Flow Matching

Camera Flow Matching refers to a self-correction constraint applied to static regions where image changes result solely

Table 1. Quantitative comparison on Nvidia Monocular dataset. We report PSNR/SSIM/LPIPS per scene; the last block shows the mean across all available scenes (including dynamicFace when available).

Method	Balloon1			Balloon2			Jumping			Playground		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3DGS [28]	17.71	0.407	0.473	20.62	0.500	0.431	14.45	0.394	0.385	15.75	0.341	0.493
STGS [36]	20.36	0.746	0.196	23.12	0.881	0.124	20.82	0.631	0.187	19.23	0.842	0.151
Ex4DGS [30]	14.69	0.351	0.503	16.29	0.471	0.457	18.93	0.661	0.321	14.16	0.442	0.437
Efficient-D3DGS [27]	23.32	0.864	0.140	26.20	0.906	0.099	21.54	0.784	0.253	24.08	0.892	0.098
Deformable-3DGS [62]	26.12	0.891	0.122	26.48	0.910	0.100	23.21	0.837	0.199	23.96	0.901	0.088
4DGS [55]	25.46	0.856	0.198	27.12	0.842	0.151	22.43	0.842	0.264	22.17	0.743	0.215
MoDec-GS [29]	26.33	0.883	0.174	27.15	0.909	0.102	23.12	0.858	0.198	23.33	0.816	0.150
Ours	27.65	0.903	0.092	29.01	0.908	0.098	24.85	0.855	0.151	25.58	0.889	0.086

Method	Skating			Truck			Umbrella			Mean		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3DGS [28]	21.20	0.628	0.508	13.81	0.413	0.627	20.05	0.410	0.530	17.69	0.451	0.497
STGS [36]	24.80	0.910	0.109	25.01	0.868	0.103	21.88	0.770	0.195	22.17	0.807	0.152
Ex4DGS [30]	21.92	0.807	0.233	19.04	0.666	0.308	19.03	0.593	0.340	17.72	0.570	0.371
Efficient-D3DGS [27]	25.44	0.892	0.161	24.33	0.829	0.232	23.95	0.805	0.183	24.92	0.868	0.150
Deformable-3DGS [62]	26.53	0.929	0.126	24.16	0.806	0.192	26.18	0.861	0.187	25.86	0.888	0.132
4DGS [55]	28.92	0.931	0.196	28.25	0.888	0.235	24.80	0.714	0.297	25.81	0.844	0.210
MoDec-GS [29]	29.29	0.942	0.156	29.19	0.910	0.185	25.02	0.760	0.224	26.63	0.879	0.160
Ours	30.17	0.946	0.076	30.18	0.917	0.115	27.21	0.831	0.170	28.20	0.903	0.103

from camera motion. By utilizing Camera Flow $\mathbf{F}_{cam}$, which contains only camera-induced motion, we enforce that static areas strictly follow this flow to ensure structural stability and prevent false motion. Following the Full Flow Matching principle, we define motion consistency and cross-time rendering constraints by warping input and rendered images with $\mathbf{F}_{cam}$, applied exclusively to static regions.

$$\mathcal{L}_{mc}^{cam} = \mathcal{L}_{photo}(I_1^{static,warped}, I_2^{static}) \quad (6)$$

$$\mathcal{L}_{cr}^{cam} = \mathcal{L}_{photo}(\hat{I}_1^{static,warped}, I_2^{static}) \quad (7)$$

$$\mathcal{L}_{camflow} = \lambda_{mc}^{cam} \mathcal{L}_{mc}^{cam} + \lambda_{cr}^{cam} \mathcal{L}_{cr}^{cam} \quad (8)$$

This mechanism ensures static regions maintain consistent 3D structure throughout the sequence and eliminates false motion caused by lighting or view-change artifacts. Together, these two flow constraints constitute a complete self-correction loop that enables reliable and dense-guidance-free 3D motion learning.

