RigRecon: Memory-Efficient Rig-Aware Reconstruction via Dual-Path Spatio-Temporal Interaction

Jiaming Guo[1,2], **Hongcheng Luo**[3,5,*], **Cheng Chi**[3,5], **Lijun Zhou**[3,5]
Bing Wang[3,5], **Guang Chen**[3,5], **Hangjun Ye**[3,5], **Jiaqi Yang**[4]
Xiaowei Zhou[1], **Haiyang Sun**[3,5], **Sida Peng**[1†]

[1]Zhejiang University [2]Xiangjiang Lab [3]Xiaomi EV [4]Northwestern Polytechnical University
[5]Beijing Key Laboratory of Intelligent Driving Foundation Models and Simulation

Project code: `https://github.com/zju3dv/RigRecon`

Abstract

Reconstructing multi-camera videos is a fundamental task in autonomous driving. However, recent learned methods either rely on costly attention-based networks for global geometry modeling or require complex post-hoc alignment for chunked predictions, limiting their scalability to long multi-view rig sequences. We propose RigRecon, a generalized rig-aware reconstruction framework that enables efficient cross-view interaction in a low-resolution space while asynchronously decoding high-resolution depth maps in a frame-wise manner. Input images are first down-sampled and processed by a transformer backbone to capture coarse global structure, which is then provided to a decoder to reconstruct frame-wise high-resolution depth maps. By fully leveraging calibration and low-resolution feature interaction, RigRecon achieves consistent-scale geometry and accurate pose estimation with significantly reduced memory cost. It achieves state-of-the-art performance on 3D reconstruction, depth estimation, and camera pose estimation, and can efficiently process videos with over 1000 images in a single feedforward pass.

1 Introduction

Rig-aware 3D Reconstruction is the task of predicting the accurate geometry of a given scene, conditioned on multi-view videos with calibrated camera settings. It serves as a fundamental capability for spatial understanding, enabling applications such as autonomous driving, robotics, and visual localization [3, 4, 5, 6, 7, 8, 9], where reliable estimation of 3D structure and motion from visual inputs is critical.

Recent feedforward multi-view reconstruction methods, such as DA3 [2], VGGT [10], and DVGT [1], leverage the strong representation capacity of large transformer models to directly regress geometry and camera poses from input images. However, attention-based interactions at the original image resolution incur $\mathcal{O}(N^2)$ memory complexity with respect to the number of tokens, making them impractical for long or high-frame-rate multi-view videos. Moreover, most methods rely heavily on image overlap for pose estimation and do not explicitly exploit known rig calibration, which can lead to degraded performance in multi-camera setups where cross-view overlap is limited.

Streaming approaches, including VGGT-Long [11] and DA3-Streaming [2], address this issue by processing videos in chunks. However, they require splitting the sequence into overlapped chunks regardless of its length, followed by post-hoc alignment of chunk-wise predicted point maps, which is computationally expensive and sensitive to noisy predictions and hyperparameter choices. More-

*Project leader.
†Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

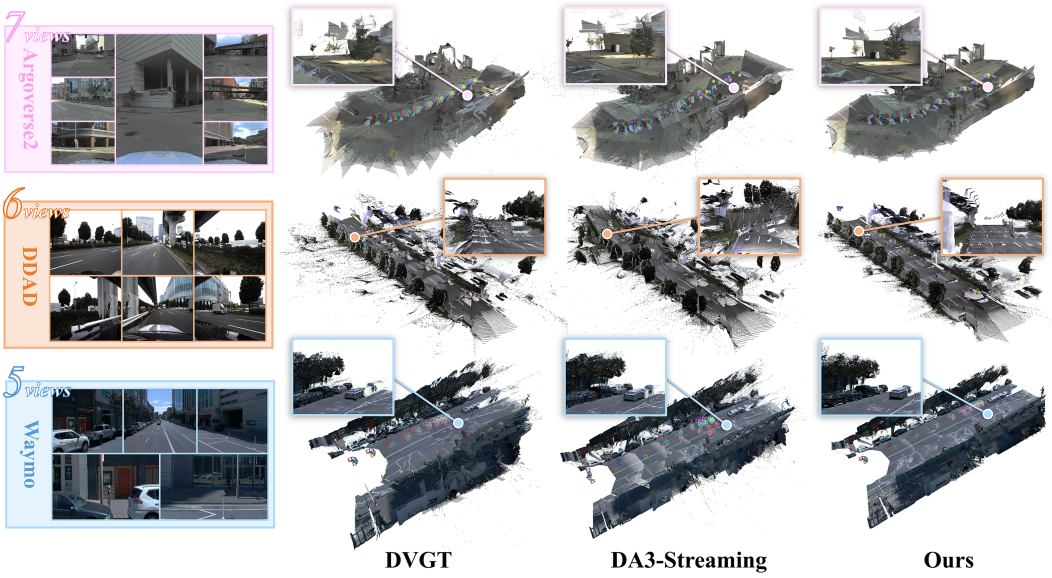

Figure 1: RigRecon is a rig-aware reconstruction framework that efficiently recovers accurate 3D geometry from multi-view rig videos. It outperforms DVGT [1] and DA3-Streaming [2], achieving more consistent 3D reconstruction and more accurate pose estimation across diverse camera rigs.

over, the alignment and concatenation process introduces cumulative errors over long sequences, leading to drift and degraded global consistency in the reconstructed geometry.

We introduce **RigRecon**, which enables efficient interaction between images in a low-resolution space while asynchronously decoding high-resolution depth maps in a frame-wise manner. Specifically, input images are first resized to a fixed lower resolution and processed by a transformer-based backbone. Instead of directly predicting dense geometry, the backbone captures coarse global structure and scale through interactions across images and poses. A separate decoder then reconstructs high-resolution depth maps for each frame conditioned on the coarse representation. Furthermore, we disentangle camera poses into ego pose (the trajectory of a reference camera) and rig pose (the transformation relative to the reference camera), allowing better utilization of the accurate calibration in rig data. The low-resolution path preserves global cross-frame interactions while reducing memory cost by a factor of d^2, where d is the down-sampling ratio. The frame-wise decoder avoids expensive high-resolution interactions and instead leverages coarse geometric guidance to maintain consistency across frames. These designs enable our method to efficiently produce detailed geometry and accurate poses with consistent scale under significantly reduced memory consumption.

Our key contributions can be summarized as:

- We propose a feedforward method for rig-aware 3D reconstruction that processes inputs in a low-resolution space and decodes high-resolution depth maps asynchronously in a frame-wise manner, achieving both efficiency and consistency while supporting videos with over 1000 images.
- We conduct extensive experiments on diverse real-world driving datasets, showing that our method achieves comparable or state-of-the-art performance.

