GRAPH3R: UNIFYING VIEW GRAPH INITIALIZATION WITH FEED-FORWARD RECONSTRUCTION

Anonymous authors
Paper under double-blind review

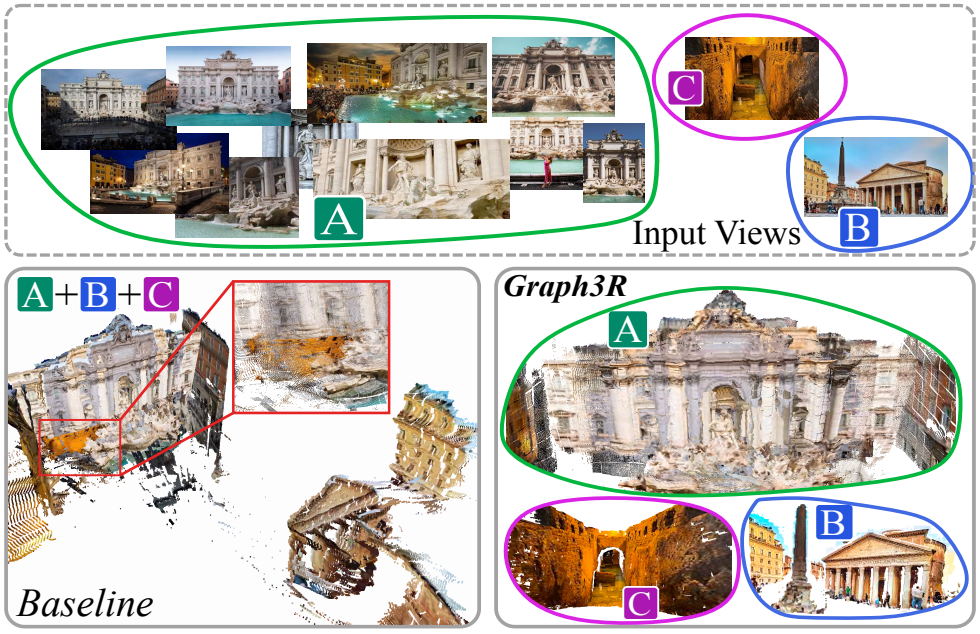

Figure 1: **Comparison with existing baselines**, given input views from three disjoint scenes (A–C), the baseline method π^3 (Wang et al., 2025b) produces an incorrect reconstruction (left). While ours (*Graph3R*) predicts a view graph from shared geometric features, partitions the views into scene groups, and reconstructs each group separately using a single backbone pass (right).

ABSTRACT

Feed-forward reconstruction recovers camera poses and dense geometry from image sets whose views are assumed to come from a single scene. In practice, uncurated image collections often contain views from disjoint scenes, and reconstructing these views jointly can corrupt camera poses and geometry. Existing pipelines typically resolve view associations in separate stages before final reconstruction, leaving view graph initialization and reconstruction inference only loosely coupled. We introduce *Graph3R*, which ***unifies view graph initialization with feed-forward reconstruction through a shared geometric representation***. A lightweight verification head predicts pairwise covisibility from the multi-view features of π^3 to construct a view graph. This graph classifies the input views into scene groups. For each group, we mask the shared features before decoding camera poses and dense geometry. Reusing the multi-view features allows *Graph3R* to reconstruct disjoint scenes separately with a single backbone pass. Across three benchmarks, *Graph3R* improves average maximum recall at 100% precision by upto 8.99 percentage points over the baseline, while improving reconstruction quality on mixed-scene inputs. Regarding the multi-session mapping task, *Graph3R* improves cross-session localization and achieves competitive pose estimation accuracy, while delivering a runtime speedup of approximately $35.9\times$ over the baseline when reconstructing the query group. By integrating learned view association with geometric inference, our approach takes a step toward end-to-end reconstruction from uncurated image collections.

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.

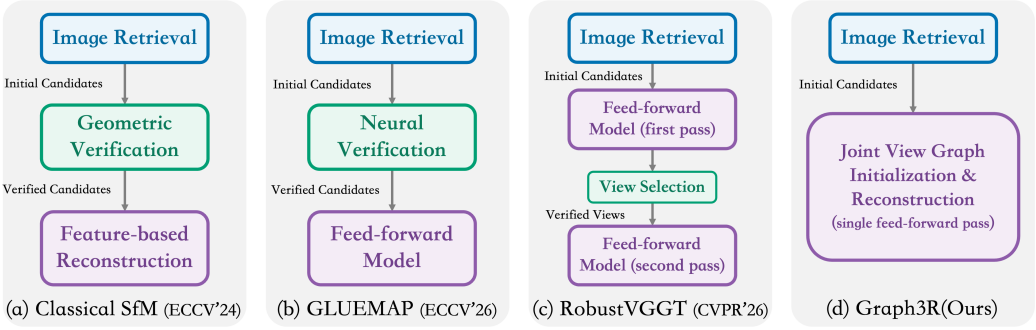

Figure 2: **Comparison of different SfM paradigms.** Classical SfM Pan et al. (2024) and GLUEMAP (Pan et al., 2026) initialize a view graph through separate verification stages before reconstruction. RobustVGGT (Han et al., 2025) selects views using VGGT's latent features and reruns the feed forward model on the selected views. ***Graph3R*** proposes a unified paradigm that jointly predicts the view graph and scene structure with a single backbone inference as detailed illustrated in Fig. 3.

1 Introduction

Feed-forward 3D reconstruction methods such as MASt3R (Leroy et al., 2024), VGGT (Wang et al., 2025a), and π^3 (Wang et al., 2025b) enable efficient recovery of camera poses and dense geometry from unordered images. These methods typically assume that the input views form a geometrically connected scene. Uncurated image collections, including Internet photos and images gathered across robot sessions, can violate this assumption by mixing views from disjoint scenes. Processing these views jointly can corrupt both camera poses and scene geometry (Han et al., 2025), as illustrated in Fig. 1. Reliable reconstruction therefore depends on identifying which views should be reconstructed together.

Classical structure-from-motion (SfM) (Schönberger et al., 2016; Schönberger & Frahm, 2016) and simultaneous localization and mapping (SLAM) (Campos et al., 2021; Mur-Artal et al., 2015) systems establish view associations through image retrieval and geometric verification. These associations define a **view graph**, whose nodes represent images and whose edges indicate covisibility. Visual place recognition (VPR) methods (Berton & Masone, 2025) propose candidate pairs, and geometric verification checks the spatial consistency of local feature correspondences using robust estimation such as RANSAC. However, correspondence consistency alone does not guarantee covisibility. GV-Bench (Yu et al., 2024) and Doppelgangers (Cai et al., 2023) document verification failures under long-term appearance changes and perceptual aliasing, including false associations between visually similar but distinct places.

Fig. 2 contrasts how representative pipelines connect view association to reconstruction. GLUEMAP (Pan et al., 2026) builds a view graph through image retrieval and pairwise verification before applying a feed-forward model to local view neighborhoods. RobustVGGT (Han et al., 2025) identifies outlier views from the reconstruction model's internal representations, then reruns the backbone on the retained inputs. These pipelines require separate feature computation for verification or a second backbone pass for reconstruction. Doppelgangers++ (Xiangli et al., 2025) already shares MASt3R features for graph construction and pruning within an SfM pipeline, demonstrating the value of feature reuse. This motivates a further question: ***Can view graph initialization and feed-forward reconstruction be unified through one shared backbone pass?***

We introduce *Graph3R*, a π^3-based model that unifies learned view graph initialization and feed-forward reconstruction through a shared geometric representation. A lightweight verification head predicts pairwise covisibility from intermediate multi-view features. The resulting view graph partitions the input into scene groups, whose masks condition pose and geometry decoding. The backbone features are computed once and reused to reconstruct each group, allowing disjoint scenes to be recovered separately without a second backbone pass.

Our contributions are threefold: (a) We unify learned view graph initialization and feed-forward reconstruction through shared backbone features, avoiding repeated feature extraction between the two tasks. (b) We develop graph-conditioned decoding that masks latent embeddings by inferred scene group to recover camera poses and dense geometry separately for disjoint scenes. (c) We

evaluate verification accuracy, reconstruction quality, and inference efficiency on uncurated image collections, and demonstrate *Graph3R* in a real-world multi-session mapping system.

2 Related Work

2.1 Feed-forward Reconstruction and Structure-from-Motion

Feed-forward reconstruction models recover scene geometry through learned multi-view representations. MASt3R (Leroy et al., 2024) combines pairwise pointmap prediction with learned descriptors for dense matching, while VGGT (Wang et al., 2025a) jointly predicts camera parameters and dense geometry from multiple views. π^3 (Wang et al., 2025b) introduces a permutation-equivariant architecture that removes the need for a designated reference view. These advances improve geometric inference, but reconstruction remains sensitive to input connectivity: unrelated views can corrupt both camera poses and scene geometry (Han et al., 2025).

GLUEMAP (Pan et al., 2026) combines feed-forward local reconstruction with global SfM optimization. It retrieves and verifies image pairs to construct a sparse view graph, applies π^3 to local graph neighborhoods, and aligns and refines the resulting reconstructions globally. The graph restricts local inference to selected views, but its initialization requires verification features computed separately from the reconstruction backbone. *Graph3R* integrates graph prediction into the reconstruction model and reuses the same backbone features for decoding each inferred scene group.

