PRETRAINING OF REGISTERS WITH COVISIBILITY FOR LONG-SEQUENCE 3D RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Feed-forward reconstruction transformers rely on all-to-all patch attention across views (*global attention*), whose quadratic cost in views and patches prevents scaling to a large number of views. Prior attempts to reduce this cost by routing cross-view information through a compact set of per-frame tokens while keeping patch attention intra-frame (*register attention*) report a significant accuracy drop. We show this gap is largely an initialization issue, not a capacity limitation of register attention. By pretraining the registers on a multi-view covisibility pretext task extended from a two-view, static-scene setting to long, dynamic sequences with a new view-conditioned, query-based segmentation head before finetuning on 3D reconstruction, our model recovers most of the global attention performance. On long sequences, where global attention's quadratic cost becomes prohibitive, our approach scales efficiently and enables training on long sequences. This curriculum makes register attention a scalable alternative that outperforms chunked global attention on hundreds of views.

1 INTRODUCTION

Reconstructing the 3D geometry of a scene from unposed, uncalibrated multi-view images or video streams is a core problem in computer vision. Recently, feed-forward reconstruction models (FRMs) such as DUSt3R (Wang et al., 2024), VGGT (Wang et al., 2025a), π^3 (Wang et al., 2026d), and CUT3R (Wang et al., 2025b) have reshaped this landscape by directly regressing camera poses and dense pointmaps in a single forward pass, offering fast inference and strong generalization across diverse scenes. Despite their success, scaling multi-view FRMs like VGGT and π^3 to long video sequences remains bottlenecked by their reliance on global cross-view attention, where every spatial patch token attends to every other patch across all frames. Because this all-to-all communication scales quadratically with both sequence length and patch resolution, processing long videos quickly becomes prohibitive in inference time and memory. Notably, VGGT-Ω (Wang et al., 2026c) observed that the attention matrices in cross-view layers are sparse, suggesting that the cross-view context could be summarized with special tokens, therefore decoupling cross-view attention from spatial resolution. However, VGGT-Ω also found that replacing all global cross-view layers with register Darcet et al. (2024) attention when training causes accuracy

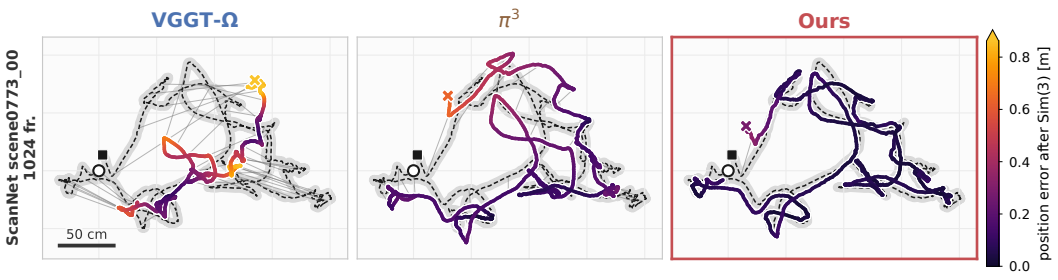


Figure 1: **Long-Sequence Camera Trajectories.** 1024-frame ScanNet sequence, colored by position error after Sim(3) alignment. Global-attention models (VGGT-Ω, π^3) are too costly to train on long sequences and are chunked at inference, which causes drift. With covisibility pretraining, our register-only model stays accurate and scales to long sequences.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

degradation (forcing them to retain 75% global layers), which leaves open the question of whether using less tokens (registers) inherently lacks the capacity for fine-grained multi-view geometry.

We show that this degradation stems largely from an optimization failure caused by random initialization rather than from a fundamental capacity limit of using fewer register tokens than patch tokens: when trained solely on dense pointmap regression, register tokens fail to organize into a structured representation of the scene. Our core insight is that registers can compress cross-view geometry if they are first pretrained on a task that explicitly forces cross-frame spatial reasoning.

Recently, Alligat0r (Loiseau et al., 2025) demonstrated that predicting pixel-level covisibility maps provide an effective geometric pretraining, before finetuning a model on camera pose estimation. Specifically, they classify source pixels as covisible, occluded, or out-of-field-of-view (out-of-FOV) with respect to a target view. Yet, Alligat0r is restricted to image pairs with global (dense) patch cross-attention and therefore does not handle multiple views. Moreover, they use static depth reprojection, which incorrectly labels moving objects. In this paper, we similarly perform a two-stage approach with a geometric pretraining using covisibility. However, we differ in our scalable multi-view covisibility objective and our explicit treatment of dynamic objects. We achieve scalability by using a *register*-based multi-view transformers architecture and designing a more efficient head to output pixel-level covisibility. To scale covisibility beyond image pairs at linear $\mathcal{O}(S)$ cost, we introduce a pair sampling mechanism with a new view-conditioned, query-based segmentation head inspired by Mask2Former (Cheng et al., 2022). We seed a small set of class queries solely from the target view's register tokens and cross-attend them to the source view's features to predict dense covisibility masks, forcing the registers to summarize the geometric context. Furthermore, we address the labeling errors introduced by static reprojection on dynamic videos by masking epipolar outliers, preventing moving objects from corrupting the static geometric prior, and enabling accurate reconstruction of both static background and dynamic objects. In contrast to Alligat0r (Loiseau et al., 2025) that only finetunes for camera pose estimation, we finetune our register-based model for a dense 3D reconstruction task. Such a model has reduced complexity and can handle significantly longer sequences (Fig. 1).

Our main contributions are summarized as follows:

- **Scalable S-view and dynamic covisibility:** We generalize pairwise static covisibility (Loiseau et al., 2025) to S-view sequences at linear cost via structured pair sampling and a register-seeded, query-based segmentation head, while handling moving objects cleanly to avoid reprojection noise.
- **Effective geometric pretraining for register attention:** We demonstrate that covisibility pretraining unlocks the representational capacity of register bottlenecks, drastically reducing the accuracy gap with global attention on standard reconstruction benchmarks.
- **Scaling to long video sequences:** We analyze the efficiency and accuracy tradeoffs across attention regimes as sequence length grows, showing that our pretrained register architecture enables training on longer sequences, mitigates pose drift, and outperforms global attention on long videos at a fraction of the latency and memory cost.

2 Related Work

Feed-Forward 3D Reconstruction. Pioneered by DUSt3R (Wang et al., 2024) and MASt3R (Leroy et al., 2024), feed-forward reconstruction regresses dense pointmaps and camera poses directly from unposed images, rapidly extending to multi-view and dynamic models (Wang et al., 2025a; 2026c; Zhang et al., 2026a; Wang et al., 2026d; Li et al., 2025b; Hu et al., 2025; Wang et al., 2026b; Yang et al., 2025; Chen et al., 2025b), and diverse downstream tasks (Kwak et al., 2026; Guédon et al., 2026; Deng et al., 2026; Li et al., 2025a; Barberteguy et al., 2026; Koch et al., 2025; Zust et al., 2025). Because standard multi-view backbones use all-to-all cross-attention across all N frames and L patches at $\mathcal{O}(S^2L^2)$ cost, scaling to long videos quickly saturates memory and inference time. To mitigate this, training-free methods prune, merge, or cache tokens (Liu et al., 2026b; Wang & Agapito, 2025; Zheng et al., 2026; Chen et al., 2026d; Dong et al., 2026; Wang et al., 2026a; Park et al., 2026; Mahdi et al., 2025; Mao et al., 2026), while streaming models maintain recurrent memory banks or sliding states (Cabon et al., 2025; Wang et al., 2025b; Liu et al., 2025; Lan et al., 2026; Cheng et al., 2026; Yan et al., 2026; Antsfeld et al., 2026; Murai et al., 2025; Duisterhof et al., 2025). Recent approaches also linearize global attention via state-space models (Gu & Dao, 2024) or via test-time training (Sun et al., 2025; Zhang et al., 2026c; Chen et al., 2026c; Zhang et al., 2026b; Liu et al., 2026a; Yuan et al., 2025; 2026; Elflein et al., 2026), though sequential state updates remain prone to latency and long-horizon drift. Our model can take a large number of views at once which enables global adjustement and more consistency in scene scale and camera pose estimation.

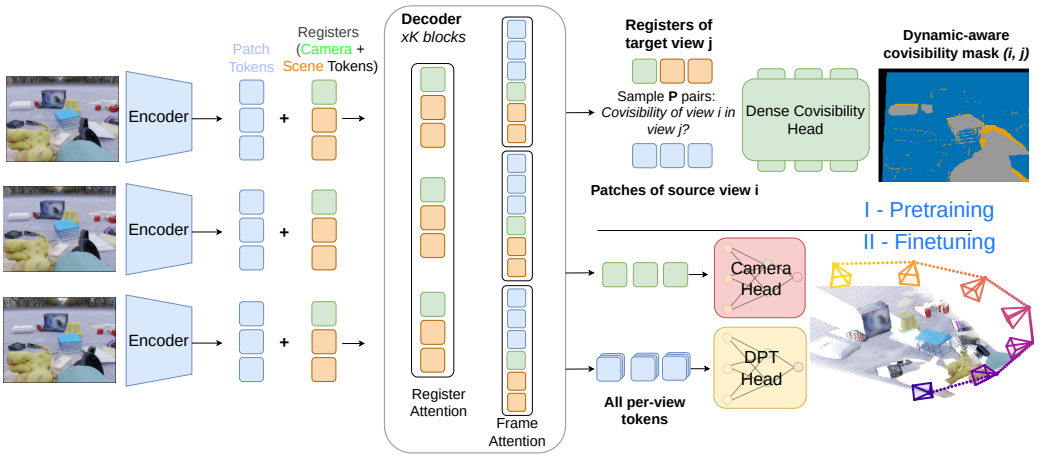


Figure 2: **Our Two-Stage Training.** Each input frame is independently encoded into spatial patch tokens (blue) and augmented with learnable camera (*CAM*) and scene (*SCE*) *register* tokens. The K-block decoder alternates between two modes: **inter-frame register** attention mode (left column of the decoder, where only *CAM* and *SCE* tokens interact across views) and **intra-frame global** attention mode (right column, where each frame's registers exchange information with its own patch tokens). **Stage I - Pretraining:** For randomly sampled frame pairs (i,j), the *Dense Covisibility Head* uses the registers of target view j to query the patch tokens of source view i, predicting a per-pixel covisibility mask (covisible in blue, occluded in orange, out-of-FOV in black, and masked-out dynamic/invalid regions in grey). **Stage II - Finetuning:** The covisibility head is replaced by a *Camera Head* and a dense prediction head (*DPT*) to directly regress multi-view camera poses and 3D pointmaps.

Off-the-Grid Tokens and Register Routing. Defining additional global tokens such as `cls` in BERT (Devlin et al., 2019) or ViT (Dosovitskiy et al., 2021) besides the 2D patch grid tokens has long been used to compress high-dimensional inputs (Jaegle et al., 2021; 2022; Sajjadi et al., 2022; van Steenkiste et al., 2024), structure set prediction (Carion et al., 2020; Cheng et al., 2022), and tokenize scenes for generation and control (Bachmann et al., 2025; Atanov et al., 2026; Asim et al., 2026; Kim et al., 2026; Lee et al., 2026). In Vision Transformers (Dosovitskiy et al., 2021), Darcet et al. (2024) introduced register tokens to absorb attention artifacts, but these global tokens exhibit semantically meaningful attention maps. Thus, they were recently repurposed as image summaries to compress cross-view communication in 3D reconstruction (Wang & Xu, 2026; Chen et al., 2026b; Wang et al., 2026c). However, LingBot-Map (Chen et al., 2026b) restricts register routing to distant frames, and VGGT-Ω (Wang et al., 2026c) replaces only 25% of global cross-view layers, reporting an accuracy drop under full replacement. We show that this degradation stems mostly from random initialization rather than capacity limits: with covisibility pretraining, our model equipped 100% register-routed cross-view attention is competitive, and enables training on long sequences.

