2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

AIM-SLAM: Dense Monocular SLAM via Adaptive and Informative Multi-View Keyframe Prioritization with Foundation Model

Jinwoo Jeon[1], Dong-Uk Seo[1], *Student Member, IEEE*, Eungchang Mason Lee[2], *Member, IEEE*, and Hyun Myung[1*], *Senior Member, IEEE*

***Abstract*— Recent advances in geometric foundation models have emerged as a promising alternative for addressing the challenge of dense reconstruction in monocular visual simultaneous localization and mapping (SLAM). Although geometric foundation models enable SLAM to leverage variable input views, the previous methods remain confined to two-view pairs or fixed-length inputs without sufficient deliberation of geometric context for view selection. To tackle this problem, we propose *AIM-SLAM*, a dense monocular SLAM framework that exploits an adaptive and informative multi-view keyframe prioritization with dense pointmap predictions from visual geometry grounded transformer (VGGT). Specifically, we introduce the selective information- and geometric-aware multi-view adaptation (SIGMA) module, which employs voxel overlap and information gain to retrieve a candidate set of keyframes and adaptively determine its size. Furthermore, we formulate a joint multi-view $\mathrm{Sim}(3)$ optimization that enforces consistent alignment across selected views, substantially improving pose estimation accuracy. The effectiveness of AIM-SLAM is demonstrated on real-world datasets, where it achieves state-of-the-art pose estimation performance and accurate dense reconstruction results. Our system supports ROS integration, with code is available at `https://aimslam.github.io/`.**

I. Introduction

Visual simultaneous localization and mapping (SLAM) has traditionally relied on geometric pipelines that exploit handcrafted features and require accurate camera calibration to estimate camera poses [1]–[3]. Recent geometry-aware foundation models such as DUSt3R [4], MASt3R [5], and VGGT [6] have emerged as compelling alternatives, directly predicting dense 3D pointmaps from uncalibrated RGB inputs. Leveraging these advantages, researchers have aimed to extend foundation models into SLAM systems that support dense reconstruction with uncalibrated monocular images [7]–[10].

As the number of input views that foundation models can accommodate has expanded, several approaches have incorporated multi-view reasoning into SLAM [9], [10]. To extend to the multi-view setting, most works have followed the sequential design, forming temporal windows of consecutive keyframes and aligning the submaps from each window with optimization. Although this design achieves decent performance, such a simple conjunction of neighboring frames does not fully exploit the potential of multi-view constraints from the foundation model; it often includes redundant frames with limited geometric information gain. While conventional SLAM approaches have explored multi-view keyframe selection methods based on map representation and covisibility [3], [11], such considerations remain largely unexplored in foundation model-based SLAM, underscoring the need for more principled keyframe prioritization.

In this context, we propose *AIM-SLAM*, a monocular SLAM framework that leverages an adaptive and informative multi-view prioritization with foundation models. By prioritizing informative views for optimization, rather than relying solely on temporally adjacent frames, our approach improves geometric consistency, mitigates scale drift, and enables consistent dense reconstruction across diverse scenes.

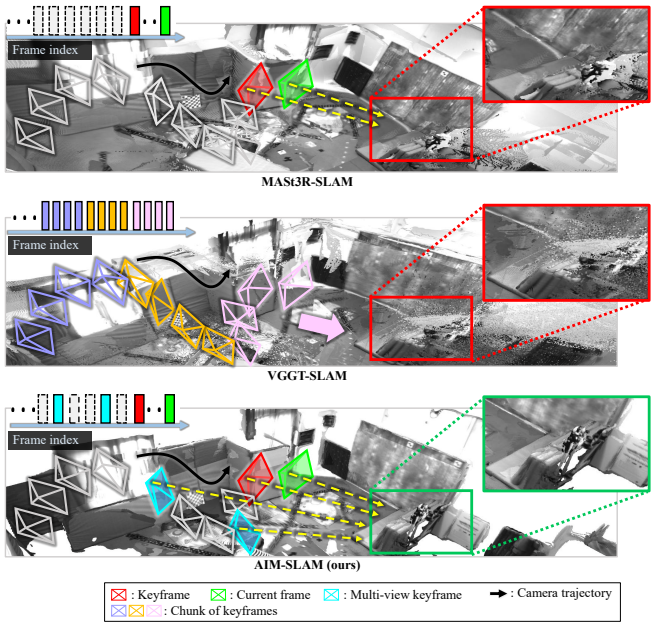


Fig. 1. Comparison among MASt3R-SLAM [8], VGGT-SLAM [9], and the proposed AIM-SLAM. MASt3R-SLAM relies on a fixed two-view input, while VGGT-SLAM processes a fixed chunk of consecutive keyframes. In contrast, AIM-SLAM adaptively prioritizes a variable number of keyframes with high viewpoint overlap and information gain. By jointly optimizing these multi-view inputs in $\mathrm{Sim}(3)$ space, AIM-SLAM achieves accurate and globally consistent dense reconstruction.

*Corresponding author: Prof. Hyun Myung
[1]The School of Electrical Engineering, KAIST (Korea Advanced Institute of Science and Technology), Daejeon, 34141, Republic of Korea, {zinuok, dongukseo, hmyung}@kaist.ac.kr
[2]KAIST InnoCORE LLM, KAIST, Daejeon, 34141, Republic of Korea, eungchang_mason@kaist.ac.kr

This work was supported in part by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2025-02303870, Software Technology for Efficient Multimodal Visual Information Processing in High-Speed Spatial Interactions), and in part by the Technology Innovation Program(or Industrial Strategic Technology Development Program-Robot Industry Technology Development)(RS-2024-00427719, Dexterous and Agile Humanoid Robots for Industrial Applications) funded By the Ministry of Trade Industry & Energy(MOTIE, Korea). The students are supported by BK21 FOUR.

As shown in Fig. 1, AIM-SLAM adaptively prioritizes candidate keyframes that maximize 3D scene overlap and information gain. A subsequent stability criterion regulates the incorporation of these views into optimization, and the prioritization process is repeated until convergence. Through this adaptive mechanism, AIM-SLAM leverages foundation models in a manner suited to SLAM, enabling both accurate and consistent reconstruction.

In summary, our key contributions are as follows:

- An adaptive and informative multi-view prioritization is introduced to construct a sparse yet highly overlapping keyframe set. To this end, we propose the selective information- and geometric-aware multi-view adaptation (SIGMA) module, while a stability criterion adaptively regulates their incorporation into the frontend visual odometry, ensuring geometric consistency and minimizing redundancy in foundation model-based SLAM.
- We present a joint multi-view $\mathrm{Sim}(3)$ optimization in foundation model-based SLAM, enabling accurate alignment across multiple views without requiring camera calibration.
- Extensive evaluations on public datasets validate the effectiveness of AIM-SLAM in pose estimation and dense reconstruction. The code is publicly released, with ROS integration also provided.

