VideoLifter: Lifting Videos to 3D with Fast and Efficient Hierarchical Stereo Alignment

Wenyan Cong[1], Hanqing Zhu[1], Kevin Wang[1], Jiahui Lei[2], Colton Stearns[3], Yuanhao Cai[4], Dilin Wang[5], Rakesh Ranjan[5], Matt Feiszli[5], Leonidas Guibas[3], Zhangyang Wang[1], Weiyao Wang[5*], Zhiwen Fan[1,6*]
[1]UT Austin [2]UPenn [3]Stanford [4]JHU [5]Meta [6]TAMU
Project Website: `https://videolifter.github.io`

Abstract

Efficiently reconstructing 3D scenes from monocular video remains a core challenge in computer vision, vital for applications in virtual reality, robotics, and scene understanding. Recently, frame-by-frame progressive reconstruction without camera poses is commonly adopted, incurring high computational overhead and compounding errors when scaling to longer videos. To overcome these issues, we introduce `VideoLifter`*, a novel video-to-3D pipeline that leverages a local-to-global strategy on a fragment basis, achieving both extreme efficiency and SOTA quality. Locally,* `VideoLifter` *leverages learnable 3D priors to register fragments, extracting essential information for subsequent 3D Gaussian initialization with enforced inter-fragment consistency and optimized efficiency. Globally, it employs a tree-based hierarchical merging method with key frame guidance for inter-fragment alignment, pairwise merging with Gaussian point pruning, and subsequent joint optimization to ensure global consistency while efficiently mitigating cumulative errors. This approach significantly accelerates the reconstruction process, reducing training time by over 82% while achieving better visual quality than SOTA methods.*

1. Introduction

Reconstructing 3D scenes from image observations is a longstanding problem in computer vision, with applications spanning AR/VR, video processing, and autonomous driving. Recently, reconstructing 3D scenes from a single video (video-to-3D) has gained significant traction. This trend is driven by two factors: the increasing accessibility of handheld capture devices, making video capture more practical for non-professional users, and recent advancements in high-fidelity 3D reconstruction methods such as Neural Radiance Fields (NeRF) [23] and 3D Gaussian Splatting (3D-GS) [14].

Most video-to-3D reconstruction methods heavily depend on Structure-from-Motion (SfM) [28] to generate initial sparse reconstructions, providing essential components like camera poses, intrinsics, and the initial point cloud to build dense 3D models using NeRF or 3DGS. However, *when applied to video data, SfM is often unreliable or even infeasible* (**Issue ❶**), because it relies on photometric assumptions that frequently break down in low-texture or challenging lighting conditions [21, 24, 28], though some works have improved SfM under certain conditions [18]. In response, recent methods [3, 7, 17, 37] have shifted toward jointly optimizing camera poses and scene representations rather than relying solely on SfM-based initializations. However, these methods still depend on accurate camera intrinsics from SfM, limiting their applicability in in-the-wild video scenarios.

More importantly, those SfM-free video-to-3D methods typically reconstruct scenes incrementally from a canonical view, leading to two critical issues. First, they are *slow and inefficient* (**Issue ❷**) due to an iterative, frame-by-frame approach that re-optimizes the entire sequence with each new frame, thereby prolonging training times ($>$ 2 hours) and complicating the handling of complex trajectories, especially in a video setting (as opposed to just a few images). Naively applying the non-incremental InstantSplat [5] to video data leads to out-of-memory (OOM) issues, preventing it from scaling to many frames (see Tab. 2). Second, they are *susceptible to incremental errors* (**Issue ❸**), as the frame-by-frame approach tends to accumulate errors over long video sequences.

To address these issues, we propose `VideoLifter`, a novel video-to-3D reconstruction pipeline that achieves a **5$\times$ speed-up** and **enhanced novel view-synthesis quality** compared to state-of-the-art methods, as demonstrated in Fig. 1. We effectively adopt the **local-to-global** approach to handle long-sequence videos on a fragment basis and then subsequently merge fragments into a final, globally consistent 3D scene. Our pipeline is driven by two key innovations that make the local-to-global concept workable with significantly boosted efficiency (Issue ❷) and much-reduced incremental errors (Issue ❸) on video-to-3D. First, in the **Fragment Registration with Learned 3D Priors** (Local) stage, we extract essential information from each fragment (e.g., pointmaps and local camera poses for 3D Gaussian

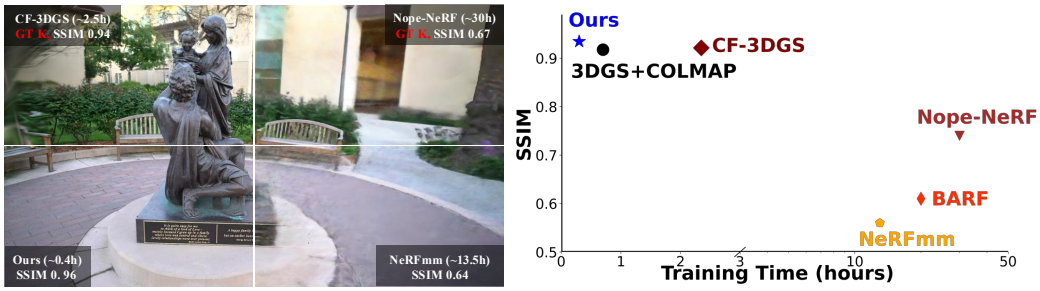

Figure 1. **Novel View Synthesis and Training Time Comparisons** on Tanks and Temples dataset. `VideoLifter` does not require precomputed camera parameters (i.e., camera intrinsics K from COLMAP), reduces the training time required by the most relevant baseline CF-3DGS [7] by 82% while improving image quality (SSIM).