Multi-View and Geometric Pretraining. While spatiotemporal extensions of masked modeling (He et al., 2022; Wei et al., 2022; Tong et al., 2022; Zoran et al., 2025; van Steenkiste et al., 2024; Gupta et al., 2023), predictive architectures (Assran et al., 2023; Mur-Labadia et al., 2026), and self-distillation (Oquab et al., 2023) learn general visual features, geometry-guided pretraining provides stronger 3D priors. CroCo (Weinzaepfel et al., 2023) and MuM (Nordström et al., 2026) use cross-view completion, Dinocular (Almukhamedov et al., 2026) supervises dense correspondences, and Alligat0r (Loiseau et al., 2025) formulates pairwise covisibility segmentation (previously used as a downstream task (Ziliotto et al., 2026; Chen et al., 2025a; Zhang et al., 2025)) into covisible, occluded, and out-of-FOV pixels. However, Alligat0r, CroCo, and Dinocular rely on quadratic pairwise cross-attention and static-scene assumptions. We extend covisibility pretraining to S-view sequences at linear cost via a register-seeded segmentation head and pair sampling while masking out dynamic objects to avoid corrupted static reprojections.

3 Method

Given a sequence of S video frames, $\mathbf{I} \in \mathbb{R}^{S \times H \times W \times 3}$, where each frame is of resolution $H \times W$, our goal is to perform reconstruction in 3D. Our model (illustrated in Fig. 2) consists of a per-frame monocular encoder and a cross-view register decoder (Sec. 3.1), which we first pretrain on scalable multi-view covisibility segmentation (**Fig. 2-I**, Sec. 3.2) and then finetune for feed-forward 3D reconstruction (**Fig. 2-II**, Sec. 3.3).

3.1 Encoder-Decoder Architecture

In the following, we first briefly describe the monocular image encoder operating per-view, and the register decoder that enables the cross-view interaction. We keep these architectures as close as possible to existing literature (Wang et al., 2026c) to be able to isolate the effect of our covisibility pretraining and the long-sequence scaling.

Monocular Encoder. Each frame is encoded independently by a DINOv3-initialized ViT (Dosovitskiy et al., 2021; Siméoni et al., 2025). We discard the encoder's special tokens (registers, `cls`) tokens and keep only the R patch embeddings $\mathbf{X} \in \mathbb{R}^{S \times P \times D}$. We use ViT-S/16 ($D$=384) for ablations and ViT-L/16 (D=1024) for our main model, with the encoder finetuned unless stated otherwise.

Register Decoder. The cross-view decoder consists of K alternating blocks following a VGGT-Ω (Wang et al., 2026c) layout: each block pairs an *intra-frame* attention layer (attention between patches with 2D RoPE (Su et al., 2021) and registers) with an *inter-frame* attention layer. At the decoder input, we prepend $T{+}1$ learnable registers per frame: one camera token (CAM) and T scene tokens (SCE), yielding $L{=}1{+}T{+}R$ tokens per frame. Inter-frame layers operate in one of two modes:

- **Global attention:** All $S \cdot L$ tokens attend across views at $\mathcal{O}(S^2L^2)$ cost (Wang et al., 2025a; 2026d; Zhang et al., 2026a).
- **Register attention:** Only the $S \times (T{+}1)$ registers (CAM and SCE) attend across views while patch tokens remain strictly intra-frame, reducing cross-view complexity to $\mathcal{O}(S^2T^2)$ with $T \ll R$.

We use 100% register attention (0% global layers) in our main model and evaluate against a backbone with 100% global layers in Sec. 4.1. Unlike standard alternating decoders (Wang et al., 2025a; 2026d;c) that concatenate intra- and inter-frame outputs into $2D$-dimensional tokens, we disable this concatenation under pure register attention since patch tokens are not updated by inter-frame blocks. In practice, we use 16 scene tokens and 1 camera token (17 registers in total) following VGGT-Ω (Wang et al., 2026c).

Rather than optimizing covisibility and 3D reconstruction jointly in a single stage, we decouple them into a two-stage training. The main reason is that joint training is prohibitively slow due to holding two DPT heads in memory simultaneously (i.e., one for covisibility, one for depth). Dropping the covisibility head during Stage II therefore frees device memory and enables finetuning on *longer* sequences, while outperforming single-stage training under the same total data, see Fig. 3(a). Furthermore, preliminary small-scale experiments with joint training did not yield any gain over two-stage training.

3.2 Stage I: Scalable Covisibility Pretraining

As mentioned in Wang et al. (2026c), training 100% register attention from scratch degrades geometric accuracy. To force registers to encode multi-view spatial relationships, we pretrain them on an S-view covisibility segmentation task (**Fig. 2-I**) inspired from Alligat0r (Loiseau et al., 2025), that we extend to mutli-view dynamic sequences by sampling interesting pairs and masking epipolar outliers.

Pair Sampling and Covisibility Head. Evaluating patch-to-patch covisibility across all $S(S{-}1)$ frame pairs is prohibitive for long sequences. Instead, we sample a set of ordered pairs $\mathcal{P}$ with $|\mathcal{P}| = \mathcal{O}(S)$ per training step. We consider three pair sampling strategies (ablated in Sec. 4.1): star sampling, where we supervise all the views $j > 0$ relative to the first view ($j \to 0$), uniform random sampling over all off-diagonal pairs (i,j), and per-view random sampling. In the latter, we ensure that each frame in the sequence is selected both as a source *and* target, so both its patches and registers get supervision signal.

Intermediate patch tokens from the decoder are reassembled once per view via a DPT pyramid (Ranftl et al., 2021) into dense feature maps $\mathbf{F} = \{\mathbf{f}_i\}_{i=1}^{S}$, with $\mathbf{f}_i \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times C}$. For each sampled pair (i,j) (source view i, target view j), we pool the target registers $\mathbf{R}_j = [\text{CAM}_j, \text{SCE}_j]$ and project them via an MLP into M initial class queries, which cross-attend to source patch tokens $\mathbf{X}_i$ to form view-conditioned queries $\mathbf{Q}_{i \leftarrow j} \in \mathbb{R}^{M \times C}$ (Cheng et al., 2022). Per-pixel covisibility logits in view i are then decoded via inner product:

$$\hat{\mathbf{Y}}_{i \leftarrow j} = \text{Upsample}_{4\times}\left(\mathbf{f}_i \cdot \mathbf{Q}_{i \leftarrow j}^{\top}\right) \in \mathbb{R}^{H \times W \times M}, \tag{1}$$

keeping per-pair decoding overhead at $\mathcal{O}(MC)$ rather than $\mathcal{O}(P^2)$. The head is illustrated in Fig. A1 and detailed in Sec. A.2.1.

Labels for Covisibility Supervision. Following Loiseau & Bourmaud (2025); Loiseau et al. (2025), for a sampled pair (i,j), we unproject valid pixels from source depth $\mathbf{D}_i$, rigidly transform them into camera

j with reprojected depth z_j, and compare against target depth $\mathbf{D}_j$ (full equations in Sec. A.1.1). Each static source pixel is assigned one of $M{=}3$ active target classes:

- **Out-of-FOV** ($m{=}0$)**:** The reprojected point falls behind camera j ($z_j \leq 0$) or outside the image bounds of view j.
- **Occluded** ($m{=}1$)**:** The point projects inside view j, but farther than target depth $(z_j - \mathbf{D}_j)/\mathbf{D}_j > \tau$ (with a $\tau{=}5\%$) or its transformed surface normal faces away from camera j by more than $85°$.
- **Covisible** ($m{=}2$)**:** The point projects inside view j, satisfies the normal check, and matches surface depth within tolerance $|z_j - \mathbf{D}_j|/\mathbf{D}_j \leq 5\%$.

Pixels with invalid depth or sky are ignored in the loss (grey in Fig. 2-I).

Handling Dynamic Scenes. On dynamic scenes, rigid reprojection violates epipolar geometry and misclassifies moving objects as occluded, which causes regression in downstream performance for dynamic datasets (Sec. 4). We explore two strategies to handle moving objects: (1) *Ignore Dynamic ($M{=}3$, default):* We mask out moving pixels to remove those regions from the loss. When instance motion masks are unavailable on real-world videos, we detect moving pixels geometrically with a depth-consistency test (details in Sec. A.1.2): a forward check ($i \to j$) flags pixels where $z_j < \mathbf{D}_j - \tau$ (detecting that an object moved away), while a backward check ($j \to i$) unprojects candidate occluders back into view i to disambiguate moving foreground occluders (that we ignore) from static walls. (2) *Explicit Dynamic Labels ($M{=}5$)*: We leverage the amount of synthetic datasets with known 3D motion. We propagate moving 3D points $\mathbf{X}_w(t_i)$ to target timestamp t_j before z-buffer testing with 6-DoF object poses $\mathbf{X}_w(t_j) = \mathbf{T}_{w\leftarrow o}(t_j)\mathbf{T}_{w\leftarrow o}(t_i)^{-1}\mathbf{X}_w(t_i)$ or dense 3D point trajectories, splitting moving pixels into dedicated *occluded-dynamic* ($m{=}3$) and *covisible-dynamic* ($m{=}4$) classes. Integrating any type of videos consequently enables scaling the training data (Zhang et al., 2026a).

We supervise all valid (i.e., non-ignored) pixels with cross-entropy ($\mathcal{L}_{\text{CE}}$) and Dice losses ($\mathcal{L}_{\text{Dice}}$) (Sudre et al., 2017):

$$\mathcal{L}_{\text{covis}} = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} \Big(\mathcal{L}_{\text{CE}}(\hat{\mathbf{Y}}_{i\leftarrow j}, \mathbf{Y}_{i\leftarrow j}) + \lambda_{\text{dice}} \mathcal{L}_{\text{Dice}}(\hat{\mathbf{Y}}_{i\leftarrow j}, \mathbf{Y}_{i\leftarrow j}) \Big) \tag{2}$$

setting equal loss weights $\lambda_{\text{dice}} = 1$ following Mask2Former (Cheng et al., 2022).

3.3 Stage II: Finetuning for 3D Reconstruction

Finetuning objective. After pretraining, the encoder and decoder are finetuned on dense feed-forward reconstruction. We append two heads as illustrated in Fig. 2-II: a camera head that supervises a 9-D vector encoding the translation, quaternion of the camera pose and field of view ($\mathcal{L}_{\text{cam}}$), and a DPT dense head (Ranftl et al., 2021) that predicts depth and confidence ($\mathcal{L}_{\text{depth}}$). One notable difference is that we do not use the matching loss to avoid heavy data preparation, which causes a minor drop in performance discussed in the ablations of the original paper (Wang et al., 2026c). The training loss is thus:

$$\mathcal{L} = \lambda_{\text{cam}}\mathcal{L}_{\text{cam}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}} + \lambda_{\text{point}}\mathcal{L}_{\text{point}}, \tag{3}$$

with the default weights from Wang et al. (2026c)$(\lambda_{\text{cam}}, \lambda_{\text{depth}}, \lambda_{\text{point}}) = (5.0, 1.0, 0.5)$. Following Wang et al. (2026c), pointmaps ($\mathcal{L}_{\text{point}}$) are supervised indirectly: the predicted depth is unprojected through the predicted camera and compared to the ground-truth point map in the coordinate system of the reference view (i.e., the first view), reusing the shared depth confidence.

