



RUBIK: A Structured Benchmark for Image Matching across Geometric Challenges

Thibaut Loiseau[†] Guillaume Bourmaud[‡]

[†] LIGM, Ecole des Ponts, Université Gustave Eiffel, CNRS, France

[‡] Laboratoire IMS, Université de Bordeaux, France

thibaut.loiseau@enpc.fr guillaume.bourmaud@u-bordeaux.fr

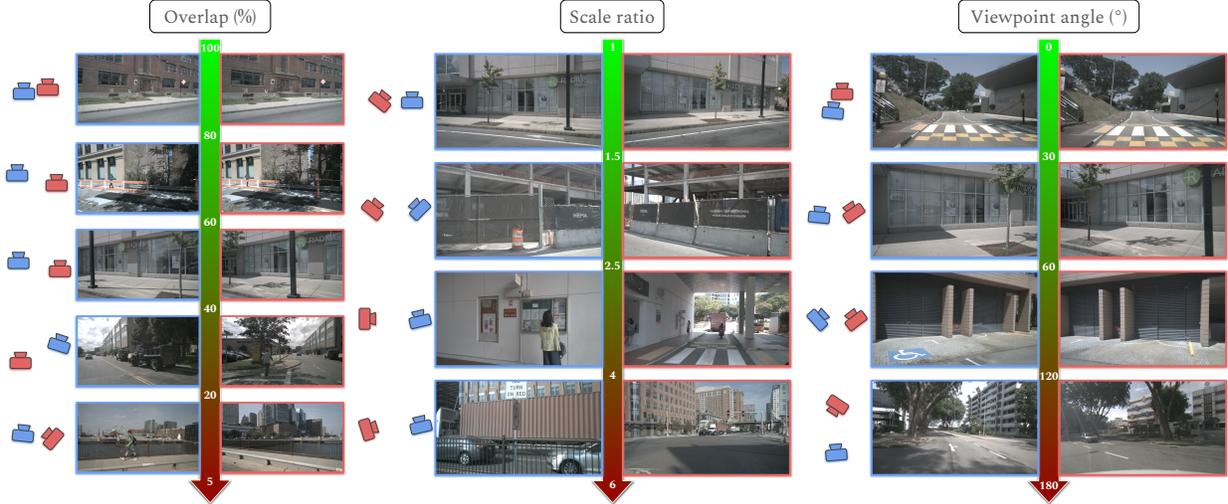


Figure 1. **We introduce RUBIK** – a benchmark based on the images from nuScenes for fine grain evaluation of camera pose estimations methods. RUBIK is made of image pairs spanning three difficulty criteria, in terms of scene overlap, scale ratio, and difference of viewpoint angles. It contains 16.5K image pairs across 33 difficulty levels. We use it to provide a comprehensive benchmarking of 14 methods.

Abstract

Camera pose estimation is crucial for many computer vision applications, yet existing benchmarks offer limited insight into method limitations across different geometric challenges. We introduce RUBIK, a novel benchmark that systematically evaluates image matching methods across well-defined geometric difficulty levels. Using three complementary criteria - overlap, scale ratio, and viewpoint angle - we organize 16.5K image pairs from nuScenes into 33 difficulty levels. Our comprehensive evaluation of 14 methods reveals that while recent detector-free approaches achieve the best performance (>47% success rate), they come with significant computational overhead compared to detector-based methods (150-600ms vs. 40-70ms). Even the best performing method succeeds on only 54.8% of the pairs, highlighting substantial room for improvement, particularly in challenging scenarios combining low overlap, large scale differences, and extreme viewpoint changes. Benchmark will be made publicly available.

1. Introduction

Camera pose estimation is a cornerstone of many computer vision applications, including augmented reality [2], robotics [6], 3D reconstruction [25, 41, 51], and autonomous navigation [34, 35, 43]. Camera pose estimators often rely on state-of-the-art image matching methods [8, 9, 13–17, 19–24, 27–29, 31, 33, 37, 38, 42, 44, 45, 48–50, 53, 55], which have achieved great performance in challenging scenarios, such as occlusions, limited overlap, and significant viewpoint changes. The creation of benchmarks [1, 40, 43, 46, 54] significantly contributed to these advancements by allowing a fair comparison between the proposed methods and pushing the development for more performing methods.

In this paper, we propose RUBIK, a benchmark based on the images from the nuScenes dataset [7], specifically designed to provide a more granular understanding of the limitations of current methods compared to existing benchmarks. Such understanding is critical to identify the weaknesses and to keep improving the performance of camera pose estimation methods. The nuScenes dataset offers an important diversity, featuring both broad and narrow streets,

large and small buildings, as well as vegetation and rivers. Additionally, the images were taken by cameras mounted on a car, oriented in multiple directions. As a result, each scene was captured from numerous viewpoints and at varying distances, making these images an ideal testbed for camera pose estimation.

More exactly, we structured RUBIK along 3 different types of challenges, as illustrated in Fig. 1. We quantified these challenges in terms of (1) overlap percentage between the two input views, (2) difference of apparent scale between the views, and (3) the difference of view angles: Small overlaps, large scale ratios, and large perspective differences make estimating the camera motion between the images challenging, and can happen alone or simultaneously. Let us highlight that we specifically focus on scenes recorded in good weather conditions to isolate and evaluate geometric difficulties without the confounding effect of adverse weathers.

We design RUBIK as follows:

- **Camera registration** – We carefully estimate the camera poses for the images in nuScenes. The nuScenes dataset already provides the camera poses but only within the ground plane, and we thus used COLMAP [41] to obtain full 6 degrees of freedom (DoF) poses. We still use the nuScenes 3 DoF camera poses to ensure the recovered 6 DoF poses are correct.
- **Dense co-visibility maps** – For each pair of nuScenes images from the same scene, we estimate co-visibility maps, as illustrated in Fig. 2. This gives us a fine measure of the co-visible regions between the two images. To do so, we developed a surprisingly simple and efficient method using the camera poses for each image pair and their depth and normal maps as predicted by state-of-the-art monocular depth estimators [36, 52].
- **Difficulty criteria** – For each pair, we evaluate our three criteria—overlap, ratio of the distances to the scene, and viewpoint angle difference—to quantify the difficulty of estimating the relative pose of a given image pair. Examples of pairs and their criteria are shown in Fig. 1. We then quantize the range of each criterion into a few bins, which results in a 3D grid of 33 boxes with varying levels of difficulty. Each box is populated with 500 image pairs, for a total of 16.5K test pairs.
- **Comprehensive benchmarking** – RUBIK allows us to provide an extensive evaluation of 14 methods: SIFT [32], SuperPoint [12], ALIKED [55], DISK [47], all using the LightGlue matcher [31], XFeat and its variants XFeat* and XFeat-LighterGlue [37], DeDoDe v2 [15], LoFTR [42], ASpanFormer [9], RoMa [16], Efficient LoFTR [50], DUST3R [49] and MAST3R [29]. Our results show that while most methods accurately estimate the poses under high overlap, similar scale, and small relative viewpoint angle, even the best performing method

struggles to correctly estimate the pose for more than 45% of the image pairs for a threshold of 5° (rotation) and 2m (translation). These findings highlight RUBIK’s utility in assessing and comparing different approaches.

We believe that RUBIK will serve as a valuable resource for the computer vision community, encouraging the development of more robust, occlusion-aware, or curriculum learning-based camera pose estimation methods. By providing both a comprehensive dataset and demonstrating the practical benefits of our co-visibility maps, we aim to advance research in this critical area of computer vision.

