ANY3D-SLAM: TAMING ANY GENERAL 3D PRIORS FOR ROBUST IN-THE-WILD SLAM

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce Any3D-SLAM, a robust visual SLAM system for jointly recovering camera poses and dense depth from streaming videos under unconstrained capture conditions. Conventional visual SLAM achieves accurate camera estimation by explicitly modeling multi-view geometric constraints, but its reliance on reliable correspondences makes it vulnerable to weak visual evidence and scene dynamics. Existing methods address these challenges with task-specific predictors, but their separate representations limit the use of complementary cues across tasks to support pose estimation. Recent 3D foundation models offer shared multi-frame representations and complementary priors, yet their feed-forward predictions may not fully satisfy multi-view geometric constraints. To harness these representations while enforcing geometric consistency, Any3D-SLAM integrates foundation-model features and complementary depth, motion, pose, and intrinsics priors into an end-to-end differential bundle adjustment (DBA) framework. We develop a warping-based updater for efficient correspondence estimation, substantially reducing memory overhead and enabling richer multi-frame representations and their associated priors to participate in online optimization. Among these priors, we introduce uncertainty-gated relative-pose constraints into online BA to support camera estimation when translational observability is weak or moving objects occlude the static background. Extensive experiments across six benchmarks spanning indoor, outdoor, dynamic, and egocentric environments demonstrate strong camera pose estimation performance in both calibrated and uncalibrated settings.

1 INTRODUCTION

Visual simultaneous localization and mapping (SLAM) jointly estimates camera motion and scene geometry from sequential observations. In-the-wild monocular videos often violate assumptions of static scenes, known intrinsics, and sufficient translational parallax. Moving objects introduce image displacements inconsistent with camera motion and may occlude the static background needed for reliable geometric estimation. Existing learning-based systems address these challenges using task-specific predictors for depth (Li et al., 2025b; 2026; Zheng et al., 2026), segmentation (Wang & Lee, 2026; Huang et al., 2025; Li et al., 2025a), or point tracking (Sun et al., 2025; Chen et al., 2025b). Although their outputs can be combined in geometric optimization, these predictors typically use separate representations, limiting the sharing of complementary geometric and temporal cues across correspondence estimation and prior prediction.

Recent 3D foundation models (Wang et al., 2025a; 2026b; Lin et al., 2025; Zhang et al., 2026a) offer a promising source of shared representations and complementary priors. Through large-scale pre-training, these models learn rich geometric features that generalize across diverse scenes and support the prediction of camera parameters and scene geometry. Recent approaches extend their capabilities to long sequences through temporal memory mechanisms (Wang et al., 2025b; Chen et al., 2026b;a) or submap-based processing (Maggio et al., 2025; Mandal et al., 2026; Maggio & Carlone, 2026). Nevertheless, feed-forward predictions and their integration across frames or submaps may not fully satisfy multi-view reprojection constraints, leading to local geometric inaccuracies and trajectory drift. This motivates exploiting shared 3D representations within a framework that explicitly enforces multi-view geometric constraints.

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.

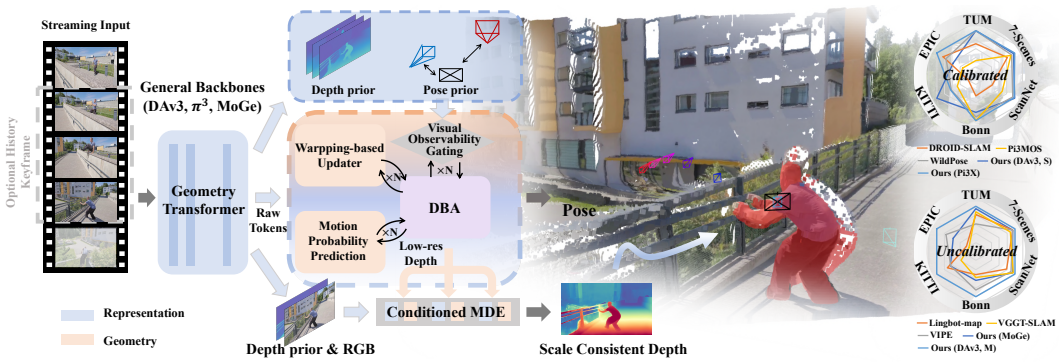


Figure 1: Overview of Any3D-SLAM. A general 3D backbone extracts shared representations and complementary priors from streaming RGB frames and optional historical keyframes. Our warping-based updater uses these representations to predict correspondence corrections and confidence weights, alternating with DBA to refine camera poses and inverse depths. Depth priors, motion-based correspondence weighting, and uncertainty-gated pose constraints support robust estimation in unconstrained scenes. Finally, an off-the-shelf conditioned depth model (Wang et al., 2025d) fuses the optimized low-resolution depth with backbone depth predictions and RGB images to produce full-resolution, scale-consistent depth.

In this paper, we introduce Any3D-SLAM, a visual SLAM framework that combines the generalization capabilities of 3D foundation models with explicit geometric optimization. Our core idea is to use a shared 3D representation for both iterative correspondence estimation and complementary prior integration within differential bundle adjustment (DBA). This brings the backbone's rich geometric and temporal cues directly into correspondence prediction, while its complementary priors provide additional geometric constraints and motion-aware weighting. Together, these cues strengthen camera tracking when correspondence-based evidence is unreliable or insufficient, while DBA refines camera parameters and scene geometry under explicit multi-view reprojection constraints. To efficiently realize this coupling across diverse 3D backbones, we develop a warping-based updater that avoids explicit all-pairs correlation volumes. This design substantially reduces memory overhead while maintaining comparable estimation accuracy, enabling richer multi-frame representations and their associated priors to support online estimation.

To adapt pose-prior integration to the reliability of visual evidence, we introduce an uncertainty-gated relative-pose regularizer into online BA. It complements reprojection constraints with direct camera-motion information, activating its translation terms only when the estimated translational observability is low. This provides targeted support under limited parallax or severe foreground occlusion without imposing translation priors when visual constraints are sufficient. We further use the optimized geometric states to guide full-resolution depth estimation, extending the framework to accurate dense reconstruction. Across six diverse benchmarks, as illustrated in Fig. 1 our uncalibrated multi-frame variant achieves the lowest or joint-lowest ATE among evaluated uncalibrated frame-based methods.

2 Related Work

In-the-Wild Visual SLAM. Considerable effort has been devoted to making visual SLAM robust in unconstrained real-world environments. For challenging visual conditions, DROID-SLAM (Teed & Deng, 2021) couples learned correspondence updates with dense differential bundle adjustment (BA). Subsequent methods (Wu et al., 2026; Zheng et al., 2026) improve correspondence estimation using 3D-aware feature extractors tailored to stereo or pairwise matching. For dynamic scenes, early methods rely on pretrained segmentation networks or object detectors, whose category-dependent predictions may miss unfamiliar moving objects or suppress static instances of potentially dynamic categories. More recent approaches model motion directly through supervised learning or online optimization. Among these, π^3_{MOS}-SLAM Zhong et al. (2026) extends the feed-forward model π^3 (Wang et al., 2026b) to provide motion probabilities and depth priors for SLAM, while retaining a separate correspondence estimator (Teed et al., 2023) for BA and leaving predicted pose priors unused in geometric optimization. Geometric degeneracies, such as those induced by insufficient

translational parallax, pose another challenge to purely visual systems. Building on direct sparse odometry (Engel et al., 2017), D3VO (Yang et al., 2020) incorporates depth and pose predictions from dedicated networks into photometric optimization to improve robustness. Among learning-based systems, MegaSaM (Li et al., 2025b) uses Hessian-based uncertainty to assess observability and guide depth and intrinsics optimization during global BA. To our knowledge, Any3D-SLAM is the first learning-based SLAM framework to support diverse general 3D representations for both iterative correspondence updates and complementary prior integration within BA. Our newly proposed uncertainty-gated pose regularization further provides direct camera-motion constraints when correspondence-based geometry is insufficient, improving robustness under weak translational observability and severe foreground occlusion.

3D Foundation Models. Recent 3D foundation models jointly predict camera parameters and scene geometry from images (Wang et al., 2026a; Lin et al., 2025; Wang et al., 2026b; Zhang et al., 2026a) in a single forward pass. Memory mechanisms extend reconstruction to long sequences (Wang et al., 2025b; Chen et al., 2026b; Zhang et al., 2026b), while large-scale pretraining further reduces trajectory drift (Chen et al., 2026a). Nevertheless, our experiments reveal remaining challenges in precise camera tracking and scenes dominated by moving foregrounds. Other approaches integrate feed-forward predictions with explicit geometric solvers (Maggio et al., 2025; Maggio & Carlone, 2026), but offer limited feedback to revise the underlying predictions. Concurrent methods instead explore learned transformers for geometric estimation (Yuan et al., 2026b; Taher et al., 2026).

3 Approach

3.1 Formulation: Deep visual SLAM

Given a streaming video $\{I_t\}_{t=0}^{N}$, when a new frame $\mathbf{I}_t \in \mathbb{R}^{H\times W}$ comes in, two state variables are registered: camera pose $\mathbf{G}_t \in SE(3)$ and disparity (inverse depth) $\mathbf{d}_t \in \mathbb{R}^{H\times W}$. Then, a co-visibility frame graph $(I_i, I_j) \in \mathcal{P}$ which is determined by overlap between frames is built dynamically both during training and inference, allowing the following differential bundle adjustment to update camera $\mathbf{G}_t$ and disparity $\mathbf{d}_t$.

Differential bundle adjustment. To handle real, unconstrained scenarios, we follow (Li et al., 2025b; Hagemann et al., 2023) to optimize camera poses, focal length and disparity by iteratively minimizing a weighted reprojection objective between network predicted correspondence and current ones derived from rigid motion of current camera parameters and disparity:

$$\mathcal{C}_{BA} = \mathcal{C}(\hat{\mathbf{G}}, \hat{\mathbf{d}}, \hat{f}) = \sum_{(i,j)\in\mathcal{P}} \left\| \hat{\mathbf{u}}_{ij} - \pi\left(\hat{\mathbf{G}}_{ij} \circ \pi^{-1}(\mathbf{p}_i, \hat{\mathbf{d}}_i, K^{-1}), K\right) \right\|^2_{\Sigma_{ij}}. \tag{1}$$

where $\Sigma_{ij} = \mathrm{diag}(\hat{\mathbf{w}}_{ij})^{-1}$ is the covariance matrix constructed from the predicted confidence weights $\hat{\mathbf{w}}_{ij}$, $\hat{\mathbf{u}}_{ij}$ denotes the network-predicted correspondences, and $\mathbf{p}_i$ denotes a grid of pixel coordinates. The operators π and π^{-1} denote perspective projection and back-projection, respectively. The relative camera pose between I_i and I_j is $\hat{\mathbf{G}}_{ij} = \hat{\mathbf{G}}_j \circ \hat{\mathbf{G}}_i^{-1}$, and $K \in \mathbb{R}^{3\times 3}$ denotes the camera intrinsic matrix.