II. Related Works

A. Classical and Learning-based Visual SLAM

Conventional visual SLAM pipelines could be categorized by their input-processing strategies. Indirect methods estimate motion by minimizing reprojection error of sparse salient primitives, such as points [3], [12], [13] and line segments [14]–[16]. Direct methods minimize photometric error over pixel intensities, yielding (semi-)dense maps from monocular [17]–[19], stereo [20], [21], or RGB-D inputs [22], [23]. Semi-direct approaches combine both paradigms [2], [24]. While these approaches have enabled decades of progress, such methods remain constrained by handcrafted modules and reliance on accurate calibration in monocular settings.

With the advent of deep learning [25], these handcrafted modules have increasingly been replaced by learned counterparts. This includes data-driven descriptors and matchers for robust feature tracking [26], [27], end-to-end networks regressing pose (and often depth) from RGB video [28]–[31], and hybrid methods injecting learned priors such as depth [32], [33], semantics [34], or optical flow [33], [35] into geometric back-ends. More recently, differentiable dense bundle adjustment (DBA) has been explored in learning-based SLAM: DROID-SLAM [11] coupled ConvGRU updates with DBA, DPVO [36] introduced a path-based formulation, and GO-SLAM [37] extended DBA with full bundle adjustment and an implicit surface model.

B. Visual Foundation Models for 3D Geometry

Recent geometry-aware foundation models have demonstrated strong ability to infer dense 3D structure directly from uncalibrated images. DUSt3R [4] first demonstrated this from image pairs, and subsequent works extended this principle to multi-view inference via learnable recurrent memory [38], [39], all-to-all attention [40], or additional priors such as intrinsics and depth [41]. For dynamic scenes, MonST3R [42] leveraged optical flow, while Easi3R [43] decomposed cross-attention maps to separate moving from static geometry. Among these, MASt3R [5] introduced per-pixel descriptors that enabled downstream systems for structure-from-motion [44] and real-time dense SLAM [8]. In a parallel line of work, VGGT [6] generalized to arbitrary multi-view inputs, jointly predicting intrinsics, poses, depth, tracks, and dense pointmaps in a single feed-forward pass.

These foundation models have recently been adapted for SLAM. MASt3R-SLAM [8] demonstrated the first real-time dense monocular SLAM leveraging a reconstruction prior, but its reliance on the adjacent two-view input restricts parallax diversity and can lead to structural inconsistencies under challenging motions. VGGT-SLAM [9] extended VGGT to online settings by batching 16–32 consecutive frames into submaps aligned via $\mathrm{SL}(4)$ optimization, while VGGT-Long [10] scaled this to larger 60–75 frame windows with $\mathrm{Sim}(3)$ refinement. Although these submap-based methods enable online SLAM with foundation models, they (i) mainly rely on the N most adjacent views, which often contain redundant overlap even when keyframes are selected, and (ii) require large, fixed window sizes to ensure sufficient geometric coverage, treating SLAM as deferred submap registration rather than continuous multi-view tracking.

To tackle these problems, we propose an overlap-aware keyframe prioritization method formulated in an adaptive and informative manner, preserving VGGT's geometric fidelity while avoiding redundant inference and retaining only informative views. As a result, the proposed approach offers a more scalable solution compared with fixed window foundation model SLAM systems.

III. AIM-SLAM

Fig. 2 shows the overall framework of AIM-SLAM, which consists of (a) adaptive and informative multi-view keyframe prioritization with VGGT inference and (b) joint multi-view $\mathrm{Sim}(3)$ pose optimization for frontend visual odometry, while loop closure with global pose-graph optimization runs asynchronously in a separate backend thread.

A. Preliminaries: VGGT

VGGT [6] is a feed-forward network that processes an arbitrary-length sequence $\mathcal{I} = \{I_i, \ldots, I_{i+N}\}$ using DINOv2 [45] patch embeddings with alternating local (within-frame) and global (across-frame) self-attention layers. Its decoder [46] predicts per-frame depth with confidence, 3D points, correspondences, and camera parameters. In our pipeline, each depth map is backprojected into a point cloud expressed in the first frame's coordinates, with its

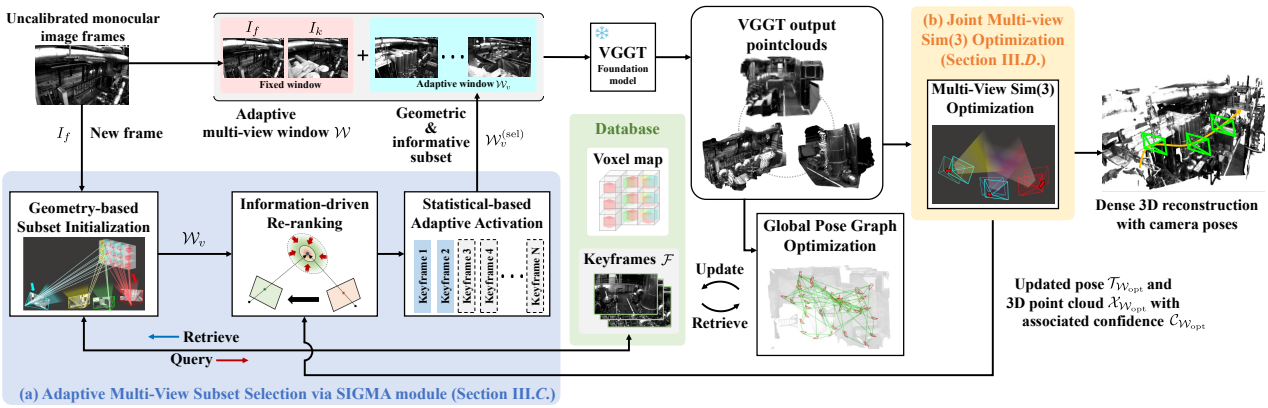


Fig. 2. Overall architecture of AIM-SLAM. The frontend consists of (a) multi-view prioritization method via the proposed SIGMA module, followed by VGGT-based dense pointmap inference, and (b) joint multi-view $\mathrm{Sim}(3)$ optimization to mitigate short- and mid-term drift. The backend loop closure module performs global pose-graph optimization to ensure global consistency.

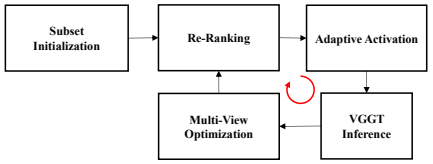


Fig. 3. Block diagram of the proposed SIGMA module, which consists of three stages: (i) geometry-based subset initialization via voxel overlap, (ii) information-driven re-ranking based on covariance reduction, and (iii) adaptive activation regulated by a stability test. After each multi-view optimization, updated poses and confidences recurrently trigger the re-ranking process.

associated confidence, as this yields higher accuracy than directly using the raw 3D points [6]. We denote the point clouds and confidences as $\mathcal{X} = \{\mathbf{X}_{i|i}, \cdots, \mathbf{X}_{i|i+N}\}$ and $\mathcal{C} = \{\mathbf{C}_{i|i}, \cdots, \mathbf{C}_{i|i+N}\}$, respectively.