2 Related works

Multi-view geometry models. Classical multi-view reconstruction follows a two-stage pipeline, where Structure-from-Motion (SfM) estimates sparse camera poses and structure, and Multi-View Stereo (MVS) densifies the reconstruction. Representative systems such as COLMAP [12, 13] rely on feature matching, triangulation, and bundle adjustment, but remain sensitive to occlusions, motion, and low-texture regions. More recent methods bypass explicit feature matching and known poses, directly predicting 3D structure from RGB images [14, 15, 16, 17]. DUSt3R [18] introduces pointmap regression from image pairs without requiring camera poses, followed by methods includ-

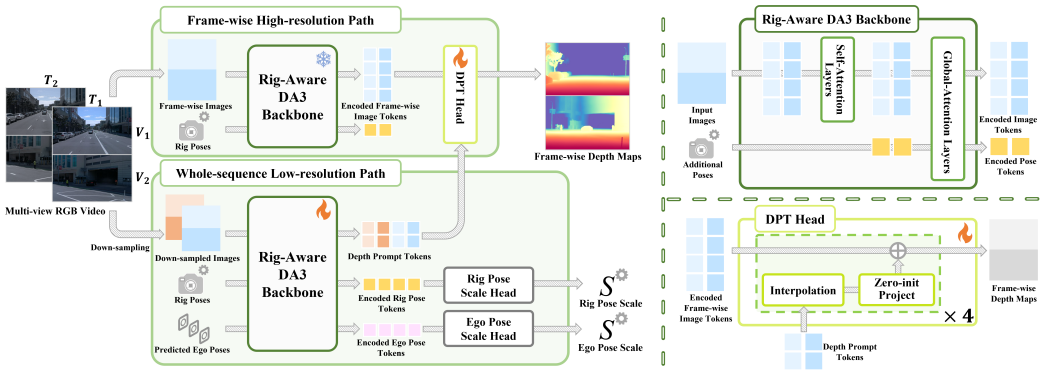

Figure 2: RigRecon employs a low-resolution path for efficient cross-view interaction and a high-resolution path to predict frame-wise depth maps with consistent scale.

ing MV-DUSt3R [19] and Fast3R [20] that scale to large numbers of views [21, 22] and handle dynamic scenes [23, 24]. Recently, geometry foundation models built on large transformer-based networks, such as VGGT [10], π^3 [25], DA3 [2], and DVGT [1], achieve strong performance by jointly modeling multi-view geometry and camera poses in a unified framework. However, their reliance on full global attention leads to memory consumption that scales as $\mathcal{O}(N^2)$ with the number of input images, making them unsuitable for long or multi-view videos. To mitigate this issue, streaming methods treat long videos as image streams and perform auto-regressive reconstruction. CUT3R [21] introduces memory tokens to retain history, while StreamVGGT [26] adopts cross-attention within a sliding window. However, such methods often suffer from inconsistency due to forgetting and cumulative errors. VGGT-long [11], DA3-Streaming [2], and Scal3R [27] further adopt chunk-wise processing with post-hoc alignment, enabling long-range reconstruction but requiring sequence partitioning regardless of length and introducing additional computational overhead. TTT3R [28], LoGeR [29] and VGG-T^3 [30] augment attention-based interaction with test-time training mechanisms, improving efficiency at the cost of reduced accuracy. RigRecon performs information interaction in a low-resolution space, ensuring consistency while maintaining efficiency.

Geometry model for driving scenes. Early driving perception approaches estimate scene geometry from multi-view depth prediction [31, 32, 33, 34]. Subsequent works improve surround-view depth estimation by explicitly exploiting geometric priors and cross-view relationships to enhance depth accuracy and multi-view consistency [35, 36, 37]. However, these methods struggle to produce globally consistent 3D reconstructions due to their 2.5D representations. To address this, TPVFormer [38] introduces voxel-based 3D occupancy prediction for dense scene modeling, and subsequent works further develop voxel-based and object-centric representations. However, discrete occupancy grids introduce quantization errors that limit geometric precision. The emergence of feedforward geometry models such as DUSt3R [18] and VGGT [10] demonstrates strong potential for 3D reconstruction, but they are not tailored for driving scenarios and cannot fully exploit meta information such as rig calibration. For driving scenarios, a line of works built on the VGGT-style [10] transformer backbone, including [39, 40, 41], extend it to 3DGS-based reconstruction frameworks for dynamic scenes, while Rig3R [42] is the first to incorporate rig calibration and temporal metadata into such architectures. However, their reliance on global attention leads to high computational overhead. DVGT [1] reduces this cost by applying attention masks to restrict interaction patterns, yet still struggles with memory constraints in long or high-resolution sequences. DVGT2 [43] further adopts a sliding-window strategy for sequential processing, but suffers from accumulated errors over long horizons. In contrast, RigRecon jointly models coarse geometry and pose scale within a unified low-resolution backbone, ensuring both consistency and efficiency in long multi-view video reconstruction.

3 Method

3.1 Overview

We address the task of rig-aware 3D reconstruction, which predicts the scene geometry from a multi-view video with calibrated cameras. To be specific, the input is a set of N RGB images $\{\mathbf{I}_i\}_{i=1}^{N}$,

where $\mathbf{I}_i \in \mathbb{R}^{3\times H\times W}$ and $N = P \cdot S$ denotes a video with P views and S frames. Each image is associated with known calibration, including the rig pose $\mathbf{T}_i^{\text{rig}} \in \mathbb{R}^{4\times4}$ (transformation to a reference camera) and intrinsics $\mathbf{K}_i \in \mathbb{R}^{3\times3}$. The goal is to predict a pixel-aligned point cloud $\mathbf{P}_i \in \mathbb{R}^{3\times H\times W}$ for each image.

We reformulate the output as a depth map $\mathbf{D}_i \in \mathbb{R}^{H\times W}$ for each image, a global scale factor $s_{\text{rig}} \in \mathbb{R}$ for the rig poses, and the ego poses $\{\mathbf{T}_i^{\text{ego}}\}_{i=1}^N$ with $\mathbf{T}_i^{\text{ego}} \in \mathbb{R}^{4\times4}$, representing the reference camera trajectory in a consistent scale. This formulation has several advantages: (1) the point cloud can be recovered via back-projection

$$\mathbf{P}_i(u,v) = \mathbf{D}_i(u,v) \cdot \mathbf{K}_i^{-1} \left[u, v, 1\right]^T$$

which maps each pixel (u, v) to 3D space; (2) it fully leverages the given calibration, avoiding errors from redundant pose prediction; (3) it enables consistent integration of geometric quantities across different scales, aligning relative-scale depth predictions and ego poses with metric-scale rig poses within a unified pipeline. The overall mapping can be expressed as:

$$\text{RigRecon} : \left\{\mathbf{I}_i, \mathbf{T}_i^{\text{rig}}, \mathbf{K}_i\right\}_{i=1}^N \rightarrow \left\{\mathbf{D}_i, \mathbf{T}_i^{\text{ego}}, s_{\text{rig}}\right\}_{i=1}^N.$$

As shown in Fig. 2, we propose a method that performs cross-view interaction in a low-resolution space and decodes high-resolution depth maps asynchronously in a frame-wise manner. A transformer backbone extracts coarse global geometry and scale from resized inputs, while a separate decoder reconstructs per-frame depth maps at full resolution. By disentangling ego and rig poses, the method better exploits rig calibration, achieving efficient and consistent reconstruction.

3.2 Whole-sequence low-resolution path

Existing feedforward multi-view reconstruction methods [1, 2, 10] rely on transformer-based architectures to model inter-image geometry cues at full resolution, ensuring consistent predictions across images. However, this design leads to prohibitive memory consumption as the number of views, frames, or image resolution increases, limiting applicability to real-world long or high-resolution videos. At the same time, cross-view interaction is essential for global geometric consistency, and attention-based modeling has demonstrated strong capability in capturing such relationships.

To address this dilemma, we introduce a low-cost interaction module that operates in a down-sampled resolution space. Specifically, we first down-sample input images to a fixed lower resolution. The resulting images are patchified and jointly processed through an attention-based encoder. This produces features that encode global geometric structure, which we denote as *depth prompt tokens*. These tokens are then injected into a depth decoder to asynchronously predict frame-wise depth maps. The encoder consists of alternating self-attention and global-attention layers and is initialized from the DA3 [2] backbone. Intuitively, this low-resolution interaction mechanism allows any pair of views and frames to exchange information through global attention. The reduction in resolution removes fine-grained visual details, and each token corresponds to a larger spatial receptive region, enabling the network to focus on coarse but globally consistent scene geometry while reducing memory cost.