Then, solving Eq. 1 can be achieved by Levenberg-Marquardt (LM) algorithm to jointly estimate the camera pose and disparity of each frame within the current active local window. Linearizing Eq. 1 leads to the following normal equation

$$\begin{bmatrix} \mathbf{H}_{G,f} & \mathbf{E}_{G,f} \\ \mathbf{E}_{G,f}^{\intercal} & \mathbf{H}_d \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\xi}_{G,f} \\ \Delta\mathbf{d} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{r}}_{G,f} \\ \tilde{\mathbf{r}}_d \end{bmatrix}. \tag{2}$$

where $\mathbf{H}_{G,f}$ and $\mathbf{H}_d$ are the Gauss–Newton Hessian blocks for camera poses and focal length and disparity, respectively. $\mathbf{E}_{G,f}$ is the cross-coupling matrix, $\Delta\boldsymbol{\xi}_{G,f}$ and $\Delta\mathbf{d}$ are the corresponding increments, and $\tilde{\mathbf{r}}_{G,f}$ and $\tilde{\mathbf{r}}_d$ are the negative gradients.

This formulation motivates our core design: **using a shared general 3D representation for correspondence prediction and prior integration**. Specifically, we predict correspondences $\hat{\mathbf{u}}_{ij}$ and confidence weights $\hat{\mathbf{w}}_{ij}$, while incorporating priors from the same representation into the corresponding camera parameter and inverse depth blocks of the normal equations.

3.2 WARPING BASED CORRESPONDENCE PREDICTION FROM GENERAL 3D PRIORS

The first bottleneck we identify is the correlation-volume-based updater Teed & Deng (2021). Constructing an explicit all-pairs correlation volume requires memory quadratic in the number of spatial locations in the matching feature maps (Wang & Deng, 2026), making correspondence estimation memory-intensive across multiple frame pairs. More importantly, this overhead competes with the backbone for GPU memory, restricting the practical use of richer multi-frame 3D representations and their associated priors. We therefore introduce a warping-based updater (Wang & Deng, 2026; Lai et al., 2026) that uses general 3D features for highly memory-efficient correspondence prediction.

Table 1: Peak GPU memory (GB) during inference on the TUM dataset at an input resolution of 384×512. This experiment is performed on a 48 GB GPU.

Updater	DA3-Single	Pi3X-Multi
Correlation	44.3	OOM
Warping	17.7	22.8

Input. Let $\mathbf{F}_i \in \mathbb{R}^{h\times w\times C}$ denote the feature map decoded from the general 3D backbone tokens for frame i using a trainable DPT (Ranftl et al., 2021). At iteration k, the current camera parameters and inverse depths induce geometric correspondences for each edge $(i,j) \in \mathcal{P}$: $\mathbf{u}_{ij}^{(k)} = \pi\left(\hat{\mathbf{G}}_{ij}^{(k)} \circ \pi^{-1}\left(\mathbf{p}_i, \hat{\mathbf{d}}_i^{(k)}, \left(K^{(k)}\right)^{-1}\right), K^{(k)}\right)$, where pixel coordinates and camera intrinsics are expressed at the feature-map resolution.

Warping. Then, inspired by (Wang & Deng, 2026), for each edge $(i,j) \in \mathcal{P}$, we directly align the target-frame features to the source-frame grid using these geometric correspondences:

$$\widetilde{\mathbf{F}}_{ij}^{(k)}(\mathbf{p}) = \mathrm{sample}\left(\mathbf{F}_j,\, \mathbf{u}_{ij}^{(k)}(\mathbf{p})\right), \tag{3}$$

where sample denotes bilinear sampling. Unlike an all-pairs correlation volume with $O((hw)^2)$ storage, the warped feature map requires $O(hwC)$ storage per edge. This reduction is particularly beneficial for dense co-visibility graphs or high-resolution feature maps.

Update. To handle large displacements, our warping-based updater uses attention to capture long-range dependencies. For each edge (i,j), we initialize a hidden state $\mathbf{h}_{ij}^{(0)} = \phi(\mathbf{F}_i \oplus \mathbf{F}_j)$, where $\oplus$ denotes channel-wise concatenation and ϕ is a linear projection. At iteration k, the updater input is $\mathbf{z}_{ij}^{(k)} = \widetilde{\mathbf{F}}_{ij}^{(k)} \oplus \mathbf{F}_i \oplus (\mathbf{u}_{ij}^{(k)} - \mathbf{p}_i) \oplus \mathbf{h}_{ij}^{(k)}$. Following (Lai et al., 2026), a video ViT (Arnab et al., 2021) aggregates spatial information within each edge and temporal information across edges sharing the same source frame, while updating the hidden states. Prediction heads produce correspondence corrections $\Delta\mathbf{u}_{ij}^{(k)}$ and confidence weights $\hat{\mathbf{w}}_{ij}^{(k)}$. The corrected correspondences $\hat{\mathbf{u}}_{ij}^{(k)} = \mathbf{u}_{ij}^{(k)} + \Delta\mathbf{u}_{ij}^{(k)}$ and weights are then passed to DBA to refine camera parameters and inverse depths, determining the warping coordinates for the next iteration. Architecture details are provided in Appendix B.

3.3 IN-THE-WILD SLAM WITH GENERAL 3D REPRESENTATIONS

A key advantage of BA is its flexibility to incorporate complementary observations and priors into a common geometric optimization framework. In total, our objective is formulated as:

$$\mathcal{C}_{\text{total}} = \mathcal{C}_{BA}(\hat{\mathbf{G}}, \hat{\mathbf{d}}, \hat{f}; \mathbf{m}) + \mathcal{C}_{\text{camera}} + \mathcal{C}_{\text{depth}}. \tag{4}$$

where $\mathcal{C}_{\text{camera}}$ and $\mathcal{C}_{\text{depth}}$ regularize camera parameters and inverse depths, respectively, and $\mathbf{m}$ denotes edge-level static reliability masks used to reweight correspondence residuals.

Uncertainty-aware pose prior regularization. Observing Eq. 2, The block structure allows camera priors to contribute directly to the camera-parameter block and its corresponding right-hand side. We combine pose constraints with optional focal-length regularization:

$$\mathcal{C}_{\text{camera}} = \mathcal{C}_{\text{pose}} + \lambda_f \mathcal{C}_f, \tag{5}$$

where $\mathcal{C}_f$ regularizes the optimized focal length toward the backbone prediction, with weight λ_f. This term is enabled only during frontend optimization in the uncalibrated setting. Its formulation and implementation details are provided in Appendix A.2. For the core pose term $\mathcal{C}_{\text{pose}}$, we consider a sparse relative-pose graph with edge set $\mathcal{E}_{\text{prior}}$, where each edge (i,j) carries a backbone-predicted

relative pose $\mathbf{G}_{ij}^{\text{prior}} = (\mathbf{R}_{ij}^{\text{prior}}, \mathbf{t}_{ij}^{\text{prior}})$. The corresponding relative pose in the current SLAM state is denoted by $\hat{\mathbf{G}}_{ij} = (\hat{\mathbf{R}}_{ij}, \hat{\mathbf{t}}_{ij})$. Since translation encodes both a dimensionless direction and a magnitude with units of length, we separate its regularization into direction and scale terms, allowing their contributions to be controlled independently. Let $\hat{\mathbf{v}}_{ij} = \hat{\mathbf{t}}_{ij}/\|\hat{\mathbf{t}}_{ij}\|_2$ and $\hat{s}_{ij} = \|\hat{\mathbf{t}}_{ij}\|_2$ denote the estimated translation direction and magnitude, with $\mathbf{v}_{ij}^{\text{prior}}$ and s_{ij}^{prior} defined analogously. The pose regularization term is defined as follows:

$$\mathcal{C}_{\text{pose}} = \frac{1}{2} \sum_{(i,j)\in\mathcal{E}_{\text{prior}}} \Big[\lambda_R \big\| \operatorname{Log}\big((\mathbf{R}_{ij}^{\text{prior}})^\top \hat{\mathbf{R}}_{ij}\big) \big\|_2^2 + g_j \lambda_t \big\| \hat{\mathbf{v}}_{ij} - \mathbf{v}_{ij}^{\text{prior}} \big\|_2^2 + g_j \lambda_s \big(\log \hat{s}_{ij} - \log s_{ij}^{\text{prior}} \big)^2 \Big]. \tag{6}$$

Here, $\operatorname{Log} : SO(3) \to \mathbb{R}^3$ is the rotation logarithm, $g_j \in \{0, 1\}$ gates the translation terms according to the observability criterion below, and λ_R, λ_t, and λ_s weight the rotation, translation-direction, and translation-scale priors, respectively. Hyperparameters and near-zero-baseline handling are detailed in Appendix D.1. A schematic of the prior graph is provided in Appendix C. In the normal equations Eq. 2, these factors contribute to the camera-parameter block and its corresponding right-hand side:

$$\begin{aligned} \mathbf{H}_{G,f} &\leftarrow \mathbf{H}_{G,f} + \mathbf{J}_{\text{pose}}^\top \mathbf{W}_{\text{pose}} \mathbf{J}_{\text{pose}}, \\ \tilde{\mathbf{r}}_{G,f} &\leftarrow \tilde{\mathbf{r}}_{G,f} - \mathbf{J}_{\text{pose}}^\top \mathbf{W}_{\text{pose}} \mathbf{e}_{\text{pose}}, \end{aligned} \tag{7}$$

where $\mathbf{e}_{\text{pose}}$ stacks the pose residuals, $\mathbf{J}_{\text{pose}}$ denotes their Jacobian with respect to the camera increments, and $\mathbf{W}_{\text{pose}}$ collects their weights, which are held fixed during each linearization. The inverse-depth and cross-coupling blocks remain unchanged. The residual Jacobians and their assembly into the normal equations are provided in Appendix A.