B. Problem Definition

Our goal is to estimate globally consistent camera poses and dense 3D reconstructions from uncalibrated monocular images. Given an image stream $\{I_1, I_2, \dots\}$, the absolute pose of j-th frame is denoted as $\mathbf{T}_j^w$, which transforms points from camera frame j into the world frame w. A relative transformation from frame i to frame j is defined as $\mathbf{T}_j^i = (\mathbf{T}_i^w)^{-1}\,\mathbf{T}_j^w$, belonging to $\mathrm{Sim}(3)$ and comprising scale $s \in \mathbb{R}^+$, rotation $\mathbf{R} \in \mathrm{SO}(3)$, and translation $\mathbf{t} \in \mathbb{R}^3$. For a given frame pair (i, j), $\mathbf{X}_{i|j}$ and $\mathbf{C}_{i|j}$ denote the 3D pointmap and its per-point confidence, predicted by VGGT from frame j and expressed in i's coordinates.

AIM-SLAM is a keyframe-based tracking framework following MASt3R-SLAM [8], but advances beyond fixed-window designs by introducing (i) adaptive, informative and overlap-aware multi-view prioritization regulated by VGGT predictions, and (ii) a joint $\mathrm{Sim}(3)$ optimization that ensures scalable and consistent reconstruction.

C. Adaptive Multi-view Prioritization via SIGMA Module

Leveraging VGGT's ability to process an arbitrary number of views, AIM-SLAM adaptively constructs a sparse yet highly overlapping and informative keyframe subset that serves as the VGGT input. To this end, we define the candidate set $\mathcal{W}_v$ for the input subset $\mathcal{W}$.

To construct $\mathcal{W}_v$, we propose the SIGMA module, which consists of three stages: (a) geometry-based initialization of candidate views using voxel-overlap scores, (b) information-driven re-ranking of these candidates based on covariance reduction, and (c) adaptive subset activation regulated by a statistical stability test. The overall procedure of the proposed SIGMA module is summarized in Fig. 3.

During the operation of the SIGMA module, VGGT input subset is denoted as $\mathcal{W} = \mathcal{W}_0 \cup \mathcal{W}_v^{(\mathrm{sel})}$, where $\mathcal{W}_0 = \{I_f, I_k, I_b\}$ contains the current frame I_f, the last keyframe I_k, and the best candidate $I_b \in \mathcal{W}_v$; $\mathcal{W}_v^{(\mathrm{sel})} \subseteq \mathcal{W}_v$ denotes the adaptively activated subset beyond the default triplet $\mathcal{W}_0$. Subsequently, $\mathcal{W}$ is fed into VGGT for multi-view inference.

The SIGMA module performs subset initialization and re-ranking with respect to the last keyframe I_k rather than the incoming frame I_f. This is because I_f and I_k inherently maintain overlap, and I_f is promoted to a new keyframe once this overlap falls below a threshold. In addition, because keyframe pointmaps are fused via confidence-weighted averaging (Section III.D), I_k serves as a stable anchor for accumulating reliable multi-view information.

1) Geometry-based Initial Subset Construction: We propose a voxel-indexed keyframe map (Fig. 4), where each voxel stores the IDs of keyframes that observe it. For the last keyframe I_k, we compute the voxel-overlap score as follows:

$$O(I_k, I_i) = |\, v(I_k) \cap v(I_i)\,|\,, \;\; I_i \in \mathcal{F} \setminus \{I_k\}, \qquad (1)$$

where $O(\cdot,\cdot)$ denotes the overlap score, $v(I)$ denotes the set of voxels observed by I, and I_i denotes a keyframe in the keyframe set $\mathcal{F} \setminus \{I_k\}$. The top-$N$ keyframes by this score form the initial candidate set $\mathcal{W}_v$. As the 3D points predicted by the foundation model is highly dense, voxel-wise associations are used instead of raw points to provide a more compact and efficient representation of co-visibility.

In contrast to prior voxel maps that index 3D landmarks for point retrieval [47], [48], our map explicitly records keyframe visibility, shifting the role of voxelization from point-level data association to view-level selection, thereby

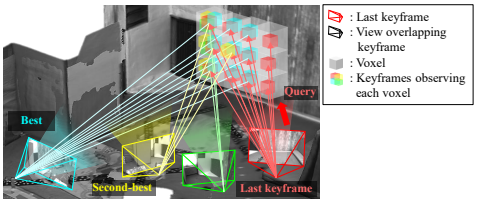


Fig. 4. Example of a voxel-indexed keyframe map for computing view overlap. Each voxel stores the IDs of keyframes that observe it. Using the last keyframe as the query, the system counts shared voxels and selects the top-N overlapping keyframes to initialize the multi-view subset.

aligning with our primary objective of adaptively constructing a multi-view input tailored for foundation model inference.

2) Information-driven Subset Re-ranking: The voxel-overlap candidates ensure sufficient co-visibility, but geometric overlap alone does not reflect the informativeness of each view. To prioritize the candidate keyframes, we re-rank $\mathcal{W}_v$ using information criterion based on the reduction of 3D point covariances, assuming that 3D points predicted by the foundation model follow Gaussian distribution. Because the last keyframe is typically the least optimized than other keyframes yet has the strongest influence on the current frame, we adopt a strategy that prioritizes candidate views to maximize the information gain of the point cloud of the last keyframe.

Formally, let $\mathbf{P}_k^-(\mathbf{x}_k)$ denote the prior covariance of a 3D point $\mathbf{x}_k$ observed in the last keyframe I_k, derived from pixel noise and fused confidence aggregated in the last keyframe under a Gaussian assumption. For brevity, we write $\mathbf{P}_k = \mathbf{P}(\mathbf{x}_k)$. Following the standard extended Kalman filter update form [49], [50], incorporating a candidate view $I_j \in \mathcal{W}_v$ updates the prior covariance as follows:

$$\mathbf{P}_{k\to j}^{+} = \mathbf{P}_k^{-} - \mathbf{P}_k^{-}\mathbf{J}_r^{\top}\left(\mathbf{R} + \mathbf{J}_r\mathbf{P}_k^{-}\mathbf{J}_r^{\top}\right)^{-1}\mathbf{J}_r\,\mathbf{P}_k^{-}, \tag{2}$$

where $\mathbf{P}_{k\to j}^{+}$ denotes the posterior covariance after adding I_j, $\mathbf{J}_r$ denotes the Jacobian of the ray-based residual [8] when reprojecting the 3D points from the last keyframe I_k into the candidate view I_j, and $\mathbf{R}$ denotes the measurement covariance. Unlike prior entropy-based methods [50], which assume 2D image-space noise, we leverage foundation-model predictions to define $\mathbf{R}$ as a full 3×3 covariance, obtained by propagating pixel noise through the inverse projection Jacobian (i.e., the Jacobian of the pinhole inverse projection function) with VGGT-predicted depth confidence. For efficiency, only a subset of points with low confidence is considered in this computation.