2. Related Work

2.1. Image Matching Benchmarks

Several benchmarks have been proposed to evaluate image matching methods. HPatches [3] focuses on homography estimation under viewpoint and illumination changes. MegaDepth1500 and ScanNet1500 are two widely used benchmarks initially proposed in [39]. MegaDepth1500 randomly sampled 1,500 image pairs from scenes “Sacre Coeur” and “St. Peter’s Square” of MegaDepth [30], discarding pairs with too small or too large overlap. ScanNet1500 randomly sampled 1,500 test pairs from ScanNet [11] similarly. The KITTI [18] dataset is also frequently used [26], where 2,710 image pairs, from consecutive frames, are sampled from the two sequences 09-10. The Image Matching Challenge 2024 [5] and its previous occurrences represent a significant advancement in comprehensive evaluation, featuring six distinct categories that cover real-world challenges: from phototourism with varying viewpoints and temporal changes, to aerial-ground matching, repeated structures, natural environments, and challenging scenarios with transparencies and reflections.

2.2. Visual Localization Benchmarks

Aachen Day-Night [40, 54] is a visual localization benchmark addressing outdoor localization in changing conditions. It consists of 4,328 sparsely sampled daytime database images, 824 daytime query images and 98 nighttime query images taken in the same environment.

InLoc [43] is a visual localization benchmark addressing large scale indoor localization with illumination and long-term changes, as well as repetitive patterns. It consists of 9,972 database images and 329 query images.

The Map-free Relocalization [1] benchmark consists of 655 small places, where each place comes with a reference image. The benchmark features changing conditions and image pairs with low to no visual overlap.

Despite these advances, existing benchmarks often lack controlled geometric variations, making it difficult to systematically analyze method performance across different difficulty levels. Our RUBIK benchmark addresses this lim-

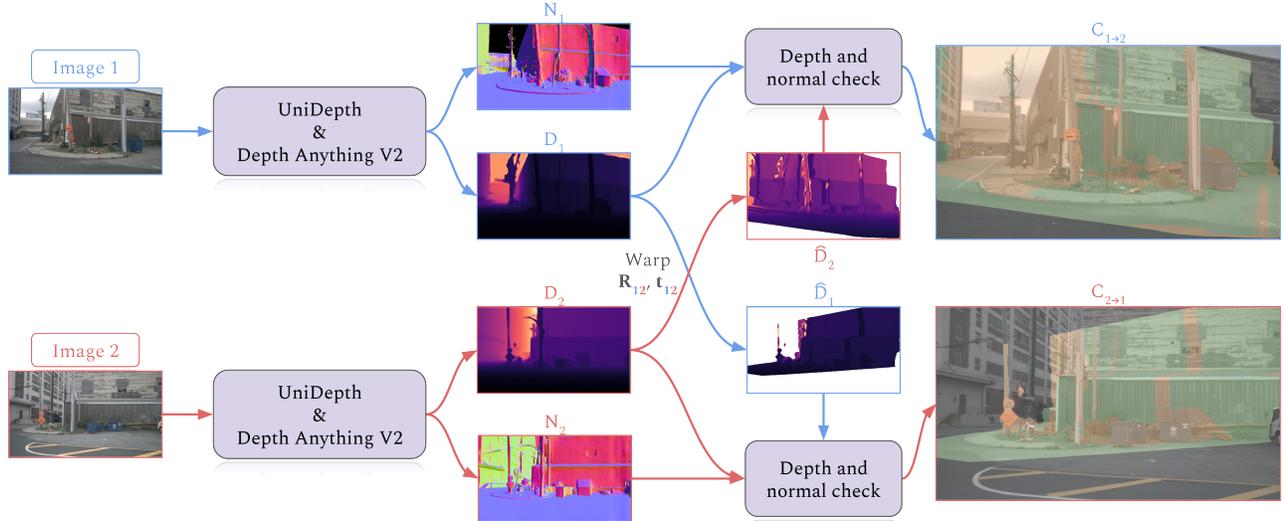


Figure 2. **Dense co-visibility map estimation** – Using normal maps (N_1, N_2) and depth maps (D_1, D_2) along with relative camera poses (R_{12}, τ_{12}), we warp depth maps between views to obtain \hat{D}_1 and \hat{D}_2 . Geometric consistency checks classify pixels as *co-visible*, *occluded*, or outside field-of-view to obtain the co-visibility maps $C_{1 \rightarrow 2}$ and $C_{2 \rightarrow 1}$ (see Sec. 3.3). We use UniDepth [36] for metric depth estimation and Depth Anything V2 [52] for normal map computation.

itation by providing 16.5K test pairs organized according to well-defined geometric criteria to obtain controlled varying levels of difficulty.

3. RUBIK

Our RUBIK benchmark is based on images from the nuScenes dataset [7]. nuScenes is a large-scale autonomous driving dataset containing 1,000 driving scenes in Boston and Singapore, each 20 seconds long, recorded at 12Hz in various weather conditions and times of day. The dataset provides high-quality synchronized camera images from 6 cameras with complete 360° coverage, along with precise camera calibration and pose information.

For our benchmark, we specifically focus on scenes recorded in good weather conditions to isolate and evaluate geometric difficulties without the confounding effect of adverse weathers. This deliberate choice allows us to systematically analyze how methods perform across different geometric challenges, without the additional complexity of environmental factors.

The creation of RUBIK consists of four main steps: (1) lifting of nuScenes ground truth 3 DoF camera poses to 6 DoF, (2) generation of depth and surface normal maps for each image in order to (3) compute the co-visibility maps between image pairs, which allow us to (4) systematically organize the image pairs based on geometric criteria.

3.1. Lifting nuScenes ground truth camera poses

While nuScenes provides high-quality metric camera poses, these are limited to 3 DoF within the ground plane,

as they are primarily intended for autonomous driving applications. However, for comprehensive camera pose estimation benchmarking, we require full 6 DoF metric camera poses that account for variation in camera height and orientation. To lift nuScenes ground truth 3 DoF metric poses to 6 DoF metric poses, we carefully process each sequence independently using a two-stage approach:

1. 3D Reconstruction – We first perform Structure-from-Motion using COLMAP [41]. This provides us with initial 6-DoF camera pose estimates. However, the translations are not metric (i.e. the scene is reconstructed up to a scale factor) and some camera poses may be erroneous.

2. Pose alignment and filtering – To obtain metric translations and filter out erroneous poses, we align the previously estimated 6 DoF poses with the nuScenes ground truth 3 DoF metric poses. To do so, we use a custom LO-RANSAC [10] approach to estimate a 7 DoF similarity transformation between the projected COLMAP poses and nuScenes ground truth poses. We set the RANSAC threshold to 1 meter to filter out erroneous poses and ensure high-quality ground truth.

An example of alignment is shown in Fig. 3. This alignment with nuScenes’ metric ground truth allows us to recover proper metric scale, which is not available from COLMAP reconstructions alone, and consequently maintain the high precision of nuScenes’ original poses while adding reliable elevation and orientation information. The resulting 6 DoF metric poses serve as the foundation for our geometric difficulty criteria and co-visibility map computa-

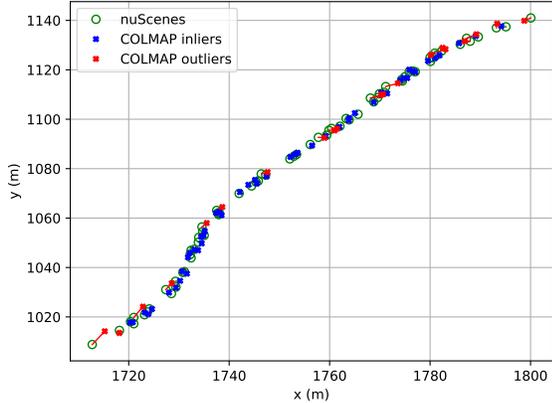


Figure 3. **Camera pose alignment and filtering** – Visualization of (subsampling) camera trajectories of scene-0266 after aligning COLMAP poses with nuScenes ground truth poses. Blue crosses (\times) indicate inlier poses (alignment error $< 1\text{m}$) that are kept for our benchmark, while red crosses (\times) show outlier poses that are discarded.

tions.

