LVSPM: Long Sequence View Synthesis and Pose Estimation Model

Xi Chen[1], Yachi Zhang[2], Linghao Chen[2], Minghua Liu[2], Hao Su[2], Zexiang Xu[2], and Xiaoshuai Zhang[2]

[1] UC San Diego
[2] Sudo AI GmbH

Abstract. We present LVSPM, a generalizable model that jointly estimates camera poses and synthesizes novel views from uncalibrated image collections. Trained with only RGB images and pose supervision, LVSPM avoids dense 3D ground truth and employs test-time training (TTT) layers to scale seamlessly to hundreds of input views. On RealEstate10k, Co3Dv2, and DL3DV, LVSPM surpasses VGGT in pose estimation across 16–256 views, with especially large margins at strict thresholds. For novel view synthesis under a practical protocol where more views cover larger scenes, LVSPM achieves state-of-the-art pose-free quality—surpassing even pose-dependent models in PSNR—and still maintains high quality as scene scale grows, while baselines collapse. The code will be available at https://burningdust21.github.io/Projects/LVSPM/.

Keywords: Novel View Synthesis · Pose Estimation · Test-Time Training

1 Introduction

Novel view synthesis (NVS) has been a long-standing challenge and a crucial driving force in 3D vision and graphics research. From per-scene optimization-based pipelines [2,7,21,29,42,43,81] to feed-forward approaches [6,8,11,20,27,80, 83], the community has developed diverse 3D representations and model architectures in recent years in pursuit of increasingly efficient, accurate, and scalable solutions. However, nearly all existing methods assume access to known camera poses, typically estimated through structure-from-motion (SfM) pipelines [53, 56], which are computationally expensive and often fragile in real-world scenarios.

To avoid the reliance on known poses, several recent attempts have explored pose-free view synthesis with feed-forward models [26, 78, 85]. While promising, these methods are mostly restricted to sparse-view input and cannot scale to long sequences, which are crucial for capturing wide scene coverage and enabling immersive experiences. In parallel, geometry-based models such as DUSt3R and VGGT [63, 68] have shown strong feed-forward performance on pose estimation, but they do not address view synthesis and rely on dense 3D supervision such

Fig. 1: LVSPM can jointly achieve high-quality novel view synthesis and accurate camera pose estimation from uncalibrated long-sequence (128) multi-view images. A subset of the 128 input views with corresponding prediction and ground-truth poses is shown in the middle, while novel view synthesis comparisons are presented on the side. Our estimated poses closely align with ground truth, while our pose-free synthesis results surpass recent baselines, including the pose-free AnySplat [26] and even the pose-required DepthSplat [73].

as point maps that are far more expensive and less scalable to obtain than the image supervision used in classical view synthesis.

This leaves open the challenge of achieving long-sequence pose-free view synthesis without requiring heavy dense 3D labels. In this work, we address this challenge by introducing **a novel feed-forward framework, LVSPM, that jointly achieves long-sequence novel view synthesis and camera pose estimation from unposed inputs**, using only RGB image and camera pose supervision. As illustrated in Fig. 1, our approach produces high-fidelity novel renderings and accurate camera poses from hundreds of captured images of a large real scene, achieving practical and scalable long-sequence view synthesis with integrated camera calibration.

To this end, our design philosophy follows the spirit of LVSM [27], aiming to minimize 3D inductive biases by framing the task as sequence-to-sequence token prediction. Specifically, given a sequence of unposed input views, our model tokenizes the images and directly predicts tokens corresponding to both input-view camera parameters and novel-view RGB images, without relying on explicit 3D representations or handcrafted geometric modules. To scale this to long sequences, we adopt a LaCT backbone [86]—a recent test-time training (TTT) architecture that performs large-chunk TTT updates for efficient and scalable long-sequence modeling. In addition, inspired by VGGT, we incorporate per-view camera tokens into the LaCT architecture, enabling direct pose estimation within the same sequence modeling framework. Together, these components lead to a novel feed-forward system that unifies long-sequence view synthesis and pose estimation within a minimal-bias, scalable framework.

We train our model on multiple large-scale synthetic and real datasets using only RGB images and camera pose supervision, and demonstrate effective scalability to 256 input views at inference. Our contributions are as follows:

- We introduce LVSPM, the first pose-free view synthesis model that scales to 256 input views, enabled by a LaCT backbone with learnable camera tokens that jointly predicts camera poses and novel views in a unified sequence-to-sequence framework.
- We demonstrate that novel view synthesis supervision provides sufficient geometric signal for accurate pose estimation, eliminating the need for dense 3D ground truth. Our ablation confirms this: removing NVS supervision reduces the AUC3 from 71.9 to 50.9, while the full model surpasses geometry-based methods trained with dense point-map supervision.
- We propose the equi-temporal evaluation protocol that isolates scene-scale effects from sampling density, providing a more realistic benchmark for long-sequence methods.
- LVSPM achieves state-of-the-art results on RE10k, Co3Dv2, and DL3DV for both long sequence pose estimation and view synthesis, surpassing even pose-dependent baselines such as DepthSplat [73], while scaling gracefully to large scenes where prior methods collapse.

These findings highlight the feasibility of addressing fundamental 3D vision problems with cheaper supervision, opening new possibilities for scalable and generalizable 3D perception systems.

2 Related Work

Novel View Synthesis. View synthesis has been extensively studied for decades in computer vision and graphics [4,13,18,20,27,29,33,42,50,88]. In recent years, NeRF [42], 3D Gaussian Splatting(3DGS) [29], and many variants of such 3D representations [7,17,21,43,59,76,81] with differentiable rendering have achieved photo-realistic results through end-to-end optimization, albeit at the cost of significant computational overhead per scene. To enable faster inference, numerous feed-forward methods have been developed for instant scene reconstruction and rendering, most of which rely on 3D representations and 3D-related architectural designs such as plane-sweep volumes [8, 11, 28, 39, 87] or epipolar priors [6,58,65,80]. Recently, large reconstruction models (LRMs) [20,34,64,70,83] have begun to reduce such architectural-level 3D biases, leveraging pure transformer architectures, while LVSM [27] further eliminates representation-level bias and achieves state-of-the-art view synthesis quality. Our method inherits this minimal-bias philosophy but extends it to the joint problem of view synthesis and pose estimation, whereas most existing approaches still assume known camera poses.

On the other hand, most prior feed-forward methods remain restricted to sparse input views. Recent works such as Long-LRM [89] and LaCT [86] have begun to explore long-sequence inputs with tens or even hundreds of images using

architectures like Mamba [12, 19] and TTT [60]; however, these approaches still assume known camera poses, which limits their practicality. Our approach builds upon LaCT and extends it to the pose-free setting, enabling joint long-sequence view synthesis and pose estimation.