3.4. Optimization

We optimize frame pairs (I_1, I_2) with supervision solely from the video frames by the loss:

$$\mathcal{L} = \mathcal{L}_{baseline} + \lambda_{ff} \mathcal{L}_{fullflow}(I_1, I_2) + \lambda_{cf} \mathcal{L}_{camflow}(I_1^{static}, I_2^{static}) \quad (9)$$

where $\mathcal{L}_{baseline}$ denotes the baseline loss from the 4DGS framework [55], which consists of a photometric rendering loss and a feature plane regularization term, and λ_{ff} , λ_{cf} weight the Full Flow Matching and Camera Flow Matching terms respectively.

4. Experiment

4.1. Experimental Setup

Implementation Details. We implement our approach within a deformation-based dynamic scene rendering framework [55], building the canonical space point cloud with a geometry foundation model [65]. Following [55], we adopt a two-phase optimization strategy: a coarse phase where we freeze the deformation fields for dynamic regions while applying camera flow matching to static regions, followed by a fine phase that jointly optimizes both static and dynamic elements with active deformation fields. All experiments run on a single NVIDIA RTX A6000 GPU.

Datasets and Metrics. We evaluate our method on three representative dynamic scene datasets. The NVIDIA Monocular dataset [18] features forward-facing captures with controlled dynamics recorded by a camera array; following [29], we use the provided monocular video sequences for benchmarking. Both Nerfies [45] and HyperNeRF [46] datasets were captured using a vrig setup with synchronized stereo cameras, allowing us to train on one camera view while using the other for testing at identical timestamps. Performance is measured using PSNR, SSIM, and LPIPS [66], with metrics computed per frame and averaged across test sequences.

4.2. Experimental Results

Quantitative Comparison. We present quantitative results in Tables 1 and 2. Our method consistently outperforms existing approaches across all eight scenes in the NVIDIA Monocular dataset on PSNR, SSIM, and LPIPS metrics,

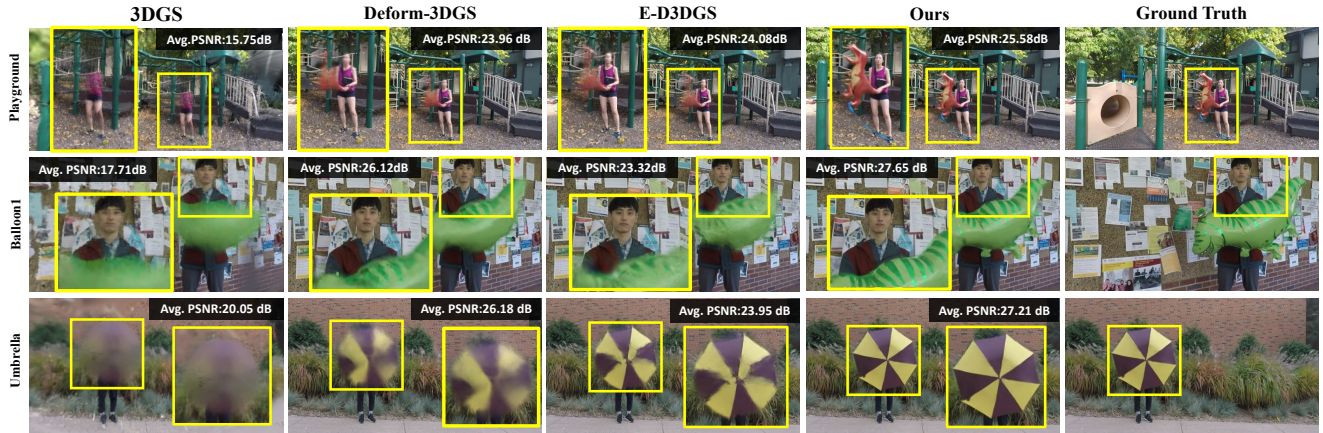


Figure 5. **Qualitative comparison on Nvidia Monocular dataset** [18]. Yellow boxes highlight zoomed-in regions for detail examination. Per-scene average PSNR values are provided.