Subsequently, the information gain of view I_j relative to keyframe I_k is then quantified as the following entropy reduction:

$$\Gamma(I_k, I_j) = \sum_{\mathbf{x}_k\in\Omega_{k\to j}} \frac{1}{2}\log\frac{\det(\mathbf{P}_k^{-})}{\det(\mathbf{P}_{k\to j}^{+})}, \tag{3}$$

where $\Gamma(\cdot,\cdot)$ denotes the information gain score, $\Omega_{k\to j}$ denotes the valid point set after reprojection, and $\det(\cdot)$ denotes the determinant. Finally, the candidate set $\mathcal{W}_v$ is re-ranked by Γ to form the ordered subset.

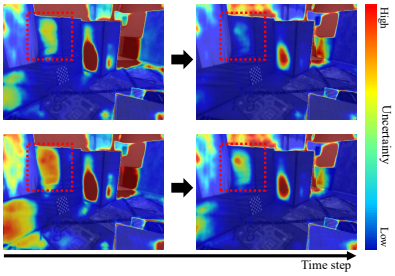


Fig. 5. Effect of the SIGMA module on keyframe uncertainty reduction. Compared with the case without re-ranking (lower), incorporating information-driven re-ranking (upper) significantly decreases keyframe uncertainty, computed as the inverse of the fused point confidence aggregated across observations during optimization. This shows that the SIGMA module effectively retrieves informative frames to refine the keyframe. Uncertainty is visualized in color, with higher values shown in warmer colors. Regions with pronounced differences are highlighted with red rectangles.

The SIGMA module effectively balances geometric co-visibility and information gain, yielding an ordered subset well-suited for multi-view inference. As shown in Fig. 5, the keyframes prioritized through the re-ranking stage of the SIGMA module significantly reduce the covariance of the last keyframe compared with the case without re-ranking, confirming that our strategy selects informative frames.

After multi-view optimization is performed on the window selected by the SIGMA module, the re-ranking is recurrently performed to reflect the effect of the updated depth confidence by the optimization. To avoid oscillations in the adaptive activation process, frames already in the input subset $\mathcal{W}$ are excluded from re-ranking in subsequent updates. Only remaining candidates are reordered, preventing an activated view from being repeatedly swapped with alternatives.

3) Adaptive Subset Activation with Stability Criterion: After the re-ranking stage, the candidate subset $\mathcal{W}_v$ contains diverse informative views, but it is not necessary to activate all of them. In practice, a smaller number of views is often sufficient, so we assess the statistical stability to determine whether candidate keyframes should be activated. As a result, only a subset of $\mathcal{W}_v$ is activated as needed, yielding a compact yet effective input $\mathcal{W}$ to VGGT. Adaptive activation starts from the default three-view baseline $\mathcal{W}_0$, and additional candidate keyframe from $\mathcal{W}_v$ is appended iteratively based on the statistical stability of the optimization.

To quantify such stability, we employ the reduced Chi-square test, a standard goodness-of-fit metric in weighted least squares. In previous SLAM methods, the test has been applied for outlier rejection [51], [52] or for directly assessing the statistical stability of the optimization itself [53]. We focus on the latter purpose to regulate the adaptive expansion of the keyframe set. Formally, let $\mathbf{b}_0 \in \mathbb{R}^M$ be the whitened residual vector, obtained by normalizing the original residuals, which will be detailed in the following section, and $\mathbf{A}_0 \in \mathbb{R}^{M\times p}$ the corresponding Jacobian at the current linearization. Under Gaussian noise assumptions, the residual sum of squares follows a Chi-square distribution with $\nu = M - \mathrm{rank}(\mathbf{A}_0)$ degrees of freedom [54], leading

to the reduced statistic as follows:

$$\varkappa \;=\; \frac{\mathbf{b}_0^\top \mathbf{b}_0}{M - \operatorname{rank}(\mathbf{A}_0)} \;\sim\; \chi^2(\,M - \operatorname{rank}(\mathbf{A}_0)), \quad (4)$$

where $\varkappa$ denotes the reduced Chi-square statistic. If $\varkappa \leq 1.0$, the configuration is considered stable and the window remains at the default three views. If $\varkappa > 1.0$, we iteratively append an additional keyframe $I_v \in \mathcal{W}_v$ in order and re-evaluate $\varkappa$ after multi-view optimization. When the inclusion of an additional keyframe results in a decrease of $\varkappa$, indicating improved stability, the keyframe is retained and further expansion is attempted. Conversely, if $\varkappa$ increases, the extension is considered unhelpful and the window reverts to the default three-view configuration.

After the assessment of statistical stability, the final input for VGGT is then defined as $\mathcal{W} = \mathcal{W}_0 \cup \mathcal{W}_v^{(\mathrm{sel})}$, where $\mathcal{W}_v^{(\mathrm{sel})} \subseteq \mathcal{W}_v$ denotes the adaptively activated subset beyond the default three views. From the VGGT inference, we obtain per-frame point clouds $\mathcal{X}_\mathcal{W} = \{\mathbf{X}_{f|i} \mid I_i \in \mathcal{W}\}$, confidences $\mathcal{C}_\mathcal{W} = \{\mathbf{C}_{f|i} \mid I_i \in \mathcal{W}\}$, and intrinsics $\mathcal{K}_\mathcal{W} = \{\mathbf{K}_i \mid I_i \in \mathcal{W}\}$, which are subsequently used in the joint multi-view optimization described in the following section.

D. Joint Multi-view $\mathrm{Sim}(3)$ *Optimization with Hybrid Residual*

1) Tracking and Keyframe Management: To perform multi-view optimization, we establish correspondences between keyframes in the subset $\mathcal{W}$ using the ray-matching strategy of MASt3R-SLAM, extended here to handle multiple views. Given dense pointmaps predicted by VGGT, each pixel $\mathbf{p}_i$ in frame I_i corresponds to a unique 3D point $\mathbf{x}_{f|i}$. Correspondences with another frame I_j are then obtained by minimizing the angular difference between their unit rays. This ray-based formulation provides scale-invariant dense matches, mitigating VGGT's mild scale inconsistency while avoiding the overhead of using raw correspondences directly estimated by VGGT across all views. If the ratio of valid correspondences in the current frame I_f falls below a threshold, I_f is promoted to a new keyframe and added to $\mathcal{F}$.