Camera Pose Estimation. Camera pose estimation has traditionally relied on Structure-from-Motion (SfM) pipelines [53,56], which remain robust for large-scale reconstructions but suffer from heavy computational cost and failure modes in textureless regions or repetitive structures. These issues persist even with the advent of learning-based feature extractors [14, 15, 47] and matchers [40, 51, 52]. Recently, many learning-based approaches have attempted to directly regress camera poses [5,35,49], though the current state of the art comes from geometry-driven transformer-based models [37, 63, 68]. In particular, DUSt3R [68] formulates pairwise 3D reconstruction as point map regression, enabling pose-free 3D reconstruction and subsequent camera estimation via PnP. Numerous follow-up works have extended DUSt3R, improving its inference quality and scalability to longer sequences [9, 32, 61, 66, 77, 82]. More recently, VGGT [63] represents a major step forward, introducing a feed-forward multi-task transformer that jointly predicts cameras, depth, and point maps, improving performance significantly across several geometry tasks, including pose estimation. DepthAnything3 [37] extends the feed-forward visual geometry paradigm of VGGT with a simpler plain-transformer architecture and a unified depth-ray prediction target, together with a teacher-student training strategy for dense geometry supervision, achieving state-of-the-art performance. However, these approaches primarily focus on geometric reconstruction, depend heavily on dense 3D labels, and do not directly address the problem of view synthesis. Our method takes inspiration from VGGT's camera estimation mechanism but integrates it into a view synthesis framework, achieving comparable or superior pose estimation results while eliminating the need for dense geometric ground truth.

Pose-Free View Synthesis. Many recent works have sought to remove the requirement of known camera poses in view synthesis to improve practicality. Early attempts integrated pose estimation into optimization-based frameworks such as NeRF, jointly estimating poses and scene representations during training [3, 10, 36, 62, 71]. Among them, RayZer [25] studies self-supervised novel view synthesis from unposed images by using self-predicted camera poses for rendering. However, these cameras are primarily designed to support view synthesis rather than being directly formulated as world-aligned pose outputs. In contrast, our method jointly estimates world-aligned cameras and synthesizes novel views over long image sequences, scaling to hundreds of input views. Recently, feed-forward approaches have gained traction. While some methods adopt a two-stage pipeline that first estimates cameras and then performs view synthesis [23, 25, 85], others aim to predict poses and novel views jointly. Early pose-free approaches typically achieve only sparse-view, object-level reconstruction and rendering [24, 64, 72]. Building on the success of DUSt3R and 3D Gaussian Splatting, subsequent works have extended this idea to scene-level pose-free view synthesis by directly reconstructing Gaussian point clouds from unposed inputs

and recovering poses via PnP [16,22,55,75,78]. However, these approaches generally rely on dense geometric supervision and remain limited to only a handful of input views (often fewer than ten). More recently, AnySplat [26] extends VGGT to the view synthesis task, supporting several tens of input views, but it depends on VGGT pretraining and still requires dense 3D labels for supervision. In contrast, our method outperforms AnySplat under its setting and scales effectively to hundreds of input images. Notably, our model is trained from scratch using only RGB image and camera pose supervision, without requiring additional dense 3D geometric labels.

3 Method

Our work, LVSPM, introduces a feed-forward model designed to jointly estimate camera poses and synthesize novel views from a sequence of unposed images. The model architecture minimizes explicit 3D inductive biases, framing the problem as a sequence-to-sequence prediction task. We leverage a Large-Chunk Test-Time Training (LaCT) backbone [86] to efficiently process long input sequences, coupled with learnable camera tokens for direct camera pose and intrinsic parameter regression. This section details the model's input representation, architecture, and the supervision strategy used for training.

3.1 Problem Formulation and Overview

Given a set of N uncalibrated input images $\mathcal{I} = \{I_i\}_{i=1}^N$, where $I_i \in \mathbb{R}^{H\times W\times 3}$, our goal is to:

- Estimate camera poses $\{E_i = (t_i, r_i)\}_{i=1}^N$, as well as intrinsic parameters $\{K_i = (f_i^x, f_i^y)\}_{i=1}^N$, where $t_i \in \mathbb{R}^3$ is the 3-dim translation, $r_i \in \mathbb{R}^4$ is the 4-dim quaternion, and $\hat{f}_i^x, \hat{f}_i^y$ are scalars denoting the normalized camera focal lengths with respect to the image size;
- Synthesize novel view I^t from target novel camera configurations (E^t, K^t).

This joint formulation eliminates the dependency on pre-computed structure-from-motion (SfM) pipelines while enabling end-to-end learning of geometric and appearance modeling.

As shown in Fig. 2, LVSPM consists of L stacked LaCT blocks, each containing three main components: (1) a large-chunk test-time training (TTT) layer for long-range multi-view reasoning, (2) a window attention layer for local spatial dependencies, and (3) a feed-forward MLP network for channel mixing. The architecture processes variable-length sequences up to 1M tokens through an efficient test-time adaptation mechanism.

3.2 Model details

Input representation and tokenization. Unlike prior pose-required view synthesis models [27,83] that typically condition both input and target views on

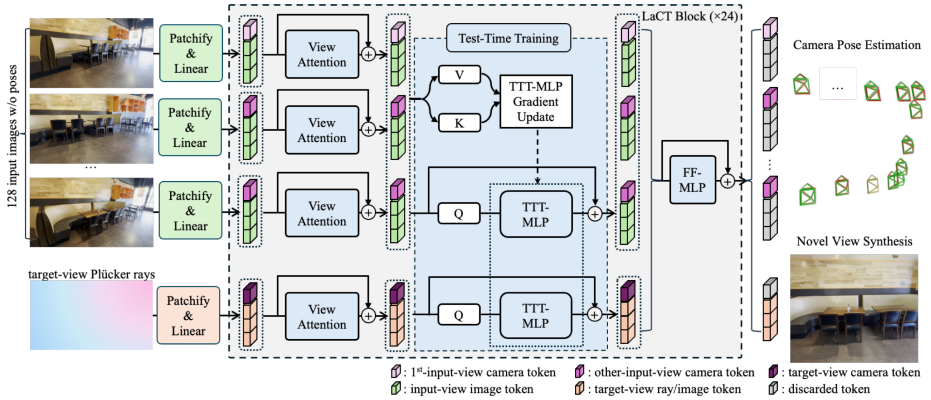

Fig. 2: Pipeline. Our approach takes uncalibrated long-sequence input views, where each image is tokenized into patch tokens and augmented with a learnable camera token, while target views are represented by Plücker-ray tokens. All tokens are processed through a stack of LaCT blocks that combine per-view self-attention, an MLP-based TTT layer, and a feed-forward MLP layer. From the final tokens, our lightweight decoders produce novel-view RGB images and camera parameters, enabling accurate pose estimation together with high-quality long-sequence view synthesis.

Plücker rays [45], our pose-free approach ignores input-view Plücker encoding and instead introduces additional learnable camera tokens to extract camera information. Specifically, each input image I_i is divided into non-overlapping patches of $p \times p$ pixels, noted as $\{I_{ij} \in \mathbb{R}^{p\times p\times 3} | j = 1, \ldots, HW/p^2\}$. These patches are then directly projected to the model dimension d with a linear layer: $x_{ij} = \text{Linear}_{input}(I_{ij})$. Inspired by VGGT [63], we append each input image with a learnable camera token c_i for pose prediction and intra-view context exchange. The first view is used as the canonical reference frame, assuming a zero translation and an identity rotation, and has a special initial learnable camera token c_r to represent the reference frame. All other input views share another learnable camera token c_x. To ensure a consistent coordinate system across scenes, the translation scale is normalized by setting the distance between the first and furthest points to unit length. This scene normalization, derived from input views, is then applied to target views, aligning both sets of views in a consistent coordinate space.