Extending to a High Number of Views. Rather than finetuning on long sequences from the start, we finetune on $S{=}24$ views, then $S{=}48$, and finally $S{=}96$, each stage starting from the previous one. This keeps finetuning tractable, similar in spirit to how video models are extended to more frames (Bain et al., 2021). Details on the curriculum training are in Sec. A.3.5.

4 Experiments

In this section, we first conduct ablation studies on the model and the number of training views (Sec. 4.1). Next, we scale up the training to obtain our final model and compare our method against state-of-the-art (Sec. 4.2). Finally, we provide qualitative analysis of the features obtained by our pretraining(Sec. 4.3).

Table 1: **Progressive Pretraining Ablation** (ViT-S/16, S=12). We sequentially ablate the pretraining objective (on fixed $j \to 0$ pairs with static reprojection), dynamic label handling, and pair sampling strategies. We report the camera pose at different thresholds (5, 10, and 20). The depth metrics AbsRel and inliers $\delta < 1.25$ are quickly saturated by all variants, and are reported in Tab. A2 for completeness.

		ScanNet++			Kubric		
Axis	**Pretraining Setup**	AUC@5↑	@10↑	20↑	@5↑	10↑	20↑
Baseline	Scratch	12.9	29.8	52.0	56.3	73.2	84.3
Objective	Scalar Covis. $[S,S]$	12.9	30.1	52.4	44.5	63.7	77.8
	Dense Covis. (Static labels)	19.1	38.9	60.2	53.5	71.2	83.1
Labels	+ Dynamic Labels	19.1	38.9	60.2	51.7	69.2	81.4
	+ Ignore Dynamic	20.8	40.6	61.5	**60.7**	**76.1**	**86.0**
Sampling	+ Random Pairs	21.1	41.6	63.0	56.7	73.3	84.2
	+ Per-view Random Pairs	**21.9**	**42.9**	**63.8**	58.6	74.3	84.9

4.1 Ablations

Unless stated otherwise, we conduct all ablation studies using our light-weight configuration using a ViT-S/16 (Dosovitskiy et al., 2021) encoder and a 9-layer cross-view decoder. To incorporate both static and dynamic environments, we pretrain and finetune all variants on a combination of static ScanNet++ (Yeshwanth et al., 2023) and dynamic Kubric (Greff et al., 2022) sequences using S=12 input views, and report downstream camera pose accuracy (Area Under Curve AUC@5/10/20) and *video* depth metrics (AbsRel and inlier ratio $\delta < 1.25$) in Tab. A.2.2. We first isolate the design choices of the model (pretraining objective, head design, and label type, pair sampling), and then study how each attention mode scales to long videos. As in Loiseau et al. (2025), we use a total budget of 40 epochs, and use 25 epochs for pretraining and 15 epochs for finetuning. For non-pretrained baselines, we use the *full* budget for training on reconstruction, ensuring that the model has seen the same number of examples overall.

Pretraining Objective. We evaluate pretraining tasks against our primary baseline: a model trained from scratch as in VGGT-Ω (Wang et al., 2026c), where the encoder is initialized from DINOv3 (Siméoni et al., 2025) and the decoder is randomly initialized. We first test a global image-level baseline that regresses the pairwise $[S,S]$ scalar covisibility overlap matrix from concatenated register tokens $[\mathbf{R}_i,\mathbf{R}_j]$ via a joint objective combining binary overlap detection which classifies the images as covisible or not. For the positive pairs, we train on a classification objective by splitting the covisibility range $[0,1]$ into 50 bins of width 0.02 (*Scalar Covis.*). Then, we pretrain on the dense covisibility objective from Alligat0r (Loiseau et al., 2025) using static reprojection. In the first block of Tab. 1, we observe that scalar overlap provides a useful coarse scene prior, but supervising pixel-level covisibility masks (*Dense Covis.*) yields a substantial jump on both datasets. Finally, even though Alligat0r (using static reprojection) reports positive results on nuScenes (Caesar et al., 2020) which contains dynamic objects (pedestrians, cars), we show that it degrades performance on highly dynamic datasets like Kubric (-3pts in AUC@5) compared to training from scratch. We attribute this to the fact that most of scenes in nuScenes are static (parked cars, buildings) and that dynamic objects are small (e.g., pedestrians). In addition, we compare pretraining with our query-based segmentation head and the learned upsampling introduced in Alligat0r in Tab. A1, showing their pretraining yields a lower downstream performance on reconstruction than our covisibility head when supervised on the same pairs.

Dynamic Scene Labels. The second block of Tab. 1 tackles the dynamic-scene issue identified above that injects contradictory geometric signal into the model. Taking *Dense Covis. (Static)* as the baseline of this block, we compare two ways of removing this corrupted signal. The first builds on the insight that synthetic data with ground-truth motion is available at large scale (Greff et al., 2022; Zheng et al., 2023; Butler et al., 2012), making it possible to supervise explicitly dynamic covisibility: we train on M=5 motion-aware labels derived from ground-truth 6-DoF object poses on Kubric (Greff et al., 2022) (+*Dynamic Labels*). The second removes moving pixels from the loss (+ *Ignore Dynamic*), using instance motion masks when available and our depth-consistency filter otherwise. Sec. A.1.2 gives the exact definition of both label sets. Explicit dynamic labels leave static reconstruction unchanged (38.9 AUC@10 on ScanNet++) and *degrade* dynamic scenes by 2.0 points (71.2 → 69.2 on Kubric): tasking the registers with re-identifying moving instances across frames injects correspondences that are inconsistent with the rigid camera motion the pose head must recover. Ignoring dynamic pixels instead improves both, by 1.7 points on ScanNet++ (38.9 → 40.6) and 4.9 points on Kubric (71.2 → 76.1), and is the first pretraining

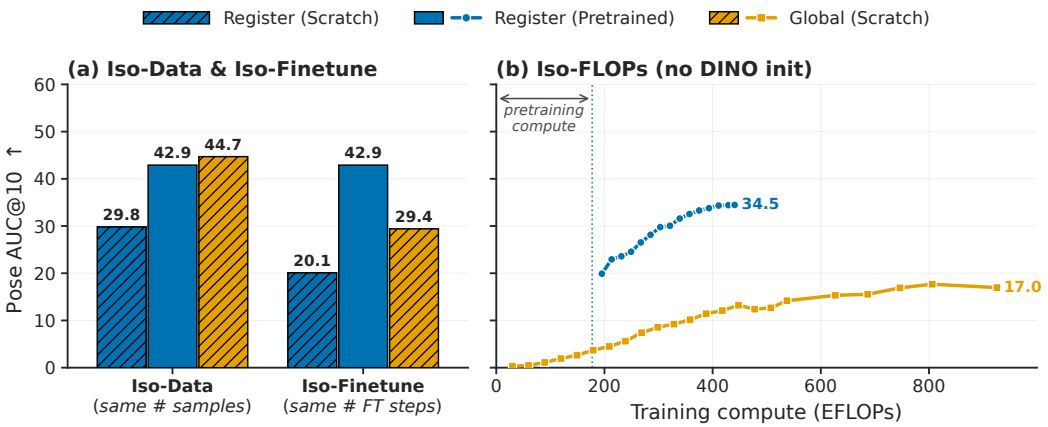


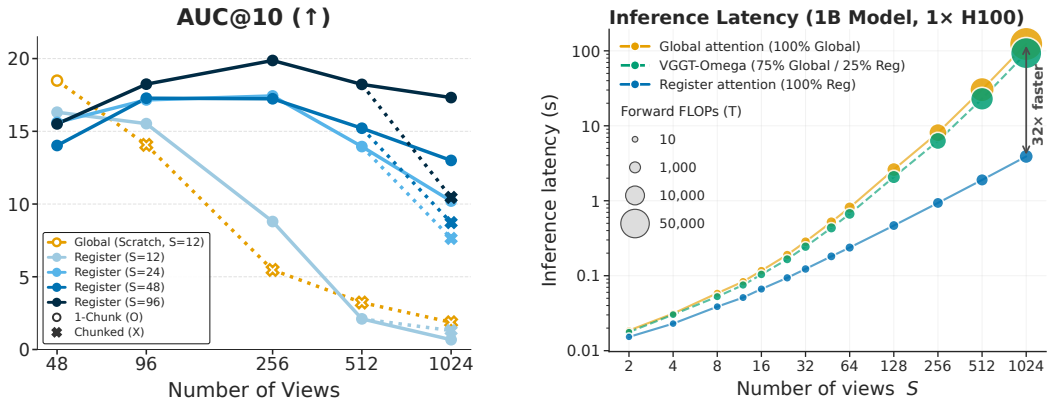
Figure 3: **Effectiveness of Covisibility Pretraining.** Pose AUC@10 on ScanNet++. (a) At matched total training samples (Iso-Data, 25+15 vs. 40 epochs) and matched finetuning steps (Iso-Finetune, 15 epochs). (b) Without DINOv3 initialization, as a function of training compute (Iso-FLOPs).

variant to surpass the from-scratch baseline on dynamic scenes (73.2). As it additionally requires no 3D motion annotation and therefore applies to real-world video, we keep it for the rest of the experiments.

Pair Sampling. In the final block of Tab. 1, we leverage the flexibility of our view-conditioned segmentation head to move beyond fixed reference-view supervision ($j \to 0$) and to sample more than $S-1$ views. Sampling $2S$ arbitrary ordered pairs (i,j) uniformly at random across the sequence (+ *Random Pairs*) distributes geometric supervision evenly along the camera trajectory and improves static reconstruction. Finally, by sampling random pairs and ensuring each frame is used both as source and target (we also supervise $2S$ pairs), the model gets a clearer signal and outperforms any other mode on static scenes (+13.1 AUC@10 over the model trained from scratch on 40 epochs). Although using fixed pairs is competitive on dynamics, as the sequence length grows, restricting covisibility to frame 0 will stop bringing useful information since most of the frames will be disjoint, hence we retain *per-view random sampling* for the main experiments.

Effectiveness of Covisibility Pretraining. We first isolate how covisibility pretraining bridges the gap between pure register attention (0% global attention layers) and global all-to-all attention (100% global) across three controlled regimes in Fig. 3 evaluated on ScanNet++. In Fig. 3(a), under the *Iso-Data* regime (where both models see the exact same total number of training samples across 40 epochs, splitting our budget into 25 pretraining and 15 finetuning epochs), pretraining boosts register pose accuracy from 29.8 to 42.9 AUC@10, closing nearly the entire gap to *Global (Scratch)* (44.7). Further experiments in Tab. A3 point to the fact that a global attention backbone pretrained with the same recipe further improves AUC@10 accuracy from 44.7 to 52.1. Under the *Iso-Finetune* regime (15 downstream finetuning epochs for all models), *Register (Pretrained)* (42.9) outperforms both *Register (Scratch)* (20.1) and *Global (Scratch)* (29.4) by a large margin. This confirms that the accuracy drop of register attention reported by Wang et al. (2026c) is mainly an initialization issue rather than a capacity bottleneck: in 3D reconstruction, depth supervision is largely monocular while camera pose provides only one sparse vector per frame, making cross-view routing hard to learn. To verify that our covisibility pretraining objective induces these geometric representations independently of pretrained 2D priors, and to compute FLOPs accurately, Fig. 3(b) evaluates training from random initialization without DINOv3 (Siméoni et al., 2025) as a function of cumulative training compute (*Iso-FLOPs*). Without DINOv3, *Global (Scratch)* drops sharply from 44.7 to 17.0 and. In contrast, our *Register (Pretrained)* remains robust without DINOv3, reaching 34.5 AUC@10 at just 440 EFLOPs, over 2× the accuracy of global attention at nearly half the FLOPs (925 for the global counterpart).