In practice, pose regularization improves robustness under limited translational parallax and static-background occlusion. However, the predicted pose priors can themselves be inaccurate. Because these factors directly constrain relative camera motion, assigning them excessive weight can bias pose estimates even when reliable visual constraints are available. Inspired by degeneracy analysis of classic state estimation problems (Zhang et al., 2016), we introduce an uncertainty gating g_j to adapt the strength of the translation priors to the reliability of the geometric estimates. Specifically, before adding the pose prior in Eq. 5, we partition the normal matrix into the current keyframe translation $\mathbf{t}_j$ and the remaining variables $\mathbf{q}$, including inverse depths, historical poses, and the current keyframe rotation. We use the Schur complement to obtain the reduced translation information matrix $\mathbf{S}_j = \mathbf{H}_{tt} - \mathbf{H}_{tq}\mathbf{H}_{qq}^{-1}\mathbf{H}_{qt}$ and assess translational observability through its smallest eigenvalue $\sigma_j = \lambda_{\min}(\mathbf{S}_j)$. We enable the translation-direction and translation-scale priors for edges targeting frame j when $\sigma_j < \tau_{\text{obs}}$ and disable them otherwise. Since $\mathbf{S}_j \in \mathbb{R}^{3\times3}$, its eigenvalue computation is inexpensive and suitable for online gating during tracking. Fig. 2 illustrates examples from KITTI (Geiger et al., 2012) and EPIC Fields (Tschernezki et al., 2023). Segments with limited translational parallax or static-background occlusion by moving objects exhibit lower values of the smallest eigenvalue, indicating weaker translational constraints. These observations support its use as an online observability indicator for activating pose priors when visual constraints are insufficient.

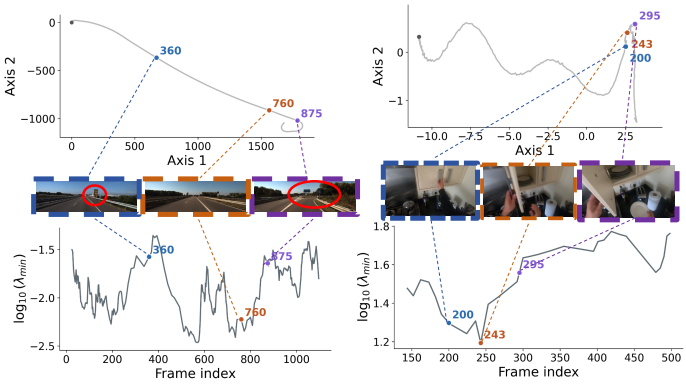


Figure 2: Translational observability with synchronized camera trajectories and RGB frames. Left: KITTI-01 contains forward-driving segments with distant scene structure and moving vehicles. Right: EPIC shows close-range cabinet interactions that occlude the static background. Higher values of $\lambda_{\min}$ indicate stronger local translational observability.

Depth prior regularization. In addition to pose constraints, we leverage the depth prior from the same representation to regularize inverse-depth estimates, particularly when limited parallax provides weak geometric constraints:

$$\mathcal{C}_{\text{depth}} = \lambda_{\text{depth}} \sum_i \left\| \hat{\mathbf{d}}_i - \mathbf{d}_i^{\text{prior}} \right\|^2 . \tag{8}$$

Here, $\mathbf{d}_i^{\text{prior}}$ is obtained by converting backbone-predicted metric depth to inverse depth at the optimization resolution for both single-frame and multi-frame configurations, and $\lambda_{\text{depth}} = 0.05$. Holding the prior fixed during each BA update, we modify the inverse-depth block of Eq. 2 as $\mathbf{H}_d \leftarrow \mathbf{H}_d + 2\lambda_{\text{depth}}\mathbf{I}$ and $\tilde{\mathbf{r}}_d \leftarrow \tilde{\mathbf{r}}_d + 2\lambda_{\text{depth}}(\mathbf{d}^{\text{prior}} - \hat{\mathbf{d}})$, leaving the camera and cross-coupling blocks unchanged.

Motion probability prediction. Finally, in dynamic scenes, independently moving objects induce image displacements that cannot be explained by camera motion alone. We therefore modulate the correspondence confidence weights $\hat{\mathbf{w}}_{ij}$ to reduce the influence of dynamic pixels on geometric estimation. To accommodate both single-frame and multi-frame backbones, we adopt a unified weighting formulation that incorporates the available motion cues:

$$\mathbf{m}_{ij} = \sigma\left(\mathbf{m}_i^{\text{base}} + \mathcal{R}_\theta\left(\mathbf{h}_{ij}, \{\mathbf{h}_{i\ell}\}_{\ell\in\mathcal{N}(i)}, \hat{\mathbf{w}}_{ij}, \mathbf{m}_i^{\text{base}}\right)\right), \qquad \widetilde{\mathbf{w}}_{ij} = \hat{\mathbf{w}}_{ij} \odot \mathbf{m}_{ij}. \tag{9}$$

where $\mathbf{m}_i^{\text{base}}$ denotes static-region logits predicted by a trainable segmentation head from the multi-frame representation, and $\mathcal{R}_\theta$ aggregates hidden states from edges sharing the same source frame to predict an edge-specific logit correction. For single-frame backbones, we replace $\mathbf{m}_i^{\text{base}}$ with a constant initialization. Implementation details are provided in Appendix B.

3.4 Training & Inference

Curriculum Learning. Our training pipeline contains three stage with diverse synthetic or real training dataset, including static TartanAir series (Wang et al., 2020), TartanGround (Patel et al., 2025), ScanNet++ (Yeshwanth et al., 2023), dynamic vKITTI2 (Cabon et al., 2020), HOI4D (Liu et al., 2022), Dynamic Replica (Karaev et al., 2023). In addition, we augment our own Kubric (Greff et al., 2022) data following the practice of (Li et al., 2025b; Zheng et al., 2026) to diversify the camera motion patterns. In the first stage, we only use static dataset to train the warping-based updater to learn correspondence induced purely by camera ego-motion. Following Teed & Deng (2021), we optimize a weighted objective $\mathcal{L}_{\text{SLAM}} = \lambda_{\text{cam}}\mathcal{L}_{\text{cam}} + \lambda_{\text{flow}}\mathcal{L}_{\text{flow}} + \lambda_{\text{res}}\mathcal{L}_{\text{res}}$, where $\mathcal{L}_{\text{cam}}$ supervises camera poses, $\mathcal{L}_{\text{flow}}$ compares geometry-induced flow with ground-truth flow, and $\mathcal{L}_{\text{res}}$ penalizes the discrepancy between geometry-induced and updater-predicted flow. In the second stage, we fine-tune the updater on a mixture of static and dynamic datasets using the same objective. Ground-truth motion masks down-weight dynamic pixels in BA, allowing the updater to adapt to dynamic scenes. In the third stage, we freeze the updater and train the motion prediction modules by replacing ground-truth masks in DBA with predicted masks. We optimize $\mathcal{L}_{\text{motion}} = \lambda_{\text{cam}}\mathcal{L}_{\text{cam}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}$, where $\mathcal{L}_{\text{mask}}$ supervises the predicted masks using ground-truth annotations. Training implementation details are presented in supplementary materials

Online inference. Following (Teed & Deng, 2021), we select keyframes based on correspondence-derived displacement relative to the previous keyframe and remove historical keyframes with insufficient co-visibility from the active window. All results are obtained **without loop-closure detection or hierarchical global optimization**. In the uncalibrated setting, camera intrinsics are initialized from the backbone prediction for the first frame. For each new keyframe, multi-frame backbones use the four preceding keyframes as context, and pose-prior edges connect the new keyframe to the most recent and oldest context keyframes.

4 Experiments

4.1 Experimental Setup

Baselines. Table 2 compares chunk-based reconstruction methods using bidirectional context with frame-based systems under calibrated and uncalibrated settings. The latter include memory-based

Table 2: Camera pose estimation measured by ATE (↓). Best and second-best results within each group are shown in bold and underlined, respectively. TL and OOM denote tracking loss and out-of-memory on a 48 GB GPU, respectively.

Setting	Method	Calib. Free	Static			Dynamic		
			TUM 307–710 fr. 2.2–17.6 m	7-Scenes 500–1,000 fr. 3.2–17.3 m	ScanNet 108–500 fr. 3.4–23.7 m	Bonn 439–937 fr. 1.7–5.6 m	KITTI (m) 271–4,661 fr. 393.6–5,067.2 m	EPIC Fields 545–1,000 fr. 19.3–182.8 m
Chunk	DAv3-Streaming (Lin et al., 2025)	✓	4.7	8.7	8.6	4.9	32.8	48.1
	AMB3R (Wang & Agapito, 2026)	✓	**3.2**	**7.3**	**6.1**	**4.1**	104.0	**37.7**
	Scal3R (Xie et al., 2026)	✓	7.4	10.0	7.6	11.3	**14.6**	154.4
	LoGeR (Zhang et al., 2026b)	✓	3.3	8.4	8.7	5.5	25.4	72.2
Frame	DROID-SLAM (Teed & Deng, 2021)	✗	3.8	8.0	13.8	4.9	100.3	11.1
	MASt3R-SLAM (Murai et al., 2025)	✗	3.0	7.0	6.2	4.0	185.5	TL
	Pi3MOS (Zhong et al., 2026)	✗	8.1	7.2	8.6	**2.2**	40.4	27.1
	DROID-W (Li et al., 2026)	✗	3.3	6.8	6.4	2.3	38.3	17.7
	WildPose (Zheng et al., 2026)	✗	**2.7**	7.0	6.2	2.4	30.4	12.2
	Ours (DAv3, Single)	✗	**2.7**	6.7	**5.9**	2.7	**19.1**	16.2
	Ours (Pi3X)	✗	**2.7**	**6.5**	6.0	**2.2**	29.9	**9.3**
	CUT3R (Wang et al., 2025b)	✓	25.2	64.2	68.8	28.7	91.6	282.8
	TTT3R (Chen et al., 2026b)	✓	14.0	39.3	27.2	22.6	72.9	257.0
	InfiniteVGGT (Yuan et al., 2026a)	✓	22.8	22.8	33.7	12.5	183.9	150.0
	LongStream (Cheng et al., 2026)	✓	7.6	15.0	13.5	7.5	51.9	193.3
	Lingbot-map (Chen et al., 2026a)	✓	4.4	6.7	8.1	4.5	28.3	44.8
	KV-Tracker (Taher et al., 2026)	✓	10.8	8.4	24.4	6.9	202.3	93.8
	SLAM-Former (Yuan et al., 2026b)	✓	3.9	**6.3**	7.1	3.4	OOM	22.2
	MegaSaM (Li et al., 2025b)	✓	6.6	7.3	9.1	3.2	OOM	OOM
	ViPE (Huang et al., 2025)	✓	5.9	6.7	7.0	2.7	32.0	13.9
	VGGT-SLAM 2.0 (Maggio & Carlone, 2026)	✓	4.1	7.2	6.6	4.0	54.5	60.2
	Ours (MoGe)	✓	3.8	6.7	6.7	3.6	31.6	18.1
	Ours (DAv3, Multi)	✓	**3.6**	**6.3**	**6.3**	**2.3**	**20.6**	**10.6**