The input-view tokens X_i for input view i are represented as:

$$X_i = \{c_i, x_{i1}, x_{i2}, \ldots, x_{iM}\},\ c_1 = c_r, \quad c_i = c_x \text{ for } i = 2, \ldots, N. \tag{1}$$

where $M = HW/p^2$. For simplicity, we denote by X_{ij} an arbitrary token from input view i, where $X_{i0} = c_i$ corresponds to the camera token, and $X_{ij} = x_{ij}$ for $j > 0$ corresponds to the image tokens.

On the other hand, we still adopt Plücker rays to condition target views, providing the 3D camera context required for novel-view synthesis. More specifically, we compute the target frame Plücker rays from its camera parameters

(E^t, K^t). For the k-th target view, its Plücker ray patches are noted as $P_k^t = \{P_{kj}^t | j = 1, \ldots, HW/p^2\}$. These Plücker ray patches are transformed into tokens using another linear layer, similar to how we encode the image patches of the input views: $x_{kj}^t = \text{Linear}_{target}(P_{kj}^t)$. To be consistent with input view encoding, these target tokens are concatenated with another learnable token c_t before feeding into the network. The target view tokens T_k for target view k is represented as:

$$T_k = \{c_k, x_{k1}^t, x_{k2}^t, \ldots, x_{kM}^t\}, \quad c_k = c_t \tag{2}$$

Similar to input-view token X_{ij}, we denote by T_{kj} an arbitrary token from target view k.

LaCT blocks. We adopt the recent Large-Chunk Test-Time Training (LaCT) architecture [86] to process both input and target view tokens for joint view synthesis and pose estimation, effectively handling unordered, unposed multi-view inputs while remaining computationally efficient. In particular, our model consists of $L = 24$ LaCT blocks with alternating (windowed) self-attention, test-time training (TTT), and Feed-forward MLP layers; each TTT layer is equipped with a SwiGLU-MLP [54], noted as $\textbf{TTT-MLP}_l$, for test-time training.

Specifically, for the l-th block, the $HW/p^2 + 1$ tokens of each view are first fed into a per-view windowed self-attention layer:

$$\hat{X}_i^l = \text{Attn}_l(X_i^{l-1}) + X_i^{l-1}, \tag{3}$$

$$\hat{T}_k^l = \text{Attn}_l(T_k^{l-1}) + T_k^{l-1}. \tag{4}$$

Then we process multi-view tokens with the TTT layer, specifically:

$$q_{l,ij} = \text{Q}_l(\hat{X}_{ij}^l), \quad q_{l,kj}^t = \text{Q}_l(\hat{T}_{kj}^l), \tag{5}$$

$$k_{l,ij} = \text{K}_l(\hat{X}_{ij}^l), \tag{6}$$

$$v_{l,ij} = \text{V}_l(\hat{X}_{ij}^l), \tag{7}$$

where Q_l, K_l, and V_l are learnable parameters to project the original tokens into Q, K, V parameters. The weight of TTT-MLP$_l$ is then updated with the gradient $G_l = \nabla_{\text{TTT-MLP}_l}(1 - \text{TTT-MLP}_l(k_l) \cdot v_l^T)$ and a learnable learning weight η. The updated $\text{TTT-MLP}_l^{\text{updated}}$ is used to process the final outputs of this TTT layer for token update:

$$\tilde{X}_{ij}^l = \text{TTT-MLP}_l^{\text{updated}}(q_{l,ij}) + \hat{X}_{ij}^l, \tag{8}$$

$$\tilde{T}_{kj}^l = \text{TTT-MLP}_l^{\text{updated}}(q_{l,kj}^t) + \hat{T}_{kj}^l \tag{9}$$

Note that only input view tokens are sent through K_l, V_l and generate gradient for TTT MLP updates, as done in [86]. This way, the target view tokens don't need to interact with one another, enabling efficient synthesis of each novel view independently. The token outputs from the l-th LaCT block come from a final feed-forward MLP, denoted as FF-MLP$_l$:

$$X_{ij}^l = \text{FF-MLP}_l(\tilde{X}_{ij}^l) + \tilde{X}_{ij}^l, \tag{10}$$

$$T_{kj}^l = \text{FF-MLP}_l(\tilde{T}_{kj}^l) + \tilde{T}_{kj}^l \tag{11}$$

Design insights. A key consequence of the asymmetric K/V routing—where only input tokens update TTT weights while target tokens are read-only—is that each novel view can be synthesized independently without interacting with other target views, enabling efficient parallel rendering at 58.8 FPS regardless of the number of targets. The same mechanism naturally supports the joint pose+NVS formulation: the TTT hidden state serves as a compressed scene representation that is queried by both camera tokens (for pose) and target ray tokens (for synthesis). As confirmed by our ablation in Sec. 4.4, this joint formulation creates a virtuous cycle where NVS supervision provides rich geometric signal that dramatically improves pose accuracy, eliminating the need for dense 3D supervision used by prior methods [63, 68].

Prediction Heads. After the 24 LaCT blocks, target view tokens are decoded with a two-layer MLP, denoted as MLP_{rgb}, and rearranged to form the final novel-view RGB predictions. On the other hand, in contrast to the heavy camera head used by VGGT [63], we adopt simple light weight MLPs for camera parameter decoding. Specifically, the camera pose of each view is decoded from the final camera token with a simple two-layer MLP_{pose}, where the output is 9-dim: a 4-dim quaternion, a 3-dim translation, and 2-dim $\hat{f}^x$ and $\hat{f}^y$ focal length concatenated. We show that such lightweight camera heads are already sufficient to produce accurate camera parameters, even surpassing VGGT in estimation accuracy.

Note that VGGT is a transformer-based architecture; to achieve camera estimation, VGGT introduces frame-attention modules coupled with full attention to process the camera token jointly with per-view image tokens. In contrast, our design leverages LaCT blocks, which natively incorporate local windowed attention, eliminating the need for extra modules. As a result, our framework achieves better camera estimation while using fewer model parameters.

3.3 Training Objective

The model is trained with photometric, camera pose, and intrinsic losses:

$$\mathcal{L}_{\text{rgb}} = ||I_{\text{pred}} - I_{\text{gt}}||_2^2 + \lambda_{\text{LPIPS}} \cdot \text{LPIPS}(I_{\text{pred}}, I_{\text{gt}}), \tag{12}$$

$$\mathcal{L}_{\text{pose}} = ||t_{\text{pred}} - t_{\text{gt}}||_2^2 + ||r_{\text{pred}} - r_{\text{gt}}||_2^2, \tag{13}$$

$$\mathcal{L}_{\text{int}} = ||\hat{f}^x_{\text{pred}} - \hat{f}^x_{\text{gt}}||_2^2 + ||\hat{f}^y_{\text{pred}} - \hat{f}^y_{\text{gt}}||_2^2, \tag{14}$$

where I_{pred}, t_{pred}, r_{pred}, $\hat{f}^x_{\text{pred}}$, and $\hat{f}^y_{\text{pred}}$ are RGB image, camera translation, rotation quaternion, and x, y focal length predictions respectively, and the subscript $_{\text{gt}}$ denotes the ground-truth values for the corresponding quantities.