2.2 View Graph Initialization

View graph initialization depends on distinguishing covisible views from visually similar distractors. VPR methods (Keetha et al., 2023; Ye et al., 2023; Berton & Masone, 2025) retrieve candidate views using global descriptors, while local matchers (Edstedt et al., 2025) provide correspondences for geometric verification. GV-Bench (Yu et al., 2024) evaluates the reliability of this verification step under long-term appearance changes and perceptual aliasing, exposing limitations of decisions based on geometric inliers.

Learned verification methods address visual ambiguity explicitly. Doppelgangers (Cai et al., 2023) classifies image pairs using appearance and correspondence cues, and Doppelgangers++ (Xiangli et al., 2025) builds a classifier on MASt3R's geometric features. Its integration with MASt3R-SfM shares features for graph construction and pruning, with the verified graph guiding subsequent SfM reconstruction. OpenNavMap (Jiao et al., 2026) leverages the learned confidence map of MASt3R to reject outlier matches while it only focuses on pixel-level verification and does not provide a global view association. RobustVGGT (Han et al., 2025) provides a training-free alternative: it selects relevant views using internal attention scores or feature similarities, then reruns the backbone on the retained inputs.

Graph3R predicts pairwise covisibility with a learned verification head operating on shared multi-view features. The predicted graph conditions group-wise pose and geometry decoding from these retained features, coupling view association and reconstruction within one backbone pass. Table 8 in Appendix A.2 compares the association features, decision rules, and reconstruction stages of representative methods.

2.3 Multi-session Mapping

Multi-session mapping relies on reliable associations across sessions to align and merge maps. A typical pipeline retrieves candidate views from existing sessions and filters outliers using a learned verifier such as Doppelgangers++ (Xiangli et al., 2025) or geometric verification (Yu et al., 2024), before estimating relative poses. MASt3R-SLAM (Murai et al., 2025) and VGGT-SLAM (Maggio et al., 2026) combine feed-forward geometry with global map optimization for single-session mapping, while FLOCC-SLAM (Lajoie et al., 2025) and OpenNavMap (Jiao et al., 2026) apply feed-forward model for relative pose estimation across sessions. Specifically, FLOCC-SLAM uses these models for inter-robot loop closing. OpenNavMap (Jiao et al., 2026) utilizes the calibrated confidence map to reject outlier matches. MR.ScaleMaster (Ju & Kim, 2026) addresses inter-session scale ambiguity through a $\mathrm{Sim}(3)$ pose-graph backend. *Graph3R* complements these systems by

integrating view association into local reconstruction. Given images from disjoint scenes, it predicts a view graph and recovers camera poses and geometry separately for each inferred scene group. The resulting reconstructions support downstream alignment and map refinement.

3 METHOD

3.1 PROBLEM FORMULATION AND SYSTEM OVERVIEW

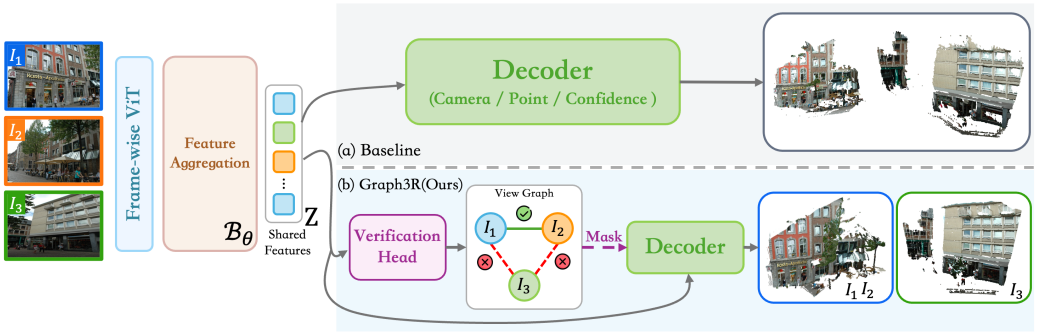

Figure 3: **System overview of *Graph3R*.** (a) The π^3 baseline (Wang et al., 2025b). (b) *Graph3R* introduces a verification head that predicts pairwise covisibility before geometry decoding. These predictions are used to construct a view graph, which guides the graph-conditioned reconstruction as described in Sec. 3.3, enabling each scene to be reconstructed individually.

Problem Formulation. Given an image collection $\mathcal{I} = \{\mathbf{I}_i\}_{i=1}^N$, our goal is to identify groups of views that should be reconstructed together and recover the corresponding camera poses and dense geometry. The collection may contain views from several disjoint scenes, and the number of scenes is unknown. We follow the classical SfM pipeline (Pan et al., 2024) and formulate the problem as a two-stage process: **(a) view graph initialization**, which predicts pairwise covisibility between images and constructs a view graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each vertex corresponds to an image and an edge indicates predicted geometric overlap, and **(b) graph-conditioned reconstruction**, which selects shared latent features by scene group and reconstructs each group separately.

System Overview. As illustrated in Fig. 3, ***Graph3R*** builds on π^3 (Wang et al., 2025b) and unifies the view verification and reconstruction with the geometric representation of the shared feature aggregation backbone. We denote the shared feature aggregation backbone by $\mathcal{B}_\theta$, which takes the input image collection $\mathcal{I}$ and produces shared features $\mathbf{Z} = \mathcal{B}_\theta(\mathcal{I}) \in \mathbb{R}^{N\times P\times D}$, where N is the number of views, P is the number of tokens per view, and D is the concatenated feature dimension. A verification head (described in Sec. 3.2) predicts pairwise covisibility from $\mathbf{Z}$ and constructs a view graph $\mathcal{G}$, whose connected components define the scene groups. For each group, we gather the corresponding view features from $\mathbf{Z}$ and pass them together to the geometry decoder, achieving graph-conditioned reconstruction (Sec. 3.3).

3.2 LEARNED VIEW GRAPH PREDICTION

Pairwise covisibility prediction. We adpot a similar design of the π^3's camera decoder for the verification head, with an additional prediction head to produce covisibility scores as illustrated in Fig. 4. In specific, the verification head takes the shared feature representation $\mathbf{Z}$ as input and processes it with a stack of transformer decoder blocks, followed by an MLP and pooling layer to generate a per-view latent descriptor $\mathbf{v}_i \in \mathbb{R}^C$. The pairwise prediction head h_ϕ maps each descriptor pair to a scalar logit, which is converted to a covisibility probability by a sigmoid:

$$S_{ij} = \sigma(h_\phi(\mathbf{v}_i, \mathbf{v}_j)), \qquad \mathbf{S} \in [0,1]^{N\times N}, \tag{1}$$

where σ is the sigmoid function and $\mathbf{S}$ is the pairwise covisibility score matrix. Here, S_{ij} estimates whether views i and j overlap geometrically, rather than merely belonging to the same scene.

View graph construction and scene grouping. We threshold the pairwise scores at τ_s to construct an undirected view graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, identifying vertices with view indices $\mathcal{V} = \{1, \ldots, N\}$.

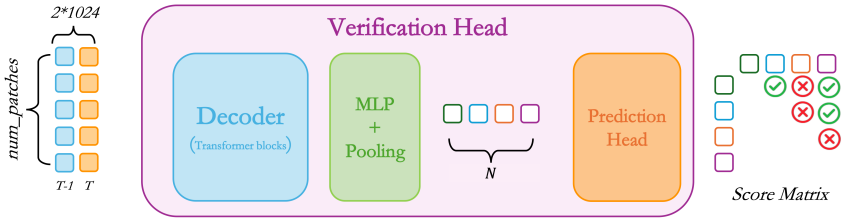

Figure 4: **Architecture of the verification head.** The head takes the shared feature representation $\mathbf{Z}$ as input and predicts pairwise covisibility scores $\mathbf{S}$. The scores are used to construct a view graph $\mathcal{G}$, which defines the mask of each scene group. The masks are applied to the shared representation $\mathbf{Z}$ prior to the geometry decoder, enabling graph-conditioned reconstruction for each scene group.

Let $\mathcal{V}_1, \ldots, \mathcal{V}_K \subseteq \mathcal{V}$ denote the vertex sets of its connected components, where K is the number of predicted components, including isolated vertices. The threshold determines accepted pairwise links, while connectivity determines the groups passed to reconstruction. Consequently, two views can belong to the same group through a path of overlapping views even when their direct score is below τ_s. For multi-view reconstruction, views with no accepted links to other images are treated as isolated inputs and excluded from graph-conditioned decoding.

3.3 Graph-Conditioned Reconstruction

The predicted view graph determines which views are decoded together. For each scene group $\mathcal{V}_k$ with $|\mathcal{V}_k| \geq 2$, we gather its view features from the shared representation along the view dimension:

$$\mathbf{Z}_k = \mathbf{Z}[\mathcal{V}_k, :, :] \in \mathbb{R}^{|\mathcal{V}_k| \times P \times D}. \tag{2}$$

All patch tokens of each selected view are retained, and the decoder receives only the features of that group. Cross-view attention during this decoding stage therefore operates within the selected group. We denote this decoding stage and its camera, point, and confidence predictors collectively by $\mathcal{D}_\omega$. The same decoder is applied separately to each group:

$$\mathcal{R}_k = \mathcal{D}_\omega(\mathbf{Z}_k) = \{(\mathbf{T}_i^{(k)}, \mathbf{P}_i^{(k)}, \mathbf{C}_i^{(k)})\}_{i \in \mathcal{V}_k}, \tag{3}$$

where $\mathbf{T}_i^{(k)}$ is a camera pose, $\mathbf{P}_i^{(k)}$ is a dense point map, and $\mathbf{C}_i^{(k)}$ is its confidence map. Outputs are associated with the original view indices used to gather $\mathbf{Z}_k$. Disjoint groups are reconstructed separately; any alignment between their reconstructions belongs to the downstream mapping stage.