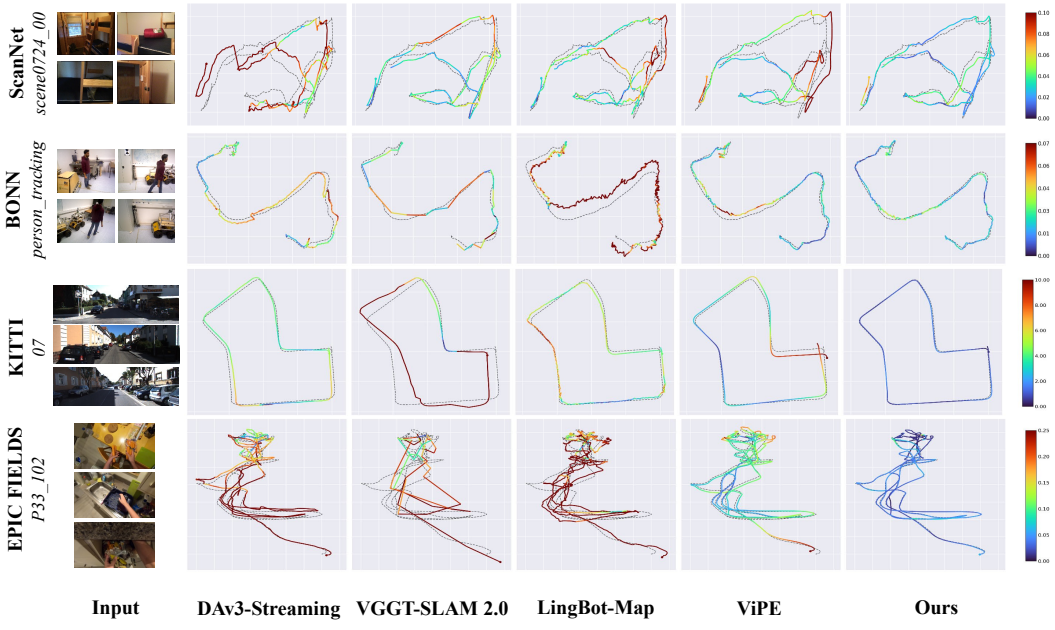


Figure 3: Qualitative camera trajectory comparisons on ScanNet, Bonn, KITTI, and EPIC Fields. Estimated trajectories are colored by positional error, with warmer colors indicating larger errors.

foundation models, transformers for geometric estimation, and integrated learning-based SLAM systems. Calibrated methods receive known camera intrinsics. To evaluate compatibility with diverse representations, we train four variants using single-frame DAv3 and MoGe (Wang et al., 2025c), and multi-frame Pi3X and DAv3 backbones.

Datasets & Metrics. We follow established SLAM protocols for TUM RGB-D and Bonn (Li et al., 2026; 2025b; Zhong et al., 2026; Teed & Deng, 2021), and foundation-model protocols for 7-Scenes and ScanNet (Chen et al., 2026b; Zhang et al., 2026b), as well as KITTI. We additionally evaluate on

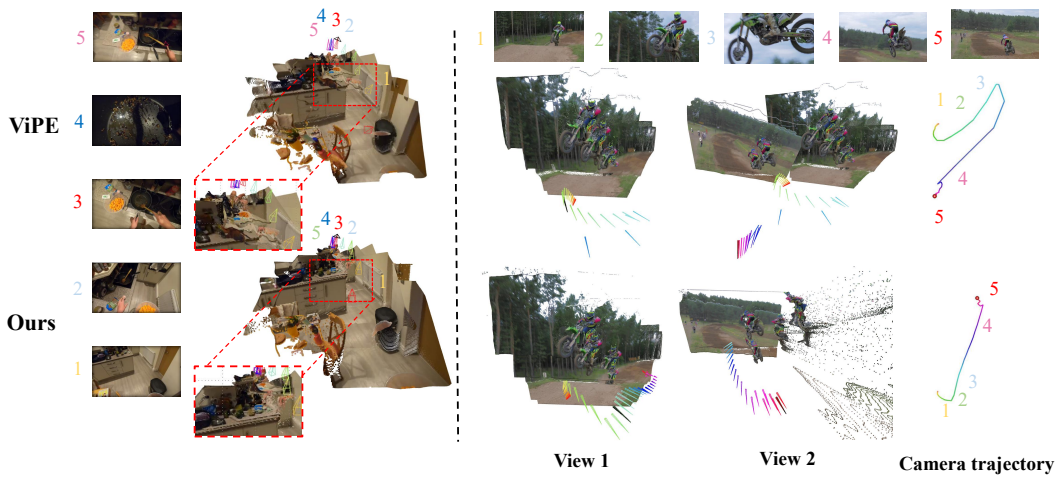


Figure 4: Qualitative reconstruction comparisons on EPIC Fields and DAVIS (Perazzi et al., 2016) under challenging visual conditions. Left: close-range object manipulation severely occludes the static background; enlarged regions highlight differences in reconstruction consistency. Right: a moving motorcycle dominates the foreground, with reconstructions shown from two viewpoints alongside the estimated camera trajectories. Colored numbers associate selected input frames with their camera poses.

EPIC Fields, covering indoor, outdoor, dynamic, and egocentric scenarios across six benchmarks. Published results are reused only where evaluation settings match; sequence selections and detailed protocols are provided in Appendix D.5. We evaluate camera trajectories using ATE (↓) and depth using AbsRel (↓) and δ_1 (↑) after scale-and-shift alignment to ground truth.

4.2 Results

Camera pose estimation. Table 2 and Fig. 3 show that Any3D-SLAM achieves competitive accuracy across diverse environments and calibration settings. Compared with calibrated SLAM baselines, our calibrated variants match or improve upon the strongest results on all three static benchmarks, while achieving lower ATE on KITTI and EPIC Fields. Compared with feed-forward foundation-model pipelines, our results support combining general 3D representations with iterative geometric optimization to improve trajectory accuracy. In particular, our uncalibrated DAv3-Multi variant achieves the lowest or joint-lowest ATE among uncalibrated frame-based methods on all six benchmarks and remains competitive with chunk-based approaches. Its advantage over memory-based models on EPIC Fields highlights robustness to challenging egocentric motion and dynamic foregrounds. Across backbones, the variants achieve comparable accuracy on static indoor benchmarks, supporting the applicability of our framework to diverse representations. Multi-frame variants show stronger results on Bonn and EPIC Fields, consistent with the benefits of their complementary temporal and geometric priors.

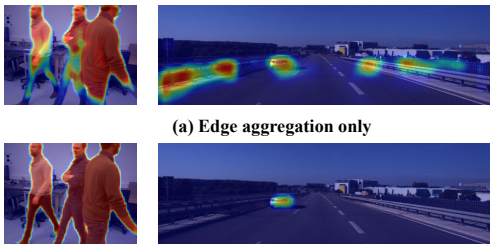
Downstream application: video depth estimation. Following the depth refinement pipeline of ViPE, we use the off-the-shelf PriorDA model (Wang et al., 2025d) to fuse BA-maintained low-resolution depth with backbone depth priors and RGB images, producing full-resolution depth maps. Table 3 reports metrics computed over complete sequences. Our method achieves the lowest AbsRel on both KITTI and Bonn, while improving δ_1 on KITTI and matching ViPE on Bonn. The gains over attention-based video depth estimation methods support the benefit of explicitly maintaining and refining geometric states over time. Comparisons in Fig. 4 further illustrate how robust camera tracking supports coherent scene reconstruction under severe occlusions, including close-range object interactions and large moving foregrounds.

Table 3: Full-resolution depth estimation on KITTI and Bonn. Best results are shown in **bold**.

Method	KITTI 271–4,661 fr.		Bonn 439–937 fr.	
	AbsRel ↓	δ_1 ↑	AbsRel ↓	δ_1 ↑
DAv3 (Lin et al., 2025)	0.235	59.6	0.122	87.2
Video Depth Anything (Chen et al., 2025a)	0.220	62.4	0.092	90.2
GemDepth (Liu et al., 2026)	Did not converge		0.088	90.6
ViPE (Huang et al., 2025)	0.144	81.2	0.072	**96.3**
Ours	**0.136**	**82.0**	**0.066**	**96.3**

Table 5: Ablation of motion, intrinsics, and pose priors measured by ATE (↓).

Component	Variant	TUM	KITTI	Bonn
Motion	(a) w/o motion mask	3.6	21.7	2.5
	(b) Edge aggregation only	3.6	20.9	2.4
	(c) Full	3.6	20.6	2.3
Intrinsics	(d) Fixed intrinsics	32.9	36.8	6.4
	(e) Global-only optimization	3.6	23.1	2.3
	(f) w/o prior regularization	3.6	TL	2.3
		TUM	**KITTI**	**EPIC Fields**
Pose prior	(g) w/o pose prior	3.6	27.3	16.8
	(h) w/o uncertainty gating	3.7	TL	10.6
	(c) Full	3.6	20.6	10.6



(a) Edge aggregation only

(b) Full

Figure 5: Motion probability visualization of ablation on multi-frame motion prior.

4.3 Ablation Study

Correspondence Updater and Training Strategy. Table 4 shows that warping maintains comparable pose accuracy while reducing peak training memory from 33.8 to 21.5 GB with DA3-Single. It also enables inference with Pi3X-Multi, for which all-pairs correlation runs out of memory. The necessity of second-stage training varies across backbones: DA3-Single already performs well after Stage I, whereas Stage II reduces Pi3X-Multi's KITTI ATE. We hypothesize that this difference reflects driving-scene coverage during pretraining, with Pi3X potentially lacking such data and DA3 exhibiting stronger transferability to this domain.

Table 4: Ablation of the correspondence updater and two-stage training. Camera pose accuracy is measured by ATE.

Backbone	Correspondence	Training		TUM	7-Scenes	KITTI	Peak Train Memory (GB)
		Stage I	Stage II				
DA3-Single	Correlation	✓	✗	2.6	6.8	20.8	33.8
	Warping	✓	✗	2.7	6.7	19.1	21.5
Pi3X-Multi	Correlation	–	–		OOM		–
	Warping	✓	✗	2.7	7.1	44.4	25.1
	Warping	✓	✓	2.7	6.5	29.9	25.1

Prior Integration. Table 5 evaluates the contributions of motion handling, intrinsics optimization, and pose priors. Multi-frame motion prediction reduces over- and under-segmentation compared with edge aggregation alone (Fig. 5), improving accuracy on KITTI and Bonn while preserving performance on static TUM sequences. Fixing intrinsics to backbone predictions substantially degrades accuracy, highlighting the importance of calibration for reprojection-based geometric estimation. Global optimization recovers much of this accuracy, while removing frontend intrinsics regularization also leads to failure. Pose priors reduce ATE from 27.3 to 20.6 on KITTI and from 16.8 to 10.6 on EPIC Fields, supporting camera estimation when reliable visual constraints are scarce. Uncertainty gating avoids imposing translation priors when geometric constraints are sufficient, preventing the KITTI failure observed with ungated priors.