2) Optimization: For optimization, the keyframes in $\mathcal{W}$ are arranged in temporal order so that each relative transformation $\mathbf{T}_j^i \in \mathrm{Sim}(3)$ naturally represents the motion from an earlier frame I_i to a later frame I_j. To this end, we define an optimization-ordered subset $\mathcal{W}_{\text{opt}}$ as the reverse of $\mathcal{W}$ as $\mathcal{W}_{\text{opt}} = \{I_m, \ldots, I_k, I_f\}$, where I_m denotes the oldest keyframe in the subset and I_f the current frame. The state vector of adjacent relative transformations is then defined as $\mathcal{T}_{\mathcal{W}_{\text{opt}}} = \left[\mathbf{T}_v^m, \quad \cdots, \quad \mathbf{T}_f^k\right]$.

For each ordered adjacent frame pair $(I_i, I_j) \in \mathcal{W}_{\text{opt}}$, we combine ray-based and pixel-based reprojection terms [8] to define the residual as follows:

$$\begin{aligned}\mathbf{r}_{ij} = &\left(\Psi_{\text{ray}}(\mathbf{X}_{i|i}) - \Psi_{\text{ray}}(\mathbf{T}_j^i\, \mathbf{X}_{j|j})\right) \\ &+ \left(\Psi_\pi(\mathbf{K}_i, \mathbf{X}_{i|i}) - \Psi_\pi(\mathbf{K}_i, \mathbf{T}_j^i\, \mathbf{X}_{j|j})\right),\end{aligned} \quad (5)$$

where $\mathbf{r}_{ij}$ denotes the reprojection residual, $\Psi_{\text{ray}}(\cdot)$ denotes the ray-normalization function that projects a 3D point onto the unit sphere, and $\Psi_\pi(\cdot)$ denotes the pinhole camera projection with VGGT estimated intrinsics $\mathbf{K}_i$. As VGGT-predicted intrinsics are not perfectly calibrated, we do not enforce a single global set; instead, each pair adopts the intrinsics of its preceding keyframe I_i, whose estimates are more stable due to repeated averaging. The total joint multi-view residual $\mathbf{r}$ is then defined as the weighted sum of all pairwise residuals as follows:

$$\mathbf{r} = \sum_{(I_i,I_j)\in\mathcal{W}_{\text{opt}}} \mathbf{r}_{ij} = \sum_{(I_i,I_j)\in\mathcal{W}_{\text{opt}}} \left(\sum_{(a,b)\in\mathbf{m}_{i\to j}} \left(\frac{\mathbf{r}_{ij}^{ab}}{{}^{ab}w_{ij}} \right)_\rho \right), \quad (6)$$

where $\mathbf{m}_{i\to j}$ denotes the set of pixel correspondences from frame i to frame j; $\mathbf{r}_{ij}^{ab}$ denotes the hybrid residual defined for each correspondence pair and $(\cdot)_\rho$ denotes the Huber norm. ${}^{ab}w_{ij}$ denotes the per-point residual weight [8], [44], respectively. Following MASt3R-SfM [44] and MASt3R-SLAM [8], ${}^{ab}w_{ij}$ is defined as the geometric mean of per-frame confidences. In our formulation, the matching confidences are replaced with per-pixel confidences predicted by VGGT.

Finally, the left Jacobian of each residual with respect to the Lie algebra perturbation $\boldsymbol{\tau} \in \mathfrak{sim}(3)$ is computed and stacked into the global Jacobian as follows:

$$\mathbf{J} = \left[\mathbf{J}_{mv}, \quad \cdots, \quad \mathbf{J}_{kf}\right] = \left[\frac{\partial \mathbf{r}}{\partial \boldsymbol{\tau}_v^m}, \quad \cdots, \quad \frac{\partial \mathbf{r}}{\partial \boldsymbol{\tau}_f^k}\right]. \quad (7)$$

The optimization problem is formulated as a Levenberg–Marquardt scheme with an iteratively reweighted least squares (IRLS) solver. We construct the Hessian matrix from the Jacobian and information weights, and solve for the pose update vector $\boldsymbol{\tau}_{\mathcal{W}_{\text{opt}}}$ to update the state as $\mathcal{T}_{\mathcal{W}_{\text{opt}}} \leftarrow \boldsymbol{\tau}_{\mathcal{W}_{\text{opt}}} \oplus \mathcal{T}_{\mathcal{W}_{\text{opt}}}$.

The operator $\oplus$ is the left-plus operator, which updates the exponential map $\mathrm{Exp}(\cdot)$ from $\mathfrak{sim}(3)$ to $\mathrm{Sim}(3)$ to update the entire state vector, and is formally defined as follows:

$$\boldsymbol{\tau}_{\mathcal{W}_{\text{opt}}} \oplus \mathcal{T}_{\mathcal{W}_{\text{opt}}} \triangleq \left[\mathrm{Exp}(\boldsymbol{\tau}_v^m) \circ \mathbf{T}_v^m, \cdots, \mathrm{Exp}(\boldsymbol{\tau}_f^k) \circ \mathbf{T}_f^k\right]. \quad (8)$$

To resolve the *gauge freedom* when converting relative poses into the world frame, the earliest frame I_m is fixed. Subsequently, keyframe pointmaps are fused via confidence-weighted averaging [8], and VGGT-predicted focal lengths are recursively averaged (principal point assumed at the image center [6]).

On the other hand, to reduce long-term drift, we implement loop closure by reusing the first-layer token z_k from VGGT. These DINOv2-based patch embeddings have proven effective for visual recognition and suffice as lightweight global descriptors [45], [55]. Loop candidates are retrieved via cosine similarity against the stored token database, with the top-2 matches defining the loop edge set $\mathcal{E}_{\mathcal{L}_k}$, after which z_k is appended for future queries. For each edge (i, j) in the pose graph, the reprojection residual is defined as similar form as in (6). The pose graph is then optimized by second-order IRLS solver, with its edge set incrementally expanded to include both sequential edges $\mathcal{E}_{\mathcal{S}_k}$ and loop edges $\mathcal{E}_{\mathcal{L}_k}$ for each keyframe I_k as $\mathcal{E}_\mathcal{G} \leftarrow \mathcal{E}_\mathcal{G} \cup (\mathcal{E}_{\mathcal{S}_k} \cup \mathcal{E}_{\mathcal{L}_k})$. The corresponding keyframe set $\mathcal{W}_{\text{pgo}}$ contains I_k and its

TABLE I. Quantitative comparison of camera pose accuracy on the TUM RGB-D dataset, measured by the RMSE of absolute trajectory error (ATE, unit: m). We indicate the top three results as first, second, and third.