The shared representation $\mathbf{Z}$ is computed once and retained throughout graph prediction and reconstruction. Each group reuses a subset of these jointly aggregated features, so group decoding does not require rerunning $\mathcal{B}_\theta$ on the original images. The computation therefore consists of one shared backbone pass, one verification-head evaluation, and decoding for each non-singleton group.

3.4 Training Objectives and Strategy

We train the verification head with two complementary objectives: a classification loss that supervises direct overlap between image pairs, and a contrastive loss that regularizes the shared view descriptors $\mathbf{v}_i$ using scene identity.

Classification loss. For each supervised pair $(i, j) \in \Omega$ within the input collection, let $t_{ij} \in \{0, 1\}$ indicate whether the two views have direct geometric overlap. Given the predicted covisibility probability S_{ij} from Eq. 1, we minimize binary cross-entropy:

$$\mathcal{L}_{\text{cls}} = -\frac{1}{|\Omega|} \sum_{(i,j) \in \Omega} \left[t_{ij} \log S_{ij} + (1 - t_{ij}) \log(1 - S_{ij}) \right]. \tag{4}$$

This loss supervises pairwise overlap rather than scene indentity: two views from the same scene can still form a negative pair if they do not overlap directly.

Contrastive loss. To complement pairwise overlap supervision, we apply a supervised contrastive loss (Khosla et al., 2020) to the same pooled descriptors $\mathbf{v}_i$ used by the pairwise prediction head, which is similar to visual place recongnition (Berton & Masone, 2025) methods. Positive samples are defined by ground truth scene identity y_i under each dataset's labeling strategy, rather than by direct overlap. For anchor i, we define $\mathcal{P}(i) = \{j \in \mathcal{V} \setminus \{i\} : y_j = y_i\}$ and retain anchors with at least one positive partner in $\mathcal{A} = \{i \in \mathcal{V} : |\mathcal{P}(i)| > 0\}$. Using normalized descriptors $\widehat{\mathbf{v}}_i = \mathbf{v}_i / \|\mathbf{v}_i\|_2$, the loss is

$$\mathcal{L}_{\text{con}} = -\frac{1}{N} \sum_{i \in \mathcal{A}} \frac{1}{|\mathcal{P}(i)|} \sum_{p \in \mathcal{P}(i)} \log \frac{\exp(\widehat{\mathbf{v}}_i^\top \widehat{\mathbf{v}}_p / \tau)}{\sum_{a \in \mathcal{V} \setminus \{i\}} \exp(\widehat{\mathbf{v}}_i^\top \widehat{\mathbf{v}}_a / \tau)}, \tag{5}$$

where τ is the temperature. The loss encourages descriptors from the same scene to align relative to other views in the collection. The definition of $\mathcal{P}$ for each dataset is provided in Appendix A.1.

Training strategy. We combine the two losses as

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda \mathcal{L}_{\text{con}}, \tag{6}$$

where λ controls the strength of contrastive regularization. We conduct exhaustive ablations studies as shown in Table 3 to estimate the optimal λ. We also explore the partial fine-tuning and full-layer LoRA (Hu et al., 2021) strategies for the backbone $\mathcal{B}_\theta$.

4 Experiments

We evaluate *Graph3R* through four complementary experiments: (a) controlled feature reuse to assess the effect of shared features on graph-conditioned reconstruction; (b) view graph initialization to measure association accuracy; (c) mixed-scene reconstruction to evaluate end-to-end geometric prediction; and (d) real-world multi-session mapping to assess cross-session localization and computational efficiency.

Datasets. We use GSV-Cities (Ali-bey et al., 2022) and 360Loc (Huang et al., 2024) to train the verification head and evaluate view graph initialization. ETH3D (Schops et al., 2017) provides high-fidelity ground-truth depth for controlled feature-reuse evaluation and supports multi-view reconstruction benchmarks under Easy and Hard overlap splits. OpenNavMap (Jiao et al., 2026) provides crowdsourced visual mapping data from two geographically distinct regions, with up to 55 sessions per region, for evaluating pose recovery and cross-session localization. The selection of these datasets and split details are provided in Appendix A.1.

Baselines. We use π^3 (Wang et al., 2025b) and VGGT (Wang et al., 2025a) as standard feed-forward reconstruction baselines. Neither explicitly verifies view associations, limiting their applicability to uncurated image collections. We also compare with GLUEMAP (Pan et al., 2026), a closely related system that constructs a view graph using Doppelgangers++ (Xiangli et al., 2025) and uses it to guide π^3 reconstruction. We denote its view graph initialization component by GLUEMAP†. For the reconstruction and pose estimation benchmarks, we omit GLUEMAP's post-optimization module for fair comparison with the direct feed-forward outputs.

Implementation details. We construct view graphs by retaining edges with $S_{ij} > \tau_s$ and partitioning the resulting graph into connected components, with $\tau_s = 0.5$ across all benchmarks. Following the ablation study in Sec. 4.2, we use full-layer LoRA, a GSV-Cities and 360Loc training mixture, and a contrastive loss weight of $\lambda = 0.25$. The training is conducted on 16 NVIDIA H200 GPUs, and evaluations are conducted on a single NVIDIA A800 GPU.

4.1 Controlled Feature Reuse

Graph3R aims to unify view graph initialization and feed-forward reconstruction through a shared feature representation. However, as shown in Fig. 3, views from unrelated scenes are processed together by $\mathcal{B}_\theta$, whose global attention aggregates across all input views. This aggregation may introduce cross-scene noise into the $\mathbf{Z}$ passed to the decoders, even when each scene is decoded

separately with masks. In principle, jointly training the backbone and geometry decoder with the verification head could help the model learn representations that are robust to such interference. This would require substantial training data and computational resources, which are beyond the scope of the question that *Graph3R* aims to answer. We therefore conduct a controlled experiment to assess how shared features affect graph-conditioned reconstruction.

We compare *Individual* and *Graph3R* on ETH3D (Schops et al., 2017), using identical π^3 weights and ground-truth scene groups to isolate the effect of feature sharing. *Individual* feeds each scene to the backbone and decoder independently, so no information is mixed across scenes in $\mathcal{B}_\theta$. In contrast, *Graph3R* processes the entire mixed-scene collection in a single backbone pass, then decodes each scene separately. We vary the number of scenes K from 2 to 4, more details are provided in Appendix A.3. As shown in Table 1, the degradation remains modest as K increases. This suggests that shared features retain sufficient geometric information for separate scene reconstruction despite cross scene interactions in the $\mathcal{B}_\theta$. Feature reuse also avoids a separate backbone pass for each scene, contributing to the efficiency of *Graph3R* as reported in Table 7, which can achieve nearly 30% speedup.

Table 1: **Controlled feature reuse on ETH3D.** We separately report AbsRel↓ and $\delta_1(\%)\uparrow$ for independent decoding and group decoding from one shared backbone representation. Δ reports relative percentage changes in each metric in comparison to the *Individual* baseline.

K	Individual		*Graph3R*		Δ (%)	
	AbsRel↓	$\delta_1(\%)\uparrow$	AbsRel↓	$\delta_1(\%)\uparrow$	AbsRel↓	$\delta_1(\%)\uparrow$
2	0.098	92.9	0.098	91.9	0.00	−1.08
3	0.097	93.4	0.097	92.1	0.00	−1.39
4	0.098	93.2	0.102	91.4	+4.08	−1.93

4.2 View Graph Initialization

As discussed in GLUEMAP (Pan et al., 2026), accurate view graph initialization is key to accurate and robust reconstruction. Following Doppelgangers++ (Xiangli et al., 2025) and GV-Bench (Yu et al., 2024), we evaluate view graph initialization as a pairwise classification task, using average precision (AP) and maximum recall at 100% precision (MR) as the evaluation metrics. Table 2 summarizes the results, with more detailed benchmarks provided in Sec. A.4. *Graph3R* achieves the best average AP and MR across the benchmarks. To achieve optimal performance with limited training resources, we conduct an ablation study to estimate the best training setup.

Table 2: **Benchmark of View Graph Initialization.**

Method	GSV-Cities		360Loc		ETH3D		Average	
	mAP ↑	MR ↑	mAP ↑	MR ↑	mAP ↑	MR ↑	mAP ↑	MR ↑
GLUEMAP† (Pan et al., 2026)	<u>91.17</u>	<u>27.29</u>	<u>96.13</u>	<u>74.79</u>	96.86	**93.17**	<u>94.72</u>	<u>65.08</u>
RobustVGGT (Han et al., 2025)	70.85	3.46	91.89	15.36	86.58	49.97	83.11	22.93
RobustPi3 (Han et al., 2025)	74.89	0.20	93.46	6.31	<u>97.76</u>	65.38	88.70	23.96
***Graph3R* (Ours)**	**98.98**	**47.17**	**99.65**	**83.64**	**99.38**	<u>91.41</u>	**99.34**	**74.07**

Ablation Study. Table 3 studies the training data mixture, contrastive loss weight, and trainable modules of the backbone aggregator $\mathcal{B}_\theta$. Mixing GSV-Cities and 360Loc improves generalization. With this mixture, contrastive regularization substantially increases MR, and full-layer LoRA achieves the best accuracy with fewer trainable parameters than partial finetuning. We therefore use the 4:1 data mixture, $\lambda = 0.25$, and full-layer LoRA in subsequent experiments.