5 Conclusion

We presented Any3D-SLAM, a framework that integrates diverse general 3D representations into visual SLAM, combining their generalization capabilities with the geometric accuracy of explicit optimization. A memory-efficient warping-based updater enables multi-frame representations to support both correspondence estimation and complementary geometric and motion priors. Uncertainty-gated pose regularization improves robustness under weak visual constraints and severe occlusions, while feeding optimized geometric states into conditioned depth estimation extends these benefits to full-resolution reconstruction. Together, these results demonstrate a flexible approach to coupling general 3D representations with geometric optimization for in-the-wild SLAM.

Limitations. First, single-frame variants run at approximately 4-5 FPS, while multi-frame variants achieve only 1–2 FPS due to repeated context extraction and dense DBA. Memory-based backbones and patch-based optimization could improve efficiency. Second, the backbone remains frozen, and geometric feedback is limited to an off-the-shelf conditioned depth model. Jointly refining representations and geometric states remains an important direction for future work.

DISCLOSURE OF LLM USE

We used generative AI tools to assist with manuscript writing and editing, code revision, and technical discussions. The authors reviewed the AI-assisted content and take responsibility for the final work.

REPRODUCIBILITY STATEMENT

We document the training hyperparameters, curriculum learning strategy, synthetic data generation, and training dataset mixtures in the paper and supplementary material. We will release our code and trained models upon acceptance. We also specify the evaluation protocols adopted from prior work, including sequence selection, frame sampling, and metric computation, with differences from published baseline settings noted where applicable.

REFERENCES

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *ICCV*, 2021.

Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020.

Lin-Zhuo Chen, Jian Gao, Yihang Chen, Ka Leong Cheng, Yipengjing Sun, Liangxiao Hu, Nan Xue, Xing Zhu, Yujun Shen, Yao Yao, et al. Geometric context transformer for streaming 3d reconstruction. *arXiv preprint arXiv:2604.14141*, 2026a.

Sili Chen, Hengkai Guo, Shengnan Zhu, Feihu Zhang, Zilong Huang, Jiashi Feng, and Bingyi Kang. Video depth anything: Consistent depth estimation for super-long videos. In *CVPR*, 2025a.

Weirong Chen, Ganlin Zhang, Felix Wimbauer, Rui Wang, Nikita Araslanov, Andrea Vedaldi, and Daniel Cremers. Back on track: Bundle adjustment for dynamic scene reconstruction. In *ICCV*, 2025b.

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

Chong Cheng, Xianda Chen, Tao Xie, Wei Yin, Weiqiang Ren, Qian Zhang, Xiaoyang Guo, and Hao Wang. Longstream: Long-sequence streaming autoregressive visual geometry. *arXiv preprint arXiv:2602.13172*, 2026.

Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *TPAMI*, 2017.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012.

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.

Annika Hagemann, Moritz Knorr, and Christoph Stiller. Deep geometry-aware camera self-calibration from video. In *ICCV*, 2023.

Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, et al. Vipe: Video pose engine for 3d geometric perception. *arXiv preprint arXiv:2508.10934*, 2025.

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

Zihang Lai, Eldar Insafutdinov, Edgar Sucar, and Andrea Vedaldi. Cowtracker: Tracking by warping instead of correlation. *arXiv preprint arXiv:2602.04877*, 2026.

Moyang Li, Zihan Zhu, Marc Pollefeys, and Daniel Barath. Droid-slam in the wild. In *CVPR*, 2026.

Yanyan Li, Youxu Fang, Zunjie Zhu, Kunyi Li, Yong Ding, and Federico Tombari. 4d gaussian splatting slam. In *ICCV*, 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. In *CVPR*, 2025b.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Yuecheng Liu, Junda Cheng, Longliang Liu, Wenjing Liao, Hanrui Cheng, Yuzhou Wang, and Xin Yang. Gemdepth: Geometry-embedded features for 3d-consistent video depth. *arXiv preprint arXiv:2605.10525*, 2026.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *CVPR*, 2022.

Dominic Maggio and Luca Carlone. Vggt-slam 2.0: Real-time dense feed-forward scene reconstruction. *arXiv preprint arXiv:2601.19887*, 2026.

Dominic Maggio, Hyungtae Lim, and Luca Carlone. Vggt-slam: Dense rgb slam optimized on the sl (4) manifold. *NeurIPS*, 2025.

Avilasha Mandal, Rajesh Kumar, Sudarshan Sunil Harithas, and Chetan Arora. Vggt-slam++. In *CVPR*, 2026.

Riku Murai, Eric Dexheimer, and Andrew J Davison. Mast3r-slam: Real-time dense slam with 3d reconstruction priors. In *CVPR*, 2025.

Manthan Patel, Fan Yang, Yuheng Qiu, Cesar Cadena, Sebastian Scherer, Marco Hutter, and Wenshan Wang. Tartanground: A large-scale dataset for ground robot perception and navigation. In *IROS*, 2025.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016.

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

Zhicong Sun, Jacqueline Lo, and Jinxing Hu. Embracing dynamics: Dynamics-aware 4d gaussian splatting slam. In *IROS*, 2025.

Marwan Taher, Ignacio Alzugaray, Kirill Mazur, Xin Kong, and Andrew Davison. Kv-tracker: Real-time pose tracking with transformers. In *CVPR*, 2026.

Zachary Teed and Jia Deng. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. *NeurIPS*, 2021.

Zachary Teed, Lahav Lipson, and Jia Deng. Deep patch visual odometry. *NeurIPS*, 2023.

Vadim Tschernezki, Ahmad Darkhalil, Zhifan Zhu, David Fouhey, Iro Laina, Diane Larlus, Dima Damen, and Andrea Vedaldi. Epic fields: Marrying 3d geometry and video understanding. *NeurIPS*, 2023.

Hengyi Wang and Lourdes Agapito. Amb3r: Accurate feed-forward metric-scale 3d reconstruction with backend. In *CVPR*, 2026.

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-$omega$. *arXiv preprint arXiv:2605.15195*, 2026a.

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

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *NeurIPS*, 2025c.

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. pi^3: Permutation-equivariant visual geometry learning. In *ICLR*, 2026b.

Yihan Wang and Jia Deng. Waft: Warping-alone field transforms for optical flow. In *ICLR*, 2026.

Yunsong Wang and Gim Hee Lee. Flow4dgs-slam: Optical flow-guided 4d gaussian splatting slam. In *CVPR*, 2026.

Zehan Wang, Siyu Chen, Lihe Yang, Jialei Wang, Ziang Zhang, Hengshuang Zhao, and Zhou Zhao. Depth anything with any prior. *arXiv preprint arXiv:2505.10565*, 2025d.

Yuchen Wu, Jiahe Li, Fabio Tosi, Matteo Poggi, Jin Zheng, and Xiao Bai. Foundationslam: Unleashing the power of depth foundation models for end-to-end dense visual slam. In *AAAI*, 2026.

Tao Xie, Peishan Yang, Yudong Jin, Yingfeng Cai, Wei Yin, Weiqiang Ren, Qian Zhang, Wei Hua, Sida Peng, Xiaoyang Guo, et al. Scal3r: Scalable test-time training for large-scale 3d reconstruction. *arXiv preprint arXiv:2604.08542*, 2026.

Nan Yang, Lukas von Stumberg, Rui Wang, and Daniel Cremers. D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry. In *CVPR*, 2020.

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

Shuai Yuan, Yantai Yang, Xiaotian Yang, Xupeng Zhang, Zhonghao Zhao, Lingming Zhang, and Zhipeng Zhang. Infinitevggt: Visual geometry grounded transformer for endless streams. *arXiv preprint arXiv:2601.02281*, 2026a.

Yijun Yuan, Zhuoguang Chen, Kenan Li, Weibang Wang, Minghui Qin, Zhijian Fang, Weicheng Zheng, and Hang Zhao. Slam-former: Putting slam into one transformer. In *ECCV*, 2026b.

Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K Barral, Raia Hadsell, et al. Efficiently reconstructing dynamic scenes one d4rt at a time. In *CVPR*, 2026a.

Ji Zhang, Michael Kaess, and Sanjiv Singh. On degeneracy of optimization-based state estimation problems. In *ICRA*, 2016.

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 preprint arXiv:2603.03269*, 2026b.

Jianhao Zheng, Liyuan Zhu, Zihan Zhu, and Iro Armeni. Wildpose: A unified framework for robust pose estimation in the wild. In *CVPR*, 2026.

Xingguang Zhong, Liren Jin, Marija Popovic, Jens Behley, and Cyrill Stachniss. Dynamic visual slam using a general 3d prior. In *CVPR*, 2026.

A Derivation

This appendix provides detailed derivations for the camera-prior terms introduced in Sec. 3.3. Although the pose and focal priors operate in different parameter spaces, they follow the same principle: we perturb the current estimate in its local tangent space, linearize the corresponding residuals, and incorporate their Jacobians into the BA normal equations.

A.1 Jacobians of Pose-Prior Factors

Perturbation convention. We represent the pose of camera k by the world-to-camera transformation $\hat{\mathbf{G}}_k \in SE(3)$, such that

$$\hat{\mathbf{G}}_k = \begin{bmatrix} \hat{\mathbf{R}}_k & \hat{\mathbf{t}}_k \\ \mathbf{0}^\top & 1 \end{bmatrix}.$$

All pose-prior Jacobians are derived under a left-multiplicative $SE(3)$ perturbation:

$$\mathbf{G}_k(\delta\boldsymbol{\xi}_k) = \operatorname{Exp}(\delta\boldsymbol{\xi}_k^\wedge)\,\hat{\mathbf{G}}_k, \qquad \delta\boldsymbol{\xi}_k = \begin{bmatrix} \delta\boldsymbol{\rho}_k \\ \delta\boldsymbol{\phi}_k \end{bmatrix} \in \mathbb{R}^6, \tag{10}$$

where $\delta\boldsymbol{\rho}_k$ and $\delta\boldsymbol{\phi}_k$ are the translation and rotation increments, respectively, and $(\cdot)^\wedge$ denotes the $se(3)$ hat operator.