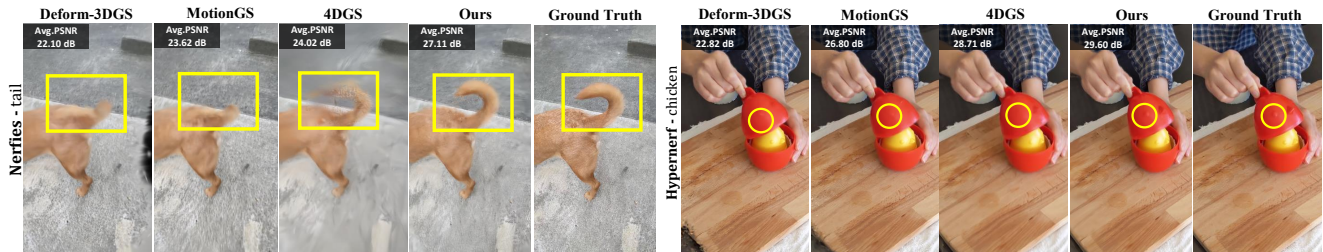


Figure 6. **Qualitative comparison on Nerfies-HyperNeRF dataset** [45, 46]. Yellow boxes highlight zoomed-in regions for detail examination. Per-scene average PSNR values are provided.

particularly in Balloon2, Playground, and Umbrella scenes, where we improve PSNR by over 2dB compared to recent methods like MoDec-GS [29] that employ complex motion modeling. On the Nerfies-HyperNeRF dataset, our approach also achieves best performance, surpassing all competing methods by significant margins.

Qualitative Comparison. Figures 5 and 6 show visual comparisons with existing methods. On the NVIDIA Monocular dataset, our method achieves better reconstruction quality, capturing the balloon and subtle ribbons in the Playground scene that other methods miss entirely, while maintaining sharp stripe patterns in Balloon1 that appear blurred in competing approaches. On the Nerfies-HyperNeRF dataset, we successfully reconstruct the complete tail structure in the Tail sequence where alternatives produce incomplete results, and preserve the critical red exterior protrusion in the Chicken sequence that remains absent in other approaches. We also visualize the process of our self-correction flow matching in Figure 10 and its effects in Figure 7, illustrating how the model captures subtle motions and aligns objects across frames. More qualitative results are included in Appendix C.

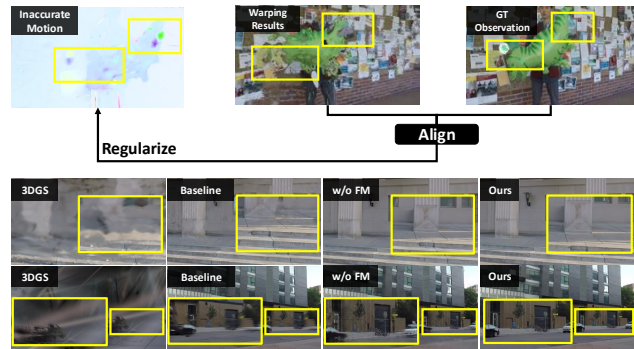


Figure 7. **Visualization of self-correction flow matching effects.** **Top:** Incorrect motion leads to erroneous warping results that do not align with the ground truth, highlighted by yellow boxes, which in turn drives motion optimization. **Bottom:** After flow matching, finer details are better reconstructed, demonstrating improved rendering quality.

4.3. Ablation Study

We analyze each component’s effectiveness through a systematic ablation study, summarized in Table 3. All experi-

Table 2. **Quantitative comparison on Nerfies-HyperNeRF dataset [45, 46].** We highlight the **best** and **second best** results.