For pose estimation, we feed front-view images into DA3 [2] to obtain accurate ego poses (see Sec. 4.5 for justification). Rig poses are directly obtained from calibration. These two poses, together with the depth maps, are defined in three different scale spaces: ego poses are predicted in the output scale space of DA3 [2], rig poses are defined in metric scale, and depth is supervised in a normalized scale space (obtained by normalizing the metric point cloud) for training stability. To unify them, we encode poses and intrinsics following VGGT [10], and let the resulting pose tokens interact with low-resolution image tokens in the backbone. Finally, the pose tokens are aggregated via attention-weighted summation and mapped to two scalar scale factors using two separate lightweight MLP heads. The two factors align the ego poses and rig poses with the normalized scale space, respectively. This design enables effective use of geometric priors while maintaining consistent scale alignment across depth and pose estimation.

The overall process of the whole-sequence low-resolution path $\mathcal{N}_\theta$ can be formulated as:

$$(\mathbf{F}_{\text{depth}}, s_{\text{ego}}, s_{\text{rig}}) = \mathcal{N}_\theta\Big(\mathcal{R}(\{\mathbf{I}_i\}_{i=1}^N), \{\mathbf{T}_i^{\text{rig}}\}_{i=1}^N, \{\hat{\mathbf{T}}_i^{\text{ego}}\}_{i=1}^N\Big),$$

where $\mathcal{R}(\cdot)$ denotes the image down-sampling operator applied to the input multi-view video. $\{\mathbf{T}_i^{\text{rig}}\}_{i=1}^N$ are metric-scale rig poses from calibration, and $\{\hat{\mathbf{T}}_i^{\text{ego}}\}_{i=1}^N$ are ego poses estimated by

DA3 [2]. $\mathbf{F}_{\text{depth}}$ represents the resulting depth prompt tokens, while s_{rig} and s_{ego} denote the predicted scale factors used to align relative and ego pose representations into a unified scale.

3.3 Frame-wise high-resolution path

With the depth prompt tokens, we leverage another backbone initialized from DA3 [2], paired with a specialized DPT head [44], to predict accurate frame-wise depth maps at the original resolution. The backbone is frozen during training. The input images are fed into the high-resolution backbone without down-sampling, together with rig poses that provide structural guidance for consistent reconstruction. The resulting features are fused with the corresponding depth prompt tokens in the DPT head, as shown in the bottom-right corner of Fig. 2. We discuss the choice of frame-wise decoding compared to image-wise and chunk-wise strategies in Sec. 4.5. Specifically, the depth prompt tokens are first spatially aligned to match the intermediate feature resolution in the DPT head and then projected through a zero-initialized convolutional layer. Finally, the resulting prompt features are directly added to the intermediate DPT representations. This design enables the high-resolution tokens to incorporate coarse geometry and scale priors encoded in the prompt tokens in an adaptive manner, resulting in consistent depth predictions and training stability.

The high-resolution path can be described as:

$$\{\mathbf{D}_i\}_{i=1}^{P} = \mathcal{G}_\theta\Big(\{\mathbf{I}_i\}_{i=1}^{P},\ \{\mathbf{T}_i^{\text{rig}}\}_{i=1}^{P},\ \mathbf{F}_{\text{depth}}\Big),$$

3.4 Training the proposed model.

Dense GT depth map construction. We train on multiple real-world autonomous driving datasets, including DDAD [45], NuScenes [46], Argoverse2 [47], KITTI [48], OpenScene [49], Waymo [50], and PandaSet [51]. However, training a dense depth prediction model requires pixel-level depth supervision [1], while most of these datasets only provide sparse LiDAR point clouds. Sparse supervision leads to artifacts such as floaters and noisy depth predictions. To mitigate this, we employ Infinidepth [52], which takes RGB images and sparse LiDAR as input and predicts dense metric-scale depth maps, serving as pseudo ground-truth supervision.

During training, we improve supervision quality by filtering out distant regions with unreliable depth estimates using a predefined depth threshold, thereby constructing distant masks. In addition, since pseudo-ground-truth depth may be noisy in some regions, we use a lightweight segmentation model, SegFormer [53], to identify sky regions and construct background masks, excluding unreliable supervision signals. Pixels masked out by either the distant masks or the background masks are considered invalid.

Loss designs. We adopt a confidence-weighted ℓ_1 loss for depth supervision following [2]. Alongside the depth map $\mathbf{D}_i$, the model jointly predicts a self-supervised, pixel-aligned confidence map $\mathbf{C}_i$ based on the aleatoric uncertainty formulation. The confidence map reflects the reliability of individual depth predictions, assigning lower confidence to ambiguous regions (e.g., sky and textureless areas) to reduce their contribution to the training loss. The loss is defined as:

$$\mathcal{L}_{\text{depth}} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \left(\left| \mathbf{D}_i - \mathbf{D}_i^{\text{gt}} \right| \cdot \mathbf{C}_i - \lambda \log \mathbf{C}_i \right),$$

where $\mathcal{M}$ denotes the set of valid pixels, and λ is a weighting coefficient (set to 1.0 in our setting) that regularizes the confidence prediction to avoid trivial solutions. This formulation encourages the model to assign higher confidence to accurate predictions while penalizing overconfident erroneous estimates, thereby improving robustness to noisy supervision.

With dense depth supervision, we further introduce a surface normal consistency loss following [14]:

$$\mathcal{L}_{\text{normal}} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \angle(\hat{\mathbf{n}}_i, \mathbf{n}_i),$$

where $\hat{\mathbf{n}}_i$ is the normal computed from the predicted point cloud via local cross products, $\mathbf{n}_i$ is the ground-truth normal, and $\angle(\cdot, \cdot)$ measures angular difference. This term encourages locally consistent surface geometry.

To suppress artifacts in regions without reliable depth supervision, we encourage the network to predict large depth values for invalid pixels. Specifically, we apply the following loss only to pixels

Table 1: Quantitative 3D reconstruction results of 2Hz videos across diverse datasets. The best and second-best results are highlighted in green and yellow, respectively.

Method	KITTI		NuScenes		Waymo		OpenScene		DDAD	
	Acc↓	Comp↓	Acc↓	Comp↓	Acc↓	Comp↓	Acc↓	Comp↓	Acc↓	Comp↓
MapAnything [54]	0.789	1.756	1.215	1.265	0.923	1.486	17.074	8.808	1.087	1.252
VGGT [10]	0.679	1.112	1.093	1.215	0.818	1.826	10.056	6.940	0.883	0.975
DA3 [2]	0.193	0.157	1.061	1.064	0.745	1.030	7.387	4.682	0.920	1.094
StreamVGGT [26]	0.625	1.335	1.322	3.330	1.095	3.718	15.348	7.841	1.148	2.319
DVGT [1]	0.334	0.433	0.827	0.709	0.746	0.922	12.894	11.902	0.838	0.730
Ours	0.139	0.108	0.565	0.458	0.323	0.289	1.803	0.941	0.482	0.458

Table 2: Quantitative depth estimation results of 2Hz videos across diverse datasets.