4.3 Reconstruction

We assess how predicted view graph affects reconstruction quality in two aspects: multi-view depth consistency and pose estimation accuracy for uncurated images.

Multi-view depth consistency. We use MVS, AbsRel, and δ_1 to measure the consistency of predicted depths across views. With reference to the reprojection-based consistency checks in MVS-

Table 3: **Ablations of the verification head.** All the *Data Mixture* uses $\lambda = 0$, *Contrastive Loss Weight* and *Trainable Modules* use the 4:1 mixture, with LoRA and $\lambda = 0.25$ fixed, respectively. Params. counts include the verification head.

Data Mixture				Contr. Loss Weight			Trainable Modules			
Backbone	Data	AP↑	MR↑	λ	AP↑	MR↑	Backbone	Params.	AP↑	MR↑
Frozen	GSV	93.75	37.32	0	99.14	57.25	Frozen	69M	98.32	57.07
Frozen	4:1	98.11	**57.46**	0.1	99.22	64.29	↪ FT2	103M	97.97	53.44
LoRA	GSV	97.67	53.07	0.25	**99.31**	**73.06**	↪ FT4	128M	98.70	38.49
LoRA	4:1	**99.14**	57.25	0.5	99.09	54.82	↪ LoRA	80M	**99.31**	**73.06**

Net (Yao et al., 2018) and the multi-view consistency evaluation in GeoVideo (Bai et al., 2025), we define MVS as the reprojection-based cross-view depth-consistency score used in this experiment. Its detailed definition and aggregation protocol are provided in Appendix A.5. We conduct evaluation on ETH3D (Schops et al., 2017), using π^3 as the feed-forward reconstruction method for all methods for fair comparison. Table 4 shows that *Graph3R* achieves the best performance across all metrics in both *Easy* and *Hard* splits, improving average MVS from 0.182 to 0.223 over RobustPi3 while also reducing AbsRel and increasing δ_1. More detailed results including the benchmark construction are provided in Appendix A.5.

Table 4: **Multi-view depth consistency** on ETH3D (Schops et al., 2017).

Method	*Easy*			*Hard*			Average		
	MVS↑	AbsRel↓	δ_1(%)↑	MVS↑	AbsRel↓	δ_1(%)↑	MVS↑	AbsRel↓	δ_1(%)↑
RobustPi3 (Han et al., 2025)	0.230	0.287	73.6	0.133	0.394	65.9	0.1815	0.3405	69.75
GLUEMAP (Pan et al., 2026)	0.127	0.549	61.2	0.078	0.586	59.9	0.1025	0.5675	60.55
***Graph3R* (ours)**	**0.271**	**0.264**	**77.5**	**0.175**	**0.321**	**70.9**	**0.2230**	**0.2925**	**74.20**

Pose estimation accuracy. We evaluate the accuracy in Table 5 that requires each method to reject unrelated views and recover accurate poses from related ones. *Graph3R* achieves the best performance on ETH3D and under the 20°/5 m criterion on 360Loc, while remaining competitive with GLUEMAP on OpenNavMap. Appendix A.6 provides a more comprehensive discussion.

Table 5: **Benchmark of pose estimation accuracy.** We report the accuracy at different thresholds.

Method	ETH3D		OpenNavMap		360Loc	
	10°/1m	20°/5m	10°/1m	20°/5m	10°/1m	20°/5m
Pi3 (Wang et al., 2025b)	27.55	41.04	19.87	31.17	1.86	16.86
VGGT (Wang et al., 2025a)	8.43	22.99	8.89	28.60	0.25	12.06
RobustPi3 (Han et al., 2025)	73.08	85.71	60.92	68.36	**54.07**	64.71
RobustVGGT (Han et al., 2025)	54.90	62.82	48.13	70.63	49.75	62.75
GLUEMAP (Pan et al., 2026)	60.00	72.04	**67.83**	**79.72**	52.79	66.37
Graph3R(Ours)	**76.19**	**88.09**	67.65	75.86	53.82	**72.70**

4.4 Multi-session Mapping

Multi-session mapping requires reliable cross-session associations to accurately align and merge maps. Following the OpenNavMap, we assess this capability through candidate reranking (Table 6) and pose estimation (Table 5), and report computational efficiency in Table 7 and Fig. 5. For reranking, we first retrieve 8 candidates per query using CosPlace (Berton et al., 2022). *Graph3R* achieves the best reranking performance in both regions. In Table 7, s/batch denotes runtime in seconds per batch of 9 images (1 query and 8 retrieved candidates). *All groups* decodes every predicted connected component, whereas *Query group* decodes only the component containing the query image. *Graph3R* is the fastest method in both settings, with approximately **27%** and **20%** lower latency, respectively, than the next-fastest method. In addition, *Graph3R* achieves competitive performance with GLUEMAP in pose estimation on OpenNavMap, with runtime speedups of approximately **29.3**× and **35.9**× over GLUEMAP when decoding all predicted groups and only the

query group, respectively. Although its peak GPU memory usage is lower than that of RobustVGGT, it remains considerably higher than that of the other baselines. Full benchmark statistics and details are provided in Appendix A.8. Additional multi-session mapping visualizations are provided in Appendix A.10.

Table 6: **Cross-session localization on OpenNavMap.** All methods rerank on CosPlace retrieved candidates.

Method	*Region A*				*Region B*			
	R@1	R@3	R@5	R@8	R@1	R@3	R@5	R@8
CosPlace (Berton et al., 2022)	63.93	69.48	73.01	76.04	66.05	71.60	73.95	76.49
RobustPi3 (Han et al., 2025)	65.83	73.77	75.28	–	70.35	74.30	75.56	–
RobustVGGT (Han et al., 2025)	71.25	74.65	75.28	–	71.35	74.21	75.26	–
GLUEMAP† (Pan et al., 2026)	72.26	75.41	75.91	–	71.33	73.74	75.19	–
Graph3R(Ours)	**74.15**	**75.54**	**76.04**	–	**72.88**	**75.00**	**75.79**	–

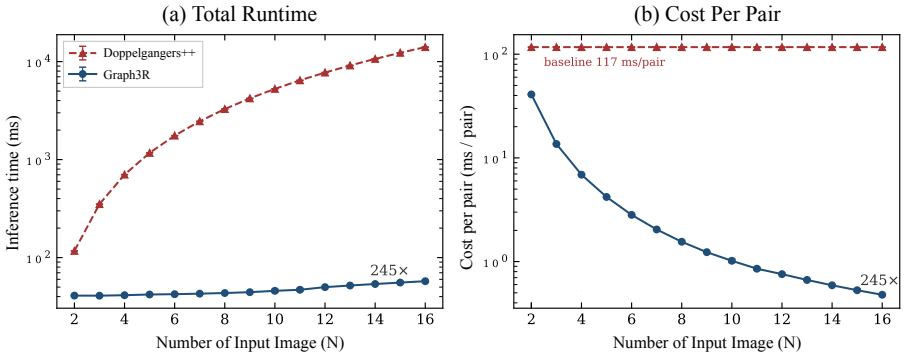

Figure 5: **Efficiency comparison** with Doppelgangers++ (Xiangli et al., 2025). Our method achieves approximately 245 ×speedup as the number of input images $N = 16$.

Table 7: **Computational efficiency on OpenNavMap.** End-to-end latency and peak GPU memory (GB) for $N = 9$ images per batch, decoding all predicted groups or only the group containing the query.

Method	All groups		Query group	
	s/batch ↓	Peak Memory ↓	s/batch ↓	Peak Memory ↓
RobustPi3 (Han et al., 2025)	0.218	4.727	0.158	4.727
RobustVGGT (Han et al., 2025)	0.693	11.065	0.421	11.065
GLUEMAP (Pan et al., 2026)	4.663	**4.635**	4.564	**4.637**
Graph3R (Ours)	**0.159**	8.087	**0.127**	8.087