3.2. Generation of metric depth and normal maps

To compute dense co-visibility maps between image pairs, we require accurate depth and normal maps for each image. Recent advances in monocular depth estimation have enable reliable geometric information extraction from single images without expensive ground truth measurements. After evaluating several state-of-the-art models, we found that combining two complementary approaches yield optimal results:

- **Metric Depth Maps** – We use UniDepth [36] for its ability to predict metric depth values. The model provides well-aligned depth predictions that are crucial for consistent cross-view measurements.
- **Surface Normal Maps** – Normal maps computed from UniDepth [36] depth predictions tend to be noisy. Instead, we compute normal maps from Depth Anything V2 [52] depth maps. Let us highlight that these depth maps are not metric, thus we align them with UniDepth depth maps before normals computation. This approach produces remarkably sharp normal maps with precise object boundaries and fine geometric details (see Fig. 4 for a comparison).

This complementary approach leverages each model’s strengths: UniDepth’s accuracy and Depth Anything’s superior normal predictions. Our experiments show that this combination provides reliable geometric information for computing co-visibility maps between views, as demonstrated in the following section.

3.3. Generation of Co-visibility maps

Given a pair of images (I_1, I_2) with known relative camera pose (R_{12}, t_{12}) from Sec. 3.1, calibration matrices (K_1, K_2), depth maps (D_1, D_2), and surface normals (N_1, N_2) from Sec. 3.2, we generate the co-visibility maps $C_{1 \rightarrow 2}$ and $C_{2 \rightarrow 1}$ (see Fig. 2). We start by warping the depth map D_2 to predict D_1 as follows:

$$\hat{D}_1(\mathbf{p}_1) = [0 \ 0 \ 1] (R_{12} D_2(\mathbf{p}_{1 \rightarrow 2}) K_2^{-1} \mathbf{p}_{1 \rightarrow 2} + \mathbf{t}_{12}), \quad (1)$$

where \mathbf{p}_1 is a pixel location (in homogeneous coordinates) in the grid of I_1 , $\mathbf{p}_{1 \rightarrow 2} = K_2 \pi (R_{12}^\top (D_1(\mathbf{p}_1) K_1^{-1} \mathbf{p}_1 - \mathbf{t}_{12}))$ and $D_2(\mathbf{p}_{1 \rightarrow 2})$ is implemented using bilinear interpolation.

Predicting D_1 enables occlusion detection through a relative depth check:

$$\frac{|\hat{D}_1(\mathbf{p}_1) - D_1(\mathbf{p}_1)|}{D_1(\mathbf{p}_1)} > \tau, \quad (2)$$

where we used $\tau = 5\%$. Let us highlight that if $\mathbf{p}_{1 \rightarrow 2}$ falls outside the image boundaries, then pixel \mathbf{p}_1 is labeled ”outside the field of view”. This occlusion detection is further refined by discarding a pixel \mathbf{p}_1 if its normal does not point towards camera 2:

$$\angle(\mathbf{z}, R_{12}^\top N_1(\mathbf{p}_1)) < 90^\circ - \epsilon, \quad (3)$$

where $\mathbf{z} = [0 \ 0 \ 1]^\top$ and we used $\epsilon = 5^\circ$.

An example of co-visibility maps $C_{1 \rightarrow 2}$ and $C_{2 \rightarrow 1}$ is shown in Fig. 2. We perform the previous steps in both directions ($I_1 \rightarrow I_2$ and $I_2 \rightarrow I_1$). Our results surprisingly show that metric monocular depth prediction networks are, from now on, accurate enough to perform cross-view 3D reasoning. We believe this finding opens a path towards camera pose estimation frameworks beyond classical combination of image matching and 3D geometry-based minimal solver, but we leave this as future work.

3.4. Geometric criteria

Using the co-visibility maps $C_{1 \rightarrow 2}$ and $C_{2 \rightarrow 1}$ previously computed, we evaluate three complementary criteria to quantify the geometric difficulty of estimating the relative pose between an image pair (I_1, I_2).

1. Overlap (ω) – The ratio of co-visible pixels to total pixels:

$$\omega = \frac{|C_{1 \rightarrow 2}| + |C_{2 \rightarrow 1}|}{|I_1| + |I_2|}, \quad (4)$$

where $|\cdot|$ is the cardinal of a set. The overlap is a classical criterion that is often used by image matching methods [9, 39, 42] to obtain a balanced training set that includes both simple pairs (with large overlaps) and challenging pairs (with small overlaps). However, this

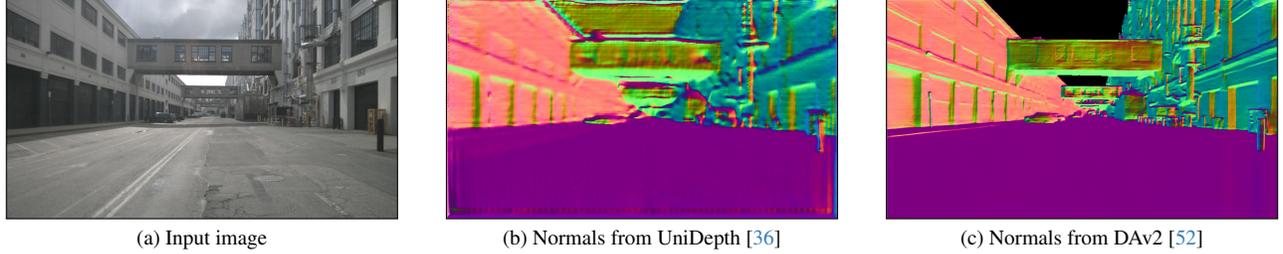


Figure 4. **Comparison of surface normal maps** – From left to right: input image (a), normal maps computed from UniDepth’s metric depth predictions (b) and from Depth Anything V2 after alignment to UniDepth depth map (c). Note the significantly sharper object boundaries and finer geometric details in Depth Anything V2’s prediction, particularly around building edges and depth discontinuities.

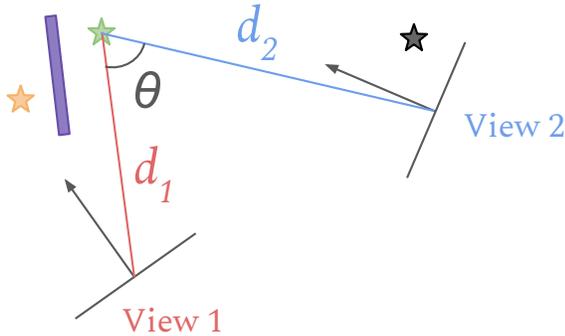


Figure 5. **Two-view setup** – Considering two views, a 3D point can either be co-visible (\star), occluded (\star), or outside the field of view (\star) in one of the views. For each co-visible 3D point, we compute its distances d_1 and d_2 to both camera centers and the angle θ between the two lines of sight.

criterion alone is somewhat limited, as an image pair with a large pure rotation (i.e. where the translation is null) may result in a small overlap, even though the underlying matching problem is not particularly difficult, since the viewpoint remains unchanged.