Method	KITTI		NuScenes		Waymo		OpenScene		DDAD	
	Abs Rel↓	σ<1.25↑	Abs Rel↓	σ<1.25↑	Abs Rel↓	σ<1.25↑	Abs Rel↓	σ<1.25↑	Abs Rel↓	σ<1.25↑
MapAnything [54]	0.305	0.489	0.508	0.366	0.301	0.579	0.471	0.394	0.534	0.384
VGGT [10]	0.053	0.968	0.171	0.820	0.065	0.949	0.190	0.770	0.171	0.786
DA3 [2]	0.046	0.988	0.178	0.804	0.088	0.921	0.199	0.750	0.145	0.835
StreamVGGT [26]	0.046	0.982	0.171	0.812	0.065	0.946	0.181	0.783	0.144	0.833
DVGT [1]	0.092	0.931	0.187	0.784	0.107	0.893	0.264	0.647	0.202	0.745
Ours	0.028	0.993	0.164	0.839	0.062	0.951	0.149	0.841	0.120	0.882

filtered out by the background masks or by the distant masks:

$$\mathcal{L}_{\text{far}} = \frac{1}{|\mathcal{M}_{\text{invalid}}|} \sum_{i \in \mathcal{M}_{\text{invalid}}} \max\left(0, d_{\max} - \mathbf{D}_i\right),$$

where $\mathcal{M}_{\text{invalid}}$ denotes the set of invalid pixels and $d_{\max}$ is a predefined upper bound (set to 41.0 in our setting). This loss encourages sufficiently large predictions in regions without reliable supervision, allowing them to be removed from the final reconstruction using a simple depth threshold.

To train the scale head, we apply an ℓ_1 loss to the 7-dimensional pose representations consisting of a 3D translation vector and a 4D rotation quaternion. Specifically, we apply the predicted scale factors, s_{ego} and s_{rig}, to the translation vectors of the corresponding predicted poses, while keeping their rotation quaternions unchanged. We denote the rescaled predicted and ground-truth pose representations at frame i as $\hat{\mathbf{p}}_i$ and $\mathbf{p}_i^{\text{gt}}$, respectively. The loss is defined as

$$\mathcal{L}_{\text{pose}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \hat{\mathbf{p}}_i - \mathbf{p}_i^{\text{gt}} \right\|_1.$$

This supervision enforces accurate pose regression and consistent scale alignment across both translation and rotation components.

The final loss is defined as:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{depth}} + \lambda_2 \mathcal{L}_{\text{normal}} + \lambda_3 \mathcal{L}_{\text{far}} + \lambda_4 \mathcal{L}_{\text{pose}}.$$

4 Experiments

We conduct extensive experiments to evaluate the performance of RigRecon. The evaluation is conducted under two settings: **2Hz**, where the number of input images is capped at 200 to ensure all compared methods can run without out-of-memory issues; and **10Hz**, where the number of images is capped at 1000 to assess performance on long video sequences. Detailed dataset usage and evaluation configurations are provided in the appendix.

Table 3: Quantitative 3D reconstruction results of 10Hz videos across diverse datasets.

Method	KITTI		Waymo		OpenScene		DDAD	
	Acc↓	Comp↓	Acc↓	Comp↓	Acc↓	Comp↓	Acc↓	Comp↓
CUT3R [21]	0.624	0.836	1.390	3.310	18.098	4.071	0.984	2.508
VGGT-Long [11]	0.590	1.590	0.689	1.234	2.674	1.243	1.208	1.586
DA3-Streaming [2]	0.561	0.933	0.605	0.702	1.064	0.617	0.855	1.365
TTT3R [28]	0.424	3.795	1.589	7.027	5.801	7.430	1.597	7.274
LoGeR [29]	0.226	0.206	1.907	4.730	24.865	8.211	1.377	7.167
Ours	0.148	0.114	0.343	0.295	1.997	0.632	0.506	0.480

Table 4: Quantitative depth estimation results of 10Hz videos across diverse datasets.

Method	KITTI		Waymo		OpenScene		DDAD	
	Abs Rel↓	σ<1.25↑	Abs Rel↓	σ<1.25↑	Abs Rel↓	σ<1.25↑	Abs Rel↓	σ<1.25↑
CUT3R [21]	0.073	0.957	0.090	0.916	0.183	0.774	0.200	0.737
VGGT-Long [11]	0.060	0.947	0.066	0.947	0.197	0.760	0.190	0.757
DA3-Streaming [2]	0.041	0.989	0.091	0.917	0.198	0.754	0.143	0.840
TTT3R [28]	0.096	0.962	0.087	0.920	0.182	0.775	0.193	0.747
LoGeR [29]	0.044	0.978	0.064	0.949	0.178	0.788	0.127	0.860
Ours	0.030	0.992	0.064	0.950	0.152	0.839	0.120	0.882

4.1 Point cloud reconstruction and depth estimation metrics

We report quantitative 3D reconstruction and depth estimation results on both 2Hz and 10Hz video clips in Table 1, Table 2, Table 3, and Table 4. All predictions are aligned to ground-truth depth capped at 40 m for robust metric scale recovery. We apply affine alignment [55] for depth maps and the Umeyama algorithm [56] for point cloud alignment. RigRecon consistently achieves state-of-the-art performance across nearly all datasets and metrics.

For 3D reconstruction, our method outperforms prior approaches by a significant margin on all datasets at 2Hz, reducing both accuracy and completeness errors substantially (e.g., improving KITTI [48] from 0.193/0.157 to 0.139/0.108). Similar trends hold for 10Hz videos, where RigRecon maintains strong performance even on long sequences with up to 1000 images. For depth estimation, RigRecon achieves consistently strong performance across all datasets, with notable improvements such as KITTI [48] (0.028, 0.993) and DDAD [45] (0.120, 0.882), demonstrating both low Abs Rel error and high $\sigma < 1.25$ accuracy.

We notice that RigRecon lags slightly behind DA3-Streaming [2] on OpenScene [49]. We attribute this to imbalanced data sampling, as OpenScene [49] contains 8 views and substantially greater scene diversity, yet is assigned a similar training weight as smaller datasets. As a result, RigRecon occasionally predicts less stable scale for this dataset, which affects point cloud consistency.

4.2 Ego pose estimation metrics

We report ego pose estimation results on 2Hz and 10Hz video clips in Table 5 and Table 6. Predicted poses are aligned to the ground truth using the Umeyama algorithm [56]. RigRecon achieves competitive and often superior performance across datasets.

At 2Hz, our method achieves the best performance on Waymo [50] and DDAD [45], and remains highly competitive on other datasets. At 10Hz, RigRecon achieves the best results on Waymo [50] and DDAD [45] by a clear margin (e.g., 0.897 and 0.944 AUC@30), while matching the best performance on KITTI [48]. These results highlight the effectiveness of using DA3 [2]-predicted front-view trajectories and demonstrate robust pose estimation across diverse scenarios.

4.3 Qualitative results

We provide quantitative results to further demonstrate the effectiveness of the proposed method, as shown in Fig. 3. By performing unified attention-based interaction over calibration and images in

Table 5: Quantitative ego pose estimation results of 2Hz videos across diverse datasets.

Method	KITTI AUC@30↑	NuScenes AUC@30↑	Waymo AUC@30↑	OpenScene AUC@30↑	DDAD AUC@30↑
MapAnything [54]	0.771	0.825	0.850	0.456	0.854
VGGT [10]	0.749	0.773	0.840	0.495	0.898
DA3 [2]	0.740	0.845	0.890	0.628	0.932
StreamVGGT [26]	0.720	0.777	0.775	0.511	0.855
DVGT [1]	0.670	0.739	0.735	0.521	0.870
Ours	0.746	0.806	0.905	0.620	0.954

Table 6: Quantitative ego pose estimation results of 10Hz videos across diverse datasets.