	Method	TUM RGB-D									Avg.
		360	desk	desk2	floor	plant	room	rpy	teddy	xyz	
Calib.	DeepV2D [59]	0.243	0.166	0.379	1.653	0.203	0.246	0.105	0.316	0.064	0.375
	DeepFactors [60]	0.159	0.170	0.253	0.169	0.305	0.364	0.043	0.601	0.035	0.233
	DROID-SLAM [11]	0.111	0.018	0.042	0.021	0.016	0.049	0.026	0.048	0.012	0.038
	DPV-SLAM [61]	0.112	0.018	0.029	0.057	0.021	0.330	0.030	0.084	0.010	0.076
	MASt3R-SLAM [8]	0.049	0.016	0.024	0.025	0.020	0.061	0.027	0.041	0.009	0.030
Uncalib.	DROID-SLAM [11]	0.202	0.032	0.091	0.064	0.045	0.918	0.056	0.045	0.012	0.158
	MUSt3R-VO [7]	0.078	0.040	0.046	0.091	0.040	0.099	0.043	0.042	0.013	0.055
	VGGT-Long [10]	0.053	0.064	0.060	0.111	0.064	0.170	0.036	0.127	0.047	0.081
	VGGT-SLAM [9]	0.071	0.025	0.040	0.141	**0.023**	0.102	0.030	**0.034**	0.014	0.053
	MASt3R-SLAM [8]	0.070	0.035	0.055	0.056	0.035	0.118	0.041	0.114	0.020	0.060
	AIM-SLAM (ours)	**0.050**	**0.017**	**0.028**	**0.024**	0.026	**0.062**	**0.021**	0.039	**0.010**	**0.031**

connected neighbors. Unlike the local frontend subset $\mathcal{W}_{\text{opt}}$, the backend jointly aligns all sequential and loop edges in $\mathcal{E}_{\mathcal{G}}$ to enforce global consistency.

IV. Experimental Results and Discussion

A. Datasets and Evaluation Metrics

We evaluate AIM-SLAM on the TUM RGB-D dataset [56] and the EuRoC MAV dataset [57]. TUM RGB-D contains room-scale indoor trajectories with cluttered scenes. EuRoC features aggressive motions and large viewpoint changes, offering a challenging benchmark for robustness; in the uncalibrated setting on EuRoC, undistorted images are used for all methods without providing calibration, following [8].

Evaluation follows two standard metrics for dense visual SLAM: (i) camera pose estimation accuracy, measured by the RMSE of absolute trajectory error (ATE), and (ii) dense reconstruction quality, measured by accuracy, completion, and chamfer distance as similar to previous works [8]–[10]. For dense reconstruction evaluations on the EuRoC dataset, we only use the Vicon Room sequences where ground truth pointclouds were obtained from laser scans.

B. Implementation Details and Baselines

We used the released pretrained VGGT model with the same parameters across all datasets. The maximum size of the VGGT input subset $\mathcal{W}$ is set to 5. All experiments are run on an NVIDIA RTX 3090 and Intel Core i9-11900K (3.50 GHz), with input images resized to 518 pixels to meet VGGT requirements.

AIM-SLAM is designed for uncalibrated settings, and baselines are primarily evaluated under this condition. We compare AIM-SLAM against state-of-the-art learning-based SLAMs, including MASt3R-SLAM [8], MUSt3R-VO [7], VGGT-SLAM [9], VGGT-Long [10], and DROID-SLAM [11]. All baselines are reported with loop closure enabled, except MUSt3R-VO, which does not support loop closure and is therefore evaluated in odometry mode. For DROID-SLAM, which assumes known intrinsics, we estimate intrinsics in the uncalibrated setting using GeoCalib [58] applied to the first frame of each sequence following prior works [9], [10]. For completeness, we also report results in calibrated settings.

TABLE II. Quantitative comparison of camera pose accuracy on the EuRoC dataset, measured by the RMSE of absolute trajectory error (ATE, unit: m). We indicate the top three results as first, second, and third. † denotes that the average is computed excluding divergent sequences.

	Method	EuRoC											Avg.
		V101	V102	V103	V201	V202	V203	MH01	MH02	MH03	MH04	MH05	
Calib.	DeepV2D [59]	0.981	0.801	1.570	0.290	2.202	2.743	0.739	1.144	0.752	1.492	1.567	1.298
	DeepFactors [60]	1.520	0.679	0.900	0.876	1.905	1.021	1.587	1.479	3.139	5.331	4.002	2.040
	DROID-SLAM [11]	0.037	0.013	0.019	0.017	0.010	0.013	0.013	0.012	0.022	0.048	0.044	0.022
	DPV-SLAM [61]	0.035	0.008	0.015	0.020	0.011	0.040	0.013	0.016	0.022	0.043	0.041	0.024
	MASt3R-SLAM [8]	0.040	0.019	0.027	0.020	0.025	0.043	0.023	0.017	0.057	0.113	0.067	0.041
Uncalib.	DROID-SLAM [11]	0.465	1.679	1.439	0.878	1.414	1.895	0.154	0.256	1.010	0.719	0.762	0.970
	MUSt3R-VO [7]	0.489	0.287	0.554	0.123	0.109	0.252	0.265	-	0.909	0.741	0.828	0.456†
	VGGT-Long [10]	0.139	0.165	0.198	0.202	0.130	0.137	0.579	0.745	0.428	0.713	0.605	0.367
	VGGT-SLAM [9]	0.098	0.184	0.353	0.068	0.903	0.431	0.400	0.701	3.599	-	-	0.749†
	MASt3R-SLAM [8]	0.101	0.134	0.096	0.133	0.100	0.170	0.180	0.124	0.156	0.282	0.327	0.164
	AIM-SLAM (ours)	**0.081**	**0.059**	**0.069**	**0.057**	**0.053**	**0.060**	**0.055**	**0.076**	**0.058**	**0.115**	**0.114**	**0.072**

TABLE III. Quantitative comparison with state-of-the-art methods for dense reconstruction on the EuRoC dataset (left) and TUM RGB-D dataset (right). The best results are highlighted in **bold**, and the second-best results are underlined.

Method	EuRoC			TUM RGB-D		
	Accuracy	Completion	Chamfer	Accuracy	Completion	Chamfer
VGGT-Long [10]	<u>0.106</u>	0.119	0.112	<u>0.094</u>	<u>0.107</u>	<u>0.100</u>
VGGT-SLAM [9]	0.246	0.216	0.231	0.109	0.127	0.118
MASt3R-SLAM [8]	0.108	**0.072**	**0.090**	0.097	0.113	0.105
AIM-SLAM (ours)	**0.103**	<u>0.102</u>	<u>0.102</u>	**0.063**	**0.098**	**0.081**

C. Analysis of Results

1) Camera Pose Estimation: On the TUM RGB-D dataset, all methods achieve relatively high accuracy, as shown in Table I. AIM-SLAM stands out, surpassing the calibrated DROID-SLAM and achieving accuracy comparable with MASt3R-SLAM, while requiring no camera intrinsics.

On the other hand, Table II reports the pose accuracy results on the EuRoC dataset. For VGGT-Long and VGGT-SLAM, errors stem from submap-based alignment, where local predictions of each submap are reliable, but alignment fails under large viewpoint changes. MASt3R-SLAM shows better robustness but remains limited by its two-view design, restricting the use of wide-baseline cues. In contrast, AIM-SLAM achieves the best overall accuracy among uncalibrated methods. This highlights the effectiveness of the adaptive multi-view prioritization under challenging wide-baseline observations.