initialization) by leveraging pretrained prior models such as MASt3R [16] to address Issue ❶. Rather than naively using 3D priors to initialize 3D Gaussians, like InstantSplat (MASt3R + 3D-GS) [5], we improve *efficiency* by (1) enforcing inter-fragment consistency via solely considering the key frames, solved on an efficient subgraph instead of a complete graph, and (2) extracting only the essential parameters (6-dimensional quaternion pose and 1-dimensional scale) for each view within each fragment, thereby avoiding costly global optimization of full point maps. Second, in the **Hierarchical Gaussian Alignment** (Global) stage, we merge fragments through a tree-based hierarchical framework that employs key frame guidance for inter-fragment alignment, pairwise merging with Gaussian point pruning, and subsequent joint optimization to ensure global consistency and mitigate cumulative errors efficiently. Overall, our main contributions are as follows:

- We introduce `VideoLifter`, an efficient, high-quality, and robust video-to-3D reconstruction framework with a local-to-global strategy.
- Our **fragment registration with learned 3D priors** efficiently extracts essential representations for subsequent dense 3D-GS with several key *efficiency-driven optimizations* along with *learned 3D priors* to remove reliance on traditional module SfM.
- Our **hierarchical 3D Gaussian alignment** minimizes incremental errors through three well-designed iterative stages, ensuring both accuracy and efficiency.
- Extensive experiments on the Tanks and Temples and CO3D-V2 datasets demonstrate that `VideoLifter` significantly enhances training efficiency, achieves over a 5× speed-up and improves rendering quality compared to state-of-the-art methods.

2. Related Works

3D Representations for Novel View Synthesis. 3D reconstruction for high-quality novel view synthesis generates unseen views of a scene or object from a set of images [1, 22]. After the seminal NeRF work [23], a wave of unstructured radiance field methods has emerged [14, 39], each adopting different scene-representation primitives. Among these, 3D-GS [14] stands out with impressive performance in efficiently reconstructing complex, real-world scenes with high fidelity. Both NeRFs and 3DGS rely on carefully captured sequential video or multi-view images to ensure sufficient scene coverage, utilizing preprocessing tools like SfM (e.g., COLMAP [28]) to compute camera parameters and provide a sparse SfM point cloud as additional input.

Traditional Structure-from-Motion (SfM). Estimating 3D structure and camera motion is a well-explored challenge [8, 25, 38]. SfM has seen significant advancements across various dimensions. Methods like [10, 19] focus on enhancing feature detection, [30] introduces innovative optimization methods, [9, 28] explore improved data representations and more robust structural solutions. Despite these advances, traditional SfM techniques remain vulnerable to issues such as low-texture regions, occlusions, moving objects, and lighting variations, limiting their overall robustness and performance.

Radiance Field without SfM. Inaccuracies from SfM can propagate through subsequent radiance field reconstruction, reducing overall quality. Various approaches have been proposed to eliminate the reliance on SfM by jointly optimizing camera parameters and scene representation, such as NeRFmm [37] and BARF [17]. GARF [12] further simplifies the joint optimization and improves both efficiency and accuracy by using Gaussian-MLP models. SPARF [33] and TrackNeRF [20] introduce a method to simulate pose noise by injecting Gaussian noise into the camera parameters. Recently, depth priors from a monocular depth estimator have been used to guide radiance field optimization [3, 4, 7, 21]. More recent work, such as [13] and InstantSplat [5], either supplement depth priors with other priors (e.g., image matching network), or integrate end-to-end stereo models like DUST3R/MASt3R to reduce the dependency on camera pose information. While these methods show promise in removing the SfM reliance, scaling them to a large number of views remains a challenge, such as the OOM issue (e.g., InstantSplat), drifting error (e.g., CF-3DGS), and unsatisfactory quality (e.g., InstantSplat).

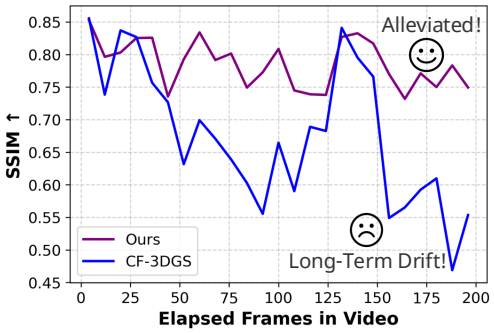

Figure 2. CF-3DGS's frame-by-frame pipeline accumulates errors with long-term drift, while our method compresses drift error along # frames. Results are tested on 247_26441_50907 from CO3D-V2.

Comparison with Simultaneous Localization and Mapping (SLAM) Although both SLAM and video-to-3D reconstruction process multiple views, their input conditions and end goals differ fundamentally. First, SLAM operates online, processing frames sequentially as they arrive, whereas video-to-3D has access to the entire sequence upfront. This offline setting enables pipelines to consider all frames together rather than a purely sequential approach. Second, our primary objective is **novel view synthesis**, generating photorealistic views from unseen viewpoints. SLAM methods are not designed to support novel view synthesis; they only re-render the training frames [27] Hence, SLAM-based techniques are not directly applicable to the video-to-3D reconstruction problem.

3. `VideoLifter`: An Efficient and Effective Video-to-3D Framework

3.1. Video-to-3D: Challenges and Our Design

We first define the video-to-3D reconstruction problem and outline current issues in delivering an *efficient and high-quality* reconstruction pipeline. We then present our high-level design philosophy that tackles these challenges.

Video-to-3D Reconstruction Given a sequence of N unposed and uncalibrated images from a monocular video, denoted as $\mathcal{I} = \{I_i \in \mathbb{R}^{H\times W\times 3}\}_{i=1}^{N}$, `VideoLifter` aims to reconstruct the scene using 3D Gaussians $\mathcal{G}$ along with estimated camera intrinsics K and extrinsics $\mathcal{T} = \{T_i \in \mathbb{R}^{3\times 4}\}_{i=1}^{N}$. We assume that all frames share a common intrinsic matrix, as they are from a single monocular video.