2. Scale Ratio (δ) – The median of the ratios of the camera distances to the co-visible 3D points:

$$\delta = \text{med} \left\{ \left\{ r_{\mathbf{p}_1}^{1 \rightarrow 2} \right\}_{\mathbf{p}_1 \in \mathcal{C}_{1 \rightarrow 2}} \cup \left\{ r_{\mathbf{p}_2}^{2 \rightarrow 1} \right\}_{\mathbf{p}_2 \in \mathcal{C}_{2 \rightarrow 1}} \right\}, \quad (5)$$

$$\text{with } r_{\mathbf{p}_i}^{i \rightarrow j} = \max \left(\frac{\| \mathbf{K}_i^{-1} \mathbf{p}_i \|}{\| \mathbf{D}_i(\mathbf{p}_i) \mathbf{K}_i^{-1} \mathbf{p}_i - \mathbf{t}_{ij} \|}, \frac{\| \mathbf{D}_i(\mathbf{p}_i) \mathbf{K}_i^{-1} \mathbf{p}_i - \mathbf{t}_{ij} \|}{\| \mathbf{K}_i^{-1} \mathbf{p}_i \|} \right).$$

Contrary to the overlap, the scale ratio is independent of the relative rotation between the two cameras (i.e. rotating camera 1 and camera 2 in Fig. 5 does not affect neither d_1 nor d_2) and only depends on the 3D geometry of the scene and the relative translation.

3. Viewpoint Angle (θ) – The median of the co-visible line-of-sight angles:

$$\theta = \text{med} \left\{ \left\{ \theta_{\mathbf{p}_1}^{1 \rightarrow 2} \right\}_{\mathbf{p}_1 \in \mathcal{C}_{1 \rightarrow 2}} \cup \left\{ \theta_{\mathbf{p}_2}^{2 \rightarrow 1} \right\}_{\mathbf{p}_2 \in \mathcal{C}_{2 \rightarrow 1}} \right\}, \quad (6)$$

where $\theta_{\mathbf{p}_i}^{i \rightarrow j} = \angle(\mathbf{K}_i^{-1} \mathbf{p}_i, \mathbf{D}_i(\mathbf{p}_i) \mathbf{K}_i^{-1} \mathbf{p}_i - \mathbf{t}_{ij})$ represents the angle between the two lines of sight. It is clear that this criterion is also independent of the relative rotation between the two cameras and only depends on the 3D geometry and the relative translation, just like the scale ratio. However, the viewpoint angle and the scale ratio complement each other, as the viewpoint angle is independent of the scale ratio (i.e. changing d_2 in Fig. 5 does not affect θ), and vice versa.

The three criteria discussed above complement each other and will be used in the next section to categorize the image pairs from the nuScenes test scenes based on their difficulty level.

3.5. Benchmark Organization

Using the three geometric criteria defined above, we can systematically organize image pairs from nuScenes test scenes according to their difficulty level. For each possible pair within a scene, we compute its overlap ω , scale ratio δ , and viewpoint angle θ . Based on the distributions of these values across the entire test set which comprises 4.2M image pairs across 85 successfully reconstructed and filtered scenes, we define meaningful bins for each criterion:

Overlap (%) – 5 bins: 5 - 20 - 40 - 60 - 80 - 100

Scale ratio – 4 bins: 1.0 - 1.5 - 2.5 - 4.0 - 6.0

Viewpoint angle ($^\circ$) – 4 bins: 0 - 30 - 60 - 120 - 180

An example of image pair for each bin is shown in Fig. 1.

While this binning strategy theoretically creates a $5 \times 4 \times 4$ grid (80 difficulty levels), not all combinations are physically possible. For instance, image pairs with both very large overlap and small scale ratio rarely exhibit large viewpoint angles, as these geometric conditions are inherently contradictory. We finally obtain 33 valid difficulty levels (see Fig. 6).

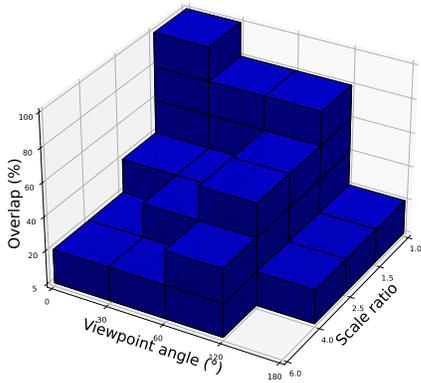


Figure 6. **Visualization of our 3D grid organization** – Each axis represents one of our geometric criteria.

To ensure statistical significance while maintaining a manageable dataset size, we populate each valid difficulty level with 500 randomly sampled image pairs. This results in a benchmark of 16.5K pairs, carefully curated to span the full spectrum of geometric challenges encountered in real-world scenarios. As shown in Fig. 7, our choice of 500 image pairs per box ensures stable and reliable evaluation metrics, with similar conclusions holding across all evaluated methods.

This structured organization enables systematic evaluation of pose estimation methods across well-defined difficulty levels, from simple cases with large overlap and similar viewpoints to challenging scenarios with minimal overlap and extreme geometric variations.

4. Results

We evaluate 14 image matching methods on our novel RUBIK benchmark to assess their performance across different geometric challenges. In this section, we first describe our evaluation protocol, then present a comprehensive analysis of both detector-based and detector-free approaches.

4.1. Evaluation Protocol

For a fair comparison of all the considered methods, for each image pair in our benchmark, we follow the same evaluation pipeline:

1. **Image Matching** – We first obtain matches between the two images using each method’s default parameters (e.g. number of keypoints, backbone size, input image resolution etc.) and pre-trained weights.
2. **Pose Estimation** – Using these matches, we estimate the essential matrix using MAGSAC++ [4] from OPENCV, with a threshold of 0.5 pixel. From this essential ma-

trix, we recover the relative rotation and the translation direction between the two views.

3. **Scale Recovery** – To obtain metric translations, we leverage depth predictions from UniDepth [36] at matched locations, following the approach in [29]. This provides us with a metric scene scale, enabling full 6 DoF pose estimation.

We consider a pose estimation successful when both the rotation error is less than 5° and the translation error is less than 2m. These thresholds were chosen based on typical requirements in real-world applications such as autonomous navigation and the precision of our ground truth poses, which were obtained through careful COLMAP reconstruction and alignment with nuScenes metric poses (see Sec. 3.1).

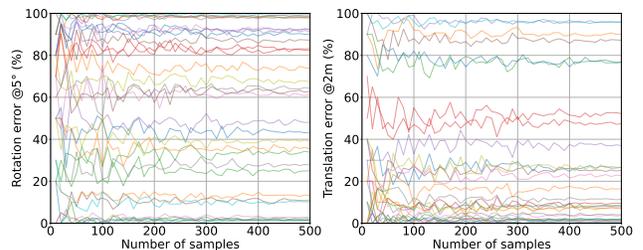


Figure 7. **Impact of sample size on evaluation reliability** – We analyze how the number of image pairs per difficulty level affects the stability of performance metrics for LoFTR [42]. Left: Percentage of pairs with rotation error $< 5^\circ$. Right: Percentage of pairs with translation error $< 2m$. The plots demonstrate that 500 pairs per difficulty level provide stable evaluation metrics, with minimal variance when increasing the sample size further than about 400 pairs. We obtain similar conclusions across all evaluated methods.

4.2. Learning-based Image Matching Methods

Before presenting our benchmark results, we first provide a brief overview of recent advances in learning-based image matching methods. Recent years have seen significant advances in learning-based image matching approaches. These can be broadly categorized into detector-based and detector-free methods.

Detector-based methods build upon traditional keypoint detection and description paradigms. SuperPoint [12] pioneered self-supervised interest point detection and description. Recent works like DISK [47], ALIKED [55], and XFeat [37] have further improved efficiency and accuracy. LightGlue [31] focuses on accelerating the matching process while maintaining high accuracy. DeDoDe v2 [15] specifically addresses the challenges of keypoint detection reliability.