5 Conclusion

We presented *Graph3R*, which unifies view graph initialization and feed-forward reconstruction through shared geometric features to reconstruct image collections containing disjoint scenes. A learned verification head predicts pairwise covisibility, and the resulting graph guides separate pose and geometry decoding for each scene group using features from a single backbone pass. Controlled experiments show that feature reuse incurs only a modest loss in depth accuracy, while verification and reconstruction benchmarks demonstrate the benefits of learned grouping for uncurated inputs. On OpenNavMap, *Graph3R* improves cross-session localization and achieves competitive pose estimation accuracy with GLUEMAP, with approximately $35.9\times$ runtime speedup when decoding only the query group. It achieves the lowest latency among the compared methods under the evaluated configurations, supporting its use as a front end for multi-session mapping. Grouping errors can still affect reconstruction by merging unrelated views or separating connected ones. Future work will explore grouping strategies that account for prediction uncertainty and more diverse training data to improve generalization across scenes and capture conditions. With sufficient computational resources, we also plan to jointly train the backbone and geometry decoder alongside the verification head on mixed-scene image collections, aiming to preserve reconstruction accuracy despite feature mixing across unrelated scenes.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on research methodology and experimental design, and to assist with the discussion and presentation of experimental results. We additionally used generative AI for literature search and summarization, language editing, debugging, code refinement, and the refinement of figures. We did not use generative AI to generate experimental data, formulate mathematical claims or proofs, or autonomously conduct experiments. All AI-assisted suggestions were critically reviewed by the authors: methodological and experimental decisions were validated through our own implementation and experiments, literature suggestions were checked against the original sources, and all AI-assisted text and figures were manually reviewed and revised. All AI-assisted code was manually reviewed, tested for correctness, and validated through experiments by the authors. We take responsibility for the final content of this work, including all text, claims, results, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We document the design and training of *Graph3R* in Sec. 3, including the verification head, view graph construction, graph-conditioned decoding, and training objectives. Sec. 4 describes the baselines and implementation settings, while Appendix A.1 summarizes the datasets and their roles in training and evaluation. Appendices A.3–A.6 detail the controlled feature-reuse protocol, verification and ablation experiments, and the depth-consistency and pose-estimation metrics, together with expanded results. Appendices A.7 and A.8 describe the cross-session localization protocol and runtime and GPU memory measurements, including the reconstruction scopes and baseline measurement procedures. The supplementary pairwise verification setting is documented in Appendix A.9. Upon acceptance of the paper, we will open-source the full source code, training and evaluation scripts, and configuration files to support independent reproduction and further research.

REFERENCES

Amar Ali-bey, Brahim Chaib-draa, and Philippe Giguère. Gsv-cities: Toward appropriate supervised visual place recognition. *Neurocomputing*, 513:194–203, 2022.

Yunpeng Bai, Shaoheng Fang, Chaohui Yu, Fan Wang, and Qixing Huang. GeoVideo: Introducing geometric regularization into video generation model. *arXiv preprint arXiv:2512.03453*, 2025.

Gabriele Berton and Carlo Masone. Megaloc: One retrieval to place them all. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 2886–2892, June 2025.

Gabriele Berton, Carlo Masone, and Barbara Caputo. Rethinking visual geo-localization for large-scale applications. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4868–4878. IEEE, 2022.

Ruojin Cai, Joseph Tung, Qianqian Wang, Hadar Averbuch-Elor, Bharath Hariharan, and Noah Snavely. Doppelgangers: Learning to disambiguate images of similar structures. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 34–44, 2023.

Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, José MM Montiel, and Juan D Tardós. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE transactions on robotics*, 37(6):1874–1890, 2021.

Johan Edstedt, David Nordström, Yushan Zhang, Georg Bökman, Jonathan Astermark, Viktor Larsson, Anders Heyden, Fredrik Kahl, Mårten Wadenbäck, and Michael Felsberg. Roma v2: Harder better faster denser feature matching. *arXiv preprint arXiv:2511.15706*, 2025.

Jisang Han, Sunghwan Hong, Jaewoo Jung, Wooseok Jang, Honggyu An, Qianqian Wang, Seungryong Kim, and Chen Feng. Emergent outlier view rejection in visual geometry grounded transformers. *arXiv preprint arXiv:2512.04012*, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Huajian Huang, Changkun Liu, Yipeng Zhu, Hui Cheng, Tristan Braud, and Sai-Kit Yeung. 360loc: A dataset and benchmark for omnidirectional visual localization with cross-device queries. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22314–22324. IEEE, 2024.

Jianhao Jiao, Changkun Liu, Jingwen Yu, Boyi Liu, Qianyi Zhang, Yue Wang, and Dimitrios Kanoulas. Opennavmap: Structure-free topometric mapping via large-scale collaborative localization. *arXiv preprint arXiv:2601.12291*, 2026.

Hyoseok Ju and Giseop Kim. MR.ScaleMaster: Scale-consistent collaborative mapping from crowdsourced monocular videos. *arXiv preprint arXiv:2604.11372*, 2026.

Nikhil Keetha, Avneesh Mishra, Jay Karhade, Krishna Murthy Jatavallabhula, Sebastian Scherer, Madhava Krishna, and Sourav Garg. Anyloc: Towards universal visual place recognition. *IEEE Robotics and Automation Letters*, 9(2):1286–1293, 2023.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Pierre-Yves Lajoie, Benjamin Ramtoula, Daniele De Martini, and Giovanni Beltrame. 3d foundation model-based loop closing for decentralized collaborative slam. *IEEE Robotics and Automation Letters*, 2025.

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. In *European conference on computer vision*, pp. 71–91. Springer, 2024.

Dominic Maggio, Hyungtae Lim, and Luca Carlone. Vggt-slam: Dense rgb slam optimized on the sl (4) manifold. *Advances in Neural Information Processing Systems*, 38:129839–129867, 2026.

Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: A versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015.

Riku Murai, Eric Dexheimer, and Andrew J Davison. Mast3r-slam: Real-time dense slam with 3d reconstruction priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16695–16705, 2025.

Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision*, pp. 58–77. Springer, 2024.

Linfei Pan, Johannes Lutz Schönberger, and Marc Pollefeys. Global structure-from-motion meets feedforward reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3260–3269, 2017.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025a.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025b.

Yuanbo Xiangli, Ruojin Cai, Hanyu Chen, Jeffrey Byrne, and Noah Snavely. Doppelgangers++: Improved visual disambiguation with geometric 3d features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. MVSNet: Depth inference for unstructured multi-view stereo. In *European Conference on Computer Vision*, pp. 767–783, 2018.

Hanjing Ye, Weinan Chen, Jingwen Yu, Li He, Yisheng Guan, and Hong Zhang. Condition-invariant and compact visual place description by convolutional autoencoder. *Robotica*, 41(6):1718–1732, 2023.

Jingwen Yu, Hanjing Ye, Jianhao Jiao, Ping Tan, and Hong Zhang. Gv-bench: Benchmarking local feature matching for geometric verification of long-term loop closure detection. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7922–7928. IEEE, 2024.

A Supplementary Material

A.1 Datasets

We distinguish the datasets used to train the multi-view verifier from those used for geometric evaluation and the supplementary pairwise study. The descriptions below specify their roles and the evaluation subsets reported in the paper; task-specific metrics are defined in the corresponding experimental sections.

GSV-Cities. GSV-Cities (Ali-bey et al., 2022) provides place-labeled imagery for supervised visual place recognition. We use it for multi-view verification training and validation. Table 2 reports the Full validation split, while Table 9 additionally reports Easy, Medium, and Hard. The selected training configuration mixes GSV-Cities and 360Loc at a 4:1 ratio; the data-composition ablation compares this mixture with GSV-Cities alone.

360Loc. 360Loc (Huang et al., 2024) is an omnidirectional visual-localization dataset with cross-device queries. It supplies the second source of multi-view verification training data and a separate verification evaluation reported alongside GSV-Cities. The pose experiment uses the atrium subset, with results in Table 13. The verification and atrium pose evaluations are distinct tasks and should not be interpreted as using an identical sample set.

ETH3D. ETH3D (Schops et al., 2017) is used to evaluate controlled feature reuse, view-graph prediction, depth consistency, and pose recovery. For the controlled depth and graph experiments, each collection contains $K \in \{2, 3, 4\}$ scenes with four views per scene, giving $4K$ input images. Each setting uses 20 fixed episodes. Easy (clique) requires pairwise within-scene overlap; Hard (connected) requires a connected within-scene overlap graph without requiring all pairs to overlap. Inputs are evaluated at 224×224 resolution. Appendix A.3 describes the oracle-group GT-depth comparison, and Appendix A.5 describes predicted-group cross-view consistency.

OpenNavMap. OpenNavMap (Jiao et al., 2026) supplies data from Region A and Region B (Aria) for pose estimation evaluation, cross-session localization, and runtime measurement. For pose estimation and cross-session localization, positive matches require a ground-truth camera-center distance of at most 25 m and relative orientation of at most 40°. Cross-session localization reranks the same eight CosPlace candidates for every method. Region-specific results and their equal-weight macro-average are reported in Table 14. The runtime protocol uses nine-image batches, as detailed in Appendix A.8.

VisymScenes and Doppelgangers. These datasets are used for the supplementary two-image verification study, not the GSV-Cities/360Loc multi-view training mixture. Doppelgangers++ and our pairwise variants are trained on a combined 210K pairs from VisymScenes (Xiangli et al., 2025) and Doppelgangers (Cai et al., 2023). VisymScenes provides the in-domain evaluation in Table 17.

GV-Bench. GV-Bench (Yu et al., 2024) is used for zero-shot evaluation of the pairwise models trained above. Table 18 reports Day, Night, Weather, Season, Nordland, and Uacampus conditions, together with their unweighted mean. This experiment tests transfer beyond the pairwise training data; it is separate from the main multi-view verification benchmark.

A.2 Comparison of Different SfM Paradigms

Table 8 complements Fig. 2 with the association features, decision rules, and reconstruction stages of representative methods. Feature context denotes the images used to compute association features, not the input requirements of the reconstruction backbone.