Key Challenges While NeRF and 3D-GS have advanced 3D reconstruction, their variants remain suboptimal for video-to-3D reconstruction in terms of **speed** and **quality**. Existing methods often adopt a frame-by-frame progressive reconstruction method, making them inherently *slow* and *prone to cumulative errors* (See long-term performance drift in Fig. 2) when processing videos. Furthermore, they typically rely on SfM to estimate camera intrinsic, which is *unreliable* or *even infeasible* for in-the-wild video sequences.

Our High-level Framework To meet the critical need for efficiency and quality in long-sequence video to 3D reconstruction, we depart from conventional frame-by-frame or holistic optimization methods by embracing a hierarchical local-to-global design philosophy. Specifically, we process long video sequences on a *fragment* basis and subsequently merge these fragments into a single, consistent 3D scene. Although the local-to-global concept is not new, adapting it to a video-to-3D reconstruction pipeline with 3DGS is new, with two key unanswered challenges:

*How can we extract 3D reconstruction information **efficiently** and **reliably** from monocular video?*

*How can we merge fragments into a **high-quality** and **consistent** 3D scene without alignment issues?*

In response, we propose `VideoLifter`, an **efficient** and **high-quality** video-to-3D reconstruction pipeline, as illustrated in Fig. 3. Our method achieves SOTA performance in both speed and quality (see Fig. 1) through two-level innovations: Local: **Fragment Registration with Learned 3D Priors** with efficiency-driven designs (Sec. 3.2) and Global: **Hierarchical Gaussian Alignment** to minimize alignment error (Sec. 3.3).

3.2. Fragment Registration with Learned 3D Priors

First, we describe our method to *efficiently* and *reliably* extract essential 3D reconstruction information on a fragment basis and how we enforce cross-fragment consistency.

Process Description: We partition the input video sequence into m disjoint windows of length k, which we refer to as **fragments**. For example, the i'th fragment is given by $\mathcal{I}_i^f = [I_{(i-1)k+1}, I_{(i-1)k+2}, \ldots, I_{ik}]$, where $i \in [1, m]$. Our objective is twofold: first, to extract essential information for subsequent intra-fragment 3D reconstruction (e.g., point cloud for Gaussian initialization, along with coarse local camera extrinsics and intrinsics); and second, to obtain inter-fragment information necessary for future local-to-global merging.

Fragmentation Method Choice: In this work, we use a straightforward fragmentation strategy by uniformly dividing the frames into disjoint windows. Although simple, this approach has proven effective on benchmarks. More sophisticated methods, such as using frame-to-frame similarity to guide fragmentation, could further enhance `VideoLifter` (e.g., in videos with abrupt view changes during capture), but this is orthogonal to our core contributions.

Fragment-level Challenges: Naively applying existing methods for the fragment level (e.g., those in LocalRF [21] or CF-3DGS [7]) is neither efficient enough nor does it adequately prepare for future merging. Challenge ①: SfM is

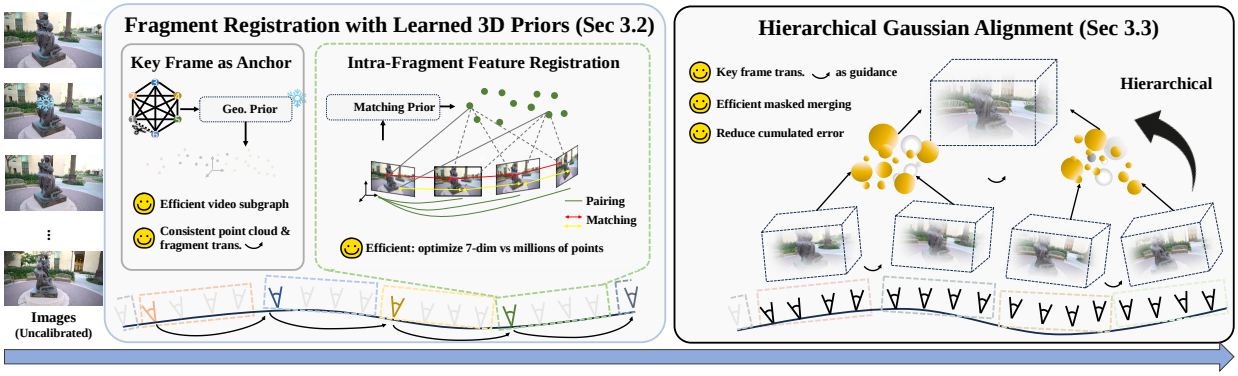

Figure 3. **Network Architecture.** Given uncalibrated images, `VideoLifter` first employs learned priors for efficient fragment registration. The independently optimized 3D Gaussians from fragment are then hierarchically aligned into a globally coherent 3D representation.

Matching	Geometric	SSIM	PSNR	LPIPS	ATE
LoFTR [32]	Metric3Dv2 [11]	0.9238	31.30	0.0757	0.005
MASt3R	MASt3R	0.9347	31.59	0.0730	0.004

Table 1. Comparison of prior models on Tanks and Temples.

heavily relied on for NeRF/3D-GS, while it is not always available or reliable—especially in our video-to-3D reconstruction with varying conditions. Challenge ②: A critical issue in the local-to-global paradigm is ensuring that fragments can be merged without incurring significant alignment errors. Challenge ③: Beyond the inherent efficiency benefits of a local-to-global design, further enhancements in efficiency are necessary for the video scenario.

Learned 3D priors (Challenge ①): 3D-GS needs *point cloud* for initialization and *camera pose* for optimization. However, in long-sequence video settings, traditional SfM methods (e.g., COLMAP) are often unavailable or unreliable. While CF-3DGS employs monocular depth estimation (a geometric prior) on each view to obtain a point cloud, it introduces scale issues that necessitate additional optimization during 3D-GS training.