Detector-free methods take a different approach by directly establishing dense correspondences between images. LoFTR [42] introduced transformer-based architectures for local feature matching without explicit keypoint detection. Recent works like RoMa [16] and Efficient LoFTR [50] have improved both the efficiency and accuracy of dense matching approaches. DUS_t3R [49] introduces a paradigm shift by reformulating the matching problem as pointmap regression without requiring camera calibration or pose information, enabling joint optimization of 3D reconstruction and matching. Building upon this, MAS_t3R [29] explicitly grounds the matching process in 3D space and introduces an efficient reciprocal matching scheme that significantly improves both speed and accuracy, particularly for challenging viewpoint changes. These 3D-aware approaches demonstrate substantial improvements over traditional 2D matching methods, especially in scenarios with extreme viewpoint variations.

4.3. Main Results

We evaluate the performance of 14 methods on our benchmark, including both detector-based approaches (SIFT [32], SuperPoint [12], DISK [47], ALIKED [55], all using LightGlue [31], XFeat [37] and its variants, DeDoDe v2 [15]) and detector-free approaches (LoFTR [42], ELofTR [50], ASpanFormer [9], RoMa [16], DUS_t3R [49] and MAS_t3R [29]). For fair runtime comparison, all experiments were conducted on the same NVIDIA RTX 4090 GPU. Tab. 1 presents the overall ranking of these methods, along with their computational efficiency.

Our benchmark, here aggregated, reveals several key findings:

1. **Detector-free dominance** – The top three performing methods (DUS_t3R, MAS_t3R and RoMa) are all detector-free approaches, suggesting that direct dense matching is more robust across varying geometric conditions.
2. **Speed-accuracy trade-off** – While detector-free methods achieve better accuracy, they generally require more computation time. Detector-based methods, especially ALIKED or DISK, combined with LightGlue, offer competitive performance with significantly lower runtime (40-70ms vs. 150-600ms).
3. **Impact of matching strategies** – The quite significant performance gap between XFeat variants (with and without LighterGlue) highlights the importance of the matching strategy, even with the same feature detector.
4. **Recent advances** – The newest methods (DUS_t3R, MAS_t3R, RoMa) show substantial improvements over their predecessors (LoFTR, ASpanFormer), demonstrating the rapid progress in the field.

These results demonstrate that while recent detector-free methods achieve the best performance across our benchmark’s diverse geometric challenges, detector-based ap-

Table 1. **Benchmark Results** – Average ranking across all difficulty levels (lower is better), based on the percentage of successful pose estimations (rotation error < 5° and translation error < 2m). We also report the overall percentage of successful pose estimations as well as the median runtime per image pair in milliseconds. Best and second-best values are shown in **bold** and underlined respectively.

Method	Avg. Rank	Success (%)	Time (ms)
<i>Detector-based methods</i>			
ALIKED+LightGlue [55]	5.3	36.8	<u>45</u>
DISK+LightGlue [47]	<u>5.4</u>	<u>35.9</u>	69
SP+LightGlue [12]	6.1	35.7	43
SIFT+LightGlue [32]	7.3	33.1	194
DeDoDe v2 [15]	8.6	30.4	282
XFeat [37]	13.1	14.2	54
XFeat* [37]	12.5	15.1	82
XFeat+LighterGlue [37]	9.0	30.1	43
<i>Detector-free methods</i>			
LoFTR [42]	10.8	24.9	185
ELofTR [50]	9.5	26.6	<u>124</u>
ASpanFormer [9]	9.8	24.8	108
RoMa [16]	2.7	47.3	614
DUS _t 3R [49]	2.4	54.8	257
MAS _t 3R [29]	<u>2.5</u>	<u>53.6</u>	173

proaches remain competitive, especially when computational efficiency is a priority.

In Fig. 8, we show for 4 methods how their performance varies with specific geometric challenges. Results for all other methods are in the Supplementary Material. Overall, detector-free methods show better performance than detector-based methods when dealing with high geometric challenges. Specifically, DUS_t3R and MAS_t3R show a better balance between speed and performance, allowing for stable results without steep degradation under tougher conditions, as is the case for most of the detector-based methods. Those fine grained findings emphasizes the recent advancements made by newer dense matching methods over traditional keypoint-based approaches. These improvements, especially in challenging scenarios, underline the value of incorporating visibility-aware features and dense matching strategies into camera pose estimation. Our findings indicate that although most methods provide accurate pose estimates under conditions of high overlap, similar scale, and small relative viewpoint angles, even the top-performing method fails to correctly estimate the pose for over 45% of image pairs when using thresholds of 5° (rotation) and 2m (translation). These results underscore the value of RUBIK in evaluating and comparing various approaches.

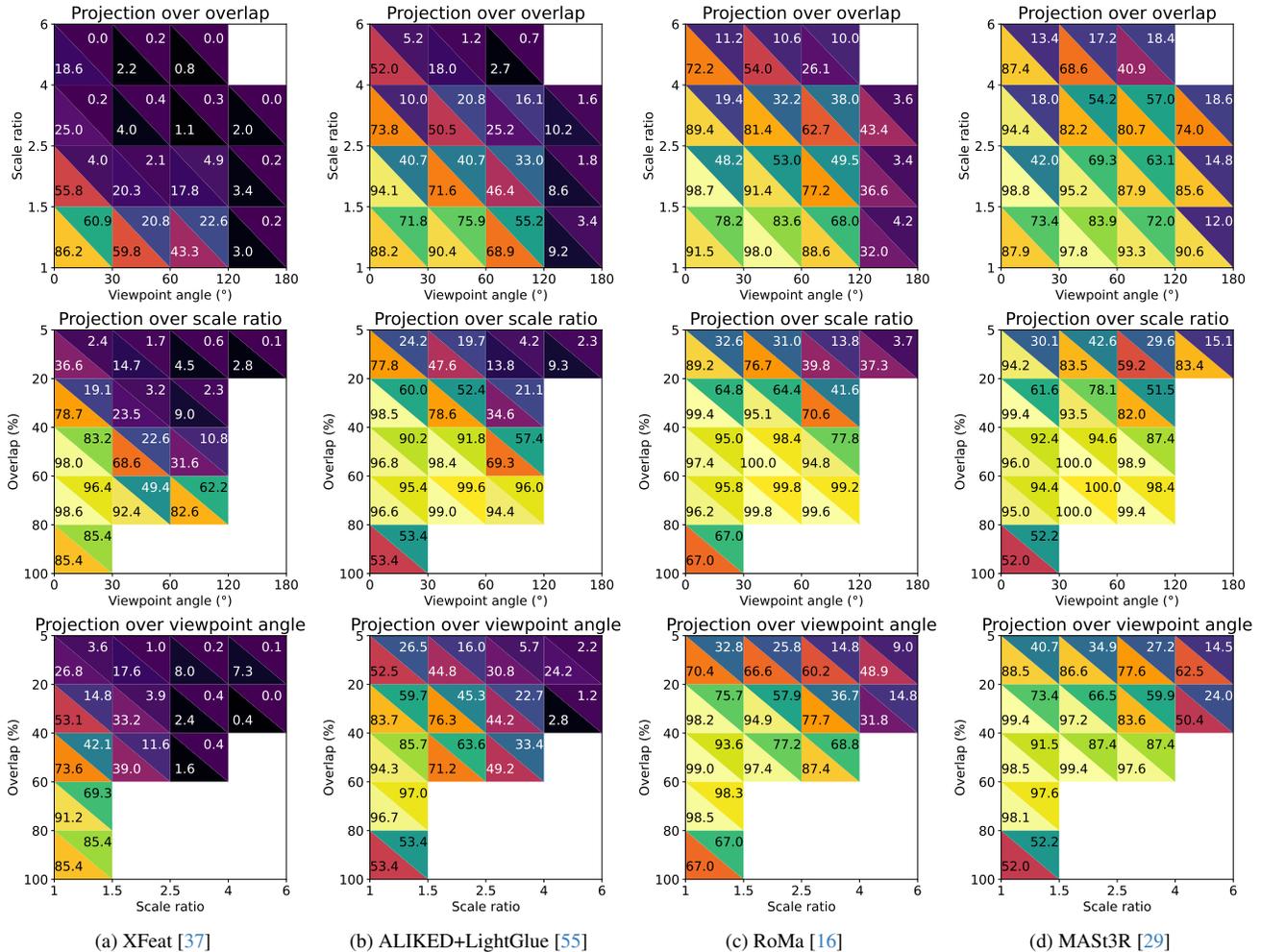


Figure 8. **Performance analysis across geometric criteria** – Success rate either for R@5° or t@2m (bottom-left and top-right of each triangle, respectively), for 4 methods (detector-based and detector-free), when projecting results onto individual geometric criteria (see Fig. 6). For each method, we show three plots corresponding to the projection over overlap (top), scale ratio (middle), and viewpoint angle (bottom).