The total training loss is:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{rgb}}\mathcal{L}_{\text{rgb}} + \lambda_{\text{pose}}\mathcal{L}_{\text{pose}} + \lambda_{\text{int}}\mathcal{L}_{\text{int}}.$$

where λ_{rgb}, λ_{pose}, and λ_{int} are the weights for the RGB, pose, and intrinsic losses, respectively.

4 Experiments

We evaluate LVSPM on multiple challenging benchmarks to demonstrate its effectiveness at joint camera pose estimation and view synthesis from unposed input views. For pose estimation, we evaluate on RealEstate10k (RE10k) [88], Co3Dv2 [46], and DL3DV-10K [38]. For novel view synthesis, we evaluate on DL3DV-10K [38], and zero-shot on Tanks and Temples [31] and MipNeRF360 [1], comparing against both pose-free and pose-dependent baselines. We evaluate our method across varying numbers of input views- from sparse (16 views) to large-scale real-world sequences with 256 views. Our experiments validate three key claims: (1) LVSPM achieves state-of-the-art pose-free view synthesis quality, (2) our pose estimation matches or exceeds specialized geometry-based methods despite using no dense 3D supervision, and (3) the model scales gracefully from small- to large-scale scenes.

4.1 Experiment Settings

Model Details. Our model uses 24 layers of alternating view-attention and test-time-training layers. Model dimension $d = 768$. We use 12 heads for self-attention layers. We follow [86] to apply RoPE embedding [57] on Q and K inputs. The model contains 312M parameters.

Training Details. We implement LVSPM with PyTorch and train on 64 NVIDIA H100 GPUs. We train our model on a mix of large-scale synthetic and real-world datasets, including Aria Synthetic Environments (ASE) [44], DL3DV-10K [38], ScanNet++ [79], Hypersim [48], and CO3Dv2 [46]. Our model is not trained on the evaluation split of these datasets. The model undergoes a three-stage training process. First, we pre-train on the synthetic ASE dataset for 90k iterations with 32 input and target views at 128×128 resolution. Second, we mix ASE with other training datasets (DL3DV-10K, ScanNet++, Hypersim, Co3Dv2), and train for another 60k iterations. Finally, we progressively increase input resolution to 512×288, and increase scale to 128 input and 64 target views. Although training uses at most 128 views, the TTT mechanism naturally generalizes to longer sequences at inference: its hidden state is updated incrementally per input token, so processing 256 views simply extends the update sequence without architectural changes. We verify this empirically—our 256-view results remain strong without any long-sequence finetuning. More details can be found in the Supplementary material.

Baselines. We compare against recent pose estimation methods, namely Fast3R [77], Cut3R [67], Flare [85], VGGT [63], DepthAnything3 (DA3) [37], and TTT3R [9]. Details for resolutions are in the supplementary. For view synthesis, we compare with both pose-free method AnySplat [26] and pose-dependent method DepthSplat [74]. Following DepthSplat [74], we evaluate LVSPM and baselines at a resolution of 448×256. We use ground truth camera poses of the target views for LVSPM and DepthSplat, while Anysplat uses its predicted camera poses to follow their evaluation protocol. We include a comparison using the LVSPM predicted target camera pose in the supplementary material for a more

Table 1: Pose estimation results (AUC↑) on RE10K and Co3Dv2 with different numbers of input views. **Best** and second-best results are marked in bold and underlined. Despite no dense geometry supervision, LVSPM is best or second-best in all entries and shows clear long-context gains.

Datasets	Method	16 views			32 views			64 views			128 views		
		AUC30	AUC5	AUC3	AUC30	AUC5	AUC3	AUC30	AUC5	AUC3	AUC30	AUC5	AUC3
RE10k	Cut3R	70.6	24.9	14.9	75.1	31.7	20.2	77.1	36.0	24.0	75.5	33.2	20.9
	Fast3R	69.3	22.3	13.0	76.4	31.5	19.7	80.6	38.6	25.2	79.1	37.8	24.1
	Flare	75.2	29.1	17.9	81.0	39.2	26.2	84.9	47.6	33.7	85.0	49.6	35.9
	VGGT	75.5	25.3	14.0	78.0	30.2	17.9	79.5	33.2	20.1	80.2	34.1	20.6
	TTT3R	73.7	27.6	16.6	78.3	35.7	23.6	79.9	40.2	28.0	76.8	35.9	23.8
	DA3	87.5	53.4	**40.9**	90.3	61.7	49.8	92.0	68.3	57.8	92.9	72.4	**62.3**
	Ours	**88.3**	**53.8**	40.0	**91.0**	**62.9**	**50.3**	**92.7**	**69.8**	**58.2**	**93.7**	**72.5**	61.3
Co3Dv2	Cut3R	67.6	25.7	15.9	76.3	35.2	22.4	76.9	32.6	18.5	73.7	25.1	12.2
	Fast3R	68.7	26.6	16.2	78.8	39.2	25.6	81.5	43.2	28.1	80.2	40.4	25.3
	Flare	68.2	24.5	14.9	74.2	33.3	21.8	72.1	33.4	22.5	70.2	34.2	23.8
	VGGT	80.2	45.2	33.9	86.8	58.6	47.2	90.0	67.2	57.0	91.3	71.4	62.1
	TTT3R	70.3	28.4	17.8	78.8	38.9	25.6	79.3	37.8	23.2	78.1	32.8	18.3
	DA3	**85.2**	**57.4**	**47.2**	**90.5**	**69.3**	**60.0**	**92.3**	**73.5**	64.0	**93.0**	75.3	65.7
	Ours	82.2	50.1	39.2	88.9	65.1	55.4	91.8	72.8	**64.3**	92.9	**76.6**	**68.8**

Table 2: Pose Estimation results on DL3DV-10K across 16–256 input views. The **best** is marked in bold, and the second best is marked with underline. Our method demonstrates particularly strong gains on long sequences.

Method	16 views			32 views			64 views			128 views			256 views		
	AUC30	AUC5	AUC3	AUC30	AUC5	AUC3	AUC30	AUC5	AUC3	AUC30	AUC5	AUC3	AUC30	AUC5	AUC3
Cut3R	88.2	60.5	48.8	89.0	66.2	55.5	88.2	64.1	52.2	82.1	48.2	34.1	73.0	31.4	19.2
Fast3R	75.2	34.0	22.7	79.5	41.9	28.8	75.9	39.6	26.9	64.5	24.7	14.8	52.4	15.3	8.0
Flare	90.4	69.9	60.6	93.2	80.0	73.6	93.8	82.5	76.2	92.8	80.8	74.0	92.8	79.1	71.1
VGGT	94.7	81.9	74.7	96.3	88.8	84.1	96.3	89.4	85.5	95.2	88.2	84.1	95.2	87.9	83.6
TTT3R	89.3	63.3	51.4	89.9	68.8	58.4	89.3	68.8	58.7	85.3	58.8	46.5	83.3	50.7	36.5
DA3	96.2	88.7	84.0	96.6	90.7	86.5	96.0	90.0	86.1	95.5	89.1	84.5	95.3	88.5	83.8
Ours	**96.9**	**91.6**	**88.4**	**97.3**	**94.0**	**92.3**	**96.3**	**93.5**	**92.0**	**95.6**	**92.0**	**90.0**	**95.7**	**92.1**	**90.1**

comprehensive evaluation. For fair comparison, we use official implementations where available and use the same evaluation datasets for all models.