Inspired by recent work on replacing SIFT with NN-based method (e.g., LoFTR [32]/GIM [29]), we eliminate dependence on SfM and instead leverage learned 3D priors from large-scale pretrained foundation models. In this work, we use MASt3R [16] as our prior model since it seamlessly integrates both geometric and matching cues. We emphasize that our video-to-3D pipeline is not exclusively tailored to MASt3R; rather, our `VideoLifter` is flexible and can incorporate any model that provides robust geometric and matching priors. For example, in Tab. 1, we demonstrate that `VideoLifter` performs well with both MASt3R and alternative approaches (e.g. using LoFTR [32] for matching cues + Metric3Dv2 [11] for geometric cues). We choose MASt3R for its simplicity and efficiency.

Key Frames as Anchor (Challenge ②, ③): In a fragment-based approach, ensuring inter-fragment consistency is critical for high-quality 3D reconstruction. To address this, we propose to "anchor" each fragment with a **key frame** (the first frame, $I_{(i-1)k+1}$) and *enforce consistency only among these key frames*. This strategy simplifies the consistency problem by limiting the problem scale to $\frac{N}{k}$ key frames instead of considering whole N frames, thereby *improving efficiency* and *reducing complexity*.

Key frame consistency is enforced at the *point-cloud* level by generating a globally optimized dense point map, along with transformation matrices across adjacent fragments $\{T^f_{i\to i+1}\}_{i=1}^{m-1}$. To achieve this, we follow a procedure similar to that in MASt3R [16], optimizing:

$$(\tilde{\boldsymbol{P}}^*, \boldsymbol{T}_e^*) = \arg\min_{\tilde{\boldsymbol{P}},\boldsymbol{T},\sigma} \sum_{e\in\mathcal{E}} \sum_{v\in e} \sum_{i=1}^{HW} \boldsymbol{O}^i_{v,e} \left\| \tilde{\boldsymbol{P}}^i_v - \sigma_e \boldsymbol{T}_e \boldsymbol{P}^i_{v,e} \right\| . \tag{1}$$

where for each image pair $e = (v, u) \in \mathcal{E}$, σ_e is scale factor, $\boldsymbol{P}_{v,e}$ and $\boldsymbol{O}_{v,e}$ is the pointmap and confidence map of v, respectively. *However, MASt3R builds a complete graph for this optimization, resulting in a complexity of $\mathcal{O}((\frac{N}{k})^2)$, which becomes prohibitively inefficient for long video sequences.*

To enhance efficiency, rather than naively using a MASt3R-like method, we build a more efficient sub-graph that connects each key frame only to its four closest neighboring frames. *This design is motivated by the observation that neighboring segments share greater co-visibility; hence, edges between key frames with distant neighbors can be safely pruned.* This sub-graph greatly reduces optimization complexity to $\mathcal{O}(4\frac{N}{k})$ (scaling linearly with the number of frames N), yet still achieves high end-to-end reconstruction quality.

Efficient intra-Fragment Feature Registration (Challenge ③): Finally, we aim to obtain an initial estimate of the local camera poses and pointmaps with depth scale factors within each non-overlapping fragment, which can accelerate and boost the quality of the subsequent 3D Gaussian construction. *A naive solution is to follow InstantSplat [5] to use MASt3R for full intra-fragment problem, but it requires optimizing millions of points and camera poses, making it inefficient.*

To enhance efficiency, we use the pre-obtained key frame

information and obtain the needed information by considering only the pairwise relationships between the key frame and all subsequent frames in the same fragment. In this way, we only need to solve for 6-dimensional camera poses (in quaternion format) and a 1-dimensional scale factor for each view. We found it sufficient to maintain end-to-end reconstruction quality with high efficiency. Moreover, this simplified, non-sequential matching approach can reduce the incremental errors that are commonly encountered in sequential matching (see Fig. 2).

Take the first fragment $\mathcal{I}_1^f = \{I_1, \ldots, I_k\}$ as an example: *Camera pose*: We refine the relative camera poses within the fragment using initial pairwise estimates. First, we identify the intersection of 2D correspondences between the key frame I_1 and each subsequent frame from index 2 to k. This process yields a consistent set of correspondences across all frames in the fragment. Using these intersected 2D correspondences, we retrieve the corresponding 3D positions from the key frame, which were previously optimized during key frame processing. These 3D-2D correspondences are then input to PnP-RANSAC [6], refining the camera poses to ensure alignment with consistent 3D points across views within the fragment. Only a 6-dim quaternion pose is optimized instead of directly optimizing pointmaps.
Scale factor: Scale variations may persist within the point clouds of the fragment due to independent inference. To address this, intersected 3D points from the key frame are utilized for scale estimation across all image pairs. Specifically, for each image pair between I_1 and $\{I_i\}_{i=2}^k$, the corresponding 3D point positions (with a total of P points) are retrieved. A one-degree-of-freedom (1-DoF) scale factor is computed between the intersected 3D points in the current pair and those from the key frame:

$$s_i = \text{median}(\left\{\|\mathbf{p}_n^{(1)}\| / \|\mathbf{p}_n^{(i)}\|\right\}_{n=1}^{P}), \tag{2}$$

which can be solved analytically (i.e., by taking the median) with *no optimization required*. This scale factor is applied to the dense pointmaps of I_i in the subsequent stage, ensuring that the point clouds within the fragment are locally aligned and maintain a consistent scale relative to the key frame.

By solving a simplified optimization problem rather than naively employing MASt3R global optimization as in InstantSplat, our method achieves a processing time of 2.97 seconds compared to 10.33 seconds—a 3× reduction in computational time within fragment k=4.