5. Limitations

Our co-visibility map generation pipeline lacks robustness to instance changes; for example, when one car is replaced by another, the depth check and normal check may still be satisfied, leading to the region being incorrectly considered co-visible. Some examples are in the Supplementary Material.

6. Conclusion

We introduced RUBIK, a novel benchmark that provides a systematic way to evaluate camera pose estimation methods across well-defined geometric challenges. By organizing 16.5K image pairs into 33 difficulty levels based on overlap, scale ratio, and viewpoint angle, our benchmark revealed

several important insights. First, recent detector-free methods (DUST3R, MAST3R, RoMa) significantly outperform traditional approaches, achieving success rates above 47%, but at the cost of higher computational requirements (150-600ms vs. 40-70ms for detector-based methods). Second, even the best performing methods (DUST3R and MAST3R) fail to correctly estimate poses more than 45% of image pairs, highlighting significant room for improvement, particularly in challenging scenarios combining low overlap, large scale differences, and extreme viewpoint changes. By providing a fine-grained understanding of method limitations, RUBIK opens new perspectives for developing more robust pose estimation approaches, particularly for challenging geometric configurations that current methods struggle with.

Acknowledgment

This project has received funding from the Bosch Research Foundation (Bosch Forschungsstiftung), and was granted access to the HPC resources of IDRIS under the allocation 2024-AD011014905R1 made by GENCI.

References

- [1] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Aron Monszpart, Victor Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *European Conference on Computer Vision*, pages 690–708. Springer, 2022. 1, 2
- [2] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. Recent advances in augmented reality. *IEEE computer graphics and applications*, 21(6):34–47, 2001. 1
- [3] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5173–5182, 2017. 2
- [4] Daniel Barath, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. Magsac++, a fast, reliable and accurate robust estimator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1304–1312, 2020. 6
- [5] Fabio Bellavia, Jiri Matas, Dmytro Mishkin, Luca Morelli, Fabio Remondino, Weiwei Sun, Amy Tabb, Eduard Trulls, Kwang Moo Yi, Sohier Dane, and Ashley Chow. Image matching challenge 2024 - hexathlon. <https://kaggle.com/competitions/image-matching-challenge-2024>, 2024. Kaggle. 2
- [6] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332, 2016. 1
- [7] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multi-modal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1, 3
- [8] Hongkai Chen, Zixin Luo, Jiahui Zhang, Lei Zhou, Xuyang Bai, Zeyu Hu, Chiew-Lan Tai, and Long Quan. Learning to match features with seeded graph matching network. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6301–6310, 2021. 1
- [9] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David Mckinnon, Yanghai Tsing, and Long Quan. Aspanformer: Detector-free image matching with adaptive span transformer. In *European Conference on Computer Vision*, pages 20–36. Springer, 2022. 1, 2, 4, 7
- [10] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *Pattern Recognition: 25th DAGM Symposium, Magdeburg, Germany, September 10-12, 2003. Proceedings 25*, pages 236–243. Springer, 2003. 3
- [11] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 2
- [12] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018. 2, 6, 7, 4
- [13] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint detection and description of local features. In *CVPR 2019-IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1
- [14] Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, and Michael Felsberg. Dkm: Dense kernelized feature matching for geometry estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17765–17775, 2023.
- [15] Johan Edstedt, Georg Bökman, and Zhenjun Zhao. Dedode v2: Analyzing and improving the dedode keypoint detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4245–4253, 2024. 2, 6, 7, 4
- [16] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19790–19800, 2024. 2, 7, 8
- [17] Miao Fan, Mingrui Chen, Chen Hu, and Shuchang Zhou. Occ²net: Robust image matching based on 3d occupancy estimation for occluded regions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9652–9662, 2023. 1
- [18] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 2
- [19] Hugo Germain, Guillaume Bourmaud, and Vincent Lepetit. S2dnet: Learning image features for accurate sparse-to-dense matching. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 626–643. Springer, 2020. 1
- [20] Hugo Germain, Vincent Lepetit, and Guillaume Bourmaud. Neural reprojection error: Merging feature learning and camera pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 414–423, 2021.
- [21] Hugo Germain, Vincent Lepetit, and Guillaume Bourmaud. Visual correspondence hallucination. In *International Conference on Learning Representations*, 2022.
- [22] Khang Truong Giang, Soohwan Song, and Sungho Jo. Topicfm: Robust and interpretable topic-assisted feature matching. In *Proceedings of the AAAI conference on artificial intelligence*, pages 2447–2455, 2023.

- [23] Pierre Gleize, Weiyao Wang, and Matt Feiszli. Silk: Simple learned keypoints. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22499–22508, 2023.
- [24] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Xingzhe He, Hossam Isack, Abhishek Kar, Helge Rhodin, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised keypoints from pretrained diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22820–22830, 2024. 1
- [25] Jared Heinly, Johannes L Schonberger, Enrique Dunn, and Jan-Michael Frahm. Reconstructing the world* in six days*(as captured by the yahoo 100 million image dataset). In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3287–3295, 2015. 1
- [26] You-Yi Jau, Rui Zhu, Hao Su, and Manmohan Chandraker. Deep keypoint-based camera pose estimation with geometric constraints. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4950–4957. IEEE, 2020. 2
- [27] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. Cotr: Correspondence transformer for matching across images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6207–6217, 2021. 1
- [28] Shinjeong Kim, Marc Pollefeys, and Daniel Barath. Learning to make keypoints sub-pixel accurate. In *European Conference on Computer Vision*, pages 413–431. Springer, 2025.
- [29] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. *arXiv preprint arXiv:2406.09756*, 2024. 1, 2, 6, 7, 8
- [30] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018. 2
- [31] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17627–17638, 2023. 1, 2, 6, 7
- [32] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004. 2, 7, 4
- [33] Runyu Mao, Chen Bai, Yatong An, Fengqing Zhu, and Cheng Lu. 3dg-stfm: 3d geometric guided student-teacher feature matching. In *European Conference on Computer Vision*, pages 125–142. Springer, 2022. 1
- [34] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE transactions on robotics*, 33(5):1255–1262, 2017. 1
- [35] 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. 1
- [36] Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10106–10116, 2024. 2, 3, 4, 5, 6
- [37] Guilherme Potje, Felipe Cadar, André Araujo, Renato Martins, and Erickson R Nascimento. Xfeat: Accelerated features for lightweight image matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2682–2691, 2024. 1, 2, 6, 7, 8, 4
- [38] Jerome Revaud, Cesar De Souza, Martin Humenberger, and Philippe Weinzaepfel. R2d2: Reliable and repeatable detector and descriptor. *Advances in neural information processing systems*, 32, 2019. 1
- [39] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 2, 4
- [40] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8601–8610, 2018. 1, 2
- [41] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1, 2, 3
- [42] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8922–8931, 2021. 1, 2, 4, 6, 7
- [43] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7199–7209, 2018. 1, 2
- [44] Dongli Tan, Jiang-Jiang Liu, Xingyu Chen, Chao Chen, Ruixin Zhang, Yunhang Shen, Shouhong Ding, and Rongrong Ji. Eco-tr: Efficient correspondences finding via coarse-to-fine refinement. In *European Conference on Computer Vision*, pages 317–334. Springer, 2022. 1
- [45] Shitao Tang, Jiahui Zhang, Siyu Zhu, and Ping Tan. Quadtree attention for vision transformers. In *International Conference on Learning Representations*, 2022. 1
- [46] Carl Toft, Will Maddern, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, et al. Long-term visual localization revisited. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4):2074–2088, 2020. 1
- [47] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. *Advances in Neural Information Processing Systems*, 33:14254–14265, 2020. 2, 6, 7, 4
- [48] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. In *Proceedings of the*