Scaling to Long Sequences. In Fig. 4a, we evaluate how our pretrained models scale from 48 to 1024 frames at inference time on a single A100 (40GB) GPU compared to a global attention model trained from scratch on 12 views. We evaluate overlapping sliding windows (chunks, with a 16-frame overlap) aligned via the Umeyama (1991) similarity transform (Zhang et al., 2026a). We evaluate the global attention model with a chunk (i.e., window) size 64 following Zhang et al. (2026b). This chunking degrades camera pose (AUC@10 drops from 18.5 at 48 views to 1.9 at 1024 views). With the same training, *Register (S=12)*, although processing frames in one chunk, suffers from a train-test shift and closely follows the trend of the global attention model. However, the small memory footprint of register routing allows us to train on longer sequences (in a curriculum, first *Register (S=24)*, then going to *Register (S=48)* and *Register (S=96)*).



(a) Camera pose AUC@10 vs. number of test views. (b) Inference runtime and FLOPs per number of views.

Figure 4: **Sequence Length Scaling.** (a) Register-attention models trained on $S \in \{12,24,48,96\}$ views vs. global attention (S=12), on 48 to 1024 frames of ScanNet++ (one A100). Solid (●): single forward pass; dotted (×): chunked (64-frame windows for global, 512 for register). (b) Latency and forward FLOPs of the 1B model on one H100 (bfloat16, FlashAttention (Dao, 2023)).

These long-context variants maintain higher accuracy up to 1024 views in a single forward pass, outperforming global attention for all settings where $S \geq 96$. Fig. 4b confirms this advantage: register attention scales near-linearly and, at 1024 views, is 32× faster and needs over 10× fewer FLOPs than global attention.

4.2 Comparison to the State of the Art

In this section, we scale up (a) the architecture size, (b) the training data, and (c) the number of views to obtain our final best-performing model. We first describe our training setup, then introduce the long-sequence benchmarks that we evaluate on. We compare this final model against the state of the art.

Training. Our final model pairs a ViT-L/16 encoder (303M parameters) with a 24-layer register decoder (605M), totaling 1B parameters with heads. We pretrain on 2 to 24 views for 250k steps (batch size 256, 128 H100 GPUs), then finetune with the S=24→48→96 curriculum (batch size 128, 128 H100 GPUs) of Sec. 3.3. Datasets, schedules, and augmentations are in Sec. A.3.

Target Benchmarks. We evaluate on datasets that have long sequences: ScanNet (Dai et al., 2017) which is an indoor setting with static scenes and KITTI (Geiger et al., 2013) that has dynamic objects and kilometer-scale scenes (but no ground truth depth). We study the performance of the models on sequences ranging from 48 to 1024 views taken contiguously. We compare against other non-causal feedforward models and set aside the variants with specially designed mechanisms like a causal memory (Cabon et al., 2025; Antsfeld et al., 2026) and test time training (TTT) (Yuan et al., 2025; Chen et al., 2026c) since they are complementary, and often build on, non-causal feedforward models. More specifically, we compare to VGGT-Ω (Wang et al., 2026c) and π^3 (Wang et al., 2026d) on a single A100 GPU. These models use global attention and we evaluate them with the chunking method used in Zhang et al. (2026b) (chunk size of 64) (same Umeyama (1991) alignment as in 4.1). We denote these baselines VGGT-Ω-chunk and π^3-chunk.

Results on Long Sequences. Tab. 2 summarizes the results. For relatively short sequences ($S \leq 96$), global methods (especially π^3) are still in-domain and perform better. However, as the sequence grows, our model clearly overtakes π^3 (Wang et al., 2026d) and VGGT-Ω (Wang et al., 2026c). On ScanNet, we improve AUC@30 by +10.3, +20.5 and +14.2 points over π^3 for S=256, 512 and 1024 (53.3 vs. 39.1 and 31.9 at S=1024). It is worth noting that all the methods evaluated include ScanNet in their training data. On KITTI, the gap to π^3 closes with sequence length, and at S=1024 we surpass it. Our depth also degrades the least with sequence length: we outperform VGGT-Ω for $S \geq 96$ (3.38 vs. 4.45 AbsRel at S=1024) although π^3 remains consistently more precise. As seen in Fig. 4, chunking quickly drops the performance of models trained on shorter sequences, since errors from stitching each window accumulate into drift. Our efficient model instead processes up to 512 views in a single forward passon on one A100, and is 32× faster than single-pass global attention at S=1024 (Fig. 4b) on one H100. Qualitative comparisons can be found in Fig. 1 (and in Fig. A4), where we observe less camera trajectory drift with our long context window than with chunking.

Table 2: **Comparison to the State of the Art on Long Sequences.** Each entry reports camera pose accuracy (AUC@30↑) and, on ScanNet, video-scale depth error (100×AbsRel↓, in %) for $S \in \{48,96,256,512,1024\}$ input views. Baselines run in sliding 64-frame windows stitched with Sim(3) (*-chunk*); ours processes up to 512 views (KITTI: 1024) in a single forward pass (ScanNet S=1024: 3×512). Best in **bold**, second underlined.

Model	S=48	S=96	S=256	S=512	S=1024
ScanNet: AUC@30↑ / AbsRel↓					
π^3-chunk (Wang et al., 2026d)	**67.8 / 2.00**	**73.6 / 2.03**	63.5 / **2.30**	53.3 / **2.50**	39.1 / **3.03**
VGGT-Ω-chunk (Wang et al., 2026c)	66.7 / 2.63	72.2 / 2.95	56.5 / 3.62	44.3 / 4.12	31.9 / 4.45
Ours	65.9 / 2.64	71.6 / 2.68	**73.8** / 2.99	**73.8** / 3.26	**53.3** / 3.38
KITTI: AUC@30↑					
π^3-chunk (Wang et al., 2026d)	**87.1**	**80.8**	**51.9**	**33.2**	19.1
VGGT-Ω-chunk (Wang et al., 2026c)	81.2	73.2	36.4	22.4	11.0
Ours	71.2	68.2	47.5	30.4	**19.8**

4.3 Qualitative Analysis of Learned Features

Fig. 5 shows the principal components of dense features from monocular DINOv3 (Siméoni et al., 2025) and from our model after covisibility pretraining. DINOv3 groups regions semantically (e.g., the whole upper cabinet becomes one part), whereas our features delineate 3D planar orientations (wall, ceiling, cabinet faces) and edges. On Kubric, they separate individual static and moving objects (highlighted in the red box), although moving pixels are ignored during pretraining (see also Fig. A2). Covisibility pretraining thus provides a useful geometric prior for reconstruction finetuning.

5 Conclusion

We showed that the accuracy drop of register attention, which routes all cross-view communication through a few per-frame tokens, stems largely from initialization rather than capacity. Pretraining the registers on multi-view covisibility segmentation of dynamic sequences, with a view-conditioned, query-based head and dynamic-pixel masking, recovers most of the accuracy of global attention. Since the cross-view cost no longer depends on the number of patches, our model is trained and evaluated on hundreds of views in a single forward pass, avoiding the drift of sliding-window chunking. A gap to global attention remains on short sequences, especially outdoors, and using explicit dynamic labels for 4D tasks such as tracking is left to future work.

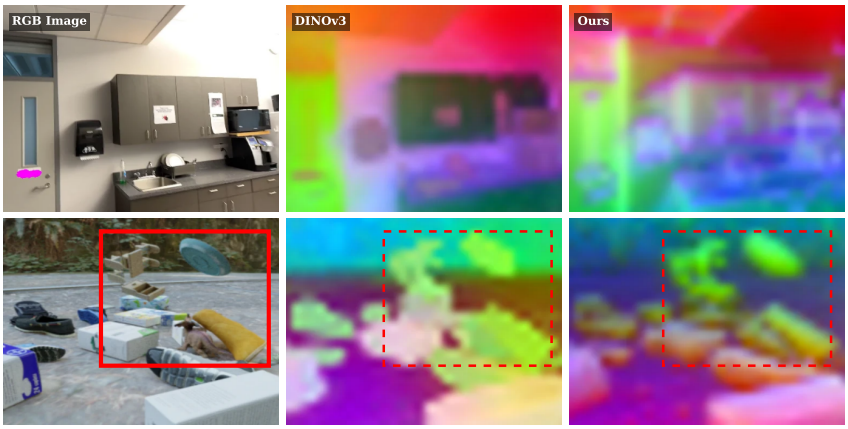


Figure 5: **PCA Visualization of Patch Features.** ScanNet++ (top) and Kubric (bottom). Compared to monocular DINOv3 (middle), our covisibility-pretrained features (right) capture object boundaries (edges of the cabinet, top row) and separate moving objects from static ones (in the red box, bottom row).

AI USE STATEMENT

In this work, we used generative AI tools to assist with software engineering (including code generation, debugging, and data analysis) as well as to assist with language editing and polishing of the manuscript. We did not use generative AI tools to formulate the core research ideas, design the model architecture or the pretraining methodology. All AI-assisted code and manuscript edits were thoroughly reviewed, tested, and verified for technical accuracy. We take full responsibility for the final content of this work, including all text, claims, and artifacts.

ETHICS STATEMENT

This work focuses on foundational 3D and 4D geometric reconstruction from multi-view images and video sequences using standard, publicly available synthetic and real-world computer vision benchmarks alongside internal static/dynamic scene datasets. Our study does not involve human subjects, personally identifiable information, biometric recognition, or sensitive demographic attributes. While efficient long-sequence 3D reconstruction can be deployed in downstream applications such as autonomous navigation, robotics, and augmented reality, we do not foresee direct ethical concerns specific to the register pretraining methodology introduced in this paper, and we have no conflicts of interest or ethics issues to declare.

REPRODUCIBILITY STATEMENT

We have made efforts to ensure the reproducibility of our architecture, pipeline, and experimental evaluations. Complete architectural specifications are provided in Sec. 3.1 and Sec. 3.2, with a detailed head diagram and memory/FLOP complexity derivations in App. A.2 (Fig. A1, Eqs. 12 13). The exact mathematical formulation of our covisibility labels generation, including the filter for unannotated videos, is detailed in App. A.1.2 (Eqs. 4- 11). Optimization schedules, loss weights, and compute budgets for both the ViT-S/16 ablation configuration and the 1B-parameter ViT-L/16 model are described in Sec. A.3.

REFERENCES

Farkhat Almukhamedov, Sami Azirar, and Hermann Blum. Dinocular: Self-supervised visuospatial representations. In *ECCV GeoInt Workshop*, 2026.

Leonid Antsfeld, Boris Chidlovskii, Yohann Cabon, Vincent Leroy, and Jerome Revaud. S-must3r: Sliding multi-view 3d reconstruction, 2026.

Mohammad Asim, Christopher Wewer, and Jan Eric Lenssen. Scenetok: A compressed, diffusable token space for 3d scenes. *CVPR*, 2026.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. *CVPR*, 2023.

Andrei Atanov, Jesse Allardice, Roman Bachmann, Oğuzhan Fatih Kar, R Devon Hjelm, David Griffiths, Peter Fu, Afshin Dehghan, and Amir Zamir. Videoflextok: Flexible-length coarse-to-fine video tokenization. *ICML*, 2026.

Roman Bachmann, Jesse Allardice, David Mizrahi, Enrico Fini, Oğuzhan Fatih Kar, Elmira Amirloo, Alaaeldin El-Nouby, Amir Zamir, and Afshin Dehghan. Flextok: Resampling images into 1d token sequences of flexible length. *ICML*, 2025.

Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *ICCV*, 2021.