Overall Efficiency/Quality Improvement with Our Novel Fragment Registration: In contrast, to naively applying MASt3R's global optimization as InstantSplat [5] to compute point clouds and camera poses for all images, our method significantly enhances **efficiency**. As in Tab. 2, their global optimization approach runs out of memory beyond 64 views and incurs up to an 18× longer inference time. Moreover, Tab. 5 demonstrates that replacing our fragment registration with MASt3R initialization yields lower-quality results. This is primarily because it tries to solve a more complex global problem with less accurate pointmaps/poses outputs, whereas our approach eases optimization complexity by providing a more robust initialization for subsequent 3D-GS, delivering **better quality** (PSNR 27.91→30.02).

# Views	MASt3R (Global Optimization)		VideoLifter (Sec. 3.2)	
	Time (min)	Peak GPU Mem (GB)	Time (min)	Peak GPU Mem (GB)
32	11	4.75	1.7	4.45
48	33	8.86	2.6	5.43
64	63	14.44	3.5	6.38
128	OOM	OOM	7.9	16.9

Table 2. Time and peak GPU memory usage on an A6000 for varying view counts: direct using MASt3R global optimization vs. our Sec. 3.2.

3.3. Hierarchical Gaussian Alignment

In this stage, we perform dense 3D scene reconstruction using Gaussian Splatting. First, we construct *local 3D Gaussians* within each fragment, and then merge these local models via *hierarchical Gaussian alignment*. The key design question is *how to construct a globally coherent 3D scene while preserving local scene details without incurring significant alignment errors*.

Local 3D Gaussian Construction: We initialize a set of Gaussians, denoted as $\mathcal{G}^f = \{G_i^f\}_{i=1}^m$, where G_i^f is independent initialized and optimize from fragment $\mathcal{I}_i^f$.
Guassian initialization: In the local fragment registration step, we obtain the key frame's dense point cloud, along with the relative poses and scale factors for the other frames, which can be used to obtain entire point map within fragment. To initialize G_i^f, we then assign a Gaussian to each point in the pixel-wise point cloud, setting its attributes as: color based on the corresponding pixel, center at the 3D point location, opacity adhering to the 3D-GS protocol [14], and scaling such that it projects as a one-pixel radius in the 2D image (by dividing the depth by the focal length). We set Gaussians as isotropic to reduce the degrees of freedom in Gaussian training.
Further refinement: The initial camera poses and point cloud positions may contain minor inaccuracies, we further refine them through joint optimization of camera poses and Gaussian parameters. Specifically, for each local Gaussian in G_i^f, we randomly sample frames within the fragment, render the current Gaussians into sampled frame, and backpropagate gradient updates to the Gaussian positions, colors, scales, opacities, and camera poses.

Hierarchical Gaussian Alignment: Next, we merge the local fragment-level 3D Gaussian sets $\mathcal{G}^f = \{G_i^f\}_{i=1}^m$ to the final consistent 3D scene. Naive pairwise progressive merging poses Challenge ①: an excessive number of Gaussians

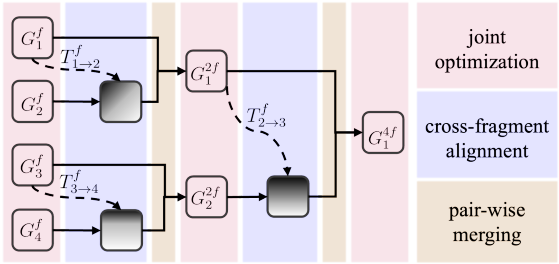

Figure 4. **Hierarchical Gaussian Alignment.** The process iteratively performs three stages: 1) joint optimization of camera poses and local Gaussians (pink), 2) cross-fragment alignment for new local Gaussian (purple), and 3) visibility masking and pairwise merging of local Gaussians (yellow), until a globally consistent scene reconstruction is achieved.

for optimization and ②: inconsistencies among various local Gaussian sets. To avoid these issues, we propose a tree-based hierarchical pipeline (Fig. 4) that iteratively performs three key processes.

1) Inter-Fragment Alignment with Key Frame Guidance (Fig. 4 purple): To merge two independently optimized fragments (e.g., G_1^f and G_2^f), we first perform **cross-fragment alignment** to ensure a faithful merging by using G_1^f as the reference coordinate system. In each fragment, the key frame, i.e., the first frame, is assigned an identity pose, and the remaining frames are defined by their relative poses to this key frame. In Section 3.2, we enforce consistency between key frames and obtain the initial transformation $T_{1\to2}^f$. We then use this information as a guide to compute the camera poses for the novel frames covered by G_2^f. By enforcing photometric loss on the next novel view while freezing all parameters in G_1^f, we further optimize $T_{1\to2}^f$ into $T_{1\to2}^{f*}$. Then, we align the Gaussians in G_2^f with the coordinate system of G_1^f by using ${T_{1\to2}^{f*}}^{-1}$. As shown in Tab. 5, omitting key frame guidance leads to prolonged optimization and degraded performance.

2) Pair-wise merging with visibility-mask-driven Gaussian pruning (Fig. 4 yellow): To avoid duplicating Gaussians in regions where G_1^f already provides adequate scene reconstruction, with $\mathbf{p}$ (the pixel position on image plane), we use a **visibility mask** to determine areas that G_1^f could faithfully reconstruct: $M(\mathbf{p}) = \text{Conf}(\mathbf{p}) > \beta + D(\mathbf{p}) > 0$. where $D(\mathbf{p})$ is rendered depth. $\text{Conf}(\mathbf{p}) = \sum_i \alpha_i(\mathbf{p}) \prod_{j=1}^{i-1} (1 - \alpha_j(\mathbf{p}))$ is the rendered confidence, e.g. if a pixel contains information from the current gaussian, and threshold β determines the masking criteria.

Fig. 5 shows that the visibility mask effectively identifies regions where Gaussians from G_1^f could provide sufficient depth and confidence. Importantly, the mask remains robust to occlusions, even when covered by large Gaussians. As our initialization is pixel-wise point clouds, the inverse of the visibility mask can be directly applied to G_2^f to select Gaussians that complement the missing regions of G_1^f. These selected Gaussians inherit previously optimized parameters, ensuring seamless integration into a unified representation.