For an edge (i, j), the relative transformation maps coordinates from camera i to camera j:

$$\hat{\mathbf{G}}_{ij} = \hat{\mathbf{G}}_j \hat{\mathbf{G}}_i^{-1} = \begin{bmatrix} \hat{\mathbf{R}}_j \hat{\mathbf{R}}_i^\top & \hat{\mathbf{t}}_j - \hat{\mathbf{R}}_j \hat{\mathbf{R}}_i^\top \hat{\mathbf{t}}_i \\ \mathbf{0}^\top & 1 \end{bmatrix}. \tag{11}$$

Under the above perturbation, it becomes

$$\mathbf{G}_{ij}(\delta\boldsymbol{\xi}_i, \delta\boldsymbol{\xi}_j) = \operatorname{Exp}(\delta\boldsymbol{\xi}_j^\wedge)\hat{\mathbf{G}}_{ij}\operatorname{Exp}(-\delta\boldsymbol{\xi}_i^\wedge). \tag{12}$$

All Jacobians below are evaluated at $\delta\boldsymbol{\xi}_i = \delta\boldsymbol{\xi}_j = \mathbf{0}$.

Rotation prior Jacobian. For each prior edge (i, j), we define the rotation residual as

$$\mathbf{e}_R^{ij} = \operatorname{Log}\left((\mathbf{R}_{ij}^{\text{prior}})^\top \hat{\mathbf{R}}_{ij}\right) \in \mathbb{R}^3, \qquad \mathcal{C}_R^{ij} = \frac{1}{2}\left\|\mathbf{e}_R^{ij}\right\|_2^2. \tag{13}$$

For brevity, let

$$\mathbf{E}_{ij} = (\mathbf{R}_{ij}^{\text{prior}})^\top \hat{\mathbf{R}}_{ij}, \qquad \mathbf{e}_R^{ij} = \operatorname{Log}(\mathbf{E}_{ij}). \tag{14}$$

Under the left-multiplicative perturbation in Eq. equation 12, the source-frame and target-frame rotation increments respectively perturb the rotation error as

$$\mathbf{E}_{ij}(\delta\boldsymbol{\phi}_i) = \mathbf{E}_{ij}\operatorname{Exp}(-\delta\boldsymbol{\phi}_i^\wedge), \tag{15}$$

$$\mathbf{E}_{ij}(\delta\boldsymbol{\phi}_j) = \operatorname{Exp}\left(\left((\mathbf{R}_{ij}^{\text{prior}})^\top \delta\boldsymbol{\phi}_j\right)^\wedge\right)\mathbf{E}_{ij}. \tag{16}$$

Applying the first-order differential of the $SO(3)$ logarithm gives

$$\mathbf{e}_R^{ij}(\delta) \approx \mathbf{e}_R^{ij} - \mathbf{J}_r^{-1}(\mathbf{e}_R^{ij})\delta\boldsymbol{\phi}_i + \mathbf{J}_l^{-1}(\mathbf{e}_R^{ij})(\mathbf{R}_{ij}^{\text{prior}})^\top \delta\boldsymbol{\phi}_j, \tag{17}$$

where $\mathbf{J}_l^{-1}$ and $\mathbf{J}_r^{-1}$ denote the inverse left and right Jacobians of $SO(3)$. Therefore, for the pose increment ordering $\delta\boldsymbol{\xi} = [\delta\boldsymbol{\rho}^\top, \delta\boldsymbol{\phi}^\top]^\top$, the Jacobians are

$$\mathbf{J}_{R,i} = \begin{bmatrix} \mathbf{0}_{3\times 3} & -\mathbf{J}_r^{-1}(\mathbf{e}_R^{ij}) \end{bmatrix}, \qquad \mathbf{J}_{R,j} = \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{J}_l^{-1}(\mathbf{e}_R^{ij})(\mathbf{R}_{ij}^{\text{prior}})^\top \end{bmatrix}. \tag{18}$$

Let $\boldsymbol{\Phi} = (\mathbf{e}_R^{ij})^\wedge$ and $\theta = \|\mathbf{e}_R^{ij}\|_2$. The inverse $SO(3)$ Jacobians are

$$\mathbf{J}_l^{-1} = \mathbf{I} - \frac{1}{2}\boldsymbol{\Phi} + a(\theta)\boldsymbol{\Phi}^2, \qquad \mathbf{J}_r^{-1} = \mathbf{I} + \frac{1}{2}\boldsymbol{\Phi} + a(\theta)\boldsymbol{\Phi}^2, \tag{19}$$

where

$$a(\theta) = \frac{1}{\theta^2} - \frac{1+\cos\theta}{2\theta\sin\theta}, \qquad \lim_{\theta\to 0} a(\theta) = \frac{1}{12}. \tag{20}$$

Translation-direction prior Jacobian. The translation-direction residual compares the normalized current and prior relative translations:

$$\mathbf{e}_t^{ij} = \hat{\mathbf{v}}_{ij} - \mathbf{v}_{ij}^{\text{prior}}, \qquad \hat{\mathbf{v}}_{ij} = \frac{\hat{\mathbf{t}}_{ij}}{\|\hat{\mathbf{t}}_{ij}\|_2}, \qquad \mathbf{v}_{ij}^{\text{prior}} = \frac{\mathbf{t}_{ij}^{\text{prior}}}{\|\mathbf{t}_{ij}^{\text{prior}}\|_2}, \tag{21}$$

such that $\mathcal{C}_t^{ij} = \frac{1}{2}\|\mathbf{e}_t^{ij}\|_2^2$. This residual is evaluated only when both translation magnitudes exceed the corresponding minimum-baseline thresholds.

For brevity, let

$$\hat{s}_{ij} = \|\hat{\mathbf{t}}_{ij}\|_2, \qquad \mathbf{N}_{ij} = \frac{\mathbf{I} - \hat{\mathbf{v}}_{ij}\hat{\mathbf{v}}_{ij}^\top}{\hat{s}_{ij}}, \tag{22}$$

where $\mathbf{N}_{ij}$ is the Jacobian of vector normalization. Under the left-multiplicative perturbation, the relative translation changes as

$$\hat{\mathbf{t}}_{ij}(\delta) \approx \hat{\mathbf{t}}_{ij} - \hat{\mathbf{R}}_{ij}\delta\boldsymbol{\rho}_i + \delta\boldsymbol{\rho}_j - [\hat{\mathbf{t}}_{ij}]_\times \delta\boldsymbol{\phi}_j. \tag{23}$$

Therefore,

$$\delta\hat{\mathbf{v}}_{ij} = \mathbf{N}_{ij}\delta\hat{\mathbf{t}}_{ij}, \tag{24}$$

and the direction residual linearizes as

$$\mathbf{e}_t^{ij}(\delta) \approx \mathbf{e}_t^{ij} + \mathbf{J}_{t,i}\delta\boldsymbol{\xi}_i + \mathbf{J}_{t,j}\delta\boldsymbol{\xi}_j, \tag{25}$$

with

$$\mathbf{J}_{t,i} = \begin{bmatrix} -\mathbf{N}_{ij}\hat{\mathbf{R}}_{ij} & \mathbf{0}_{3\times 3} \end{bmatrix}, \qquad \mathbf{J}_{t,j} = \begin{bmatrix} \mathbf{N}_{ij} & -\mathbf{N}_{ij}[\hat{\mathbf{t}}_{ij}]_\times \end{bmatrix}. \tag{26}$$

Translation-scale prior Jacobian. The translation-scale residual measures the discrepancy between the current and backbone-predicted relative baseline magnitudes in logarithmic space:

$$e_s^{ij} = \log\|\hat{\mathbf{t}}_{ij}\|_2 - \log\|\mathbf{t}_{ij}^{\text{prior}}\|_2, \qquad \mathcal{C}_s^{ij} = \frac{1}{2}\left(e_s^{ij}\right)^2. \tag{27}$$

The residual is evaluated only when both translation magnitudes exceed the scale minimum-baseline threshold. Let

$$\hat{s}_{ij} = \|\hat{\mathbf{t}}_{ij}\|_2. \tag{28}$$

The differential of the logarithmic baseline magnitude is

$$\delta\log\hat{s}_{ij} = \frac{\hat{\mathbf{t}}_{ij}^\top}{\hat{s}_{ij}^2}\delta\hat{\mathbf{t}}_{ij}. \tag{29}$$

Substituting the perturbed relative translation yields

$$e_s^{ij}(\delta) \approx e_s^{ij} + \mathbf{J}_{s,i}\delta\boldsymbol{\xi}_i + \mathbf{J}_{s,j}\delta\boldsymbol{\xi}_j, \tag{30}$$

where

$$\mathbf{J}_{s,i} = \begin{bmatrix} -\dfrac{\hat{\mathbf{t}}_{ij}^\top\hat{\mathbf{R}}_{ij}}{\hat{s}_{ij}^2} & \mathbf{0}_{1\times 3} \end{bmatrix}, \qquad \mathbf{J}_{s,j} = \begin{bmatrix} \dfrac{\hat{\mathbf{t}}_{ij}^\top}{\hat{s}_{ij}^2} & \mathbf{0}_{1\times 3} \end{bmatrix}. \tag{31}$$

Normal-equation assembly. For each pose-prior edge (i, j), we stack the rotation, translation-direction, and translation-scale residuals as

$$\mathbf{e}_{ij} = \begin{bmatrix} \mathbf{e}_R^{ij} \\ \mathbf{e}_t^{ij} \\ e_s^{ij} \end{bmatrix}, \qquad \mathbf{J}_i = \begin{bmatrix} \mathbf{J}_{R,i} \\ \mathbf{J}_{t,i} \\ \mathbf{J}_{s,i} \end{bmatrix}, \qquad \mathbf{J}_j = \begin{bmatrix} \mathbf{J}_{R,j} \\ \mathbf{J}_{t,j} \\ \mathbf{J}_{s,j} \end{bmatrix}. \tag{32}$$

The corresponding block-diagonal information matrix is

$$\mathbf{W}_{ij} = \operatorname{diag}\left(\lambda_R\mathbf{I}_3,\, g_j\lambda_t\mathbf{I}_3,\, g_j\lambda_s\right). \tag{33}$$

Thus, the first-order approximation of the edge cost is

$$\mathcal{C}_{\text{pose}}^{ij}(\delta) \approx \frac{1}{2}\left\|\mathbf{W}_{ij}^{1/2}\left(\mathbf{e}_{ij} + \mathbf{J}_i\delta\boldsymbol{\xi}_i + \mathbf{J}_j\delta\boldsymbol{\xi}_j\right)\right\|_2^2. \tag{34}$$