A.3 Controlled Feature Reuse

Table 1 uses the ETH3D Easy (clique) setting with $K \in \{2, 3, 4\}$ scenes, four views per scene, and 20 fixed episodes per setting. Both variants use oracle scene groups and identical π^3 weights at 224×224 resolution. Independent reconstruction computes features separately for each scene; shared

Table 8: **View association and subsequent reconstruction.** DG denotes Doppelgangers; feature context refers to association.

Method	Feature context	Association	Reconstruction
MegaLoc	Per-image	Similarity	External stage
RoMAv2 + RANSAC	Pairwise	Geometric verification	External stage
DG / DG++	Pairwise	Learned classifier	SfM
RobustVGGT	Multi-view	Attention / features	Backbone rerun
GLUEMAP	Image + pair	Retrieval + verification	Local inference + global optimization
Graph3R (ours)	Multi-view	Learned classifier	Group decoding from retained features

reconstruction gathers each group's features from a single mixed-collection backbone pass. Depth is evaluated against ground truth with per-scene median-ratio scale alignment, a shared GT mask, and equal episode weighting. This GT-depth evaluation is distinct from the cross-view consistency evaluation in Appendix A.5.

A.4 View Graph Initialization

Table 9 expands Table 2 with all four GSV-Cities validation splits and the attention-only ($\mathcal{A}$) and feature-only ($\mathcal{F}$) robust baselines. Its Average weights the four GSV-Cities columns and 360Loc equally; it is not the three-dataset average in the main table, which uses GSV-Cities Full, 360Loc, and ETH3D. The ETH3D breakdown is provided in Table 12.

Table 9: **Full benchmark of view graph initialization.** We report mean average precision (mAP) and maximum recall at 100% precision (MR) for every GSV-Cities validation split and 360Loc, together with the overall average.

	Method	GSV-Cities								360Loc		Average	
		Easy		Medium		Hard		Full					
		mAP ↑	MR ↑	mAP ↑	MR ↑	mAP ↑	MR ↑	mAP ↑	MR ↑	mAP ↑	MR ↑	mAP ↑	MR ↑
Pairwise	RoMAv2 (Edstedt et al., 2025)	95.26	3.63	91.64	1.62	87.89	0.21	91.60	0.16	68.78	3.13	87.03	1.75
Pairwise	GLUEMAP† (Pan et al., 2026)	**99.48**	45.87	91.33	44.69	88.39	26.98	91.17	27.29	96.13	74.79	92.26	43.92
Multi-view	RobustVGGT-$\mathcal{A}$	72.09	12.99	41.98	10.98	25.68	2.38	46.13	2.58	53.97	0.00	47.97	5.79
Multi-view	RobustVGGT-$\mathcal{F}$	89.59	3.43	68.76	0.41	49.10	0.12	68.81	0.24	91.88	2.09	73.63	1.26
Multi-view	RobustVGGT (Han et al., 2025)	91.32	19.85	69.56	10.17	49.08	3.17	70.85	3.46	91.89	15.36	74.54	10.40
Multi-view	RobustPi3-$\mathcal{A}$	84.71	0.88	63.05	1.11	43.11	0.04	60.32	0.03	82.59	6.09	66.76	1.63
Multi-view	RobustPi3-$\mathcal{F}$	91.87	0.00	78.88	0.16	61.86	0.03	75.28	0.01	93.48	6.39	80.27	1.32
Multi-view	RobustPi3 (Han et al., 2025)	93.34	0.10	78.67	0.52	60.46	0.35	74.89	0.20	93.46	6.31	80.16	1.50
Multi-view	Ours	99.43	**80.33**	**99.18**	**79.73**	**98.25**	**42.72**	**98.98**	**47.17**	**99.65**	**83.64**	**99.10**	**66.72**

Ablation Study. Table 10 gives per-split results for the configurations in Table 3. The data-composition comparison uses classification loss alone ($\lambda = 0$); the contrastive-weight comparison uses full-layer LoRA and the 4:1 mixture; the adaptation comparison uses the 4:1 mixture and $\lambda = 0.25$. FT2 and FT4 update the last two and four backbone layers, respectively. The reported checkpoints have different step counts, so these results compare the saved configurations rather than a matched-step training budget.

Table 10: **Per-split results for the main ablations.** Each metric entry is AP/MR@100P (%). Data is the GSV-Cities:360Loc ratio; $\lambda = 0$ disables the contrastive term. Steps identify the saved checkpoints. The main-table mean weights the mean over four GSV-Cities splits and 360Loc equally. Configurations shared by multiple ablations appear only once.

Adaptation	Data	λ	Steps	GSV-Cities				360Loc
				Easy	Medium	Hard	Full	
Frozen	1:0	0.0	9840	99.45/67.62	99.55/58.38	98.38/39.16	98.59/19.58	88.51/28.46
Frozen	4:1	0.0	4800	99.12/58.53	98.10/44.05	94.78/23.66	97.00/19.93	98.96/78.37
LoRA	1:0	0.0	7600	99.83/84.86	99.69/90.41	98.92/66.67	99.14/54.01	95.94/32.15
LoRA	4:1	0.0	7200	99.17/41.04	99.10/42.44	97.93/17.81	98.69/18.56	99.56/84.54
LoRA	4:1	0.1	9840	99.08/75.44	99.13/77.92	98.07/8.48	98.70/8.33	99.69/86.04
LoRA	4:1	0.25	11800	99.43/80.33	99.18/79.73	98.25/42.72	98.98/47.17	99.65/83.64
LoRA	4:1	0.5	8610	99.35/44.50	98.98/34.70	97.84/14.18	97.84/16.30	99.68/82.22
Frozen	4:1	0.25	7380	99.06/64.16	98.25/24.70	95.11/24.70	97.50/23.42	99.15/79.90
FT last 2	4:1	0.25	4305	98.86/57.11	97.50/22.53	94.28/14.31	97.08/16.18	99.01/79.35
FT last 4	4:1	0.25	11685	99.43/73.64	99.05/18.87	97.20/28.36	98.35/21.27	98.89/41.44

A.5 Multi-view Depth Consistency

Table 11 extends Table 4 with RobustVGGT as an additional grouping scorer. Table 12 provides the per-K breakdown, including raw-score AP/MR and thresholded edge F1. All scorers share π^3 reconstruction weights and features; only the predicted grouping changes. Easy requires pairwise within-scene overlap, whereas Hard requires only a connected within-scene overlap graph.

MVS definition. We evaluate cross-view geometric consistency using a reprojection-based score, denoted by MVS. For a source pixel $\mathbf{p}$ in view i, we reproject its predicted depth into view j using the predicted relative pose $\mathbf{T}_j^{-1}\mathbf{T}_i$ and the camera intrinsics recovered from the predicted point maps. Let $\mathbf{u}_{i\to j}(\mathbf{p})$ and $z_{i\to j}(\mathbf{p})$ denote the projected location and the transformed depth in the target camera, respectively. We bilinearly sample the target depth prediction as

$$\widehat{D}_j(\mathbf{p}) = D_j(\mathbf{u}_{i\to j}(\mathbf{p}))\,. \tag{7}$$

The directional consistency score is

$$c_{i\to j} = \frac{1}{H_i W_i} \sum_{\mathbf{p}\in\Omega_i} \mathbf{1}\left[v_{i\to j}(\mathbf{p}) \wedge \left| \frac{z_{i\to j}(\mathbf{p}) - \widehat{D}_j(\mathbf{p})}{\widehat{D}_j(\mathbf{p})} \right| < 0.05 \right], \tag{8}$$

where Ω_i is the set of all $H_i W_i$ source pixels and $v_{i\to j}(\mathbf{p})$ requires a positive source depth and a projection inside the target image. Invalid projections and undefined depth comparisons contribute zero. The normalization therefore uses all source pixels, rather than only valid projections.

For an input episode e, let $\mathcal{E}_e$ contain all unordered view pairs within the predicted connected components. We compute

$$\mathrm{MVS}_e = \frac{1}{|\mathcal{E}_e|} \sum_{\{i,j\}\in\mathcal{E}_e} \frac{c_{i\to j} + c_{j\to i}}{2}. \tag{9}$$

We report the unweighted mean over episodes containing at least one retained pair. Higher scores indicate greater cross-view consistency. MVS uses only predicted geometry, without ground-truth depth or ground-truth scale alignment. Because the retained pair set depends on the predicted grouping, we interpret MVS together with the corresponding view-graph prediction metrics.

The depth metrics measure agreement between predicted depths across views, not accuracy against GT depth. AbsRel and δ_1 use valid reprojections. Retained pair sets differ between methods, so consistency should be read alongside graph-prediction quality rather than as a coverage measure. RobustVGGT attains the best Easy consistency and average MVS/δ_1, while *Graph3R* achieves the lowest average AbsRel.

Table 11: **ETH3D depth consistency benchmark.** Average is the equal-weight mean of the displayed Easy and Hard results. Bold and underline mark the best and second-best results in each column, respectively.