Figure 5. **Visibility Mask** showing rendering G_1^f into next two novel frames in G_2^f. White region denotes faithful reconstruction using G_1^f, while black represents pixels unseen from G_1^f. With visibility mask, we select complementary gaussians from G_2^f, merging them with G_1^f into G_1^{2f}.

3) Joint optimization (Fig. 4 pink): After merging pair of local Gaussians, further **joint optimization** is needed to ensure the merged Gaussian set G_1^{2f} meets our consistency objectives. We jointly optimize Gaussian properties and camera poses for G_1^{2f}. Specifically, we randomly sample frames within $[I_1, I_8]$, render all Gaussians into each frame, and backpropagate gradients to Gaussian positions, colors, scales, opacities, and camera poses.

4. Experiments

4.1. Experimental Setup

Datasets: We carry out comprehensive experiments on various real-world datasets, including Tanks and Temples [15] and CO3D-V2 [26]. For Tanks and Temples, we evaluate under **two settings**. The first follows prior works such as NoPe-NeRF [3] and CF-3DGS [7], using a subset of up to 350 frames. The second leverages much longer video sequences (∼1,000 frames), which cover substantially larger scene areas. This longer-sequence setting is designed to stress-test scalability and robustness in handling extended trajectories. In both settings, we assess novel view synthesis quality and pose estimation accuracy across 8 scenes spanning both indoor and outdoor environments. In each case, we use 7 out of every 8 frames from the video clips as training data and evaluate the novel view synthesis on the remaining frames except Family. For CO3D-V2, which contains thousands of object-centric videos with cameras orbiting the object, recovering camera poses is significantly more challenging due to the complex, large camera motions *We follow the experimental settings of CF-3DGS [7]* to select the same 10 scenes from different object categories and apply the same procedure to divide training and testing sets.

Metrics: We assess our approach on two key tasks: generating novel views and estimating camera poses. For the task of novel view synthesis, we evaluate performance using common metrics such as PSNR, SSIM [36], and LPIPS [40].

	Camera Param.	Train Time	SSIM ↑	PSNR ↑	LPIPS ↓	ATE ↓
COLMAP+3DGS	GT K & Pose	∼50min	0.9175	30.20	0.1025	-
InstantSplat [5]* (MASt3R MVS+3DGS)	-	14min56s	0.5617	18.28	0.488	0.021
NeRF-mm [37]	-	∼13h33min	0.5313	20.02	0.5450	0.035
BARF [17]	GT K	∼20h	0.6075	23.42	0.5362	0.078
NoPe-NeRF [3]	GT K	∼30h	0.7125	25.49	0.4113	0.013
CF-3DGS [7]	GT K	∼2h20min	0.9213	31.14	0.0859	**0.004**
Ours	**-**	**26min20s**	**0.9347**	**31.59**	**0.0730**	**0.004**

Table 3. **Quantitative Evaluations on Tanks and Temples Dataset.** Our method achieves superior rendering quality and pose accuracy while requiring minimal training time and no camera parameters. (-) indicates no camera parameters required, GT K indicates known intrinsics, GT K & Pose indicates both known intrinsics and extrinsics. * **InstantSplat cannot process dense views directly due to OOM (see Tab. 2); thus, we adopt its chunk-by-chunk version, which yields inferior quality on long-sequence videos.**

In terms of camera pose estimation, we evaluate Absolute Trajectory Error (ATE) [31] and utilize COLMAP-generated poses from all dense views as our ground-truth. While Relative Pose Error (RPE) [31] evaluates the local consistency of relative transformations between consecutive frames, it can be sensitive to discrepancies in intrinsic parameters such as focal length. ATE provides a more comprehensive measure of global trajectory accuracy and is better aligned with the goals of our method, which emphasizes globally consistent 3D reconstruction [31]. As such, we prioritize ATE as the primary metric for evaluating the poses of `VideoLifter`.

Implementation Details: During fragment registration, each fragment consists of $k = 4$ frames. For depth map prediction, we utilize MASt3R with a resolution of 512 on the longer side. We run 200 iterations for key frame optimization. For hierarchical Gaussian alignment, we initialize each local Gaussian using the number of pixels within the fragment and train it for 200 steps. Camera poses are represented in quaternion format. For pair-wise merging, the transformation matrix from key frame optimization is applied to the camera poses and Gaussian points of the subsequent local Gaussian. First, the camera poses are optimized with a learning rate of 1e-3 for 200 steps. Next, a mask is rendered to identify inadequately reconstructed regions, where new Gaussians are added. This process is repeated iteratively until a globally consistent Gaussian representation is achieved. We uniformly sample $\frac{1}{2}$ and $\frac{1}{4}$ training views on Tanks and Temples and CO3D-V2, respectively. All experiments were conducted on a single Nvidia A6000 GPU to maintain fair comparison.

4.2. Quantitative Evaluations

To quantitatively evaluate the quality of synthesized novel views, we present the results in Tab. 3 for the Tanks and Temples dataset and Tab. 4 for the CO3D-V2 dataset. Baseline models were re-trained using their officially released code to ensure a fair comparison of training time. Compared to other self-calibrating radiance field methods, our approach achieves superior performance in terms of efficiency and rendering quality, which is largely thanks to our decoupled fragment registration and hierarchical alignment process. Compared to the most relevant baseline CF-3DGS [7], we reduce >80% training time yet get >0.012 LPIPS improvement on Tanks and Temples, and reduce >85% training time yet get >0.12 LPIPS improvement on CO3D-V2 dataset. Note that our `VideoLifter` does not require any ground-truth camera parameters, making it more adaptable to scenes that do not have or fail to get precomputed intrinsics from COLMAP. Compared to NeRFmm [37], which also does not need ground-truth camera parameters, our `VideoLifter` delivers much better quality and much less training time. Quantitative results on the longer-sequence setting, as well as detailed per-scene breakdowns, are provided in the Supplementary.