Each edge contributes the following blocks to the camera normal equations:

$$\begin{aligned} &\mathbf{H}_{ii} \leftarrow \mathbf{H}_{ii} + \mathbf{J}_i^\top\mathbf{W}_{ij}\mathbf{J}_i, \quad \mathbf{H}_{ij} \leftarrow \mathbf{H}_{ij} + \mathbf{J}_i^\top\mathbf{W}_{ij}\mathbf{J}_j, \\ &\mathbf{H}_{ji} \leftarrow \mathbf{H}_{ji} + \mathbf{J}_j^\top\mathbf{W}_{ij}\mathbf{J}_i, \quad \mathbf{H}_{jj} \leftarrow \mathbf{H}_{jj} + \mathbf{J}_j^\top\mathbf{W}_{ij}\mathbf{J}_j, \\ &\tilde{\mathbf{r}}_i \leftarrow \tilde{\mathbf{r}}_i - \mathbf{J}_i^\top\mathbf{W}_{ij}\mathbf{e}_{ij}, \qquad \tilde{\mathbf{r}}_j \leftarrow \tilde{\mathbf{r}}_j - \mathbf{J}_j^\top\mathbf{W}_{ij}\mathbf{e}_{ij}. \end{aligned} \tag{35}$$

A.2 JACOBIANS OF FOCAL-LENGTH PRIOR

In the uncalibrated setting, we optimize a single focal variable shared across the sequence while keeping the principal point fixed. Following the parameterization used by the focal BA solver, we define

$$u = \frac{f_x}{2c_x}, \qquad u^{\text{prior}} = \frac{f_x^{\text{prior}}}{2c_x}, \tag{36}$$

where f_x^{prior} is predicted by the backbone. The vertical focal length preserves its initial aspect ratio $\rho = f_y/f_x$, such that

$$f_x = 2c_x u, \qquad f_y = 2c_x \rho u. \tag{37}$$

The focal prior is defined as

$$\mathcal{C}_f = \frac{1}{2}\left(u - u^{\text{prior}}\right)^2. \tag{38}$$

Since the focal residual has scalar Jacobian

$$J_f = \frac{\partial(u - u^{\text{prior}})}{\partial u} = 1, \tag{39}$$

its contribution to the focal normal-equation block is

$$H_{uu} \leftarrow H_{uu} + \lambda_f, \qquad \tilde{r}_u \leftarrow \tilde{r}_u + \lambda_f \left(u^{\text{prior}} - u\right). \tag{40}$$

To make the backbone prior comparable across windows with different visual constraint densities, we scale its information weight by the current visual focal information:

$$\lambda_f = \max\left(\epsilon_f, \gamma H_{uu}^{\text{vis}}\right), \tag{41}$$

where H_{uu}^{vis} is the focal Hessian accumulated from visual reprojection factors, $\gamma = 1$ in all experiments, and $\epsilon_f = 10^{-6}$ prevents a degenerate zero weight. The focal prior is used only during frontend optimization.

A.3 RUNTIME ANALYSIS

We report online tracking FPS in Table 6. Our framework obtains shared features and complementary priors through a single backbone invocation per input frame, reducing the need for separate task-specific predictors. Under the evaluated configurations, our method achieves comparable or slightly higher tracking throughput than ViPE and WildPose. The remaining speed gap to Lingbot-map and DROID-SLAM motivates future work on efficient multi-frame feature extraction and geometric optimization.

Table 6: Tracking runtime analysis on KITTI and Bonn. All methods are evaluated on NVIDIA RTX A6000.

Method	**KITTI**		**Bonn**	
	FPS ↑	**ATE** ↓	**FPS** ↑	**ATE** ↓
DROID-SLAM	21.4	100.3	6.9	4.9
WildPose	1.1	30.4	3.6	2.4
ViPE	2.3	32.0	5.9	2.7
Lingbot-map	6.0	28.3	4.3	4.5
Ours (DAv3, Single)	4.5	19.1	3.1	2.7
Ours (DAv3, Multi)	2.7	20.6	1.9	2.3

B NETWORK ARCHITECTURE

During Stages I and II, backbone features are extracted independently for each frame, including for variants that use multi-frame context at inference. Although this introduces a train–test mismatch in the feature distribution, we empirically find that the updater remains robust when applied to context-enriched multi-frame features.

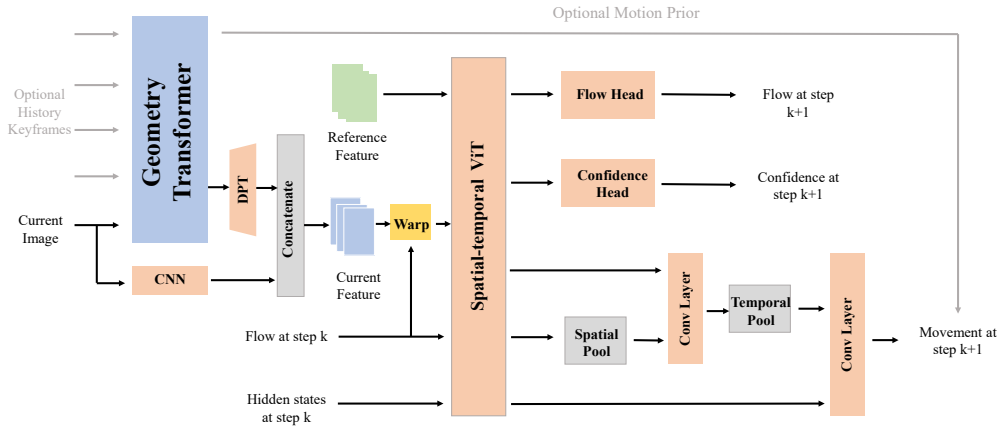


Figure 6: Architecture of the warping-based updater. Backbone features decoded by a DPT are combined with CNN features and aligned using the current flow estimate. A spatiotemporal ViT processes the reference and warped features together with flow and recurrent hidden states to predict flow corrections and correspondence confidence. A motion branch aggregates spatial and temporal information and incorporates an optional backbone motion prior to estimate motion probabilities. Yellow-shaded modules indicate trainable components.

C SCHEMATIC OF POSE PRIOR REGULARIZATION

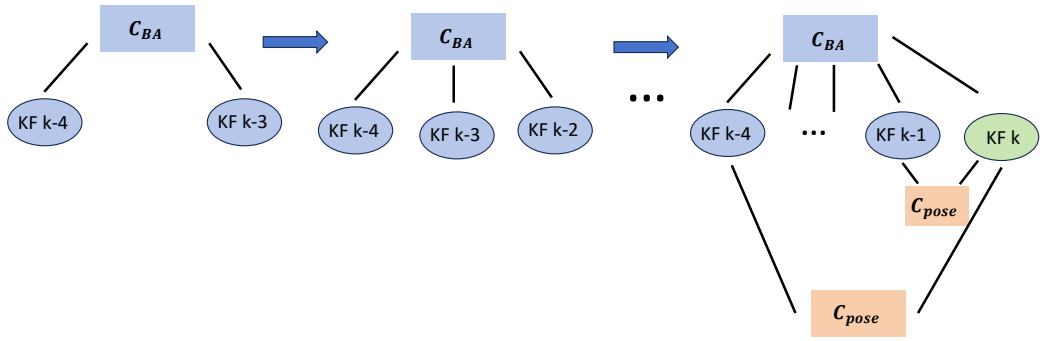


Figure 7: Incremental integration of pose priors. As new keyframes enter the active window, visual BA constraints (blue) are augmented with relative-pose priors (orange) connecting the current keyframe (green) to the oldest and most recent context keyframes. Each pose prior constrains relative rotation, translation direction, and baseline magnitude, with uncertainty gating applied to the translation terms. Only pose-prior factors involving the newest keyframe are shown.

D ADDITIONAL IMPLEMENTATION DETAILS

D.1 POSE PRIOR

Translation observability computation. Let $\mathbf{H}^0$ denote the normal matrix of the standard local DBA update before adding the pose-prior factors or Levenberg–Marquardt damping. We first eliminate the inverse-depth increments through the standard DBA Schur complement, yielding a pose-only system $\bar{\mathbf{H}} \in \mathbb{R}^{6M\times 6M}$ for the M poses in the local window. Thus, the proposed diagnostic does not require constructing a separate geometric system.

Let j be the newest keyframe and partition the pose increments into its 6-DoF increment $\delta\boldsymbol{\xi}_j$ and all historical pose increments $\delta\boldsymbol{\xi}_{\text{hist}}$. Marginalizing the historical poses gives

$$\mathbf{H}_j = \bar{\mathbf{H}}_{jj} - \bar{\mathbf{H}}_{j\text{hist}}\bar{\mathbf{H}}_{\text{histhist}}^{-1}\bar{\mathbf{H}}_{\text{hist}j} \in \mathbb{R}^{6\times 6}. \tag{42}$$

Finally, we partition the current-pose increment as $\delta\boldsymbol{\xi}_j = [\delta\boldsymbol{\rho}_j^\top, \delta\boldsymbol{\phi}_j^\top]^\top$, where $\delta\boldsymbol{\rho}_j$ and $\delta\boldsymbol{\phi}_j$ are the translational and rotational components. Using subscripts t and r for the corresponding blocks, we write

$$\mathbf{H}_j = \begin{bmatrix} \mathbf{H}_{tt} & \mathbf{H}_{tr} \\ \mathbf{H}_{rt} & \mathbf{H}_{rr} \end{bmatrix}. \tag{43}$$

Eliminating the current rotation yields

$$\mathbf{S}_j = \mathbf{H}_{tt} - \mathbf{H}_{tr}\mathbf{H}_{rr}^{-1}\mathbf{H}_{rt} \in \mathbb{R}^{3\times 3}. \tag{44}$$

This sequential elimination corresponds to eliminating the nuisance variables q jointly in the main text. We use $\sigma_j = \lambda_{\min}(\mathbf{S}_j)$ as the translation observability score and set $g_j = \mathbb{I}[\sigma_j < \tau_{\text{obs}}]$. The gate affects only translation-direction and translation-scale factors; the rotation prior remains active independently. In implementation, all linear systems are solved directly.

Regularization weights. We set $\lambda_R = 1.0$, $\lambda_t = 1.0$, and $\lambda_s = 0.1$ for the rotation, translation-direction, and translation-scale terms, respectively. The observability gate g_j controls both translation terms, while the rotation term remains ungated.

Near-zero baselines. Normalizing translation vectors and evaluating their log magnitudes become numerically unstable near zero. We therefore evaluate the translation-direction term only when both $\|\hat{\mathbf{t}}_{ij}\|_2$ and $\|\mathbf{t}_{ij}^{\text{prior}}\|_2$ exceed $\epsilon_t = 10^{-3}$. The translation-scale term requires both magnitudes to exceed the stricter threshold $\epsilon_s = 0.05$. Each term is omitted when its baseline condition is not satisfied, independently of the observability gate.