2) Dense Reconstruction: On the TUM RGB-D dataset, AIM-SLAM reconstructs fine object details more accurately than baselines, as shown in the bottom row of Fig. 6, achieving the strongest reconstruction performance across all metrics. The same multi-view strategy also preserves global consistency in large-scale sequences, as demonstrated on the EuRoC dataset (top row of Fig. 6). Here, baseline methods often suffer from ghosting artifacts on planar surfaces due to scale inconsistency, which persist even after two-view $\mathrm{Sim}(3)$ optimization [8] or accumulated submap alignment errors [10]. Consistent with this observation, Table III shows that AIM-SLAM achieves the best accuracy and competitive completion and chamfer distance. We attribute this behavior to a trade-off between stable informative-view selection and dense surface coverage, as AIM-SLAM focuses on a compact subset of reliable and overlapping views, which improves pose stability and geometric accuracy.

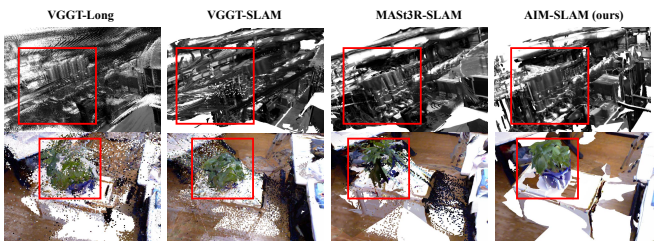
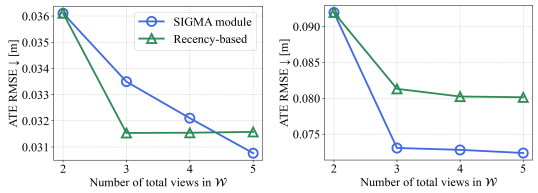


Fig. 6. Dense reconstruction results on the EuRoC dataset (top row) and TUM RGB-D dataset (bottom row). With its adaptive multi-view design, the proposed method achieves robust dense reconstructions across diverse environments.



(a) (b)

Fig. 7. Ablation study of the proposed SIGMA module showing how the total number of views affects pose accuracy (ATE RMSE) for the (a) TUM RGB-D dataset and (b) EuRoC dataset. Recency-based denotes selecting the most recent consecutive keyframes to form the input subset $\mathcal{W}$, as in prior methods [8]–[10].

D. Ablation Study

1) Effect of the Total Number of Views: Fig. 7 compares pose accuracy as the maximum limit size of the input keyframe subset $\mathcal{W}$ increases, between the cases where $\mathcal{W}$ is constructed in a recency-based manner and the case where the SIGMA module is employed. For the 2-view and 3-view cases, the subset size is fixed without adaptive expansion for the comparison. Increasing the limit naturally improves accuracy by introducing additional multi-view constraints, but the gain quickly saturates beyond 4–5 views. On the TUM dataset, both methods achieve comparable performance. The difference between the two methods becomes evident on the EuRoC dataset, which involves larger baselines and rapid viewpoint changes (Fig. 7(b)). While both methods eventually saturate beyond 4–5 views, the SIGMA module-based method maintains substantially higher accuracy throughout, as it consistently leverages more informative keyframes than the recency-based strategy.

2) Effect of the Hybrid Residual: Table IV compares alternative residual formulations. Ray-only residuals result in the largest errors, showing that the lack of pixel-level constraints limits geometric precision. Projection-only residuals with VGGT-estimated intrinsics reduce error but remain sensitive to calibration noise. The hybrid formulation yields the best performance on both datasets by combining the angular robustness of rays with the pixel-level accuracy of projections, highlighting the complementarity of the two terms.

V. Conclusions

In summary, we presented AIM-SLAM, a dense monocular SLAM framework for uncalibrated settings that leverages geometry-aware foundation models. The proposed SIGMA module adaptively prioritizes a sparse but overlap-rich and informative keyframe subset. The prioritized multi-view subset is then jointly optimized in $\mathrm{Sim}(3)$ space to reduce both short- and mid-term drift. Together, these components enable accurate pose estimation and geometrically consistent dense reconstruction under uncalibrated conditions, offering a more scalable solution for foundation model-based SLAM. The current limitation of AIM-SLAM is its reliance on VGGT inference, yielding an overall runtime of about 3 Hz in our environment. Excluding VGGT inference, the remaining components of our method run at about 17 Hz. Future work will explore accelerating the current foundation model or integrating faster alternatives.

TABLE IV. Ablation study of the proposed hybrid residual for joint multi-view $\mathrm{Sim}(3)$ pose optimization on the EuRoC dataset (left) and the TUM RGB-D dataset (right).

Method	ATE RMSE	
	EuRoC	TUM RGB-D
Ray only	0.138	0.061
Projection only	0.081	0.032
Hybrid (ray + projection)	**0.072**	**0.031**

References

[1] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2007, pp. 3565–3572.

[2] C. Forster, M. Pizzoli, and D. Scaramuzza, "SVO: Fast semi-direct monocular visual odometry," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 15–22.

[3] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 1147–1163, 2015.

[4] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, "DUSt3R: Geometric 3D vision made easy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 20 697–20 709.

[5] V. Leroy, Y. Cabon, and J. Revaud, "Grounding image matching in 3D with MASt3R," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 71–91.

[6] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, "VGGT: Visual geometry grounded transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2025, pp. 5294–5306.

[7] Y. Cabon, L. Stoffl, L. Antsfeld, G. Csurka, B. Chidlovskii, J. Revaud, and V. Leroy, "MUSt3R: Multi-view network for stereo 3D reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2025, pp. 1050–1060.

[8] R. Murai, E. Dexheimer, and A. J. Davison, "MASt3R-SLAM: Real-time dense SLAM with 3D reconstruction priors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2025, pp. 16 695–16 705.

[9] D. Maggio, H. Lim, and L. Carlone, "VGGT-SLAM: Dense RGB SLAM optimized on the SL (4) manifold," *arXiv preprint arXiv:2505.12549*, 2025.

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

[11] Z. Teed and J. Deng, "DROID-SLAM: Deep visual SLAM for monocular, stereo, and RGB-D cameras," *Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 16 558–16 569, 2021.

[12] T. Qin, P. Li, and S. Shen, "VINS-Mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, 2018.

[13] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1874–1890, 2021.

[14] H. Lim, Y. Kim, K. Jung, S. Hu, and H. Myung, "Avoiding degeneracy for monocular visual SLAM with point and line features," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 11 675–11 681.

[15] J. Lee and S.-Y. Park, "PLF-VINS: Real-time monocular visual-inertial SLAM with point-line fusion and parallel-line fusion," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 7033–7040, 2021.

[16] H. Lim, J. Jeon, and H. Myung, "UV-SLAM: Unconstrained line-based SLAM using vanishing points for structural mapping," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 1518–1525, 2022.

[17] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 834–849.