Victor Barberteguy, Ahmet Iscen, Mathilde Caron, Alireza Fathi, Gül Varol, and Cordelia Schmid. Pano3D: Unified 3D reconstruction and panoptic segmentation. *ECCV*, 2026.

Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012.

Yohann Cabon, Lucas Stoffl, Leonid Antsfeld, Gabriela Csurka, Boris Chidlovskii, Jerome Revaud, and Vincent Leroy. MUSt3R: Multi-view network for stereo 3D reconstruction. In *CVPR*, 2025.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. *CVPR*, 2020.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

Caroline Chen, Sayna Ebrahimi, Fedor Kitashov, Ming-Hsuan Yang, Leonidas Guibas, Viorica Pătrăucean, and Maks Ovsjanikov. Seese3: Emergence of 3d space in vision features, 2026a.

Chao Chen, Nobel Dang, Juexiao Zhang, Wenkai Sun, Pengfei Zheng, Xuhang He, Yimeng Ye, Jiasheng Zhang, Taarun Srinivas, and Chen Feng. Co-vision: Co-visibility reasoning on sparse image sets of indoor scenes, 2025a.

Lin-Zhuo Chen, Jian Gao, Yihang Chen, Ka Leong Cheng, Yipengjing Sun, Liangxiao Hu, Nan Xue, Xing Zhu, Yujun Shen, Yao Yao, and Yinghao Xu. Geometric context transformer for streaming 3d reconstruction. *arXiv*, 2026b.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Easi3r: Estimating disentangled motion from dust3r without training. *ICCV*, 2025b.

Xingyu Chen, Yue Chen, Yuliang Xiu, Andreas Geiger, and Anpei Chen. Ttt3r: 3d reconstruction as test-time training. *ICLR*, 2026c.

Yutian Chen, Yuheng Qiu, Ruogu Li, Ali Agha, Shayegan Omidshafiei, Jay Patrikar, and Sebastian Scherer. Co-me: Confidence-guided token merging for visual geometric transformers, 2026d.

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022.

Chong Cheng, Xianda Chen, Tao Xie, Wei Yin, Weiqiang Ren, Qian Zhang, Xiaoyang Guo, and Hao Wang. Longstream: Long-sequence streaming autoregressive visual geometry. *CVPR*, 2026.

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *ICLR*, 2024.

Tianchen Deng, Xun Chen, Ziming Li, Hongming Shen, Shuhao Zhai, Danwei Wang, Javier Civera, and Hesheng Wang. Unipr-3d: Towards universal visual place recognition with visual geometry grounded transformer. In *ECCV*, 2026.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics (ACL)*, 2019.

Jiacheng Dong, Huan Li, Sicheng Zhou, Wenhao Hu, Weili Xu, and Yan Wang. Memix: Writing less, remembering more for streaming 3d reconstruction, 2026.

Siyan Dong, Shuzhe Wang, Shaohui Liu, Lulu Cai, Qingnan Fan, Juho Kannala, and Yanchao Yang. Reloc3r: Large-scale training of relative camera pose regression for generalizable, fast, and accurate visual localization. *CVPR*, 2025.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Bardienus Pieter Duisterhof, Lojze Zust, Philippe Weinzaepfel, Vincent Leroy, Yohann Cabon, and Jerome Revaud. MASt3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. In *3DV*, 2025.

Sven Elflein, Ruilong Li, Sérgio Agostinho, Zan Gojcic, Laura Leal-Taixé, Qunjie Zhou, and Aljosa Osep. Vgg-t^3: Offline feed-forward 3d reconstruction at scale. *CVPR*, 2026.

A Gaidon, Q Wang, Y Cabon, and E Vig. Virtual worlds as proxy for multi-object tracking analysis. In *CVPR*, 2016.

Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *IJRR*, 2013.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *CVPR*, 2022.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *COLM*, 2024.

Agrim Gupta, Jiajun Wu, Jia Deng, and Li Fei-Fei. Siamese masked autoencoders. In *NeurIPS*, 2023.

Antoine Guédon, Shu Nakamura, Nicolas Dufour, Jiahui Lei, Ko Nishino, and Angjoo Kanazawa. Surflo: Consistent 3d surface flow model with global state, 2026.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Doll'ar, and Ross B. Girshick. Masked autoencoders are scalable vision learners. *CVPR*, 2022.

Yu Hu, Chong Cheng, Sicheng Yu, Xiaoyang Guo, and Hao Wang. Vggt4d: Mining motion cues in visual geometry transformers for 4d scene reconstruction. *ArXiv*, 2025.

Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *CVPR*, 2018.

Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and Joao Carreira. Perceiver: General perception with iterative attention. *ICML*, 2021.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier Hénaff, Matthew M. Botvinick, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver io: A general architecture for structured inputs & outputs. *ICLR*, 2022.

Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Dynamicstereo: Consistent dynamic depth from stereo videos. In *CVPR*, 2023.

Dongwon Kim, Gawon Seo, Jinsung Lee, Minsu Cho, and Suha Kwak. Planning in 8 tokens: A compact discrete tokenizer for latent world model. *CVPR*, 2026.

Sebastian Koch, Johanna Wald, Hidenobu Matsuki, Pedro Hermosilla, Timo Ropinski, and Federico Tombari. Unified semantic transformer for 3d scene understanding. *arXiv preprint arXiv:2512.14364*, 2025.

Min-Seop Kwak, Junho Kim, Sangdoo Yun, Dongyoon Han, Taekyoung Kim, Seungryong Kim, and Jin-Hwa Kim. Cross-modal attention instillation for aligned novel view image and geometry synthesis. *ICLR*, 2026.

Yushi Lan, Yihang Luo, Fangzhou Hong, Shangchen Zhou, Honghua Chen, Zhaoyang Lyu, Shuai Yang, Bo Dai, Chen Change Loy, and Xingang Pan. STream3R: Scalable sequential 3D reconstruction with causal transformer. In *ICLR*, 2026.

Seokmin Lee, Yunghee Lee, Byeonghyun Pak, and Byeongju Woo. Pixel-level scene understanding in one token: Visual states need what-is-where composition. *CVPR Workshop*, 2026.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3D with MASt3R. In *ECCV*, 2024.

Hao Li, Zhengyu Zou, Fangfu Liu, Xuanyang Zhang, Fangzhou Hong, Yukang Cao, Yushi Lan, Manyuan Zhang, Gang Yu, Dingwen Zhang, and Ziwei Liu. IGGT: Instance-grounded geometry transformer for semantic 3D reconstruction. In *ICLR*, 2025a.

Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast, and robust structure and motion from casual dynamic videos. *CVPR*, 2025b.

Changkun Liu, Jiezhi Yang, Zeman Li, Yuan Deng, Jiancong Guo, and Luca Ballan. Mem3r: Streaming 3d reconstruction with hybrid memory via test-time training, 2026a.

Xuanyi Liu, Chunan Yu, Deyi Ji, Qi Zhu, Lingyun Sun, Xuanfu Li, Jin Ma, Tianrun Chen, and Lanyun Zhu. Streamcachevggt: Streaming visual geometry transformers with robust scoring and hybrid cache compression, 2026b.

Yuzheng Liu, Siyan Dong, Shuzhe Wang, Yanchao Yang, Qingnan Fan, and Baoquan Chen. Slam3r: Real-time dense scene reconstruction from monocular rgb videos. *CVPR*, 2025.

Thibaut Loiseau and Guillaume Bourmaud. Rubik: A structured benchmark for image matching across geometric challenges. In *CVPR*, 2025.

Thibaut Loiseau, Guillaume Bourmaud, and Vincent Lepetit. Alligat0r: Pre-training through covisibility segmentation for relative camera pose regression. *NeurIPS*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *ICLR*, 2019.

Soroush Mahdi, Fardin Ayar, Ehsan Javanmardi, Manabu Tsukada, and Mahdi Javanmardi. Evict3r: Training-free token eviction for memory-bounded streaming visual geometry transformers, 2025.

Hongbo Mao, Junjun Jiang, Youyu Chen, Jiaxin Zhang, Zhemeng Dong, and Xianming Liu. Regvggt: Sustainable visual geometry grounding for streaming via regulated memory. *ECCV*, 2026.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mahmoud Assran, Koustuv Sinha, Michael Rabbat, Yann LeCun, Nicolas Ballas, and Adrien Bardes. V-jepa 2.1: Unlocking dense features in video self-supervised learning. *arXiv*, 2026.

Riku Murai, Eric Dexheimer, and Andrew J. Davison. MASt3R-SLAM: Real-time dense SLAM with 3D reconstruction priors. In *CVPR*, 2025.

David Nordström, Johan Edstedt, Fredrik Kahl, and Georg Bökman. Mum: Multi-view masked image modeling for 3d vision. In *CVPR*, 2026.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Jinsoo Park, Donggyu Choi, Ahyun Seo, Minsu Cho, and Jeany Son. Diversity-aware view partitioning for scalable vggt. *ECCV*, 2026.

René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *CVPR*, 2021.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *ICCV*, 2021.

Mehdi S. M. Sajjadi, Henning Meyer, Etienne Pot, Urs Bergmann, Klaus Greff, Noha Radwan, Suhani Vora, Mario Lucic, Daniel Duckworth, Alexey Dosovitskiy, Jakob Uszkoreit, Thomas Funkhouser, and Andrea Tagliasacchi. Scene representation transformer: Geometry-free novel view synthesis through set-latent scene representations. *CVPR*, 2022.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *ArXiv*, 2021.

Carole Helene Sudre, Wenqi Li, Tom Kamiel Magda Vercauteren, Sébastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Deep learning in medical image analysis and multimodal learning for clinical decision support*, 2017.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *CVPR*, 2020.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, Tatsunori Hashimoto, and Carlos Guestrin. Learning to (learn at test time): Rnns with expressive hidden states. *ICML*, 2025.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *NeurIPS*, 2022.

Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 1991.

Sjoerd van Steenkiste, Daniel Zoran, Yi Yang, Yulia Rubanova, Rishabh Kabra, Carl Doersch, Dilara Gokay, Joseph Heyward, Etienne Pot, Klaus Greff, Drew Hudson, Thomas Albert Keck, João Carreira, Alexey Dosovitskiy, Mehdi S. M. Sajjadi, and Thomas Kipf. Moving off-the-grid: Scene-grounded video representations. *NeurIPS*, 2024.

Chung-Shien Brian Wang, Christian Schmidt, Jens Piekenbrinck, and Bastian Leibe. Block-sparse global attention for efficient multi-view geometry transformers. *CVPR*, 2026a.

Haonan Wang, Hanyu Zhou, Haoyue Liu, and Luxin Yan. 4d-vggt: A general foundation model with spatiotemporal awareness for dynamic scene geometry estimation. *ECCV*, 2026b.

Hengyi Wang and Lourdes Agapito. 3d reconstruction with spatial memory. *3DV*, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *CVPR*, 2025a.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-Ω. *CVPR*, 2026c.

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

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D vision made easy. In *CVPR*, 2024.

Wenshan Wang, Delong Zhu, Xiangwei Wang, Yaoyu Hu, Yuheng Qiu, Chen Wang, Yafei Hu, Ashish Kapoor, and Sebastian Scherer. TartanAir: A dataset to push the limits of visual slam. In *IROS*, 2020.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *ICLR*, 2026d.

Zipeng Wang and Dan Xu. Flashvggt: Efficient and scalable visual geometry transformers with compressed descriptor attention. *CVPR*, 2026.

Chen Wei, Haoqi Fan, Saining Xie, Chaoxia Wu, Alan Loddon Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. *CVPR*, 2022.

Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *CVPR*, 2023.