Threshold scaling. The numerical scale of our observability score depends on how camera translation translates into pixel displacement. For example, under a small horizontal camera translation Δt_x with fixed rotation, a static point at depth Z undergoes an image displacement $\Delta u \approx -f_x \Delta t_x / Z$, where f_x is the focal length in pixels at the optimization resolution. Thus, the same physical translation produces a smaller pixel displacement for more distant points or a shorter focal length, making translation less observable from reprojection measurements. For this component, the Jacobian magnitude is f_x/Z, and its contribution to the translation-information block scales as $w f_x^2/Z^2$, where w is the residual weight.

This motivates an approximate threshold scaling rule:

$$\tau_{\text{obs}}^b \approx \tau_{\text{obs}}^a \left(\frac{\tilde{f}_b}{\tilde{f}_a}\right)^2 \left(\frac{\tilde{Z}_a}{\tilde{Z}_b}\right)^2, \tag{45}$$

where $\tilde{f}$ and $\tilde{Z}$ denote representative focal-length and depth scales. This rule provides only an initial operating scale: the reduced eigenvalue also depends on the number, weights, and spatial distribution of observations, as well as the variable coupling removed by the Schur complement. We fix one threshold for all sequences within each benchmark, without per-sequence depth-dependent adaptation at inference. We use $\tau_{\text{obs}} = 1.0$ for Bonn, TUM, 7-Scenes, ScanNet, and EPIC Fields, and $\tau_{\text{obs}} = 0.025$ for KITTI, reflecting the lower translation-information scale in its more distant outdoor scenes.

D.2 Training Dataset Details.

TartanAir v2 & TartanGround. We use the front-facing left camera and exclude a shared set of environments from training: `Amusement`, `AmusementPark`, `CarWelding`, `Cyberpunk`, `CyberpunkDowntown`, `Ocean`, `SoulCity`, `SeasonalForestAutumn`, `WaterMillDay`, and `WaterMillNight`. We additionally discard one known problematic trajectory from `OldBrickHouseDay`: `P004` for TartanAir v2 and `P0004` for TartanGround. All remaining sequences with valid synchronized RGB images, depth maps, and camera poses are retained for training.

Kubric. We generate synthetic RGB-D sequences using Kubric with Blender rendering and PyBullet simulation. Foreground objects are sampled from the training split of Google Scanned Objects (GSO), while HDRI Haven environments provide the background illumination and KuBasic provides the hemispherical floor dome. Each scene contains 8–16 static objects and 1–4 dynamic objects; the latter are assigned random horizontal velocities. We render 25 frames at 512×512 resolution with a 35 mm focal length. Camera centers are initialized on an upper hemispherical shell with radius sampled from $[7, 9]$ m and observe the scene center. We sample linear trajectories with up to 3 m displacement, pure-rotation trajectories with yaw and pitch amplitudes of $8°$–$25°$ and $4°$–$15°$, respectively, and target-locked lateral, vertical, orbital, and spiral trajectories with translational motion up to 6 m or orbital angles between $22.5°$ and $60°$. The renderer provides exact RGB images, metric depth, camera poses, instance segmentations, and binary masks for dynamic objects.

Data Sampling. Table 7 summarizes the dataset mixtures, crop resolutions, and per-GPU batch sizes. Before training, we compute geometric correspondences from depth and camera parameters and use dataset-specific thresholds $(f_{\min}, f_{\max})$ to filter candidate frame pairs when constructing the training co-visibility graphs. Seven-frame clips are sampled using these precomputed graphs.

D.3 Supervision.

In Stages I and II, we optimize $\mathcal{L}_{\text{SLAM}} = 10\mathcal{L}_{\text{cam}} + 0.05\mathcal{L}_{\text{flow}} + 0.01\mathcal{L}_{\text{res}}$. All losses are applied at each BA iteration, with weight γ^{K-k} for iteration k, where K is the total number of iterations and $\gamma = 0.9$. The camera loss supervises relative poses through their Lie-algebra errors: $\mathcal{L}_{\text{cam}} = \text{mean}_{(i,j)}(\|\boldsymbol{\rho}_{ij}\|_2 + \|\boldsymbol{\phi}_{ij}\|_2)$, where $[\boldsymbol{\rho}_{ij}, \boldsymbol{\phi}_{ij}] = \text{Log}_{SE(3)}(\hat{\mathbf{G}}_{ij}(\mathbf{G}_{ij}^{gt})^{-1})$ The flow loss compares geometry-induced correspondences after BA with ground truth: $\mathcal{L}_{\text{flow}} = \text{mean}_{(i,j),\mathbf{p}} \|\mathbf{u}_{ij}(\mathbf{p}) - \mathbf{u}_{ij}^{gt}(\mathbf{p})\|_2$, evaluated over valid static pixels in temporally adjacent frames. In uncalibrated training, $\mathbf{u}_{ij}$ is computed using the updated camera poses, inverse depths, and current optimized intrinsics, allowing flow supervision to jointly constrain geometry and focal length. The residual loss $\mathcal{L}_{\text{res}} = \text{mean}\|\mathbf{r}_{ij}\|_1$ penalizes the discrepancy between updater-predicted and geometry-induced correspondences after BA. We omit the flow loss for Kubric because its rendered depth lacks valid background geometry. In Stage II, ground-truth dynamic masks exclude moving pixels from DBA and the pixel-wise flow and residual losses.

In Stage III, we freeze the correspondence updater and train only the motion prediction modules using camera-pose and motion-mask supervision. Ground-truth masks are replaced by predicted static reliabilities in DBA, and we optimize $\mathcal{L}_{\text{motion}} = \lambda_{\text{cam}}\mathcal{L}_{\text{cam}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}$, where $\mathcal{L}_{\text{mask}} = \lambda_{\text{rel}} \frac{1}{K} \sum_{k=1}^{K} \mathcal{L}_{\text{seg}}(\hat{s}_{ij}^{(k)}, s_i) + \lambda_{\text{base}}\mathcal{L}_{\text{seg}}(\hat{s}_i^{\text{base}}, s_i)$ and $\mathcal{L}_{\text{seg}} = \mathcal{L}_{\text{BCE}} + \mathcal{L}_{\text{Dice}}$.

D.4 Training schedule.

We use a OneCycle learning-rate schedule in each stage. Stages I and II each run for 50,000 steps, with peak learning rates of 10^{-4} and 5×10^{-5}, respectively. Stage III freezes the non-motion components and trains the motion prediction modules for 20,000 steps with a peak learning rate of 10^{-4}. All stages use four NVIDIA RTX A6000 GPUs and seven-frame clips. Table 7 reports per-GPU batch sizes, corresponding to global batch sizes of 16 or 8 depending on the dataset.

D.5 Benchmarking details

We follow established dataset-specific evaluation protocols to facilitate comparisons with both SLAM systems and 3D foundation models. Published results are reused only when sequence selection and metric computation match our evaluation settings; otherwise, we evaluate the methods using their official implementations.

TUM, Bonn & EPIC Fields For TUM RGB-D, we evaluate on all nine Freiburg1 sequences: `360`, `room`, `plant`, `xyz`, `desk2`, `teddy`, `floor`, `desk`, and `rpy`, where each name is prefixed with `rgbd_dataset_freiburg1_`. We process the complete sequences with a frame stride of 2. For Bonn, we evaluate our method on eight sequences: `balloon`, `balloon2`, `crowd`, `crowd2`, `person_tracking`, `person_tracking2`, `moving_nonobstructing_box`, and `moving_nonobstructing_box2`, each prefixed with `rgbd_bonn_`. we process the complete sequences with a frame stride of 1. For EPIC Fields, We evaluate our method on 55 sequences from

Dataset	Crop size	Sampling probability (%)			Batch size			$(f_{\min}, f_{\max})$	Flow sup.
		I	II	III	I	II	III		
TartanAir	384×512	33.3	15	10	4	4	4	$(8, 96)$	Yes
TartanAir v2	384×512	–	15	–	–	4	–	$(8, 96)$	Yes
TartanGround	384×512	–	15	–	–	4	–	$(4, 96)$	Yes
ScanNet++	512×768	66.7	15	10	2	2	2	$(8, 256)$	Yes
Virtual KITTI 2	320×960	–	20	20	–	2	2	$(4, 96)$	Yes
Dynamic Replica	384×672	–	10	20	–	2	2	$(8, 96)$	Yes
Kubric	384×512	–	10	20	–	2	4	$(1, 128)$	No
HOI4D	384×512	–	–	20	–	–	2	$(4, 128)$	No

Table 7: Dataset mixtures and sampling configurations for the three training stages. All clips contain seven frames. The final two columns report the dataset-specific sampling range and whether the ground-truth flow supervision is enabled. We disable flow supervision for Kubric because its rendered depth maps do not provide valid background depth.

the official test split, excluding 12 sequences with missing ground-truth camera poses: `P33_111`, `P12_102`, `P34_101`, `P09_102`, `P33_107`, `P26_123`, `P30_105`, `P07_104`, `P26_120`, `P07_108`, `P34_103`, and `P02_137`. For each retained sequence, we use up to the first 3,000 original frames and subsample them with a stride of 3, yielding at most 1,000 input frames.

7-Scenes, ScanNet & KITTI For 7-Scenes, we evaluate on all 42 sequences across the seven scenes, processing every frame of each complete sequence without temporal subsampling. For ScanNet, we evaluate on all 100 sequences in the ScanNet v2 test split. Starting from the beginning of each sequence, we sample frames with a stride of 3 and retain up to 500 sampled frames, covering approximately the first 1,500 original frames. For KITTI, we evaluate on odometry sequences `00–10`, processing every frame of each complete sequence without temporal subsampling.

Evaluation metrics. We evaluate camera trajectories using the root mean squared absolute trajectory error (ATE) after similarity alignment, including scale correction: $\mathrm{ATE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T} \|\hat{\mathbf{c}}_t^{\mathrm{align}} - \mathbf{c}_t^{\mathrm{gt}}\|_2^2}$, where $\hat{\mathbf{c}}_t^{\mathrm{align}}$ and $\mathbf{c}_t^{\mathrm{gt}}$ are the aligned estimated and ground-truth camera centers. For depth evaluation on KITTI and Bonn, we estimate a single scale and shift per sequence using median-based alignment over valid pixels across all evaluated frames, yielding $\tilde{D}_q = a\hat{D}_q + b$. We report $\mathrm{AbsRel} = \frac{1}{|\Omega|}\sum_{q\in\Omega} |\tilde{D}_q - D_q^{\mathrm{gt}}|/D_q^{\mathrm{gt}}$ and $\delta_1 = \frac{1}{|\Omega|}\sum_{q\in\Omega} \mathbb{I}[\max(\tilde{D}_q/D_q^{\mathrm{gt}}, D_q^{\mathrm{gt}}/\tilde{D}_q) < 1.25]$, where Ω contains valid evaluation pixels after alignment and clipping. We report δ_1 as a percentage and average sequence-level metrics across each dataset.