[18] R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2011, pp. 2320–2327.

[19] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 611–625, 2017.

[20] B. Zhang and D. Zhu, "A stereo SLAM system with dense mapping," *IEEE Access*, vol. 9, pp. 151 888–151 896, 2021.

[21] R. Wang, M. Schworer, and D. Cremers, "Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2017, pp. 3903–3911.

[22] C. Kerl, J. Sturm, and D. Cremers, "Dense visual SLAM for RGB-D cameras," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2013, pp. 2100–2106.

[23] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "ElasticFusion: Real-time dense SLAM and light source estimation," *Int. J. Robot. Res.*, vol. 35, no. 14, pp. 1697–1716, 2016.

[24] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "SVO: Semidirect visual odometry for monocular and multicamera systems," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 249–265, 2016.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 25, 2012.

[26] D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: Self-supervised interest point detection and description," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. WorkShops*, 2018, pp. 224–236.

[27] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: Learning feature matching with graph neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4938–4947.

[28] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1851–1858.

[29] S. Wang, R. Clark, H. Wen, and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 2043–2050.

[30] R. Li, S. Wang, Z. Long, and D. Gu, "UnDeepVO: Monocular visual odometry through unsupervised deep learning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 7286–7291.

[31] G. Iyer, J. Krishna Murthy, G. Gupta, M. Krishna, and L. Paull, "Geometric consistency for self-supervised end-to-end visual odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. WorkShops*, 2018, pp. 267–275.

[32] M. Bloesch, J. Czarnowski, R. Clark, S. Leutenegger, and A. J. Davison, "CodeSLAM—learning a compact, optimisable representation for dense visual SLAM," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2560–2568.

[33] N. Yang, L. v. Stumberg, R. Wang, and D. Cremers, "D3VO: Deep depth, deep pose and deep uncertainty for monocular visual odometry," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1281–1292.

[34] K. Tateno, F. Tombari, I. Laina, and N. Navab, "CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6243–6252.

[35] H. Zhan, C. S. Weerasekera, J.-W. Bian, and I. Reid, "Visual odometry revisited: What should be learnt?" in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 4203–4210.

[36] Z. Teed, L. Lipson, and J. Deng, "Deep patch visual odometry," *Adv. Neural Inf. Process. Syst.*, vol. 36, pp. 39 033–39 051, 2023.

[37] Y. Zhang, F. Tosi, S. Mattoccia, and M. Poggi, "GO-SLAM: Global optimization for consistent 3D instant reconstruction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 3727–3737.

[38] Q. Wang, Y. Zhang, A. Holynski, A. A. Efros, and A. Kanazawa, "Continuous 3D perception model with persistent state," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2025, pp. 10 510–10 522.

[39] H. Wang and L. Agapito, "3D reconstruction with spatial memory," *arXiv preprint arXiv:2408.16061*, 2024.

[40] J. Yang, A. Sax, K. J. Liang, M. Henaff, H. Tang, A. Cao, J. Chai, F. Meier, and M. Feiszli, "Fast3R: Towards 3D reconstruction of 1000+ images in one forward pass," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2025, pp. 21 924–21 935.

[41] W. Jang, P. Weinzaepfel, V. Leroy, L. Agapito, and J. Revaud, "Pow3R: Empowering unconstrained 3D reconstruction with camera and scene priors," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2025, pp. 1071–1081.

[42] J. Zhang, C. Herrmann, J. Hur, V. Jampani, T. Darrell, F. Cole, D. Sun, and M.-H. Yang, "MonST3R: A simple approach for estimating geometry in the presence of motion," 2025.

[43] X. Chen, Y. Chen, Y. Xiu, A. Geiger, and A. Chen, "Easi3R: Estimating disentangled motion from DUSt3R without training," *arXiv preprint arXiv:2503.24391*, 2025.

[44] B. P. Duisterhof, L. Zust, P. Weinzaepfel, V. Leroy, Y. Cabon, and J. Revaud, "MASt3R-SfM: a Fully-integrated solution for unconstrained structure-from-motion," in *Proc. IEEE Int. Conf. 3D Vis.*, 2025.

[45] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "DINOv2: Learning robust visual features without supervision," *Trans. Mach. Learn. Res.*, pp. 1–31, 2024.

[46] R. Ranftl, A. Bochkovskiy, and V. Koltun, "Vision transformers for dense prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 12 179–12 188.

[47] M. Muglikar, Z. Zhang, and D. Scaramuzza, "Voxel map for visual SLAM," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 4181–4187.

[48] Z. Yuan, F. Lang, J. Deng, H. Luo, and X. Yang, "Voxel-SVIO: Stereo visual-inertial odometry based on voxel map," *IEEE Robot. Automat. Lett.*, 2025.

[49] G. Welch, G. Bishop *et al.*, "An introduction to the Kalman filter," 1995.

[50] A. Das and S. L. Waslander, "Entropy based keyframe selection for multi-camera visual SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2015, pp. 3676–3681.

[51] S. Leutenegger, P. Furgale, V. Rabaud, M. Chli, K. Konolige, and R. Siegwart, "Keyframe-based visual-inertial SLAM using nonlinear optimization," *Robot. Sci. Syst.*, 2013.

[52] D. Zou, Y. Wu, L. Pei, H. Ling, and W. Yu, "StructVIO: Visual-inertial odometry with structural regularity of man-made environments," *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 999–1013, 2019.

[53] N. Keivan and G. Sibley, "Asynchronous adaptive conditioning for visual-inertial SLAM," *Int. J. Robot. Res.*, vol. 34, no. 13, pp. 1573–1589, 2015.

[54] B. K. Springer, "Parameter estimation and hypothesis testing in linear models," *Springer, BerlinKotsakis C (2012) Reference frame stability and nonlinear distortion in minimum-constrained network adjustment. J Geod*, vol. 86, no. 9, p. 755774, 1999.

[55] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, "AnyLoc: Towards universal visual place recognition," *IEEE Robot. Automat. Lett.*, vol. 9, no. 2, pp. 1286–1293, 2023.

[56] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2012, pp. 573–580.

[57] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The EuRoC micro aerial vehicle datasets," *Int. J. Robot. Res.*, vol. 35, no. 10, pp. 1157–1163, 2016.

[58] A. Veicht, P.-E. Sarlin, P. Lindenberger, and M. Pollefeys, "GeoCalib: Learning single-image calibration with geometric optimization," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 1–20.

[59] Z. Teed and J. Deng, "DeepV2D: Video to depth with differentiable structure from motion," 2020.

[60] J. Czarnowski, T. Laidlow, R. Clark, and A. J. Davison, "DeepFactors: Real-time probabilistic dense monocular SLAM," *IEEE Robot. Automat. Lett.*, vol. 5, no. 2, pp. 721–728, 2020.

[61] L. Lipson, Z. Teed, and J. Deng, "Deep patch visual SLAM," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 424–440.