Yibin Yan, Jilan Xu, Shangzhe Di, Haoning Wu, and Weidi Xie. Omnistream: Mastering perception, reconstruction and action in continuous streams. In *ECCV*, 2026.

Jianing Yang, Alexander Sax, Kevin J. Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *CVPR*, 2025.

Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *CVPR*, 2020.

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *CVPR*, 2023.

Shuai Yuan, Yantai Yang, Xiaotian Yang, Xupeng Zhang, Zhonghao Zhao, Lingming Zhang, and Zhipeng Zhang. Infinitevggt: Visual geometry grounded transformer for endless streams, 2026.

Yuheng Yuan, Qiuhong Shen, Shizun Wang, Xingyi Yang, and Xinchao Wang. Test3r: Learning to reconstruct 3d at test time, 2025.

Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K. Barral, Raia Hadsell, Zoubin Ghahramani, Andrew Zisserman, Junlin Zhang, and Mehdi S. M. Sajjadi. Efficiently reconstructing dynamic scenes one D4RT at a time. In *CVPR*, 2026a.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Chen Sun, Ming-Hsuan Yang, Forrester Cole, Trevor Darrell, and Deqing Sun. Loger: Long-context geometric reconstruction with hybrid memory. *arXiv*, 2026b.

Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T. Freeman, and Hao Tan. Test-time training done right. *ICML*, 2026c.

Yuchen Zhang, Nikhil Keetha, Chenwei Lyu, Bhuvan Jhamb, Yutian Chen, Yuheng Qiu, Jay Karhade, Shreyas Jha, Yaoyu Hu, Deva Ramanan, Sebastian Scherer, and Wenshan Wang. Ufm: A simple path towards unified dense correspondence with flow. In *NeurIPS*, 2025.

Yang Zheng, Adam W. Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J. Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *ICCV*, 2023.

Zhijie Zheng, Xinhao Xiang, and Jiawei Zhang. Ttsa3r: Training-free temporal-spatial adaptive persistent state for streaming 3d reconstruction, 2026.

Filippo Ziliotto, Luciano Serafini, Lamberto Ballan, and Tommaso Campari. What vggt knows about overlap: Probing geometric foundation models for co-visibility. *ECCV*, 2026.

Daniel Zoran, Nikhil Parthasarathy, Yi Yang, Drew A Hudson, João Carreira, and Andrew Zisserman. Recurrent video masked autoencoders. *arXiv*, 2025.

Lojze Zust, Yohann Cabon, Juliette Marrie, Leonid Antsfeld, Boris Chidlovskii, Jerome Revaud, and Gabriela Csurka. Panst3r: Multi-view consistent panoptic segmentation. In *CVPR*, 2025.

APPENDIX

A.1 COVISIBILITY

A.1.1 STATIC REPROJECTION

We formulate pairwise covisibility between an ordered source-target frame pair $(\mathbf{I}_i, \mathbf{I}_j)$ with camera calibration matrices $\mathbf{K}_i, \mathbf{K}_j \in \mathbb{R}^{3\times3}$, metric depth maps $\mathbf{D}_i, \mathbf{D}_j \in \mathbb{R}^{H\times W}$, surface normal maps $\mathbf{N}_i \in \mathbb{R}^{H\times W\times3}$, and relative rigid transformation $(\mathbf{R}_{ij}, \mathbf{t}_{ij}) \in \mathrm{SE}(3)$ mapping 3D coordinates from the camera frame of view i into view j. For any valid source pixel coordinate $\mathbf{p}_i = [u_i, v_i, 1]^\top$, its 3D position in camera coordinates is obtained by unprojection:

$$\mathbf{X}_i = \mathbf{D}_i(\mathbf{p}_i)\mathbf{K}_i^{-1}\mathbf{p}_i \in \mathbb{R}^3. \tag{4}$$

Assuming a static scene, this 3D point is rigidly transformed into the coordinate frame of camera j:

$$\mathbf{X}_{i\to j} = \mathbf{R}_{ij}\mathbf{X}_i + \mathbf{t}_{ij} = [x_j, y_j, z_j]^\top, \tag{5}$$

and projected onto the target image sensor via perspective projection π:

$$\mathbf{p}_{i\to j} = \pi(\mathbf{K}_j\mathbf{X}_{i\to j}) = \left[\frac{(\mathbf{K}_j\mathbf{X}_{i\to j})_1}{z_j}, \frac{(\mathbf{K}_j\mathbf{X}_{i\to j})_2}{z_j}\right]^\top. \tag{6}$$

Let $\Omega_j = [0, W-1] \times [0, H-1]$ denote the image domain of view j. Source pixel $\mathbf{p}_i$ is categorized into one of four mutually exclusive visibility states:

1. **out-of-FOV ($m=0$):** The reprojected point lies behind camera j ($z_j \leq 0$) or outside the target image sensor boundaries, such that $\mathbf{p}_{i\to j} \notin \Omega_j$.
2. **Occluded ($m=1$):** The point projects inside the target sensor ($\mathbf{p}_{i\to j} \in \Omega_j, z_j > 0$), but fails the relative depth-buffer consistency check against the recorded target depth $\mathbf{D}_j(\mathbf{p}_{i\to j})$:

$$\frac{z_j - \mathbf{D}_j(\mathbf{p}_{i\to j})}{\mathbf{D}_j(\mathbf{p}_{i\to j})} > \tau_{\text{depth}}, \tag{7}$$

   where $\tau_{\text{depth}} = 5\%$. Furthermore, we reject surfaces that graze or face away from target camera j by checking the transformed surface normal against the target line-of-sight vector $\mathbf{e}_z = [0,0,1]^\top$:

$$\angle(\mathbf{e}_z, \mathbf{R}_{ij}\mathbf{N}_i(\mathbf{p}_i)) < 90^\circ - \epsilon_{\text{normal}}, \tag{8}$$

   with $\epsilon_{\text{normal}} = 5^\circ$. Points failing the normal check are also assigned to the occluded class.
3. **Covisible ($m=2$):** The point projects within bounds ($\mathbf{p}_{i\to j} \in \Omega_j$), satisfies the surface normal check, and passes the symmetric relative depth test:

$$\frac{|z_j - \mathbf{D}_j(\mathbf{p}_{i\to j})|}{\max(\mathbf{D}_j(\mathbf{p}_{i\to j}), \epsilon)} \leq \tau_{\text{depth}}. \tag{9}$$

4. **Undefined ($m=-1$):** Pixels with invalid source depth ($\mathbf{D}_i(\mathbf{p}_i) \leq 0$), sky, or boundary artifacts are masked out and ignored by all loss functions.

Given the set of valid source pixels $\mathcal{V}_i = \{\mathbf{p}_i \mid \mathbf{D}_i(\mathbf{p}_i) > 0\}$, the scalar asymmetric covisibility overlap $M_{ij} \in [0,1]$ is defined as the fraction of valid source pixels that are covisible in view j:

$$M_{ij} = \frac{1}{|\mathcal{V}_i|} \sum_{\mathbf{p}_i \in \mathcal{V}_i} \mathbb{I}(\text{class}(\mathbf{p}_i, j) = \text{covisible}), \tag{10}$$

with $M_{ii} = 1.0$ by definition. In general, $M_{ij} \neq M_{ji}$.

A.1.2 Deriving Dynamic Covisibility Labels

Static reprojection assumes all 3D points remain rigid across time. When objects move between timestamp t_i and t_j, static reprojection causes epipolar violations, mistakenly classifying moving foreground surfaces as occluded. We resolve this limitation across dynamic sequences depending on the available ground-truth annotations:

- **Rigid Object-Pose Transfer (e.g., Kubric (Greff et al., 2022)):** For synthetic dynamic sequences with known rigid instance segments and 6-DoF object-to-world transformations $\mathbf{T}_{w\leftarrow o}(t) \in \text{SE}(3)$, moving pixels are identified when the object pose delta between frames exceeds a motion tolerance: $\|\mathbf{T}_{w\leftarrow o}(t_i) - \mathbf{T}_{w\leftarrow o}(t_j)\|_F > \tau_{\text{obj}}$. For each moving source point $\mathbf{X}_w(t_i)$, its target 3D coordinate at timestamp t_j is transferred via:

$$\mathbf{X}_w(t_j) = \mathbf{T}_{w\leftarrow o}(t_j)\mathbf{T}_{w\leftarrow o}(t_i)^{-1}\mathbf{X}_w(t_i). \tag{11}$$

  The motion-transferred point is then projected into target camera j and evaluated against the target depth buffer $\mathbf{D}_j$.
- **Dense 3D Point Tracks (e.g., PointOdyssey (Zheng et al., 2023)):** For sequences with dense ground-truth 3D point tracks $\mathbf{X}_w(t)$, a source track with motion displacement $\|\mathbf{X}_w(t_j) - \mathbf{X}_w(t_i)\|_2 > \tau_{\text{track}}$ is projected into camera j using its ground-truth target coordinate $\mathbf{X}_w(t_j)$. Visibility is established either directly from ground-truth track visibility flags or via z-buffer testing against rendered target depth.
- **Unannotated Datasets:** On purely static sequences, standard rigid reprojection applies directly. However, unconstrained real-world scenes often contain moving objects without ground-truth 3D poses or point tracks. We detect these moving regions purely from depth maps and camera poses by exploiting a simple visibility asymmetry: a camera ray can be blocked by a closer surface, but it can never pass through a solid static surface to observe geometry behind it. Consequently, comparing a rigidly reprojected depth $Z_{i\to j}$ against the target depth buffer $\mathbf{D}_j(\mathbf{p}_j)$ yields two distinct regimes depending on the sign of the depth residual, which we resolve via bidirectional reprojection ($i \to j$ and $j \to i$):

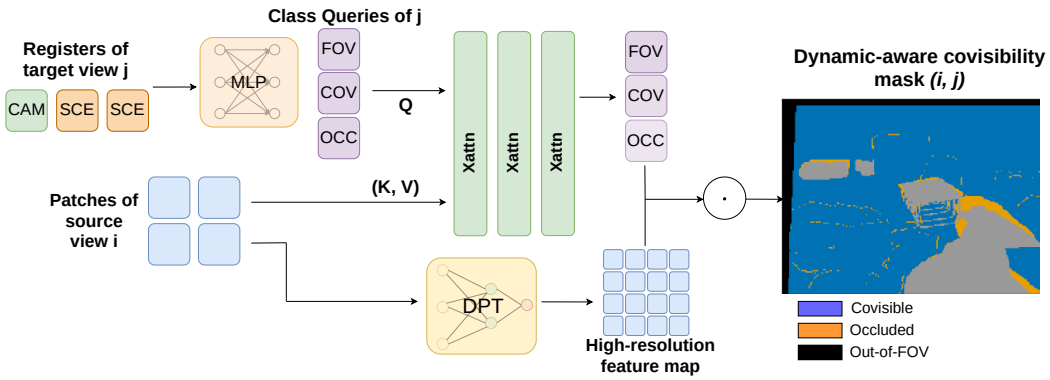


Figure A1: **Query-Based Covisibility Head.** First, for each frame, we use a shared DPT Head (the same design as the one in Fig. 2 for finetuning). Each frame k gets a unique high-resolution feature map F_k of size $\frac{H}{4} \times \frac{W}{4}$ that is used every time that the frame is sampled as a source. Then, for a given sequence of frames S, we sample P pairs out of the possible $S(S-1)$. For every pair $(source, target) = (i, j)$, we project the registers of the target j into M class queries Q_j ($M = 3$). These queries attend to the patch tokens of the source view i. Finally, we get the dot product between F_i and Q_j to get the class logits.