Evaluation Metrics. For pose estimation, we follow VGGT [63] to report Area Under Curve (AUC) metrics at thresholds of 3° (AUC3), 5° (AUC5), and 30° (AUC30). The pose AUC evaluates both rotation and translation accuracy jointly—a pose is considered correct at a given threshold only if both relative rotation error (RRE) and relative translation error (RTE) are below the threshold. Higher AUC values indicate more accurate pose estimation. AUC3 is the strictest metric, requiring sub-3-degree precision for both rotation and translation; we order tables by AUC3 first to highlight fine-grained accuracy. For view synthesis, we measure PSNR, SSIM [69], and LPIPS [84] on held-out test views, reporting average metrics across scenes. We also report network runtime across 16-256 views.

4.2 Camera Pose Estimation

Evaluation Protocol. We adopt an equi-temporal evaluation protocol that reflects practical capture scenarios: N input views are uniformly sampled from the first $N/N_{\max}$ portion of the video sequence, so that more views correspond to larger scene coverage rather than denser sampling of the same area. This makes the task progressively harder with more views, as the model must handle a larger, more diverse scene. Standard uniform sampling over the full sequence conflates two variables—scene coverage and sampling density—making it unclear whether improvements come from seeing more of the scene or from denser observations of the same area. Our equi-temporal protocol isolates the effect of scene scale by holding sampling density approximately constant, providing a cleaner evaluation of a model's ability to handle diverse, large-scale environments. We also evaluate under standard sparse-view protocols following prior work [74, 78]; these results are provided in the supplementary material and confirm our method's competitiveness across evaluation settings. For RE10k and Co3Dv2, $N_{\max} = 128$ with $N \in \{16, 32, 64, 128\}$. For the large-scale DL3DV dataset, $N_{\max} = 256$ with $N \in \{16, 32, 64, 128, 256\}$.

Our pose estimation results in Tab. 1 and Tab. 2 demonstrate LVSPM's strong performance across all datasets and view counts, with especially large advantages at strict accuracy thresholds (AUC3 and AUC5).

On RealEstate10k (RE10k), LVSPM dominates nearly all metrics. At 16 views, we achieve 40.0 AUC3 and 53.8 AUC5, compared to VGGT's 14.0 AUC3 and 25.3 AUC5—nearly 3× and 2× the precision. At 128 views (full sequence), the gap grows further: 61.3 vs. 20.6 AUC3 and 72.5 vs. 34.1 AUC5. Compared with DA3, LVSPM achieves higher AUC30 and AUC5 across all view counts and remains competitive on AUC3, winning 10 out of 12 metrics, despite using only RGB images and camera poses rather than large-scale dense geometric supervision. Flare emerges as the third-best method at longer sequences (35.9 AUC3 at 128 views) but remains far behind ours.

On Co3Dv2, LVSPM outperforms VGGT at all metrics. At 16 views, we achieve 39.2 AUC3 vs. VGGT's 33.9 and 50.1 AUC5 vs. 45.2. At 128 views, the margins remain consistent: 68.8 vs. 62.1 AUC3 and 76.6 vs. 71.4 AUC5. DA3 sets a strong pose-estimation reference with dense geometry supervision, especially in a sparse-view setting. LVSPM remains competitive under this comparison and closes the gap as context grows, surpassing DA3 on AUC3 at 64/128 views and AUC_5 at 128 views while using no dense geometry labels and jointly supporting NVS. This trend supports our design goal of scalable long-sequence aggregation rather than optimizing only short-context pose estimation.

On the challenging DL3DV dataset featuring large-scale scenes, LVSPM demonstrates both high accuracy and strong scalability. At 16 views, we achieve 88.4 AUC3 vs. VGGT's 74.7 (+13.7 absolute) and DA3's 84.0, and 91.6 AUC5 vs. 81.9 and 88.7. At 256 views (full sequence), we maintain 90.1 AUC3 vs. 83.6 and 83.8, while other baselines collapse: Fast3R drops to 8.0 AUC3 and Cut3R to 19.2 AUC3. These results validate that joint NVS supervision and

Table 3: Novel view synthesis on DL3DV-10K (PSNR↑ / SSIM↑ / LPIPS↓) across 16–256 input views. The **best** is marked in bold. DepthSplat requires GT poses while others are pose-free. DepthSplat fails to run on more than 64 views.

Method	16 views			64 views			128 views			256 views		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
DepthSplat	<u>24.80</u>	**0.849**	**0.134**	18.39	0.617	0.363	–	–	–	–	–	–
AnySplat	23.15	0.757	0.158	<u>20.52</u>	<u>0.643</u>	<u>0.248</u>	<u>19.45</u>	<u>0.593</u>	<u>0.294</u>	<u>18.55</u>	<u>0.543</u>	<u>0.339</u>
Ours	**25.77**	<u>0.811</u>	<u>0.152</u>	**24.33**	**0.752**	**0.182**	**23.31**	**0.710**	**0.208**	**22.09**	**0.654**	**0.250**

the TTT architecture enable both robust long-sequence scaling and fine-grained pose accuracy without dense geometric supervision.

4.3 View Synthesis

Table 3 shows that LVSPM achieves state-of-the-art pose-free view synthesis on DL3DV-10K. At 16 views (the smallest scene coverage), LVSPM achieves 25.77 PSNR, surpassing even the pose-dependent DepthSplat (24.80) and far exceeding the pose-free AnySplat (23.15). This demonstrates strong synthesis quality in the relatively sparse regime without requiring any pose input.

Under our equi-temporal protocol, increasing the number of input views corresponds to larger scene coverage and hence harder tasks. All methods naturally degrade, but LVSPM shows the most graceful scaling: our PSNR decreases from 25.77 to 22.09 (−3.68 dB) across a 16× increase in scene scale, while AnySplat drops from 23.15 to 18.55 (−4.60 dB) and DepthSplat collapses from 24.80 to 18.39 at just 64 views (−6.41 dB) before failing due to its model constraints. Critically, the gap between LVSPM and baselines *widens* with scale: +2.62 dB over AnySplat at 16 views grows to +3.54 dB at 256 views. LPIPS shows similar robustness (ours 0.152→0.250 vs. AnySplat 0.158→0.339). Qualitative results in Fig. 3 confirm that our method renders sharper reflections and avoids the layered-surface artifacts of AnySplat [26]. **Zero-shot Evaluation.** We zero-shot generalize to the challenging Tanks and Temples (TnT) [30] dataset and the indoor MipNeRF360 dataset [1] under both dense (64–128 views) and sparse (3–6 views) input settings. Full quantitative tables and qualitative comparisons are provided in the supplementary material. LVSPM consistently outperforms baselines across all settings and view counts.

Sparse view Evaluation. We also evaluate LVSPM under sparse view setting on unseen datasets RE10K [88], MipNeRF360 [1] and LLFF [41], and compare with both posed (e.g. MVSplat [11]) and unposed (e.g. NoPoSPlat [78]) baselines. We finetune LVSPM with sparse input views and zero-shot generalize to these datasets. For RE10K, the results are shown in Table 4. LVSPM surpasses AnySplat under the zero-shot setting. When fine-tuned on RE10K (Ours - F.T.), it also outperforms other baselines trained exclusively on this dataset. Full MipNeRF360/LLFF sparse-view results and finetuning details are in the supplementary.