Methods	(a) Nerfies-HyperNeRF	
	PSNR $\uparrow$	SSIM $\uparrow$
T-NeRF [47]	20.43	0.508
NSFF [34]	19.41	0.423
Nerfies [45]	20.08	0.462
HyperNeRF [46]	20.40	0.467
Deformable-3DGS [62]	22.19	0.558
4DGS [55]	24.46	0.588
Ours	25.97	0.674

Methods	(b) HyperNeRF	
	PSNR $\uparrow$	SSIM $\uparrow$
T-NeRF [47]	21.31	0.583
NSFF [34]	19.02	0.456
Nerfies [45]	20.22	0.535
HyperNeRF [46]	20.73	0.544
Deformable-3DGS [62]	22.48	0.611
SC-GS [21]	21.20	0.576
4DGS [55]	25.16	0.684
MotionGS [68]	24.80	0.69
MoDec-GS [29]	25.02	0.679
Ours	26.15	0.736

Table 3. **Ablation study of key components in our method.** CompInit.: Complete Canonical Space Initialization; Sep.: Separation-based Dynamic Scene Modeling; FullFM: Full Flow Matching; CamFM: Camera Flow Matching. We highlight best performance in **bold**.

#	CompInit.	Sep.	FullFM	CamFM	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
Baseline					25.81	0.844	0.210
1			✓		26.45	0.865	0.172
2		w/o align			24.50	0.815	0.260
3		✓			26.60	0.870	0.167
4		✓	✓		27.00	0.880	0.159
5		✓	✓	✓	27.85	0.897	0.116
Ours		✓	✓	✓	28.20	0.903	0.103

ments are conducted on the NVIDIA Monocular dataset.

Effectiveness of flow matching. *Configuration #1* applies our self-correction flow matching directly on the baseline model, already yielding a notable improvement to 26.45dB PSNR (+0.64dB). This highlights the immediate effectiveness of our approach—without requiring additional motion guidance or architectural modifications, the model benefits from motion-aware supervision derived solely from 2D video cues even in a sparse initialization setup. As illustrated in Fig. 7, our self-correction flow matching clearly correct the process of motion learning using only the 2D motion cues

from input video.

Impact of initialization and modeling. We investigate the effectiveness of our self-correction flow matching framework in building a complete and disentangled scene representation. Starting from the baseline, naively incorporating raw, unaligned point clouds (*configuration #2*) leads to significant degradation (−0.71dB), indicating that misaligned geometric priors introduce noise rather than providing useful guidance, as also reflected in increased LPIPS scores. In contrast, our hierarchical alignment strategy (*configuration #3*) constructs a coherent canonical space, improving synthesis quality to 26.60dB (+0.79dB). Building upon this, further disentangling the static and dynamic components of the scene (*configuration #4*) yields a notable performance boost (27.00dB, +1.19dB), demonstrating that motion-aware modeling enables more faithful and generalizable scene encoding. These results collectively highlight the importance of structured scene initialization and motion decomposition in enhancing representation quality under our self-correction paradigm.

Flow matching thrives with appropriate foundation. Combining hierarchical alignment with static-dynamic separation (*configuration #4*) establishes a robust foundation at 27.00dB. Upon this foundation, Full Flow Matching (*configuration #5*) delivers +0.85dB gain to reach 27.85dB. This confirms flow matching is most effective when applied to representations that distinguish between motion types, allowing the model to respect real-world motion precisely where and how it matters most. Finally, adding Camera Flow Matching (*Ours*) further improves static region quality, resulting in an additional gain. The quantitative gains of each step are summarized in Fig. 10 in Appendix, illustrating the progressive benefits of our design choices.

5. Conclusion

We present ReFlow, a unified dynamic scene reconstruction framework that learns 3D motion in a self-correcting manner from monocular video. By exploiting the inherent relationship between 3D motion and 2D frame-to-frame differences, ReFlow introduces a simple yet effective flow matching mechanism to directly supervise scene dynamics from video. To enhance Gaussian initialization and support separate modeling of static and dynamic objects, ReFlow incorporates a Complete Canonical Space Construction module and a Separation-Based Dynamic Scene Modeling module. These components enable ReFlow to effectively perform region-specific motion learning and scene reconstruction, with experiments demonstrating superior reconstruction quality in diverse dynamic scenarios, establishing a new self-correction paradigm for 4D reconstruction.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. U25A20536), the Youth Innovation Promotion Association of CAS, and the Key Technology Research Project of TW-3 (No. TW3003).