Method	KITTI AUC@30↑	Waymo AUC@30↑	OpenScene AUC@30↑	DDAD AUC@30↑
CUT3R [21]	0.536	0.286	0.151	0.308
VGGT-Long [11]	0.730	0.773	0.550	0.788
DA3-Streaming [2]	0.743	0.703	0.679	0.453
TTT3R [28]	0.743	0.825	0.539	0.875
LoGeR [29]	0.594	0.732	0.489	0.781
Ours	0.743	0.897	0.619	0.944

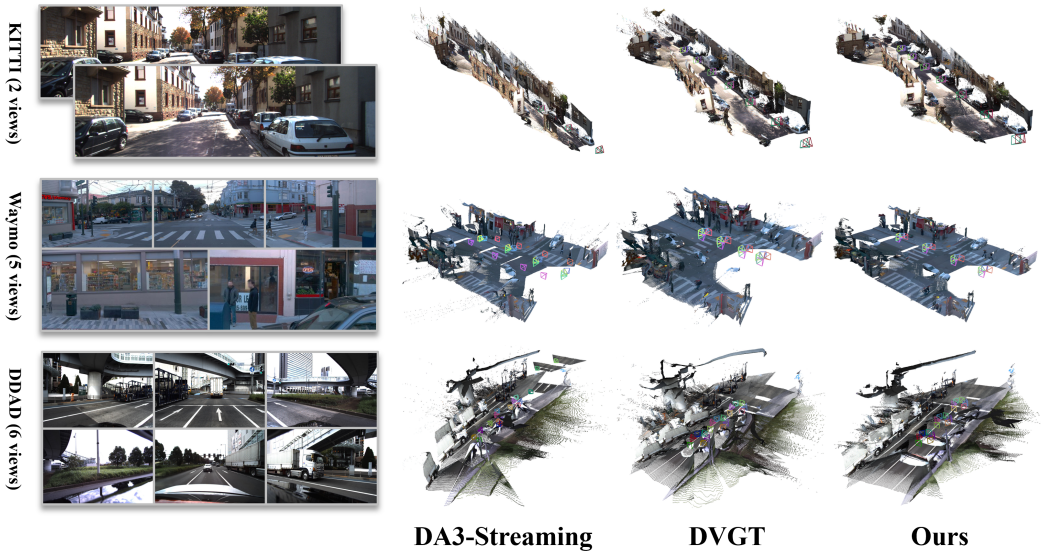

Figure 3: Qualitative comparison of RigRecon with DVGT [1] and DA3-Streaming [2] on rig-aware 3D reconstruction. Camera poses are depicted as colored wireframe frustums.

a low-resolution space, RigRecon achieves improved consistency and pose accuracy compared to DVGT [1] and DA3-Streaming [2], while maintaining high efficiency.

4.4 Inference Memory Consumption Evaluation

We report quantitative results on inference memory consumption in Table 7. The experiments are conducted on the Waymo dataset [50] at 2 Hz with an input resolution of 504×672. Our primary focus is peak memory consumption, which determines whether longer video clips can be processed at higher frame rates without out-of-memory (OOM) errors. The results demonstrate that our design effectively reduces peak inference memory consumption.

We also note that, while RigRecon's peak memory consumption increases with sequence length, streaming-based methods (e.g., VGGT-Long [11], DA3-Streaming [2], and LoGeR [29]) maintain

Table 7: Comparisons of peak inference memory consumption on the Waymo dataset [50] at 2Hz with an input resolution of 504×672.

Method	Peak Memory (GB)↓
MapAnything [54]	40.85
VGGT [10]	40.20
StreamVGGT [26]	26.34
VGGT-Long [11]	26.26
DA3 [2]	28.22
DA3-Streaming [2]	22.68
DVGT [1]	27.65
Ours	20.72

Table 8: Ablation studies on the Waymo dataset [50].

Method	AUC@30↑
Left front views	0.751
Left views	0.463
All views low resolution	0.841
Backbone predictions	0.718
Front views (Ours)	0.901

(a) Ego pose estimation methods.

Method	Acc↓	Comp↓	Abs Rel↓	σ<1.25↑	AUC@30↑
168×224	0.395	0.338	0.070	0.945	0.905
84×112	0.424	0.328	0.075	0.941	0.905
126×168 (Ours)	0.416	0.334	0.073	0.943	0.905

(b) Resolution choices.

Method	Acc↓	Comp↓	Abs Rel↓	σ<1.25↑	AUC@30↑
Image-wise	0.423	0.338	0.075	0.940	0.905
Chunk-wise (chunk size 2)	0.404	0.340	0.070	0.946	0.905
Frame-wise (Ours)	0.416	0.334	0.073	0.943	0.905

(c) Decoding designs.

Method	Acc↓	Comp↓	Abs Rel↓	σ<1.25↑	AUC@30↑
MapAnything [54] + DA3 pose	0.961	1.118	0.312	0.562	0.792
DA3 [2] + DA3 pose	0.768	1.009	0.088	0.921	0.888
Ours	0.323	0.289	0.062	0.951	0.905

(d) Methods using additional DA3 pose estimates.

nearly constant memory usage. However, the two approaches serve complementary purposes: RigRecon jointly reconstructs video clips to ensure geometric consistency across point cloud chunks, whereas streaming methods align reconstructed chunks to support much longer sequences. RigRecon can therefore be integrated with streaming frameworks to extend reconstruction to longer videos.

4.5 Ablations

Ablations on ego pose estimation methods are evaluated on Waymo [50] at 10Hz to assess the ability of different methods to predict long pose sequences. All other ablation experiments are conducted at 2Hz. Models are trained with a batch size of 8 on 8 H20 GPUs, using a smaller-scale setting to reduce computational cost for large-scale experiments.

Ablations on ego pose estimation methods. We compare strategies for ego pose estimation:

- Left front/Left/Front views: DA3 [2] predict poses using images from specified views.
- All views low resolution: all images are used to predict poses with DA3 [2], where inputs are resized to $1/4$ of the original resolution to avoid OOM.
- Backbone predictions: the low-resolution backbone is trained to directly predict poses for each image, following VGGT [10].

Results are shown in Table 8a. The first two rows indicate that front views outperform other view selections, likely due to stronger overlap. The next two rows show that down-sampling degrades pose accuracy. Based on these observations, we adopt front views for ego pose estimation.

Ablations on resolution choices. We evaluate different resolution settings for the low-resolution path. As shown in Table 8b, although higher resolution yields better performance, it incurs a higher memory cost. Therefore, we choose 126×168 to balance accuracy and efficiency.

Ablations on decoding designs. We compare frame-wise decoding with image-wise and chunk-wise strategies in Table. 8c. Image-wise decoding performs slightly worse, likely due to the lack of high-resolution interaction across images, which weakens consistency in fine-grained structures. Frame-wise and chunk-wise decoding achieve comparable performance, while frame-wise decoding is more flexible as it does not require handling sequences with length not divisible by the chunk size.

Comparisons with methods using additional DA3 pose estimates. We conduct additional experiments by providing DA3 pose estimates to baseline methods that optionally accept camera poses as input. Specifically, we first use DA3 [2] to estimate the global camera poses for all input frames. We then align the estimated poses with the rig poses via least-squares scale fitting and provide the aligned poses to the evaluated methods. The results are reported in Table 8d.