Fig. 3: Novel-view rendering comparison on the DL3DV dataset with long-sequence input. We visualize results for 64 inputs using our method, AnySplat and DepthSplat. Unlike DepthSplat, which requires known poses, our method and AnySplat are pose-free.

Table 4: NVS on RE10K dataset. We do not use test time pose alignment for a fair comparison. We highlight the **best** and the second best. "*" indicates methods require known intrinsics.

Method	Overlap Settings											
	Small			Medium			Large			Average		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Pose-Required												
pixelNeRF	18.417	0.601	0.526	19.930	0.632	0.458	20.869	0.639	0.485	19.824	0.626	0.485
AttnRend	19.151	0.663	0.368	22.532	0.763	0.186	25.897	0.845	0.269	22.664	0.762	0.269
pixelSplat	20.263	0.717	0.266	23.711	0.809	0.181	27.151	0.879	0.122	23.848	0.806	0.185
MVSplat	20.353	0.724	0.250	23.778	0.812	0.173	**27.408**	**0.884**	0.116	23.977	0.811	0.176
Pose-Free												
DUSt3R	14.101	0.432	0.468	15.419	0.451	0.402	16.427	0.453	0.432	15.382	0.447	0.432
MASt3R	13.534	0.407	0.494	14.945	0.436	0.418	16.028	0.444	0.452	14.907	0.431	0.452
Splatt3R	14.352	0.475	0.472	15.529	0.502	0.425	15.817	0.483	0.421	15.318	0.490	0.436
CoPoNeRF	17.393	0.585	0.462	18.813	0.616	0.392	20.464	0.652	0.318	18.938	0.619	0.388
SelfSplat	17.506	0.550	0.461	19.357	0.704	0.378	20.868	0.672	0.256	19.33	0.656	0.363
NoPoSplat*	21.814	0.765	0.220	23.044	0.787	0.178	25.408	0.844	0.126	23.424	0.798	0.173
CocaSplat	21.312	0.741	0.234	22.891	0.773	0.189	25.312	0.829	0.138	23.200	0.781	0.185
CocaSplat*	22.843	**0.781**	**0.192**	23.394	0.804	0.175	26.270	0.872	**0.110**	24.093	**0.818**	**0.160**
Flare	20.779	0.695	0.247	22.412	0.743	0.193	23.791	0.774	0.151	22.404	0.740	0.194
Ours - F.T.	**22.951**	0.768	0.203	**25.047**	**0.814**	**0.163**	27.193	0.856	0.129	**25.146**	0.815	0.164
Zero-Shot												
AnySplat	15.171	0.587	0.412	17.352	0.622	0.344	19.828	0.663	0.277	17.526	0.625	0.341
Ours	**20.664**	**0.693**	**0.262**	**22.537**	**0.746**	**0.211**	**24.026**	**0.773**	**0.180**	**22.501**	**0.741**	**0.214**

4.4 Ablation Study

We conduct ablations on DL3DV and ASE using 32 input and target views. First, we evaluate the effect of large-scale synthetic pretraining by directly training on real data ("w/o Syn."). As shown in Table 5, synthetic pretraining consistently improves both pose estimation and novel-view rendering on DL3DV.

We then validate the model design on the ASE synthetic dataset, due to computational constraints. We first validate the effect of NVS supervision on pose estimation ("w/o RGB"). As shown in Table 5, removing NVS supervision causes AUC3 to drop from 71.9 to 50.9—a nearly half degradation. This dramatic gap demonstrates that NVS supervision provides rich geometric signal that effectively replaces the dense 3D supervision (point maps, depth) used by methods like VGGT [63] and DUSt3R [68].

We also ablate the TTT chunk size and the number of TTT layers. Our full model uses a chunk size of 8224 tokens (aggregated from all input images and cameras) and performs one MLP update per TTT layer. In the chunk-size ablation ('Half Chunk size'), we halve the chunk size to 4112 and apply two updates per layer; as shown in Table 5, this causes a slight drop in NVS and noticeable degradation in pose estimation. We hypothesize that repeated updates on smaller chunks promote catastrophic forgetting, causing TTT layers to overfit local geometry and lose the global context needed for consistent long-sequence alignment. In contrast, a single-pass update over a larger context window is more stable for joint pose–NVS optimization and is also more efficient. We further ablate the number of layers ('12 TTT layers') by reducing from 24 to 12 layers, which causes worse performance in Table 5, indicating that a smaller TTT capacity is insufficient. Our full models give the best overall performance.

Dataset	Method	Pose AUC30	Pose AUC5	Pose AUC3	NVS PSNR	NVS SSIM	NVS LPIPS
DL3DV	w/o Syn	92.7	84.3	79.9	20.08	0.589	0.267
	Ours	**94.5**	**85.6**	**80.2**	**21.12**	**0.638**	**0.222**
ASE	w/o RGB	87.9	62.6	50.9	–	–	–
	Half Chunk size	88.3	74.3	67.9	24.98	**0.702**	0.263
	12 TTT layers	90.1	69.2	59.3	24.18	0.673	0.288
	Ours Full	**93.7**	**79.5**	**71.9**	**25.01**	0.701	**0.257**

Table 5: Ablation study on DL3DV-10K and ASE datasets.

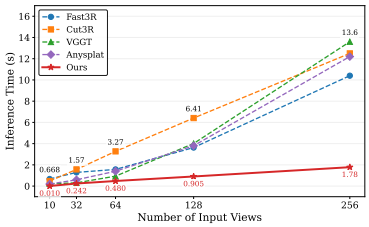

Fig. 4: Inference time comparison.

4.5 Inference Time comparison

We report inference time for LVSPM and all baselines across 10–256 input views. We use a resolution of 512×288 for LVSPM, Cut3R, Fast3R, and AnySplat, and 518×280 for VGGT due to its 14 patch size. For each method, we measure the time required to predict poses for all input views, using an H100 GPU. As shown in Fig. 4, LVSPM scales linearly with the number of input views and processes 256 images in under 2 seconds, whereas baselines require over 10 seconds. Additionally, our method renders at **58.8** FPS, regardless of the number of input views.

5 Conclusion and Limitations

We presented LVSPM, a unified framework for pose-free view synthesis from long unposed image sequences. LVSPM jointly estimates world-aligned camera poses and synthesizes novel views, trained with RGB images and camera-pose supervision, avoiding dense 3D geometric supervision while matching or outperforming methods that rely on it. Enabled by an efficient test-time training mechanism, LVSPM scales to sequences of 200+ views and maintains strong rendering quality under the practical equi-temporal evaluation protocol, where additional input views correspond to larger scene coverage. Across challenging benchmarks such as DL3DV-10K, this leads to graceful scaling as scene complexity increases, while existing baselines degrade more rapidly. LVSPM represents a significant step toward practical, scalable 3D scene understanding from unposed image collections, bringing us closer to systems that can operate on real-world data without expensive preprocessing or annotation requirements.

Despite these advances, several limitations remain. The multi-stage training pipeline introduces additional complexity. Performance may degrade under extreme lighting changes or in weakly textured scenes, where reliable geometric reasoning is difficult. In addition, although LVSPM achieves competitive pose estimation while jointly supporting view synthesis, specialized geometry-focused models may still perform better in certain edge cases. Future work includes exploring fully self-supervised training to further reduce supervision requirements and investigating how far scalable 3D scene understanding can be pushed with minimal geometric priors.