- Asian Conference on Computer Vision*, pages 2746–2762, 2022. [1](#)
- [49] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. [2](#), [7](#), [4](#)
- [50] Yifan Wang, Xingyi He, Sida Peng, Dongli Tan, and Xiaowei Zhou. Efficient lofr: Semi-dense local feature matching with sparse-like speed. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21666–21675, 2024. [1](#), [2](#), [7](#), [4](#)
- [51] Changchang Wu. Towards linear-time incremental structure from motion. In *2013 International Conference on 3D Vision-3DV 2013*, pages 127–134. IEEE, 2013. [1](#)
- [52] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024. [2](#), [3](#), [4](#), [5](#)
- [53] Rui Yin, Yulun Zhang, Zherong Pan, Jianjun Zhu, Cheng Wang, and Biao Jia. Srpose: Two-view relative pose estimation with sparse keypoints. *arXiv preprint arXiv:2407.08199*, 2024. [1](#)
- [54] Zichao Zhang, Torsten Sattler, and Davide Scaramuzza. Reference pose generation for long-term visual localization via learned features and view synthesis. *International Journal of Computer Vision*, 129(4):821–844, 2021. [1](#), [2](#)
- [55] Xiaoming Zhao, Xingming Wu, Weihai Chen, Peter CY Chen, Qingsong Xu, and Zhengguo Li. Aliked: A lighter keypoint and descriptor extraction network via deformable transformation. *IEEE Transactions on Instrumentation and Measurement*, 72:1–16, 2023. [1](#), [2](#), [6](#), [7](#), [8](#)



RUBIK: A Structured Benchmark for Image Matching across Geometric Challenges

Supplementary Material

7. Additional Results

We provide detailed performance metrics for all evaluated methods across our benchmark’s geometric criteria. In Tab. 2, we break down the success rates according to individual geometric bins showing the percentage of successful pose estimations for each method across the different ranges of overlap, scale ratio, and viewpoint angle. This granular analysis complements the aggregated results presented in the main paper (see Tab. 1).

The performance analysis across geometric criteria for methods not shown in Fig. 8 is presented in Fig. 10. These triangular plots follow the same visualization approach as in the main paper, with success rates for rotation (bottom-left) and translation (top-right) thresholds projected onto individual geometric criterion: overlap (top), scale ratio (middle), and viewpoint angle (bottom).

To provide additional context for the cumulative results analysis, we present in Tab. 3 the complete ordering of all 33 difficulty levels, sorted by decreasing average success rate across all methods. This ordering reveals clear patterns in what makes image pairs challenging: the easiest pairs typically combine high overlap (60-80%), small scale changes (1.0-1.5), and small viewpoint changes (0-30°), while the most challenging pairs involve minimal overlap (5-20%), large scale changes (4.0-6.0), and significant viewpoint changes (60-120°). This ordering was used to generate the cumulative plot in Fig. 9, which shows how performance evolves when starting from the easiest geometric configurations (1 box) and gradually incorporating more difficult image pairs up to the complete benchmark (33 boxes). This visualization complements the fine-grained analysis by showing the overall robustness of each method across the full spectrum of geometric challenges.

These additional results further support and refine the conclusions drawn in the main paper. The detailed breakdown in Tab. 2 reveal several noteworthy patterns:

1. **Extreme conditions handling** – While the best detector-free methods generally outperform the best detector-based ones, this gap becomes particularly pronounced in extreme geometric conditions. For instance, at very low overlap (5-20%), DUS_t3R and MAS_t3R maintain success rates of 30.4% and 28.4% respectively, while the best detector-based method (ALIKED+LightGlue) achieves only 12.7%.
2. **Detector-based methods vs LoFTR-like detector-free methods** – LoFTR-like methods (LoFTR, ELoFTR

and ASpanFormer) are almost systematically outperformed by several detector-based methods (DeDoDe v2, XFeat+LighterGlue, ALIKED+LighGLue, DISK+LightGlue, SP+LightGlue, SIFT+LightGlue).

3. **Performance degradation patterns** – The cumulative plot in Fig. 9 reveals distinct patterns in how different methods handle increasing geometric difficulty. Detector-free methods, particularly DUS_t3R and MAS_t3R, show a more gradual performance degradation compared to detector-based approaches. This is quantitatively confirmed in Tab. 2, where these methods maintain relatively high success rates across all geometric criteria: overlap (>28% even at 5-20%), scale ratio (>40% up to 4.0), and viewpoint angle (>50% up to 120°). In contrast, detector-based methods show steeper performance drops, especially in challenging conditions, suggesting that recent dense matching approaches are inherently more robust to various geometric transformations (as some of the older detector-free approaches are beaten by most of the detector-based ones).
4. **High overlap performance paradox** – Interestingly, almost all methods perform better on image pairs with 60-80% overlap compared to those with 80-100% overlap. This seemingly counter-intuitive behavior could be explained by the geometric configuration of these pairs. Very high overlap (>80%) often occurs in image pairs taken from nearly identical positions, resulting in very small baselines (i.e. small distance between camera centers). While these pairs have strong visual similarity, the small baseline makes both rotation and translation estimation challenging: small errors in matching lead to large uncertainties in triangulation geometry, affecting both the essential matrix estimation and the subsequent pose decomposition. In contrast, pairs with 60-80% overlap typically have larger baselines while maintaining sufficient visual correspondences, creating more favorable conditions for pose estimation.

These findings highlight the importance of comprehensive evaluation across different geometric criteria, as methods can exhibit significantly different behaviors depending on the specific challenges they encounter.

8. Limitations

While our benchmark provides comprehensive evaluations across various geometric challenges, there are some inherent limitations in how we determine co-visibility between