RigRecon consistently outperforms these baselines even when augmented with DA3 pose estimates. We attribute this to discrepancies between the relative camera geometry estimated by DA3 [2] and the ground truth, which cannot be fully corrected by a single global scale alignment. These results further highlight the advantage of jointly modeling reconstruction and scale consistency within our framework.

5 Conclusion

We introduce RigRecon, a feedforward approach for rig-aware 3D reconstruction that efficiently processes long multi-view videos. It employs a low-resolution path to exploit rig calibration and down-sampled inputs for coarse geometric reasoning, and a high-resolution path that asynchronously decodes frame-wise depth maps conditioned on these cues. Extensive experiments demonstrate its effectiveness across videos with diverse camera rigs and frame rates.

Limitations. The main limitation of RigRecon is its limited compatibility with different backbones. Adapting the method to a new backbone requires retraining to jointly optimize the high-resolution decoder and depth prompt injection mechanism. A promising direction for future work is to replace latent prompt tokens with explicit coarse geometric representations, potentially improving the flexibility of the framework across different backbones.

Acknowledgments

This work was partially supported by NSFC (No. 62402427), Zhejiang Provincial Natural Science Foundation of China (No. LZ26F020003), Xiangjiang Laboratory (No. 24XJJCYJ01004), "Innovation Yongjiang 2035" Key R&D Program (No. 2025Z061), and Information Technology Center and State Key Lab of CAD&CG, Zhejiang University.

References

[1] S. Zuo, Z. Xie, W. Zheng, S. Xu, F. Li, S. Jiang, L. Chen, Z.-X. Yang, and J. Lu, "Dvgt: Driving visual geometry transformer," *arXiv preprint arXiv:2512.16919*, 2025.

[2] H. Lin, S. Chen, J. Liew, D. Y. Chen, Z. Li, G. Shi, J. Feng, and B. Kang, "Depth anything 3: Recovering the visual space from any views," *arXiv preprint arXiv:2511.10647*, 2025.

[3] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[4] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[5] B. Kerbl, G. Kopanas, T. Leimkühler, G. Drettakis, *et al.*, "3d gaussian splatting for real-time radiance field rendering.," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[6] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang, "Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction," *arXiv preprint arXiv:2106.10689*, 2021.

[7] H. Fu, D. Zhang, Z. Zhao, J. Cui, D. Liang, C. Zhang, D. Zhang, H. Xie, B. Wang, and X. Bai, "Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 24823–24834, 2025.

[8] S. Gao, J. Yang, L. Chen, K. Chitta, Y. Qiu, A. Geiger, J. Zhang, and H. Li, "Vista: A generalizable driving world model with high fidelity and versatile controllability," *Advances in Neural Information Processing Systems*, vol. 37, pp. 91560–91596, 2024.

[9] A. Hu, L. Russell, H. Yeo, Z. Murez, G. Fedoseev, A. Kendall, J. Shotton, and G. Corrado, "Gaia-1: a generative world model for autonomous driving (2023)," *URL https://arxiv. org/abs/2309.17080*, vol. 3.

[10] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, "Vggt: Visual geometry grounded transformer," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025.

[11] K. Deng, Z. Ti, J. Xu, J. Yang, and J. Xie, "Vggt-long: Chunk it, loop it, align it–pushing vggt's limits on kilometer-scale long rgb sequences," *arXiv preprint arXiv:2507.16443*, 2025.

[12] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4104–4113, 2016.

[13] L. Pan, D. Baráth, M. Pollefeys, and J. L. Schönberger, "Global structure-from-motion revisited," in *European Conference on Computer Vision*, pp. 58–77, Springer, 2024.

[14] R. Wang, S. Xu, C. Dai, J. Xiang, Y. Deng, X. Tong, and J. Yang, "Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5261–5271, 2025.

[15] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[16] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10371–10381, 2024.

[17] J. Wang, N. Karaev, C. Rupprecht, and D. Novotny, "Vggsfm: Visual geometry grounded deep structure from motion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 21686–21697, 2024.

[18] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "Dust3r: Geometric 3d vision made easy," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20697–20709, 2024.

[19] Z. Tang, Y. Fan, D. Wang, H. Xu, R. Ranjan, A. Schwing, and Z. Yan, "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.

[20] J. Yang, A. Sax, K. J. Liang, M. Henaff, H. Tang, A. Cao, J. Chai, F. Meier, and M. Feiszli, "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.

[21] Q. Wang, Y. Zhang, A. Holynski, A. A. Efros, and A. Kanazawa, "Continuous 3d perception model with persistent state," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 10510–10522, 2025.

[22] J. Zhang, C. Herrmann, J. Hur, V. Jampani, T. Darrell, F. Cole, D. Sun, and M.-H. Yang, "Monst3r: A simple approach for estimating geometry in the presence of motion," *arXiv preprint arXiv:2410.03825*, 2024.

[23] L. Jin, R. Tucker, Z. Li, D. Fouhey, N. Snavely, and A. Holynski, "Stereo4d: Learning how things move in 3d from internet stereo videos," *arXiv preprint arXiv:2412.09621*, 2024.

[24] X. Chen, Y. Chen, Y. Xiu, A. Geiger, and A. Chen, "Easi3r: Estimating disentangled motion from dust3r without training," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9158–9168, 2025.

[25] Y. Wang, J. Zhou, H. Zhu, W. Chang, Y. Zhou, Z. Li, J. Chen, J. Pang, C. Shen, and T. He, "π^3: Permutation-equivariant visual geometry learning," *arXiv preprint arXiv:2507.13347*, 2025.

[26] D. Zhuo, W. Zheng, J. Guo, Y. Wu, J. Zhou, and J. Lu, "Streaming 4d visual geometry transformer," *arXiv preprint arXiv:2507.11539*, 2025.

[27] T. Xie, P. Yang, Y. Jin, Y. Cai, W. Yin, W. Ren, Q. Zhang, W. Hua, S. Peng, X. Guo, *et al.*, "Scal3r: Scalable test-time training for large-scale 3d reconstruction," *arXiv preprint arXiv:2604.08542*, 2026.

[28] X. Chen, Y. Chen, Y. Xiu, A. Geiger, and A. Chen, "Ttt3r: 3d reconstruction as test-time training," *arXiv preprint arXiv:2509.26645*, 2025.

[29] J. Zhang, C. Herrmann, J. Hur, C. Sun, M.-H. Yang, F. Cole, T. Darrell, and D. Sun, "Loger: Long-context geometric reconstruction with hybrid memory," in *European Conference on Computer Vision*, pp. 263–281, Springer, 2026.

[30] S. Elflein, R. Li, S. Agostinho, Z. Gojcic, L. Leal-Taixé, Q. Zhou, and A. Osep, "Vgg-t^3: Offline feed-forward 3d reconstruction at scale," *arXiv preprint arXiv:2602.23361*, 2026.

[31] A.-Q. Cao and R. De Charette, "Scenerf: Self-supervised monocular 3d scene reconstruction with radiance fields," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9387–9398, 2023.

[32] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3828–3838, 2019.

[33] Y. Wei, L. Zhao, W. Zheng, Z. Zhu, Y. Rao, G. Huang, J. Lu, and J. Zhou, "Surrounddepth: Entangling surrounding views for self-supervised multi-camera depth estimation," in *Conference on robot learning*, pp. 539–549, PMLR, 2023.

[34] J. Guo, Y. Ding, X. Chen, S. Chen, B. Li, Y. Zou, X. Lyu, F. Tan, X. Qi, Z. Li, *et al.*, "Dist-4d: Disentangled spatiotemporal diffusion with metric depth for 4d driving scene generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27231–27241, 2025.