References

1. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. CVPR (2022)
2. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Zip-nerf: Anti-aliased grid-based neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19697–19705 (2023)
3. Bian, W., Wang, Z., Li, K., Bian, J.W., Prisacariu, V.A.: Nope-nerf: Optimising neural radiance field with no pose prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4160–4169 (2023)
4. Buehler, C., Bosse, M., McMillan, L., Gortler, S., Cohen, M.: Unstructured lumigraph rendering. In: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. pp. 425–432 (2001)
5. Cai, R., Hariharan, B., Snavely, N., Averbuch-Elor, H.: Extreme rotation estimation using dense correlation volumes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14566–14575 (2021)
6. Charatan, D., Li, S.L., Tagliasacchi, A., Sitzmann, V.: pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19457–19467 (2024)
7. Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H.: Tensorf: Tensorial radiance fields. In: European Conference on Computer Vision (ECCV) (2022)
8. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 14124–14133 (2021)
9. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Ttt3r: 3d reconstruction as test-time training. arXiv preprint arXiv:2509.26645 (2025)
10. Chen, Y., Chen, X., Wang, X., Zhang, Q., Guo, Y., Shan, Y., Wang, F.: Local-to-global registration for bundle-adjusting neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8264–8273 (2023)
11. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In: European Conference on Computer Vision. pp. 370–386. Springer (2025)
12. Dao, T., Gu, A.: Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. arXiv preprint arXiv:2405.21060 (2024)
13. Debevec, P.E., Taylor, C.J., Malik, J.: Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach. Seminal Graphics Papers: Pushing the Boundaries, Volume 2 (1996), https://api.semanticscholar.org/CorpusID:2609415
14. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 224–236 (2018)
15. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-net: A trainable cnn for joint description and detection of local features. In: Proceedings of the ieee/cvf conference on computer vision and pattern recognition. pp. 8092–8101 (2019)

16. Fan, Z., Cong, W., Wen, K., Wang, K., Zhang, J., Ding, X., Xu, D., Ivanovic, B., Pavone, M., Pavlakos, G., et al.: Instantsplat: Sparse-view gaussian splatting in seconds. arXiv preprint arXiv:2403.20309 (2024)
17. Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5501–5510 (2022)
18. Gortler, S.J., Grzeszczuk, R., Szeliski, R., Cohen, M.F.: The lumigraph. Proceedings of the 23rd annual conference on Computer graphics and interactive techniques (1996), https://api.semanticscholar.org/CorpusID:2036193
19. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752 (2023)
20. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: Lrm: Large reconstruction model for single image to 3d. In: The Twelfth International Conference on Learning Representations (2024)
21. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2d gaussian splatting for geometrically accurate radiance fields. In: ACM SIGGRAPH 2024 conference papers. pp. 1–11 (2024)
22. Huang, R., Mikolajczyk, K.: No pose at all: Self-supervised pose-free 3d gaussian splatting from sparse views. arXiv preprint arXiv:2508.01171 (2025)
23. Jiang, H., Jiang, Z., Grauman, K., Zhu, Y.: Few-view object reconstruction with unknown categories and camera poses. ArXiv **2212.04492** (2022)
24. Jiang, H., Jiang, Z., Zhao, Y., Huang, Q.: Leap: Liberate sparse-view 3d modeling from camera poses. arXiv preprint arXiv:2310.01410 (2023)
25. Jiang, H., Tan, H., Wang, P., Jin, H., Zhao, Y., Bi, S., Zhang, K., Luan, F., Sunkavalli, K., Huang, Q., et al.: Rayzer: A self-supervised large view synthesis model. arXiv preprint arXiv:2505.00702 (2025)
26. Jiang, L., Mao, Y., Xu, L., Lu, T., Ren, K., Jin, Y., Xu, X., Yu, M., Pang, J., Zhao, F., et al.: Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. arXiv preprint arXiv:2505.23716 (2025)
27. Jin, H., Jiang, H., Tan, H., Zhang, K., Bi, S., Zhang, T., Luan, F., Snavely, N., Xu, Z.: Lvsm: A large view synthesis model with minimal 3d inductive bias. In: The Thirteenth International Conference on Learning Representations (2025), https://openreview.net/forum?id=QQBPWtvtcn
28. Johari, M.M., Lepoittevin, Y., Fleuret, F.: Geonerf: Generalizing nerf with geometry priors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18365–18375 (2022)
29. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. **42**(4), 139–1 (2023)
30. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Transactions on Graphics (TOG) **36**(4), 78:1–78:13 (2017). https://doi.org/10.1145/3072959.3073599
31. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. ACM Transactions on Graphics **36**(4) (2017)
32. Leroy, V., Cabon, Y., Revaud, J.: Grounding image matching in 3d with mast3r. In: European Conference on Computer Vision. pp. 71–91. Springer (2024)
33. Levoy, M., Hanrahan, P.: Light field rendering. Proceedings of the 23rd annual conference on Computer graphics and interactive techniques (1996), https://api.semanticscholar.org/CorpusID:1363510
34. Li, J., Tan, H., Zhang, K., Xu, Z., Luan, F., Xu, Y., Hong, Y., Sunkavalli, K., Shakhnarovich, G., Bi, S.: Instant3d: Fast text-to-3d with sparse-view generation

and large reconstruction model. In: The Twelfth International Conference on Learning Representations (2023)
35. Lin, A., Zhang, J.Y., Ramanan, D., Tulsiani, S.: Relpose++: Recovering 6d poses from sparse-view observations. arXiv preprint arXiv:2305.04926 (2023)
36. Lin, C.H., Ma, W.C., Torralba, A., Lucey, S.: Barf: Bundle-adjusting neural radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5741–5751 (2021)
37. Lin, H., Chen, S., Liew, J., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
38. Ling, L., Sheng, Y., Tu, Z., Zhao, W., Xin, C., Wan, K., Yu, L., Guo, Q., Yu, Z., Lu, Y., et al.: Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22160–22169 (2024)
39. Liu, T., Wang, G., Hu, S., Shen, L., Ye, X., Zang, Y., Cao, Z., Li, W., Liu, Z.: Fast generalizable gaussian splatting reconstruction from multi-view stereo. arXiv preprint arXiv:2405.12218 (2024)
40. Liu, Y., Liu, L., Lin, C., Dong, Z., Wang, W.: Learnable motion coherence for correspondence pruning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3237–3246 (2021)
41. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (TOG) (2019)
42. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
43. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM transactions on graphics (TOG) **41**(4), 1–15 (2022)
44. Pan, X., Charron, N., Yang, Y., Peters, S., Whelan, T., Kong, C., Parkhi, O., Newcombe, R., Ren, Y.C.: Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 20133–20143 (October 2023)
45. Plucker, J.: Xvii. on a new geometry of space. Philosophical Transactions of the Royal Society of London (155), 725–791 (1865)
46. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: International Conference on Computer Vision (2021)
47. Revaud, J., Weinzaepfel, P., De Souza, C., Pion, N., Csurka, G., Cabon, Y., Humenberger, M.: R2d2: repeatable and reliable detector and descriptor. arXiv preprint arXiv:1906.06195 (2019)
48. Roberts, M., Ramapuram, J., Ranjan, A., Kumar, A., Bautista, M.A., Paczan, N., Webb, R., Susskind, J.M.: Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10912–10922 (2021)
49. Rockwell, C., Johnson, J., Fouhey, D.F.: The 8-point algorithm as an inductive bias for relative pose prediction by vits. In: 2022 International Conference on 3D Vision (3DV). pp. 1–11. IEEE (2022)
50. Sajjadi, M.S., Meyer, H., Pot, E., Bergmann, U., Greff, K., Radwan, N., Vora, S., Lučić, M., Duckworth, D., Dosovitskiy, A., et al.: Scene representation trans-