1. Forward reprojection ($i \to j$) catches objects that moved away (negative depth residual): When a 3D point $\mathbf{X}_i$ from frame i rigidly projects to pixel $\mathbf{p}_j$ in camera j with $Z_{i\to j} < \mathbf{D}_j(\mathbf{p}_j) - \tau$, the reprojected point lies in front of the surface actually observed by camera j. Because camera j's line of sight passed unobstructed through the 3D coordinates of $\mathbf{X}_i$ to hit a deeper background ($\mathbf{D}_j(\mathbf{p}_j)$), the surface that occupied $\mathbf{X}_i$ at time t_i must have vacated that location by time t_j. Thus, a negative depth residual in $i \to j$ directly proves that the source pixel in frame i belongs to a moving object.
2. Backward reprojection ($j \to i$) disambiguates static vs. dynamic occluders: Conversely, when $Z_{i\to j} > \mathbf{D}_j(\mathbf{p}_j) + \tau$, the reprojected point $\mathbf{X}_i$ falls behind the surface $\mathbf{Q}_j = (\mathbf{p}_j, \mathbf{D}_j(\mathbf{p}_j))$ observed in frame j. From the forward check alone, the sign of this occlusion is ambiguous: $\mathbf{X}_i$ could be genuinely occluded by static scene geometry (e.g., a wall or pillar due to viewpoint change), or it could be a static background point that became blocked by a moving object entering camera j's view. To break this ambiguity, we unproject the occluder $\mathbf{Q}_j$ from frame j back into camera i at pixel $\mathbf{p}'_i$ and apply the same test in reverse. If $Z_{j\to i} < \mathbf{D}_i(\mathbf{p}'_i) - \tau$, camera i saw right through $\mathbf{Q}_j$'s 3D position at time t_i to a deeper surface $\mathbf{D}_i(\mathbf{p}'_i)$, proving that $\mathbf{Q}_j$ was empty space at t_i and is therefore a moving foreground occluder rather than static geometry.

A.2 Pretraining

A.2.1 Comparison to Alligat0r Head

Covisibility Head Design. In Sec. 3.2, we introduced a new view-conditioned query-based head to allow for flexible pair selection and more efficiency. We give a detailed figure of the head in Fig. A1. This head computes only one feature map per view with a DPT (Ranftl et al., 2021) that is reused every time the frame is sampled as a source. Then, for all the pairs sampled, we instantiate class queries (similar to object queries in Mask2Former (Cheng et al., 2022)) with the register tokens of the target view. The class queries cross attend to the patch tokens to capture the geometry of the source and compare it against their own geometry. The last operation is a dot product with the high-resolution feature map to get the class logits. It is worth noting that we fix the queries to be assigned to a class name (query 0 detects out-of-FOV, 1 detects occluded and 2 detects covisible) and we know exactly the number of masks in the ground truth (exactly 3 masks, one for each class), so we do not have to compute Hungarian matching during training.

In the pairwise setting of (Loiseau et al., 2025), no head is needed as there are only two views. A straightforward extension would be to predict the mask from view $j \to 0$ without any head. In Tab. A1 (*Head Design* section, we compare this pretraining to our head design on the same sampled pairs. We observe that our design surpasses the straightforward Alligat0r extension.

Table A1: **Learned Upsampling vs. Query Head.** We compare extending Alligat0r (Loiseau et al., 2025) to more than two views by using the same learned upsamling and computing the mask of view $j>0$ in the reference view 0. Pretrained on the same pairs, our introduced head yields better downstream performance both in depth and camera pose.

		ScanNet++		
Axis	**Pretraining Setup**	AUC@5↑	AbsRel↓	$\delta_1<1.25$↑
Head Design	Learned Upsampling (Alligat0r)	18.3	0.042	97.3
	Query Head (Ours)	**19.1**	**0.041**	**97.4**
Objective	Camera Pose (Dong et al., 2025)	13.7	**0.043**	97.2
	Combined Covis Head	**17.8**	**0.043**	**97.4**

Memory and Compute Complexity. A natural way to extend Alligat0r (Loiseau et al., 2025) to multi-view sequences is to add a dedicated head that takes the patch tokens of a given pair, applies cross-attention between the two views' patches to simultaneously predict the masks for both directions $(i\to j)$ and $(j\to i)$, and maps the transformed patches to dense masks via learned upsampling. We did not choose this design because it goes against our core principle of removing direct patch-to-patch interactions across frames, and incurs a higher per-pair memory complexity. To formalize this trade-off, let S denote the number of views, P an even number of directed pairs $(i\to j)$ (corresponding to $P/2$ unordered pairs $\{i,j\}$), $p=\frac{H}{16}\frac{W}{16}$ the number of patches per view ($p=768$ at 384×512, so quarter resolution $\frac{H}{4}\times\frac{W}{4}$ contains $16p$ pixels), $Q=3$ the number of queries per directed pair, and $C=3$ the number of covisibility classes, with head feature dimension $D=256$, $L=3$ cross-attention layers, and $h=8$ attention heads. In the patch-to-patch + learned upsampling head, the upsampling step itself is lightweight because learned upsampling maps the transformed tokens directly to $16PpC$ quarter-resolution logits (or $256PpC$ at full resolution) without a convolutional feature pyramid, and each of the $P/2$ pairwise passes yields logits for both $(i\to j)$ and $(j\to i)$, requiring:

$$\mathcal{M}_{\text{patch}} = \underbrace{SLpD}_{\text{source K/V}} + \underbrace{\frac{P}{2}L(pD+hp^2)}_{\text{bidirectional patch-to-patch X-attn}} + \underbrace{16PpC}_{\text{upsampled logits (res/4)}} . \tag{12}$$

In contrast, our per-view DPT (Ranftl et al., 2021) + query-to-patch head forces the backbone's register tokens $\mathbf{R}_j$ to encode target geometry during inter-frame attention, pairing a shared per-view DPT feature map $\mathbf{F}_i\in\mathbb{R}^{\frac{H}{4}\times\frac{W}{4}\times D}$ with L layers of cross-attention between $Q=3$ register-seeded queries $\mathbf{Q}_{i\leftarrow j}\in\mathbb{R}^{Q\times D}$ and the p source patches for each of the P directed pairs:

$$\mathcal{M}_{\text{query}} = \underbrace{SLpD}_{\text{source K/V}} + \underbrace{16SpD}_{\text{shared per-view DPT}} + \underbrace{PL(QD+hQp)}_{\text{per-pair query-to-patch X-attn}} + \underbrace{16PpC}_{\mathbf{F}_i\mathbf{Q}_{i\leftarrow j}^{\top}\text{ logits (res/4)}} . \tag{13}$$

Although the DPT canvas introduces a fixed per-view cost of $16SpD$ floats (50 MB at $S=8$ in `fp16`) that direct learned upsampling avoids, this overhead scales only with S rather than P. Meanwhile, even after accounting for the factor of 2 gained by predicting both $(i\to j)$ and $(j\to i)$ per pairwise pass, replacing patch-to-patch cross-attention with $Q=3$ register-derived queries reduces the per-pair cross-attention activations ($\frac{1}{2}PLpD\to PLQD$), score maps ($\frac{1}{2}PLhp^2\to PLhQp$), and $QK^{\top}$/FFN FLOPs by a factor of $p/(2Q)=\mathbf{128}\times$ (33 MB → 0.26 MB in query tokens and 0.80 GB → 6.2 MB in score maps at $S=8,P=56$). Our head requires $128\times$ fewer cross-attention FLOPs per pair while channeling all cross-view geometric supervision through the register bottleneck.

A.2.2 Pretraining Objective

Additional Ablations. Tab. A2 shows the depth metrics (absolute relative error AbsRel and inliers at 25%) that correspond to the camera pose reported in Tab. 1. In every setting, the depth metrics are almost saturated, all the variants being very close to each other, even though the scratch baseline has a little edge. This shows that depth is essentially a monocular task and doesn't need cross-view reasoning, thus does not benefit a lot from covisibility pretraining, whereas camera pose requires the model to learn cross-view relationships.

Alternative Pretraining Objectives. In Tab. A1 (*Objective* section), we investigate alternative pretraining objectives to understand what geometric signals best structure the register bottleneck. First, we pretrain

Table A2: **Progressive Pretraining Ablation (ViT-S/16, S=12).** We sequentially ablate the pretraining objective (on fixed $j \to 0$ pairs with static reprojection), dynamic label handling, and pair sampling strategies. We report the depth metrics AbsRel and inliers $\delta < 1.25$.

Axis	Pretraining Setup	ScanNet++ AbsRel↓	ScanNet++ $\delta<1.25$↑	Kubric AbsRel↓	Kubric $\delta<1.25$↑
Objective	Scratch	0.036	97.6	0.019	98.7
	Scalar Covis. $[S,S]$	0.043	97.2	0.021	98.7
	Dense Covis. (Static)	0.042	97.3	0.024	98.7
Labels	+ Dynamic Labels	0.044	97.2	0.022	98.7
	+ Ignore Dynamic	0.040	97.5	0.020	98.7
Sampling	+ Random Pairs	0.040	97.6	0.021	98.7
	+ Per view Random Pairs	0.040	97.5	0.022	98.7

Table A3: **Impact of Covisibility Pretraining with Global Attention on ScanNet++.** Pretraining consistently improves both camera pose estimation and video depth accuracy over training from scratch.

Method	AUC@5↑	AUC@10↑	AUC@20↑	AbsRel↓	$\delta<1.25$↑
Global (Scratch)	23.8	44.7	65.3	0.034	97.8
Global (Pretrain)	**31.2**	**52.1**	**70.4**	**0.029**	**97.9**

the model using direct relative camera pose supervision using the pairwise pose head introduced in Reloc3r (Dong et al., 2025). While pose-only pretraining improves downstream camera pose estimation slightly over the from-scratch baseline in Tab. 1, it still lags behind dense covisibility pretraining on pose accuracy and leads to a degradation in dense depth prediction. This result is logical: supervising global 6-DoF camera pose does not provide an incentive to encode depth. Second, we explore combining a global weak signal, namely $S \times S$ pairwise scalar covisibility matrix regression, with our dense per-pixel covisibility segmentation loss under equal loss weighting. Adding this global scalar objective does not improve over dense covisibility alone and slightly hurts downstream accuracy, suggesting that regressing a coarse view-level overlap score introduces a shortcut signal in the registers that competes with the fine-grained spatial and occlusion reasoning enforced by dense mask prediction.

What Does the Model Learn for Dynamic Objects? Under static reprojection, moving objects violate epipolar consistency and are labeled as occluded, whereas under explicit dynamic supervision ($m \in \{3,4\}$) the model is directly tasked with segmenting object motion. A natural question is what the network predicts on moving objects when they are masked out during training via the ignore label ($m=-1$). As shown in Fig. A2, while the network primarily matches visible moving objects as covisible ($m=2$, 59.6%), whenever object motion causes a dynamic region to be classified as non-covisible (growing from 30.3% at $\Delta t=2$ to 51.5% at $\Delta t=20$), the model consistently predicts out-of-FOV ($m=0$, 32.5%) rather than occluded ($m=1$, 7.9%). We attribute this behavior to how the two non-covisible classes decompose $\mathrm{SE}(3)$ camera motion: out-of-FOV regions are governed primarily by camera rotation (as a pure rotation sweeps the field of view without inducing depth parallax), whereas occlusion requires reasoning about 3D depth ordering and parallax induced by camera translation. Empirically, and in line with recent findings from SeeSE3 (Chen et al., 2026a), camera translation and parallax are substantially harder to learn than rotation. Consequently, when encountering dynamic foreground regions where static epipolar correspondences break down without a foreground depth occluder, the model defaults to the simpler rotational cue and explains the unmatched visual content as having left the target field of view (out-of-FOV) rather than hallucinating 3D parallax occlusions.