[35] V. Guizilini, I. Vasiljevic, R. Ambrus, G. Shakhnarovich, and A. Gaidon, "Full surround monodepth from multiple cameras," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5397–5404, 2022.

[36] S. Zhang, J. Li, C. Liu, J. Li, and D. Yang, "E^{2}depth: Efficient self-supervised surround-view depth estimation with explicit geometric enhancement," *IEEE Robotics and Automation Letters*, 2026.

[37] S. Abualhanud, C. Grannemann, and M. Mehltretter, "Cylinderdepth: Cylindrical spatial attention for multi-view consistent self-supervised surround depth estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 169–179, 2026.

[38] Y. Huang, W. Zheng, Y. Zhang, J. Zhou, and J. Lu, "Tri-perspective view for vision-based 3d semantic occupancy prediction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9223–9232, 2023.

[39] Z. Yu, Z. Wang, Y. Xie, Y. Wang, X. Zhang, Y. Zhan, and K. Zhan, "Streetforward: Perceiving dynamic street with feedforward causal attention," *arXiv preprint arXiv:2603.19552*, 2026.

[40] X. Chen, Z. Xiong, Y. Chen, G. Li, N. Wang, H. Luo, L. Chen, H. Sun, B. Wang, G. Chen, *et al.*, "Dggt: Feedforward 4d reconstruction of dynamic driving scenes using unposed images," *arXiv preprint arXiv:2512.03004*, 2025.

[41] H. Yu, K. Xiao, J. Wang, R. Hao, Y. Huang, G. Hu, H. Qin, B. Jing, Y. Bo, and P. Luo, "Recondrive: Fast feed-forward 4d gaussian splatting for autonomous driving scene reconstruction," *arXiv preprint arXiv:2603.07552*, 2026.

[42] S. Li, P. Kachana, P. Chidananda, S. Nair, Y. Furukawa, and M. Brown, "Rig3r: Rig-aware conditioning for learned 3d reconstruction," *arXiv preprint arXiv:2506.02265*, 2025.

[43] S. Zuo, Z. Xie, W. Zheng, S. Xu, F. Li, H. Li, L. Chen, Z.-X. Yang, and J. Lu, "Dvgt-2: Vision-geometry-action model for autonomous driving at scale," *arXiv preprint arXiv:2604.00813*, 2026.

[44] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12179–12188, 2021.

[45] V. Guizilini, R. Ambrus, S. Pillai, A. Raventos, and A. Gaidon, "3d packing for self-supervised monocular depth estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2485–2494, 2020.

[46] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

[47] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, *et al.*, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," *arXiv preprint arXiv:2301.00493*, 2023.

[48] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The international journal of robotics research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[49] O. Contributors, "Openscene: The largest up-to-date 3d occupancy prediction benchmark in autonomous driving." `https://github.com/OpenDriveLab/OpenScene`, 2023.

[50] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2446–2454, 2020.

[51] P. Xiao, Z. Shao, S. Hao, Z. Zhang, X. Chai, J. Jiao, Z. Li, J. Wu, K. Sun, K. Jiang, *et al.*, "Pandaset: Advanced sensor suite dataset for autonomous driving," in *2021 IEEE international intelligent transportation systems conference (ITSC)*, pp. 3095–3101, IEEE, 2021.

[52] H. Yu, H. Lin, J. Wang, J. Li, Y. Wang, X. Zhang, Y. Wang, X. Zhou, R. Hu, and S. Peng, "Infinidepth: Arbitrary-resolution and fine-grained depth estimation with neural implicit fields," *arXiv preprint arXiv:2601.03252*, 2026.

[53] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," *Advances in neural information processing systems*, vol. 34, pp. 12077–12090, 2021.

[54] N. Keetha, N. Müller, J. Schönberger, L. Porzi, Y. Zhang, T. Fischer, A. Knapitsch, D. Zauss, E. Weber, N. Antunes, *et al.*, "Mapanything: Universal feed-forward metric 3d reconstruction," *arXiv preprint arXiv:2509.13414*, 2025.

[55] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," *Advances in neural information processing systems*, vol. 27, 2014.

[56] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 13, no. 4, pp. 376–380, 2002.

[57] B. Kerbl, G. Kopanas, T. Leimkühler, G. Drettakis, *et al.*, "3d gaussian splatting for real-time radiance field rendering.," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[58] D. Charatan, S. L. Li, A. Tagliasacchi, and V. Sitzmann, "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.

[59] M. Roberts, J. Ramapuram, A. Ranjan, A. Kumar, M. A. Bautista, N. Paczan, R. Webb, and J. M. Susskind, "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.

[60] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

A Appendix

We provide details regarding implementation, evaluation settings, and training settings of the work in the appendix.

A.1 Combination with 3DGS

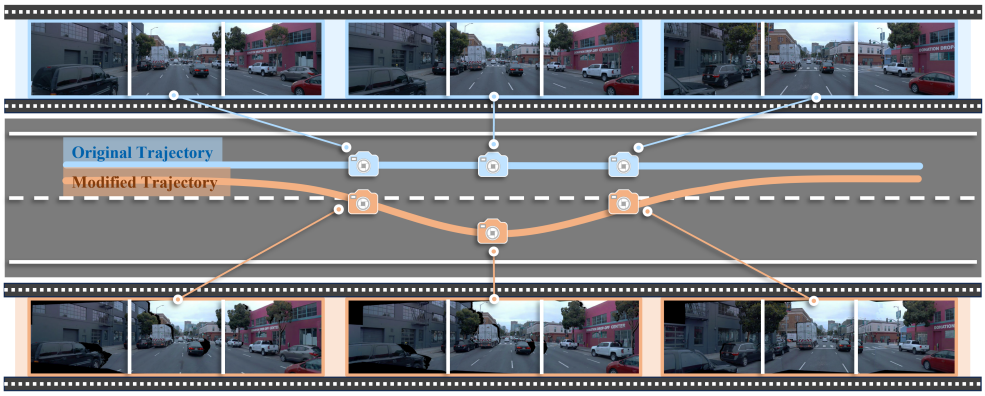

Figure 4: Qualitative results of 3DGS-based novel view rendering. The upper row (blue) shows input video frames, while the lower row (orange) presents renderings from RigRecon along a novel trajectory modified from the predicted ego poses.

We introduce a Gaussian head to demonstrate that the accurate reconstruction from RigRecon can facilitate advanced neural rendering methods such as 3DGS [57]. The head is implemented as a single DPT [44] module that takes encoded RGB tokens from the high-resolution path to predict pixel-aligned 3DGS attributes [58], without using the depth prompt token. The final 3DGS representation is constructed by combining the predicted Gaussian attributes with the reconstructed point cloud from RigRecon. Due to the lack of multi-view supervision in open-source driving datasets, the head is trained separately on the synthetic indoor dataset Hypersim [59]. Training details are provided in the appendix. As shown in Fig. 4, given a multi-view rig video, RigRecon reconstructs the scene frame by frame and enables consistent multi-view rendering along a specified novel trajectory.

A.2 Implementation details

The backbones in both paths of RigRecon are initialized from the attention-based backbone of DA3 [2]. The backbone consists of 40 attention layers: the first 13 layers use self-attention only, while the remaining layers alternate between self-attention and global attention (refer to [2] for details). The depth prompt token is taken from the final layer of the low-resolution path. The encoded image tokens used by the DPT heads are extracted from layers 20, 28, 34, and 40 of the high-resolution path. Input images are down-sampled to a resolution of 126×168 for global interaction in the low-resolution path. All resolutions are reported in pixels.