References

- [1] Kara-Ali Aliev, Artem Sevastopolsky, Maria Kolos, Dmitry Ulyanov, and Victor Lempitsky. Neural point-based graphics. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXII 16*, pages 696–712. Springer, 2020. [1](#)
- [2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021.
- [3] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022.
- [4] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Zip-nerf: Anti-aliased grid-based neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19697–19705, 2023. [1](#)
- [5] Andrew Bond, Jui-Hsien Wang, Long Mai, Erkut Erdem, and Aykut Erdem. Gaussianvideo: Efficient video representation via hierarchical gaussian splatting. *arXiv preprint arXiv:2501.04782*, 2025. [3](#)
- [6] Ang Cao and Justin Johnson. Hexplane: A fast representation for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 130–141, 2023. [3](#)
- [7] Qizhi Chen, Delin Qu, Yiwen Tang, Haoming Song, Yiting Zhang, Dong Wang, Bin Zhao, and Xuelong Li. Freegaussian: Guidance-free controllable 3d gaussian splats with flow derivatives. *arXiv preprint arXiv:2410.22070*, 2024. [2](#), [5](#)
- [8] Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *arXiv preprint arXiv:2311.18561*, 2023. [3](#)
- [9] Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *International Journal of Computer Vision*, 134(3):83, 2026. [1](#)
- [10] Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo De Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, et al. Omnire: Omni urban scene reconstruction. *arXiv preprint arXiv:2408.16760*, 2024. [1](#)
- [11] Zhixin Cheng, Jiacheng Deng, Xinjun Li, Baoqun Yin, and Tianzhu Zhang. Bridge 2d-3d: Uncertainty-aware hierarchical registration network with domain alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2491–2499, 2025. [1](#)
- [12] Zhixin Cheng, Jiacheng Deng, Xinjun Li, Xiaotian Yin, Bohao Liao, Baoqun Yin, Wenfei Yang, and Tianzhu Zhang. Ca-i2p: Channel-adaptive registration network with global optimal selection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 27739–27749, 2025.
- [13] Zhixin Cheng, Bohao Liao, Jiacheng Deng, Xiaotian Yin, Xinjun Li, Yujia Chen, Baoqun Yin, and Tianzhu Zhang. Rethinking 2d-3d registration: A novel network for high-value zone selection and representation consistency alignment. *Authorea Preprints*, 2026.
- [14] Zhixin Cheng, Xiaotian Yin, Jiacheng Deng, Bohao Liao, Yujia Chen, Xu Zhou, Baoqun Yin, and Tianzhu Zhang. Adaptive agent selection and interaction network for image-to-point cloud registration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3335–3343, 2026. [1](#)
- [15] Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. Tapir: Tracking any point with per-frame initialization and temporal refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10061–10072, 2023. [2](#)
- [16] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023. [3](#)
- [17] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5712–5721, 2021. [6](#)
- [18] Chen Gao, Ayush Saraf, Johannes Kopf, and Jia-Bin Huang. Dynamic view synthesis from dynamic monocular video, 2021. [6](#), [7](#), [4](#)
- [19] Quankai Gao, Qiangeng Xu, Zhe Cao, Ben Mildenhall, Wenchao Ma, Le Chen, Danhang Tang, and Ulrich Neumann. Gaussianflow: Splatting gaussian dynamics for 4d content creation. *arXiv preprint arXiv:2403.12365*, 2024. [2](#), [3](#), [5](#)
- [20] Bingbing Hu, Yanyan Li, Rui Xie, Bo Xu, Haoye Dong, Junfeng Yao, and Gim Hee Lee. Learnable infinite Taylor gaussian for dynamic view rendering. *arXiv preprint arXiv:2412.04282*, 2024. [3](#)
- [21] Yi-Hua Huang, Yang-Tian Sun, Ziyi Yang, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Sc-gs: Sparse-controlled gaussian splatting for editable dynamic scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4220–4230, 2024. [3](#), [8](#)
- [22] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. In *European conference on computer vision*, pages 668–685. Springer, 2022. [2](#), [3](#)
- [23] DaDong Jiang, Zhihui Ke, Xiaobo Zhou, Zhi Hou, Xianghui Yang, Wenbo Hu, Tie Qiu, and Chunchao Guo. Timeformer:

- Capturing temporal relationships of deformable 3d gaussians for robust reconstruction. *arXiv preprint arXiv:2411.11941*, 2024. 3
- [24] Ying Jiang, Chang Yu, Tianyi Xie, Xuan Li, Yutao Feng, Huamin Wang, Minchen Li, Henry Lau, Feng Gao, Yin Yang, et al. Vr-gs: A physical dynamics-aware interactive gaussian splatting system in virtual reality. In *ACM SIGGRAPH 2024 conference papers*, pages 1–1, 2024. 1
- [25] Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Co-tracker3: Simpler and better point tracking by pseudo-labelling real videos. *arXiv preprint arXiv:2410.11831*, 2024. 2, 3
- [26] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Co-tracker: It is better to track together. In *European Conference on Computer Vision*, pages 18–35. Springer, 2024. 2, 3
- [27] Kai Katsumata, Duc Minh Vo, and Hideki Nakayama. A compact dynamic 3d gaussian representation for real-time dynamic view synthesis. In *European Conference on Computer Vision*, pages 394–412. Springer, 2024. 3, 6
- [28] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023. 1, 5, 6
- [29] Sangwoon Kwak, Joonsoo Kim, Jun Young Jeong, Won-Sik Cheong, Jihyong Oh, and Munchurl Kim. Modec-gs: Global-to-local motion decomposition and temporal interval adjustment for compact dynamic 3d gaussian splatting. *arXiv preprint arXiv:2501.03714*, 2025. 6, 7, 8, 4
- [30] Junoh Lee, ChangYeon Won, Hyunjun Jung, Inhwan Bae, and Hae-Gon Jeon. Fully explicit dynamic gaussian splatting. *Advances in Neural Information Processing Systems*, 37:5384–5409, 2024. 6
- [31] Jiahui Lei, Yijia Weng, Adam Harley, Leonidas Guibas, and Kostas Daniilidis. Mosca: Dynamic gaussian fusion from casual videos via 4d motion scaffolds. *arXiv preprint arXiv:2405.17421*, 2024. 2, 3
- [32] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European Conference on Computer Vision*, pages 71–91. Springer, 2024. 2
- [33] Yue Li, Qi Ma, Runyi Yang, Huapeng Li, Mengjiao Ma, Bin Ren, Nikola Popovic, Nicu Sebe, Ender Konukoglu, Theo Gevers, et al. Scenesplat: Gaussian splatting-based scene understanding with vision-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4961–4972, 2025. 1
- [34] Zhengqi Li, Simon Niklaus, Noah Snavely, and Oliver Wang. Neural scene flow fields for space-time view synthesis of dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6498–6508, 2021. 8
- [35] Zhengqi Li, Qianqian Wang, Forrester Cole, Richard Tucker, and Noah Snavely. Dynibar: Neural dynamic image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4273–4284, 2023. 3
- [36] Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8508–8520, 2024. 1, 3, 6
- [37] Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast, and robust structure and motion from casual dynamic videos. *arXiv preprint arXiv:2412.04463*, 2024. 5
- [38] Yiqing Liang, Numair Khan, Zhengqin Li, Thu Nguyen-Phuoc, Douglas Lanman, James Tompkin, and Lei Xiao. Gafre: Gaussian deformation fields for real-time dynamic novel view synthesis. *arXiv preprint arXiv:2312.11458*, 2023. 3
- [39] Chieh Hubert Lin, Zhaoyang Lv, Songyin Wu, Zhen Xu, Thu Nguyen-Phuoc, Hung-Yu Tseng, Julian Straub, Numair Khan, Lei Xiao, Ming-Hsuan Yang, et al. Dgs-irm: Real-time deformable 3d gaussian reconstruction from monocular videos. *arXiv preprint arXiv:2506.09997*, 2025. 5
- [40] Youtian Lin, Zuozhuo Dai, Siyu Zhu, and Yao Yao. Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21136–21145, 2024. 3
- [41] Jiahao Lu, Tianyu Huang, Peng Li, Zhiyang Dou, Cheng Lin, Zhiming Cui, Zhen Dong, Sai-Kit Yeung, Wenping Wang, and Yuan Liu. Align3r: Aligned monocular depth estimation for dynamic videos. *arXiv preprint arXiv:2412.03079*, 2024. 2, 5
- [42] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. *arXiv preprint arXiv:2308.09713*, 2023. 1
- [43] 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. 1
- [44] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, 2022. 1
- [45] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5865–5874, 2021. 6, 7, 8, 4
- [46] Keunhong Park, Utkarsh Sinha, Peter Hedman, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Ricardo Martin-Brualla, and Steven M Seitz. Hypernerf: A higher-dimensional representation for topologically varying neural radiance fields. *arXiv preprint arXiv:2106.13228*, 2021. 6, 7, 8, 4
- [47] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 3, 8