Pretraining a Global Attention Model. In the main experiments, we prioritized investigating geometric pretraining as a way to improve register attention. Preliminary experiments on ScanNet++ (Yeshwanth et al., 2023) and with ViT-S showed that pretraining the dense model with the same objective and the same model design yield substantial gain over the same model trained only on 3D reconstruction. As shown in Tab. A3, after having seen the same number of examples (with 25/15 epochs of pretraining/finetuning vs 40 epochs of finetuning), the global attention variant benefits from pretraining and scores +7.4 points on AUC@5. This suggests that this training recipe could be applied to train even stronger models.

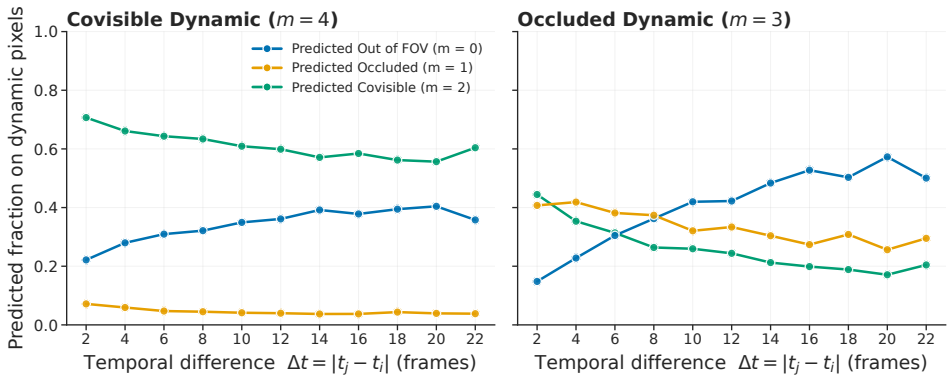


Figure A2: **Model Predictions on Unsupervised Dynamic Objects Across Temporal Difference Δt.** On Kubric (Greff et al., 2022) held-out sequences, moving regions masked during pretraining ($m=-1$) are predominantly matched as covisible ($m=2$) when visible (left, $m=4$), while non-covisible dynamic regions are consistently classified as out-of-FOV ($m=0$) rather than occluded ($m=1$), especially as Δt increases (right, $m=3$).

A.3 Training

A.3.1 Datasets

The training mixture of our final model spans static and dynamic indoor/outdoor scenes from BlendedMVS (Yao et al., 2020), Co3Dv2 (Reizenstein et al., 2021), Dynamic Replica (Karaev et al., 2023), Kubric (Greff et al., 2022), MVS-Synth (Huang et al., 2018), PointOdyssey (Zheng et al., 2023), ScanNet++ (Yeshwanth et al., 2023), ScanNet (Dai et al., 2017), TartanAir (Wang et al., 2020), VirtualKITTI (Gaidon et al., 2016), Waymo Open (Sun et al., 2020), Cub3 (Loiseau & Bourmaud, 2025) with an internal dataset. We use the same augmentations in all stages. For each sample, we draw a single centered crop shared by all views, whose area is a fraction of the image area drawn uniformly in $[0.3,1]$ and whose aspect ratio is drawn log-uniformly in the range that keeps the crop inside the image. We resize the crop to 384×512 and update the intrinsics accordingly. We then jitter the brightness, contrast and saturation of each view independently, with multiplicative factors drawn uniformly in $[0.6,1.4]$ and applied in random order. We do not use flips, hue jitter, blur, or view shuffling.

A.3.2 Training Schedule

We train our 1B model in four consecutive runs, one for Stage I and three for Stage II. Stage I pretrains the encoder and the register decoder with the covisibility objective of Sec. 3.2 on sequences of 2 to 24 views. The encoder is initialized from DINOv3 ViT-L/16 (Siméoni et al., 2025), the decoder and the covisibility head are randomly initialized, and all parameters are trained from the first step. Stage II then finetunes the model for 3D reconstruction with the objective of Eq. 3 in three phases of increasing sequence length, with up to S=24, 48 and 96 views. The S=24 phase loads the encoder and the decoder of Stage I and randomly initializes the camera and DPT heads, while the S=48 and S=96 phases load all the weights of the previous phase, heads included. At the beginning of every stage and phase, we reset the optimizer state and restart the learning rate schedule with a linear warmup.

A.3.3 Optimization

We use AdamW (Loshchilov & Hutter, 2019) with β_1=0.9, β_2=0.95 and a weight decay of 0.05, , and we clip gradients at a global norm of 1.0. The learning rate follows a linear warmup and a cosine decay. In Stage I, all modules share a base learning rate of 1.5×10^{-4} and a decay to zero at 250k steps. In Stage II, we lower the learning rate to 5×10^{-5} and scale it per module: the encoder uses half of the decoder learning rate in all phases, and the heads use twice the decoder learning rate in the S=24 phase, where they are trained from scratch for 150k steps. The $S=48$ and $S=96$ phases last 60k steps each. We train with data parallelism on 128 GPUs (batch size 256 in Stage I and 128 in Stage II), in `bfloat16` mixed

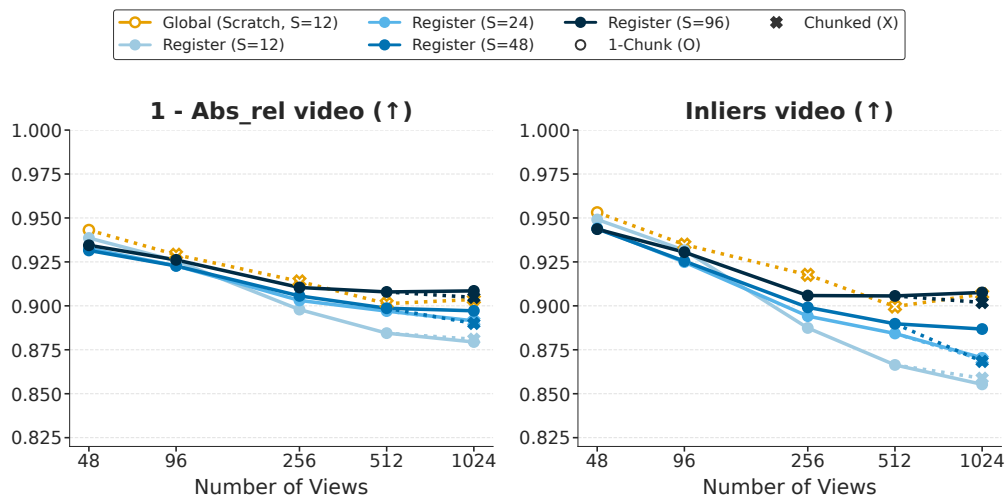


Figure A3: **Depth metrics when evaluating on long sequences.** Similarly to Fig. 4 on camera pose, a register-attention model finetuned on 96 views has better depth performance on large view counts. We use 64-frame windows for global attention and 512 for register attention.

precision and we use FlashAttention (Dao, 2023). To fit long sequences in memory, we rematerialize the activations every decoder layer. From S=48 on, we also run the DPT head on chunks of 8 views.

A.3.4 Covisibility Supervision

In Stage I, we compute the covisibility labels introduced in Sec. 3.2 on the fly on the CPU hosts, with the Ignore Dynamic regime (M=3) and the bidirectional depth-consistency filter of Sec. A.1.2. We use per-view random sampling: we draw a random cyclic ordering of the n views of the sample and supervise both directions of each consecutive pair, which yields $2n$ distinct ordered pairs in which every view is twice a source and twice a target.

A.3.5 Sequence Length Curriculum

We train on S frames sampled from videos with a random stride between each view. When a sample has fewer views than S, which happens when a video has fewer than S frames, we pad it to S views and mask the padded views out of the cross-view attention and of all losses. We first train on samples between 2 and 24 views. The, the S=48 phase draws 24 to 48 views. Finally, the S=96 phase draws 48 to 96 views.

A.4 Inference

A.4.1 Scaling Sequence Length at Inference

In Sec. 4.1, a global-attention model trained on few views underperformed pretrained register models trained on more views in camera pose. Fig. A3 shows that depth metrics are more stable and follow a similar but much weaker trend: the register-attention model trained on 96 views is on par with or slightly better than chunked global attention at 512 and 1024 views.

A.4.2 Qualitative Results

In Fig. A4, we provide additional trajectory comparisons on ScanNet (last two rows) where our model clearly outperforms VGGT-Ω-chunk and π^3 baselines. On large-scale outdoor scenes such as KITTI (top row), our model sometimes struggle to recover metric translations, thus explaining the lower scores in Tab. 2. We attribute this to an imbalanced data mix between indoor and outdoor rather than a fundamental limitation.

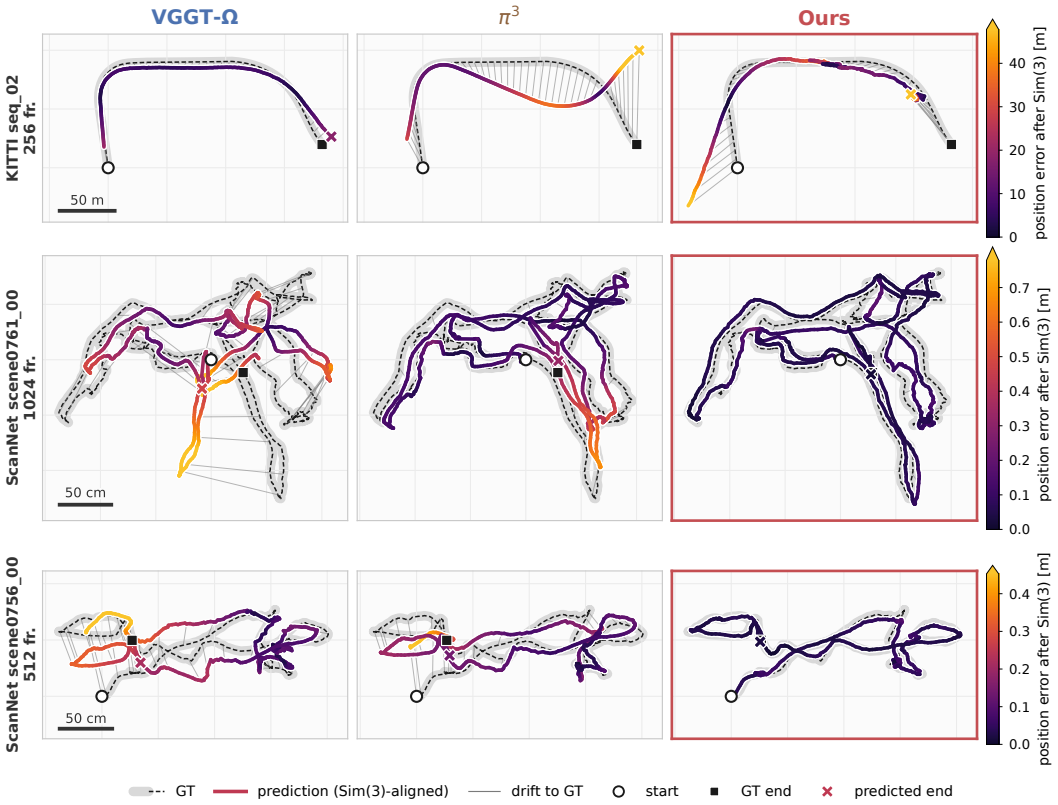


Figure A4: **Qualitative camera trajectories.** Our model (right column) performs best in indoor scenes for large view counts (middle and last rows). In some outdoor scenes (that are challenging for other models like π^3 in the middle column), we observe that the overall the shape of the trajectory is faithful but translation degenerates towards the end of the sequence.