A.3 Ablations on prompt token injection designs

We compare our DPT-injection design with an alternative that injects depth prompt tokens into the 33rd, 35th, 37th, and 39th layers of the high-resolution attention backbone using the same zero-initialized convolutional design. As shown in Table 9, both settings achieve similar performance, while our design is more computationally efficient during training.

Table 9: Comparisons between prompt token injection designs on the Waymo dataset [50].

Method	Acc↓	Comp↓	Abs Rel↓	σ<1.25↑	AUC@30↑
Attention injection	0.416	0.344	0.072	0.943	0.905
DPT injection (Ours)	0.416	0.334	0.073	0.943	0.905

A.4 Training details

Training datasets. We summarize the training datasets in Table 10, including the number of scenes, and the distributions of views, frame counts, and frame rates. During training, we first randomly sample a dataset according to the weights in the table. Each video is then clipped into segments of 20 to 50 frames, from which we uniformly sample 8 frames with 3 views. This ensures sufficient overlap among the selected frames.

Table 10: Statistics of training datasets.

Dataset	Scenes	Min Frames	Max Frames	Avg Frames	Views	Fps	Weights
DDAD [45]	150	50	100	84	6	10	1
NuScene [46]	700	39	41	40	6	2	1
Argoverse2 [47]	700	301	339	320	7	10	1
KITTI [48]	58	12	5167	681	2	10	1
OpenScene [49]	943	78	5962	565	8	10	2
Waymo [50]	798	171	200	198	5	10	3
Pandaset [51]	103	80	80	80	6	10	1

Training configurations. We use the AdamW [60] optimizer with a learning rate of 5×10^{-5} and a weight decay of 0.01. Training is performed for 20k steps with a batch size of 64 on 64 H20 GPUs. Input images are resized to a fixed resolution of 504×672 for high-resolution training. To reduce memory consumption, gradient checkpointing is applied to all attention layers. All loss weighting coefficients λ are set to 1.0. All resolutions are reported in pixels.

Training details of the Gaussian head. The Gaussian head is trained separately on the Hypersim [59] dataset. During training, it takes encoded tokens from a single view of the DA3 [2] backbone and predicts 3DGS attributes. The predicted Gaussians are rendered using ground-truth geometry, and supervision is applied via an ℓ_1 loss on novel views. Training runs for 40k steps with a batch size of 32 on 8 H20 GPUs. Once trained, the DPT head can be directly integrated with RigRecon without further adaptation.

A.5 Evaluation details

Evaluation datasets. We summarize the evaluation datasets in Table 10. The maximum number of frames used in each experiment is reported to ensure that a single batch contains fewer than approximately 200 and 1000 frames for the 2Hz and 10Hz settings, respectively. For datasets originally sampled at 10Hz, we down-sample them to 2Hz for the corresponding experiments.

Evaluation configurations. The image resolution is fixed to 504×672 for all evaluations. For 3D reconstruction and depth estimation, depth maps are aligned to ground-truth depth values capped at 40 m to recover metric scale while reducing the influence of noisy regions in predictions. We apply affine alignment [55] for depth maps and the Umeyama algorithm [56] for point cloud alignment. For efficiency, we uniformly sample 24 frames from full predictions for computing 3D reconstruction metrics. For ego pose estimation, predicted poses are aligned to ground truth using the Umeyama algorithm [56]. All resolutions are reported in pixels.

Table 11: Statistics of evaluation datasets.

Dataset	Scenes	Views	Fps	Frames Used (2Hz)	Frames Used (10Hz)
DDAD [45]	50	6	10	20	100
NuScene [46]	150	6	2	30	N/A
KITTI [48]	13	2	10	40	200
OpenScene [49]	134	8	10	25	125
Waymo [50]	150	5	10	40	200

Metric definitions. We follow prior works [1, 54, 10] and evaluate geometric quality using Accuracy and Completeness. Let $P = \{p_i\}$ and $G = \{g_j\}$ denote valid point sets from the predicted point map P_{pred} and ground-truth P_{gt}, respectively. The metrics are defined as

$$\text{Accuracy} = \frac{1}{|P|} \sum_{p \in P} \min_{g \in G} \|p - g\|_2, \quad \text{Completeness} = \frac{1}{|G|} \sum_{g \in G} \min_{p \in P} \|g - p\|_2,$$

where $\|\cdot\|_2$ denotes the L2 distance. Both metrics are measured in meters.

For depth evaluation, we report Absolute Relative error (Abs Rel) and threshold accuracy ($\delta < 1.25$). Let Ω denote the set of valid pixels with ground truth, and $u \in \Omega$ be a pixel index. Given predicted depth D_{pred} and ground-truth depth D_{gt}, we compute

$$\text{Abs Rel} = \frac{1}{|\Omega|} \sum_{u \in \Omega} \frac{|D_{\text{pred}}(u) - D_{\text{gt}}(u)|}{D_{\text{gt}}(u)},$$

$$\text{Acc}_\delta = \frac{1}{|\Omega|} \sum_{u \in \Omega} \mathbb{I}\left(\max\left(\frac{D_{\text{pred}}(u)}{D_{\text{gt}}(u)}, \frac{D_{\text{gt}}(u)}{D_{\text{pred}}(u)} \right) < \delta \right),$$

where $\mathbb{I}(\cdot)$ is the indicator function.

For ego-pose estimation, we report AUC@30 based on pose accuracy. We compute Relative Rotation Accuracy (RRA) and Relative Translation Accuracy (RTA) for all frame pairs, where RRA measures geodesic distance between rotation matrices and RTA measures angular deviation between translation vectors. The accuracy at threshold τ is defined as the percentage of pairs satisfying $\max(\text{RRA}, \text{RTA}) < \tau$. The final metric is

$$\text{AUC@30}^\circ = \frac{1}{30} \int_0^{30} \text{Acc}(\tau)\, d\tau,$$

where $\text{Acc}(\tau)$ denotes the fraction of pairs satisfying the threshold.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately summarize the key contributions, which are elaborated in Sec. 3 and empirically validated in Sec. 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Sec. 5 explicitly discusses limitations in handling dynamic scenes and outlines potential directions for future work.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper focuses on empirical modeling and leverages established geometric formulations (e.g., back-projection), without introducing new theoretical results requiring formal proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides sufficient implementation details, training configurations, evaluation protocols, and dataset statistics in the appendix to support reproducibility of the main results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Although code is not released at submission time, the method is described in sufficient detail and relies on public datasets; code and model weights are planned for future release.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Detailed training and testing settings, including hyperparameters, optimizer configuration, and data splits, are specified in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are not reported due to the high computational cost of large-scale experiments; instead, robustness is supported through extensive ablations and evaluations across diverse datasets.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The appendix specifies compute resources, including GPU type, training duration, and batch size, sufficient to reproduce the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research adheres to the NeurIPS Code of Ethics in all respects.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

Justification: The work focuses on foundational 3D vision research without direct societal deployment, and thus does not explicitly analyze broader societal impacts.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not introduce high-risk assets. It builds on standard 3D vision models and publicly available datasets, making safeguards not applicable.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All public datasets used and third-party methods compared are properly cited, and their usage follows the respective licenses and terms.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new datasets, models, or other assets at submission time. We plan to release the trained models and corresponding code in the future.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve human subjects or crowdsourcing experiments.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects, so there are no participant risks or IRB considerations.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method in this research does not involve LLMs in any important, original, or non-standard way.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.