4.3. Qualitative Evaluations

As shown in Fig. 6, for large-scale scenes from the Tanks and Temples dataset, thanks to the hierarchical design in `VideoLifter`, our method consistently produces sharper details among all test views, and preserves fine details that are well-optimized within each fragment. For the CO3D-V2 dataset, which includes 360-degree scenes with complex trajectories, achieving a globally consistent 3D reconstruction without any COLMAP initialization is even more challenging. Baselines that rely on monocular depth prediction to unproject images into point clouds often suffer from depth scale inconsistencies, making them fragile and prone to failure. Even CF-3DGS, which uses the more robust ZoeDepth monocular depth estimator [2], encounters severe failures on CO3D-V2. In contrast, `VideoLifter` leverages 3D geometry priors to achieve robust registration, making it highly adaptable and resilient in challenging settings. More results can be found in the Supplementary.

4.4. Ablation Study

Tab. 5 reports the impact of various design choices on training time and reconstruction quality.

Local Fragment Registration. Replacing it with direct MASt3R multi-view stereo initialization increases training time and lowers reconstruction quality, suggesting that out-

	Camera Param.	Train Time	SSIM ↑	PSNR ↑	LPIPS ↓	ATE ↓
COLMAP+3DGS	GT K & Pose	15min44s	0.9211	32.26	0.1662	-
InstantSplat [5]* (MASt3R MVS+3DGS)	-	19min3s	0.6400	18.48	0.5355	0.045
NeRF-mm [37]	-	∼17h22min	0.4380	13.43	0.7058	0.061
NoPe-NeRF [3]	GT K	∼35h	0.7030	25.54	0.5190	0.055
CF-3DGS [7]	GT K	∼2h55min	0.6821	22.98	0.3515	0.014
Ours	-	**24min58s**	**0.8502**	**28.37**	**0.2237**	**0.012**

Table 4. **Quantitative Evaluations on CO3D-V2 Datasets.** (-) indicates no camera parameters required, GT K indicates known intrinsics.

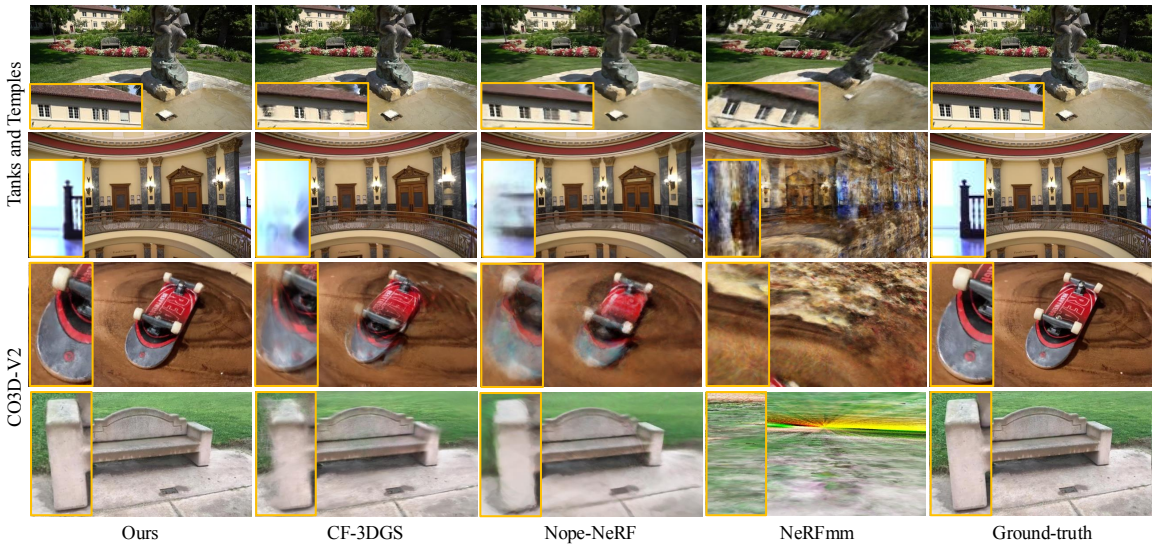

Figure 6. **Visual Comparisons** with other baselines on TT and CO3D-V2 datasets. The insets highlight the details of renderings. `VideoLifter` achieves faithful 3D reconstruction, preserves better details, and alleviates incremental error in progressive learning.

Model	Train Time	SSIM ↑	PSNR ↑	LPIPS ↓
Ours($k = 4$, $\beta = 0.9$)	28min49s	0.8957	30.02	0.1745
Local: Use MASt3R MVS Init. [5]	35min	0.8582	27.91	0.1768
Global: Hierarchy → sequential	53min12s	0.6969	20.22	0.3876
Global: Remove Key Frame Guidance	35min42s	0.7629	24.35	0.3433
$k = 2$	35min2s	0.8936	29.77	0.2138
$k = 4$	28min49s	0.8957	30.02	0.1745
$k = 8$	38min5s	0.8787	27.51	0.2743
$\beta = 0.5$	23min18s	0.6529	18.50	0.4457
$\beta = 0.9$	28min49s	0.8957	30.02	0.1745
$\beta = 0.99$	43min55s	0.8325	29.08	0.2691

Table 5. **Ablation studies on 34_1403_4393 scene from CO3D-V2 Dataset.** k denotes the number of frames in local fragment. β denotes the rendering confidence threshold in Gaussian merging.

puts directly from MASt3R are less accurate and introduce errors to Gaussian optimization, especially in the challenging video setup. While MASt3R serves as the most direct baseline, we further compare against more recent end-to-end multi-view stereo methods, including CUT3R [35] and VGGT [34], to contextualize our approach within stronger alternatives (refer to the Supplementary).