Grouping scorer	Easy			Hard			Average		
	MVS↑	AbsRel↓	δ_1(%)↑	MVS↑	AbsRel↓	δ_1(%)↑	MVS↑	AbsRel↓	δ_1(%)↑
RobustPi3 (Han et al., 2025)	0.230	0.287	73.6	0.133	0.394	65.9	0.1815	0.3405	69.75
RobustVGGT	**0.290**	**0.239**	**78.8**	<u>0.157</u>	<u>0.393</u>	**71.0**	**0.2235**	<u>0.3160</u>	**74.90**
GLUEMAP (Pan et al., 2026)	0.127	0.549	61.2	0.078	0.586	59.9	0.1025	0.5675	60.55
***Graph3R* (ours)**	<u>0.271</u>	<u>0.264</u>	<u>77.5</u>	**0.175**	**0.321**	<u>70.9</u>	<u>0.2230</u>	**0.2925**	<u>74.20</u>

Table 12: **View graph initialization and multi-view depth consistency on ETH3D.** Each setting contains 20 fixed episodes with K scenes and four views per scene. Easy (clique) requires pairwise within-scene overlap; Hard (connected) requires a connected overlap graph. Only the grouping scorer varies: reconstruction uses the same π^3 weights, shared features, and grouped decoding at 224×224.

K	Grouping scorer	View graph initialization (%)			Depth consistency		
		AP↑	MR↑	F1↑	MVS↑	AbsRel↓	δ_1(%)↑
(a) Easy (clique)							
2	RobustPi3	99.74	95.42	93.57	0.350	0.189	83.1
	RobustVGGT	99.86	89.58	70.62	**0.388**	**0.181**	**84.2**
	GLUEMAP	97.99	93.75	79.86	0.216	0.360	72.4
	***Graph3R* (ours)**	**99.92**	**99.17**	**98.09**	0.339	0.191	82.8
3	RobustPi3	97.03	63.06	89.56	0.214	0.272	75.4
	RobustVGGT	92.22	33.33	60.48	**0.293**	**0.247**	**80.4**
	GLUEMAP	96.80	93.33	66.04	0.109	0.570	61.2
	***Graph3R* (ours)**	**99.22**	**94.17**	**94.21**	0.241	0.330	75.7
4	RobustPi3	96.31	49.17	82.02	0.127	0.401	62.3
	RobustVGGT	76.03	25.00	47.98	0.189	0.290	71.7
	GLUEMAP	96.06	92.92	57.78	0.057	0.718	50.0
	***Graph3R* (ours)**	**99.54**	**94.58**	**98.34**	**0.234**	**0.270**	**74.0**
(b) Hard (connected)							
2	RobustPi3	**99.95**	**97.39**	96.27	**0.222**	**0.277**	**77.7**
	RobustVGGT	99.47	94.77	68.10	0.194	0.282	76.3
	GLUEMAP	97.54	92.16	75.25	0.132	0.350	73.3
	***Graph3R* (ours)**	99.92	94.77	**99.03**	0.213	0.281	77.1
3	RobustPi3	97.79	52.16	85.33	0.115	0.411	64.5
	RobustVGGT	79.00	31.47	55.08	0.148	**0.326**	**71.1**
	GLUEMAP	96.19	**92.67**	57.47	0.064	0.593	59.1
	***Graph3R* (ours)**	**99.34**	87.50	**96.51**	**0.165**	0.330	70.8
4	RobustPi3	95.75	35.06	74.57	0.061	0.493	55.6
	RobustVGGT	72.91	25.65	50.42	0.128	0.571	**65.7**
	GLUEMAP	96.57	**94.16**	49.55	0.037	0.816	47.2
	***Graph3R* (ours)**	**98.34**	78.25	**91.92**	**0.145**	**0.352**	64.6

A.6 Pose Estimation Accuracy

Table 13 restores the relative-pose metrics omitted from Table 5, under the same strict score > 0.5 verification protocol. AP-RPE is the median defined angular error on accepted positive pairs. SA-RPE is the class-balanced system-aware mean: correctly rejected negatives receive $0°$, and decision or pose failures receive $180°$. Rotation and translation-direction errors are reported separately. System accuracy is $\frac{1}{2}(\mathrm{TNR} + \text{pose recall})$, with pose recall evaluated at the joint rotation/translation thresholds shown in the table. Thus, AP-RPE measures conditional pose quality, whereas SA-RPE and system accuracy also account for verification failures.

OpenNavMap positives satisfy a GT camera-center distance of at most 25 m and relative orientation of at most $40°$. Pi3, RobustPi3, and GLUEMAP use Pi3 geometry, as does *Graph3R*; VGGT and RobustVGGT use VGGT geometry.

Table 13: **Full pose estimation accuracy.** Supplement to Table 5, using the same strict score > 0.5 verification rule. AP-RPE and SA-RPE are defined in Appendix A.6. Bold and underline mark the best and second-best results per dataset and metric.

Method	AP-RPE median ↓		SA-RPE mean ↓		System accuracy (%) ↑	
	Rot. (°)	Dir. (°)	Rot. (°)	Dir. (°)	10°/1 m	20°/5 m
ETH3D						
Pi3 (Wang et al., 2025b)	4.67	16.04	99.60	107.58	27.55	41.04
VGGT (Wang et al., 2025a)	14.76	43.25	117.39	126.79	8.43	22.99
RobustPi3 (Han et al., 2025)	**4.46**	<u>15.52</u>	<u>20.70</u>	<u>28.53</u>	<u>73.08</u>	<u>85.71</u>
RobustVGGT (Han et al., 2025)	8.57	25.90	61.44	65.45	54.90	62.82
GLUEMAP (Pan et al., 2026)	4.52	**15.49**	45.05	52.33	60.00	72.04
Graph3R (Ours)	<u>4.48</u>	15.56	**15.39**	**23.02**	**76.19**	**88.09**
OpenNavMap						
Pi3 (Wang et al., 2025b)	3.01	5.84	107.10	111.51	19.87	31.17
VGGT (Wang et al., 2025a)	5.83	15.10	109.70	117.43	8.89	28.60
RobustPi3 (Han et al., 2025)	**2.65**	<u>5.41</u>	54.27	57.74	60.92	68.36
RobustVGGT (Han et al., 2025)	5.50	14.66	45.07	51.98	48.13	70.63
GLUEMAP (Pan et al., 2026)	<u>2.74</u>	**5.21**	**32.18**	**36.71**	**67.83**	**79.72**
Graph3R (Ours)	2.81	6.46	<u>41.18</u>	<u>45.22</u>	<u>67.65</u>	<u>75.86</u>
360Loc atrium						
Pi3 (Wang et al., 2025b)	12.18	21.11	125.04	126.58	1.86	16.86
VGGT (Wang et al., 2025a)	15.26	31.13	127.35	134.89	0.25	12.06
RobustPi3 (Han et al., 2025)	**5.52**	**10.93**	60.97	62.61	**54.07**	64.71
RobustVGGT (Han et al., 2025)	12.22	27.45	57.62	61.12	49.75	62.75
GLUEMAP (Pan et al., 2026)	<u>7.95</u>	<u>13.75</u>	<u>54.09</u>	<u>55.47</u>	52.79	<u>66.37</u>
Graph3R (Ours)	10.48	18.76	**38.46**	**39.56**	<u>53.82</u>	**72.70**

A.7 Multi-session Mapping

Table 14 expands Table 6 with the macro-average over Region A and Region B. All methods rerank the same eight CosPlace candidates, so R@8 is a shared retrieval ceiling, not a verifier-dependent result. Correctness uses the same 25 m/40° criterion as the OpenNavMap pose estimation evaluation. These experiments evaluate cross-session localization for mapping; they do not measure final map-merging or trajectory-alignment accuracy.

Table 14: Cross-session localization on OpenNavMap. CosPlace retrieves the top eight candidates per query, which each method reranks using its saved verification scores. Recall@K is the fraction of answerable queries with at least one correct result among the top K; correctness uses ground-truth camera-center distance ≤ 25 m and relative orientation $\leq 40°$. All entries are percentages. The CosPlace R@8 columns show the retrieval ceiling shared by every reranker. OpenNavMap is the equal-weight mean of Region A and Region B.

	Region A				Region B				OpenNavMap (macro)			
Ranking	R@1	R@3	R@5	R@8	R@1	R@3	R@5	R@8	R@1	R@3	R@5	R@8
CosPlace	63.93	69.48	73.01	76.04	66.05	71.60	73.95	76.49	64.99	70.54	73.48	76.26
RobustPi3	65.83	73.77	75.28	–	70.35	74.30	75.56	–	68.09	74.04	75.42	–
RobustVGGT	71.25	74.65	75.28	–	71.35	74.21	75.26	–	71.30	74.43	75.27	–
GLUEMAP†	72.26	75.41	75.91	–	71.33	73.74	75.19	–	71.79	74.58	75.55	–
Graph3R (Ours)	**74.15**	**75.54**	**76.04**	–	**72.88**	**75.00**	**75.79**	–	**73.52**	**75.27**	**75.92**	–

A.8 MULTI-SESSION MAPPING: COMPUTATIONAL EFFICIENCY

Tables 15 and 16 expand Table 7. Each batch contains nine images and 36 unordered pairs. Methods use their own scores and the same strict score > 0.5 connected-component rule. Output scopes are all predicted groups or only the group containing query image 0; predicted memberships can differ across methods. Equivalent pair throughput is 36 times batch throughput, with displayed values independently rounded. Peak memory subtracts each run's measured idle baseline and uses decimal GB (10^9 bytes); both average and maximum peaks are reported.

For GLUEMAP, Doppelgangers++ verification time and peak memory come from its original 36-pair scoring run; each group's Pi3 reconstruction was measured anew. RobustPi3 and RobustVGGT verification and reconstruction were measured anew for both scopes. *Graph3R* uses its validated dual-stream runs and cached base geometry features.