former: Geometry-free novel view synthesis through set-latent scene representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6229–6238 (2022)
51. Sarlin, P.E., Cadena, C., Siegwart, R., Dymczyk, M.: From coarse to fine: Robust hierarchical localization at large scale. In: CVPR (2019)
52. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4938–4947 (2020)
53. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)
54. Shazeer, N.: Glu variants improve transformer. arXiv preprint arXiv:2002.05202 (2020)
55. Smart, B., Zheng, C., Laina, I., Prisacariu, V.A.: Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs. arXiv preprint arXiv:2408.13912 (2024)
56. Snavely, N., Seitz, S.M., Szeliski, R.: Photo tourism: exploring photo collections in 3d. In: ACM siggraph 2006 papers, pp. 835–846 (2006)
57. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568**, 127063 (2024)
58. Suhail, M., Esteves, C., Sigal, L., Makadia, A.: Generalizable patch-based neural rendering. In: European Conference on Computer Vision. pp. 156–174. Springer (2022)
59. Sun, C., Sun, M., Chen, H.T.: Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5459–5469 (2022)
60. Sun, Y., Li, X., Dalal, K., Xu, J., Vikram, A., Zhang, G., Dubois, Y., Chen, X., Wang, X., Koyejo, S., et al.: Learning to (learn at test time): Rnns with expressive hidden states. arXiv preprint arXiv:2407.04620 (2024)
61. Tang, Z., Fan, Y., Wang, D., Xu, H., Ranjan, R., Schwing, A., Yan, Z.: Mv-dust3r+: Single-stage scene reconstruction from sparse views in 2 seconds. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5283–5293 (2025)
62. Truong, P., Rakotosaona, M.J., Manhardt, F., Tombari, F.: Sparf: Neural radiance fields from sparse and noisy poses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4190–4200 (2023)
63. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5294–5306 (2025)
64. Wang, P., Tan, H., Bi, S., Xu, Y., Luan, F., Sunkavalli, K., Wang, W., Xu, Z., Zhang, K.: Pf-lrm: Pose-free large reconstruction model for joint pose and shape prediction. In: The Twelfth International Conference on Learning Representations (2023)
65. Wang, Q., Wang, Z., Genova, K., Srinivasan, P.P., Zhou, H., Barron, J.T., Martin-Brualla, R., Snavely, N., Funkhouser, T.: Ibrnet: Learning multi-view image-based rendering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4690–4699 (2021)
66. Wang, Q., Zhang, Y., Holynski, A., Efros, A.A., Kanazawa, A.: Continuous 3d perception model with persistent state. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 10510–10522 (2025)
67. Wang, Q., Zhang, Y., Holynski, A., Efros, A.A., Kanazawa, A.: Continuous 3d perception model with persistent state. In: CVPR (2025)

68. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20697–20709 (2024)
69. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004). https://doi.org/10.1109/TIP.2003.819861
70. Wei, X., Zhang, K., Bi, S., Tan, H., Luan, F., Deschaintre, V., Sunkavalli, K., Su, H., Xu, Z.: Meshlrm: Large reconstruction model for high-quality mesh. arXiv preprint arXiv:2404.12385 (2024)
71. Xia, Y., Tang, H., Timofte, R., Van Gool, L.: Sinerf: Sinusoidal neural radiance fields for joint pose estimation and scene reconstruction. arXiv preprint arXiv:2210.04553 (2022)
72. Xu, C., Li, A., Chen, L., Liu, Y., Shi, R., Su, H., Liu, M.: Sparp: Fast 3d object reconstruction and pose estimation from sparse views. In: European Conference on Computer Vision. pp. 143–163. Springer (2024)
73. Xu, H., Peng, S., Wang, F., Blum, H., Barath, D., Geiger, A., Pollefeys, M.: Depthsplat: Connecting gaussian splatting and depth. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16453–16463 (2025)
74. Xu, H., Peng, S., Wang, F., Blum, H., Barath, D., Geiger, A., Pollefeys, M.: Depthsplat: Connecting gaussian splatting and depth. In: CVPR (2025)
75. Xu, J., Gao, S., Shan, Y.: Freesplatter: Pose-free gaussian splatting for sparse-view 3d reconstruction. arXiv preprint arXiv:2412.09573 (2024)
76. Xu, Q., Xu, Z., Philip, J., Bi, S., Shu, Z., Sunkavalli, K., Neumann, U.: Point-nerf: Point-based neural radiance fields. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5438–5448 (2022)
77. Yang, J., Sax, A., Liang, K.J., Henaff, M., Tang, H., Cao, A., Chai, J., Meier, F., Feiszli, M.: Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 21924–21935 (2025)
78. Ye, B., Liu, S., Xu, H., Li, X., Pollefeys, M., Yang, M.H., Peng, S.: No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. arXiv preprint arXiv:2410.24207 (2024)
79. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the International Conference on Computer Vision (ICCV) (2023)
80. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4578–4587 (2021)
81. Yu, Z., Chen, A., Huang, B., Sattler, T., Geiger, A.: Mip-splatting: Alias-free 3d gaussian splatting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19447–19456 (2024)
82. Yuan, Y., Shen, Q., Wang, S., Yang, X., Wang, X.: Test3r: Learning to reconstruct 3d at test time. arXiv preprint arXiv:2506.13750 (2025)
83. Zhang, K., Bi, S., Tan, H., Xiangli, Y., Zhao, N., Sunkavalli, K., Xu, Z.: Gs-lrm: Large reconstruction model for 3d gaussian splatting. In: European Conference on Computer Vision. pp. 1–19. Springer (2024)
84. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)

85. Zhang, S., Wang, J., Xu, Y., Xue, N., Rupprecht, C., Zhou, X., Shen, Y., Wetzstein, G.: Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 21936–21947 (2025)
86. Zhang, T., Bi, S., Hong, Y., Zhang, K., Luan, F., Yang, S., Sunkavalli, K., Freeman, W.T., Tan, H.: Test-time training done right. arXiv preprint arXiv:2505.23884 (2025)
87. Zhang, X., Bi, S., Sunkavalli, K., Su, H., Xu, Z.: Nerfusion: Fusing radiance fields for large-scale scene reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5449–5458 (2022)
88. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: learning view synthesis using multiplane images. ACM Transactions on Graphics (TOG) **37**(4), 1–12 (2018)
89. Ziwen, C., Tan, H., Zhang, K., Bi, S., Luan, F., Hong, Y., Fuxin, L., Xu, Z.: Long-lrm: Long-sequence large reconstruction model for wide-coverage gaussian splats. arXiv preprint arXiv:2410.12781 (2024)