Hierarchical Gaussian Alignment. Removing our hierarchical alignment and instead adding local Gaussians sequentially (as in CF-3DGS [7]) prolongs training and hurts performance, showing the efficiency and accuracy gains from our hierarchical design.

Key Frame Guidance. Omitting key frame guidance forces additional time for pose optimization without achieving optimal performance, showing the crucial role of key frames in stabilizing and accelerating the merging process.

Fragment Size k. A smaller k yields more precise intra-fragment registration but complicates fragment alignment, whereas a larger k reduces joint correspondences within fragment and degrades relative pose estimation, thus degrading the performance.

Confidence Threshold β. Low β allows fewer Gaussians to merge, leading to under-reconstructed areas, while a high β merges too many Gaussians, slowing down training.

5. Conclusion

We presented VideoLifter, a framework for efficient 3D scene reconstruction from monocular videos without relying on pre-computed camera poses or known intrinsics. By leveraging learning-based stereo priors and a hierarchical alignment strategy with 3D Gaussian splatting, VideoLifter produces dense, globally consistent reconstructions with significantly reduced computational overhead compared to prior methods [3, 7].

References

[1] Shai Avidan and Amnon Shashua. Novel view synthesis in tensor space. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1034–1040. IEEE, 1997. 2

[2] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 7

[3] Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4160–4169, 2023. 1, 2, 6, 7, 8

[4] Zezhou Cheng, Carlos Esteves, Varun Jampani, Abhishek Kar, Subhransu Maji, and Ameesh Makadia. Lu-nerf: Scene and pose estimation by synchronizing local unposed nerfs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18312–18321, 2023. 2

[5] Zhiwen Fan, Wenyan Cong, Kairun Wen, Kevin Wang, Jian Zhang, Xinghao Ding, Danfei Xu, Boris Ivanovic, Marco Pavone, Georgios Pavlakos, et al. Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds. *arXiv preprint arXiv:2403.20309*, 2024. 1, 2, 4, 5, 7, 8

[6] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 5

[7] Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A Efros, and Xiaolong Wang. Colmap-free 3d gaussian splatting. *arXiv preprint arXiv:2312.07504*, 2023. 1, 2, 3, 6, 7, 8

[8] Yasutaka Furukawa, Brian Curless, Steven M Seitz, and Richard Szeliski. Towards internet-scale multi-view stereo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 1434–1441. IEEE, 2010. 2

[9] Riccardo Gherardi, Michela Farenzena, and Andrea Fusiello. Improving the efficiency of hierarchical structure-and-motion. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1594–1600. IEEE, 2010. 2

[10] Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3279–3286, 2015. 2

[11] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 4

[12] Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Anima Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5846–5854, 2021. 2

[13] Kaiwen Jiang, Yang Fu, Yash Belhe, Xiaolong Wang, Hao Su, Ravi Ramamoorthi, et al. A construct-optimize approach to sparse view synthesis without camera pose. *arXiv preprint arXiv:2405.03659*, 2024. 2

[14] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (ToG)*, 42(4): 1–14, 2023. 1, 2, 5

[15] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4): 1–13, 2017. 6

[16] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. *arXiv preprint arXiv:2406.09756*, 2024. 2, 4

[17] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5741–5751, 2021. 1, 2, 7

[18] Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5987–5997, 2021. 1

[19] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 2

[20] Jinjie Mai, Wenxuan Zhu, Sara Rojas, Jesus Zarzar, Abdullah Hamdi, Guocheng Qian, Bing Li, Silvio Giancola, and Bernard Ghanem. Tracknerf: Bundle adjusting nerf from sparse and noisy views via feature tracks. *arXiv preprint arXiv:2408.10739*, 2024. 2

[21] Andreas Meuleman, Yu-Lun Liu, Chen Gao, Jia-Bin Huang, Changil Kim, Min H Kim, and Johannes Kopf. Progressively optimized local radiance fields for robust view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16539–16548, 2023. 1, 2, 3

[22] Ben Mildenhall, Pratul P Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 2

[23] 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, 2

[24] Pierre Moulon, Pascal Monasse, and Renaud Marlet. Global fusion of relative motions for robust, accurate and scalable structure from motion. In *Proceedings of the IEEE international conference on computer vision*, pages 3248–3255, 2013. 1

[25] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. Dtam: Dense tracking and mapping in real-time. In *2011 international conference on computer vision*, pages 2320–2327. IEEE, 2011. 2

[26] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10901–10911, 2021. 6

[27] Erik Sandström, Keisuke Tateno, Michael Oechsle, Michael Niemeyer, Luc Van Gool, Martin R Oswald, and Federico Tombari. Splat-slam: Globally optimized rgb-only slam with 3d gaussians. *arXiv preprint arXiv:2405.16544*, 2024. 3

[28] 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, 2

[29] Xuelun Shen, Zhipeng Cai, Wei Yin, Matthias Müller, Zijun Li, Kaixuan Wang, Xiaozhi Chen, and Cheng Wang. Gim: Learning generalizable image matcher from internet videos. *arXiv preprint arXiv:2402.11095*, 2024. 4

[30] Noah Snavely. Scene reconstruction and visualization from internet photo collections: A survey. *IPSJ Transactions on Computer Vision and Applications*, 3:44–66, 2011. 2

[31] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 573–580. IEEE, 2012. 7

[32] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021. 4

[33] Prune Truong, Marie-Julie Rakotosaona, Fabian Manhardt, and Federico Tombari. Sparf: Neural radiance fields from sparse and noisy poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4190–4200, 2023. 2

[34] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 8

[35] Qianqian Wang*, Yifei Zhang*, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *CVPR*, 2025. 8

[36] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6

[37] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021. 1, 2, 7, 8

[38] Changchang Wu et al. Visualsfm: A visual structure from motion system, 2011. *URL http://www. cs. washington. edu/homes/ccwu/vsfm*, 14:2, 2011. 2

[39] 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. 2

[40] 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