Table 15: **OpenNavMap runtime and peak GPU memory: all predicted groups.** Each batch contains nine images and 36 unordered pairs. Each method uses its own verification scores with the strict score > 0.5 connected-component rule. Equivalent pairs/s is 36 times batch/s. Average and maximum peak memory subtract each run's measured idle baseline and use decimal GB. Best and second-best results are bold and underlined, respectively.

Method	s/batch ↓	batch/s ↑	Equivalent pairs/s ↑	Avg peak GB ↓	Max peak GB ↓
RobustPi3	<u>0.218</u>	<u>4.588</u>	<u>165.180</u>	<u>4.727</u>	<u>4.727</u>
RobustVGGT	0.693	1.442	51.912	10.994	11.065
GLUEMAP	4.663	0.214	7.721	**4.633**	**4.635**
Graph3R (Ours)	**0.159**	**6.305**	**226.995**	8.087	8.087

Table 16: **OpenNavMap runtime and peak GPU memory: query group only.** The output scope is restricted to the connected component containing query image 0, using the same nine-image, 36-pair batches and strict score > 0.5 rule as Table 15. Equivalent pairs/s is 36 times batch/s. Average and maximum peak memory subtract each run's measured idle baseline and use decimal GB. Best and second-best results are bold and underlined, respectively.

Method	s/batch ↓	batch/s ↑	Equivalent pairs/s ↑	Avg peak GB ↓	Max peak GB ↓
RobustPi3	<u>0.158</u>	<u>6.327</u>	<u>227.773</u>	<u>4.727</u>	<u>4.727</u>
RobustVGGT	0.421	2.376	85.550	10.995	11.065
GLUEMAP	4.564	0.219	7.887	**4.634**	**4.637**
Graph3R (Ours)	**0.127**	**7.851**	**282.621**	8.087	8.087

A.9 View Graph Initialization: Pairwise Verification

For a two-image comparison, Doppelgangers++ and *Graph3R* are trained on 210K pairs from VisymScenes (Xiangli et al., 2025) and Doppelgangers (Cai et al., 2023), separately from the multi-view training setting in the main paper. Table 17 reports VisymScenes results and zero-shot transfer to GV-Bench (Yu et al., 2024); Table 18 gives the available per-condition breakdown. *Graph3R* improves VisymScenes AP and MR over Doppelgangers++, but this advantage does not carry over to GV-Bench, where MegaLoc has the best mean AP and MR. These results distinguish in-domain pairwise verification from cross-domain generalization.

Table 17: **Pairwise verification.** We evaluate on VisymScenes (Xiangli et al., 2025) and GV-Bench (Yu et al., 2024) with average precision (AP) and maximum recall at 100% precision (MR), both expressed as percentages. Avg. reports the mean of each metric across the two benchmarks. The first three methods are not trained on VisymScenes; the remaining methods are. Bold marks the best result per column.

Method	VisymScenes		GV-Bench		Avg.	
	AP ↑	MR ↑	AP ↑	MR ↑	AP ↑	MR ↑
MegaLoc (Berton & Masone, 2025)	91.17	26.42	**91.74**	**32.38**	91.46	29.40
RoMAv2 (Edstedt et al., 2025)	92.75	2.45	84.63	19.49	88.69	10.97
RobustVGGT (Han et al., 2025)	93.57	18.81	52.13	2.14	72.85	10.48
Doppelgangers (Cai et al., 2023)	97.89	38.21	73.95	17.28	85.92	27.75
Doppelgangers++ (Xiangli et al., 2025)	98.42	41.76	89.70	25.75	**94.06**	33.76
Graph3R (FT last 2)	**99.05**	**74.34**	81.45	11.18	90.25	**42.76**
Graph3R (LoRA)	98.72	60.92	85.17	15.79	91.95	38.36

Table 18: **Pairwise verification on GV-Bench (per condition).** We report AP and MR (%) across the six GV-Bench conditions; Avg. is the mean over conditions. Best result per column in **bold**.

Method	Day		Night		Weather		Season		Nordland		Uacampus		Avg.	
	AP ↑	MR ↑	AP ↑	MR ↑	AP ↑	MR ↑	AP ↑	MR ↑	AP ↑	MR ↑	AP ↑	MR ↑	AP ↑	MR ↑
MegaLoc	99.64	**57.84**	97.81	**28.58**	99.77	**49.26**	**99.81**	**48.72**	**77.32**	**3.15**	76.09	6.72	**91.74**	**32.38**
RoMAv2	**99.78**	50.31	**97.89**	13.47	99.82	14.89	99.75	7.33	29.30	0.11	**81.22**	**30.85**	84.63	19.49
Graph3R (FT last 2)	98.53	32.39	80.00	1.63	99.57	13.68	99.34	16.20	46.21	0.17	65.06	2.99	81.45	11.18
Graph3R (LoRA)	99.20	40.63	87.15	3.79	**99.83**	18.90	99.46	27.90	52.19	1.30	73.17	2.24	85.17	15.79
RobustVGGT	78.74	7.66	36.19	0.19	94.93	1.08	88.76	0.06	6.42	0.11	7.76	3.73	52.13	2.14

A.10 VISUALIZATION OF MULTI-SESSION MAPPING

Fig. 6 visualizes multi-session mapping results for three OpenNavMap cases, showing cross-session associations and the corresponding mapping results for each method.

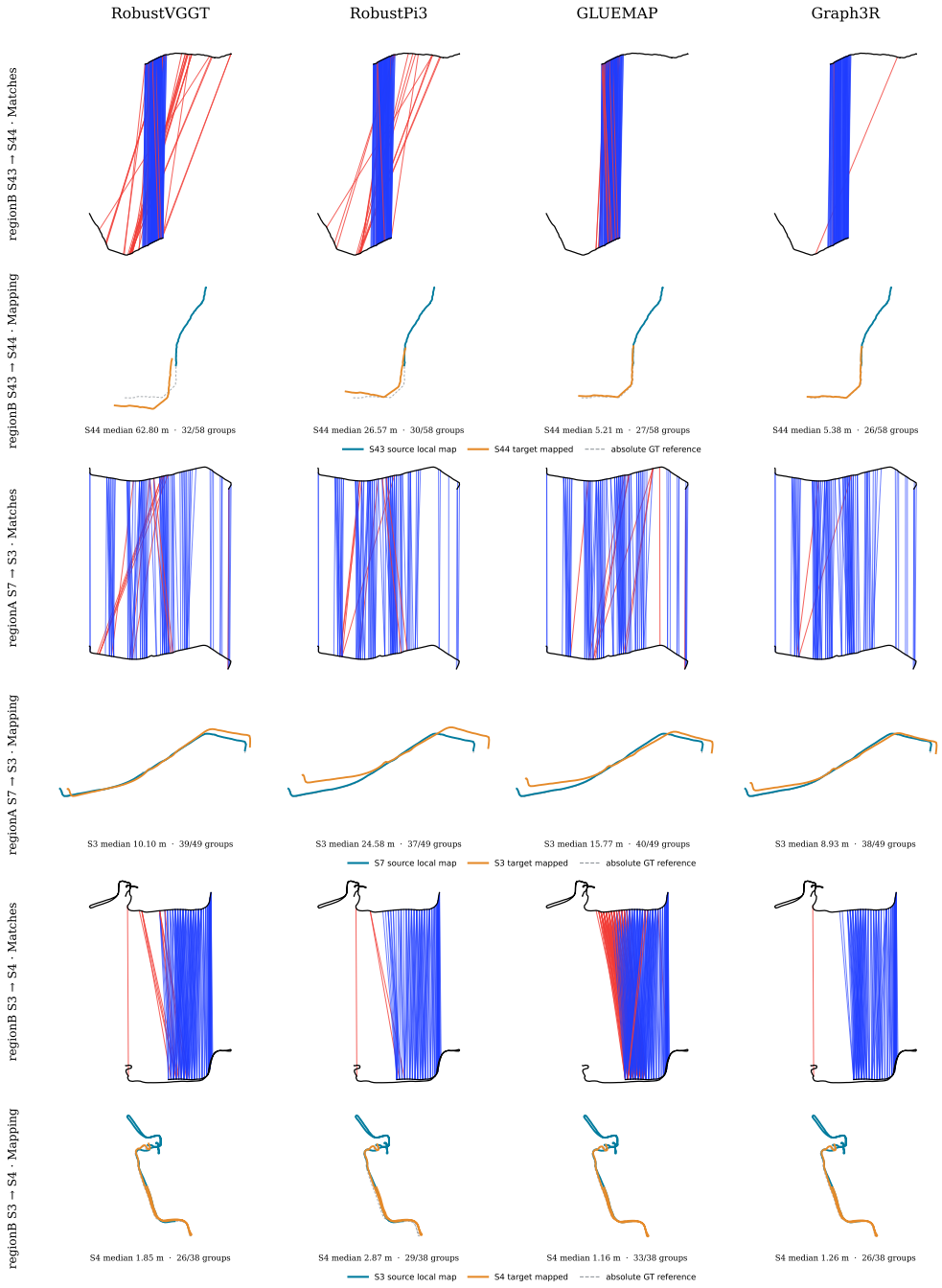

Figure 6: **Visualization of multi-session mapping on OpenNavMap.** Three cases are shown, each with cross-session associations above and the corresponding mapping results below.