- [48] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3), 2022. 3
- [49] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 1
- [50] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1
- [51] Yang-Tian Sun, Yihua Huang, Lin Ma, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Splatter a video: Video gaussian representation for versatile processing. *Advances in Neural Information Processing Systems*, 37:50401–50425, 2025. 2
- [52] Qianqian Wang, Vickie Ye, Hang Gao, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. *arXiv preprint arXiv:2407.13764*, 2024. 2, 3
- [53] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 2
- [54] Shizun Wang, Xingyi Yang, Qihong Shen, Zhenxiang Jiang, and Xinchao Wang. Gflow: Recovering 4d world from monocular video. *arXiv preprint arXiv:2405.18426*, 2024. 2, 3
- [55] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. *arXiv preprint arXiv:2310.08528*, 2023. 1, 3, 6, 8, 2, 5
- [56] Wenqi Xian, Jia-Bin Huang, Johannes Kopf, and Changil Kim. Space-time neural irradiance fields for free-viewpoint video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9421–9431, 2021. 3
- [57] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8121–8130, 2022. 2, 3
- [58] Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, Fisher Yu, Dacheng Tao, and Andreas Geiger. Unifying flow, stereo and depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):13941–13958, 2023. 2
- [59] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5438–5448, 2022. 1
- [60] Zhen Xu, Zhengqin Li, Zhao Dong, Xiaowei Zhou, Richard Newcombe, and Zhaoyang Lv. 4dgt: Learning a 4d gaussian transformer using real-world monocular videos. *arXiv preprint arXiv:2506.08015*, 2025. 5
- [61] Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In *European Conference on Computer Vision*, pages 156–173. Springer, 2024. 1
- [62] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. *arXiv preprint arXiv:2309.13101*, 2023. 1, 3, 6, 8
- [63] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023. 1, 3
- [64] Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K Barral, Raia Hadsell, et al. Efficiently reconstructing dynamic scenes one d4rt at a time. *arXiv preprint arXiv:2512.08924*, 2025. 5
- [65] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*, 2024. 2, 3, 6, 1
- [66] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6
- [67] Kaichen Zhou, Jia-Xing Zhong, Sangyun Shin, Kai Lu, Yiyuan Yang, Andrew Markham, and Niki Trigoni. Dynpoint: Dynamic neural point for view synthesis. *Advances in Neural Information Processing Systems*, 36, 2023. 2, 3
- [68] Ruijie Zhu, Yanzhe Liang, Hanzhi Chang, Jiacheng Deng, Jiahao Lu, Wenfei Yang, Tianzhu Zhang, and Yongdong Zhang. Motions: Exploring explicit motion guidance for deformable 3d gaussian splatting. *Advances in Neural Information Processing Systems*, 37:101790–101817, 2025. 2, 3, 5, 8