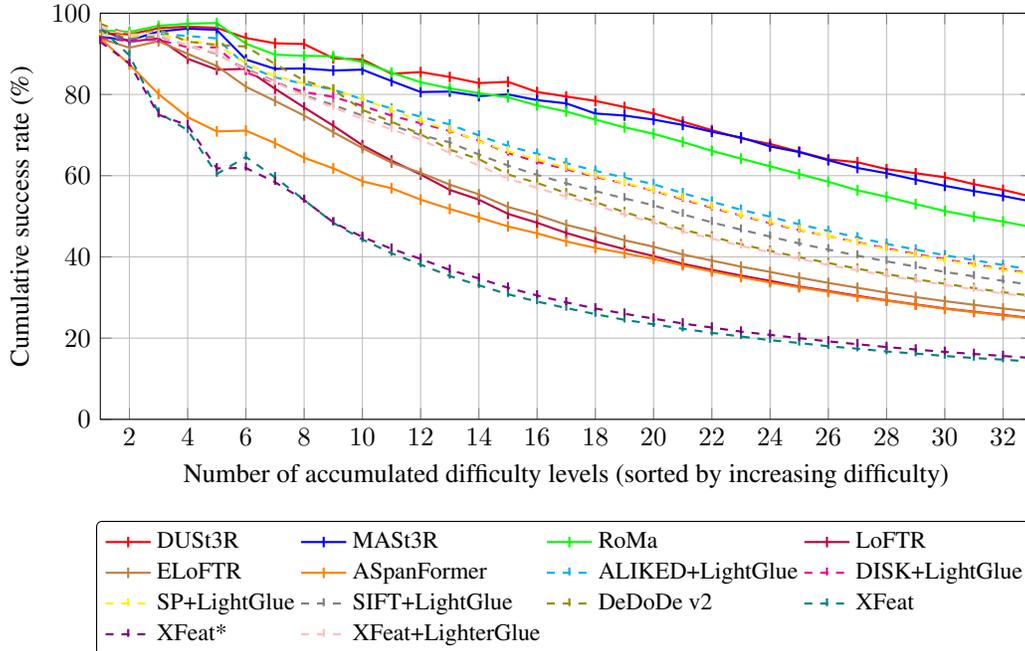


Figure 9. **Cumulative success rates across difficulty levels** – Methods are evaluated on increasingly difficult image pairs, sorted by the average success rate across all methods. Solid lines represent detector-free methods while dashed lines represent detector-based methods. The plot shows how performance degrades as more challenging pairs are included in the evaluation.

Table 2. **Detailed Results by Geometric Criterion** – Success rate (in %) for each method across individual geometric criterion bins. Best and second-best values for each column are shown in **bold** and underlined respectively.

	Overlap (%)					Scale Ratio				Viewpoint Angle (°)				Whole Dataset
	80–100	60–80	40–60	20–40	5–20	1.0–1.5	1.5–2.5	2.5–4.0	4.0–6.0	0–30	30–60	60–120	120–180	
Number of boxes	1	3	5	9	15	14	8	7	4	9	9	12	3	33
<i>Detector-based methods</i>														
ALIKED+LightGlue [55]	53.4	95.8	68.2	<u>38.0</u>	12.7	62.0	31.0	13.1	1.6	50.6	46.0	28.3	<u>2.0</u>	36.8
DISK+LightGlue [47]	54.2	91.4	65.9	38.7	11.8	60.4	<u>30.8</u>	11.6	2.4	<u>50.3</u>	<u>43.8</u>	27.4	2.7	<u>35.9</u>
SP+LightGlue [12]	64.8	<u>93.3</u>	<u>68.0</u>	36.4	10.9	<u>61.2</u>	28.4	<u>12.5</u>	1.4	49.9	43.0	<u>28.2</u>	0.9	35.7
SIFT+LightGlue [32]	68.2	92.1	61.4	32.3	9.9	57.3	26.9	9.6	1.7	49.8	39.7	23.7	0.5	33.1
DeDoDe v2 [15]	89.8	<u>93.3</u>	54.8	26.7	7.9	60.4	16.3	3.2	0.9	49.3	35.4	19.9	0.3	30.4
XFeat [37]	<u>85.4</u>	67.4	24.3	5.2	0.9	32.1	2.4	0.1	0.0	34.4	8.3	7.1	0.0	14.2
XFeat* [37]	62.4	69.1	27.6	7.6	1.5	32.8	4.5	0.6	0.0	33.8	9.4	9.2	0.0	15.1
XFeat+LighterGlue [37]	64.6	91.7	59.1	26.2	8.1	56.6	20.9	4.6	0.2	48.0	33.4	21.4	1.2	30.1
<i>Detector-free methods</i>														
LoFTR [42]	87.2	88.4	47.2	17.5	5.0	51.6	10.1	2.3	0.6	43.2	27.9	15.1	0.0	24.9
ELoFTR [50]	56.4	90.3	50.8	22.1	6.3	51.2	15.6	4.4	0.7	42.2	30.8	18.2	0.1	26.6
ASpanFormer [9]	72.2	72.3	44.5	21.9	7.4	46.0	14.9	6.9	1.6	42.5	27.2	16.0	0.1	24.8
RoMa [16]	67.0	98.3	84.5	52.7	20.2	<u>71.2</u>	43.2	26.6	8.3	<u>57.5</u>	<u>56.2</u>	44.1	3.0	47.3
DUS3R [49]	<u>81.8</u>	97.4	90.8	<u>58.4</u>	30.4	73.3	57.9	<u>40.1</u>	<u>9.9</u>	67.4	55.3	<u>50.0</u>	35.2	54.8
MAS3R [29]	52.0	<u>97.5</u>	<u>89.6</u>	61.0	<u>28.4</u>	<u>71.2</u>	<u>52.3</u>	42.5	13.8	53.5	65.6	54.5	<u>14.1</u>	<u>53.6</u>

image pairs. The main challenge stems from dynamic objects in the scenes, as illustrated in Fig. 11.

Our co-visibility computation relies on static scene geometry, which cannot properly account for moving objects. When dynamic objects (such as vehicles or pedestrians) appear in different positions in image pairs, our method may incorrectly label pixels as co-visible simply because they occupy the same 3D space, even though they correspond to

different objects. This limitation particularly affects urban scenes where temporary occlusions and moving objects are common.

While this does not invalidate our benchmark’s utility for evaluating the methods, it does suggest potential areas for improvement in co-visibility estimation, particularly for dynamic scene understanding. Future work could explore incorporating instance segmentation or temporal consistency

Table 3. **Difficulty Level Ordering** – All 33 difficulty levels sorted by decreasing average success rate across all methods. Each level is defined by its overlap range (%), scale ratio range, and viewpoint angle range (°).

Level	Overlap (%)	Scale Ratio	Viewpoint (°)	Success (%)
1	60–80	1.0–1.5	0–30	95.2
2	40–60	1.0–1.5	0–30	89.9
3	60–80	1.0–1.5	30–60	88.0
4	60–80	1.0–1.5	60–120	82.2
5	40–60	1.0–1.5	30–60	75.5
6	80–100	1.0–1.5	0–30	68.5
7	20–40	1.0–1.5	0–30	60.6
8	40–60	1.0–1.5	60–120	57.9
9	20–40	1.0–1.5	30–60	52.7
10	40–60	1.5–2.5	60–120	47.1
11	5–20	1.0–1.5	0–30	40.6
12	20–40	1.5–2.5	0–30	40.4
13	20–40	1.5–2.5	30–60	36.7
14	20–40	1.0–1.5	60–120	33.0
15	40–60	2.5–4.0	60–120	28.3
16	5–20	1.0–1.5	30–60	27.6
17	20–40	1.5–2.5	60–120	25.3
18	5–20	1.5–2.5	0–30	22.5
19	20–40	2.5–4.0	30–60	22.2
20	5–20	1.5–2.5	30–60	20.5
21	20–40	2.5–4.0	60–120	12.2
22	5–20	1.0–1.5	60–120	10.6
23	5–20	2.5–4.0	30–60	9.3
24	5–20	2.5–4.0	0–30	9.0
25	5–20	1.5–2.5	60–120	6.4
26	5–20	4.0–6.0	0–30	5.4
27	5–20	1.0–1.5	120–180	5.0
28	5–20	2.5–4.0	60–120	4.1
29	5–20	1.5–2.5	120–180	4.1
30	5–20	2.5–4.0	120–180	3.8
31	5–20	4.0–6.0	30–60	3.0
32	20–40	4.0–6.0	60–120	2.9
33	5–20	4.0–6.0	60–120	1.0

checks to better handle dynamic objects when computing co-visibility maps.

9. Visualization of Geometric Criteria

We provide visual examples of image pairs for each geometric criterion bin, along with 100 randomly sampled matches from different methods in Figs. 12 to 14. For each bin, we show results on two image pairs, from the two best methods in either detector-based (ALIKED+LightGlue) or detector-free (DUST3R) approaches.

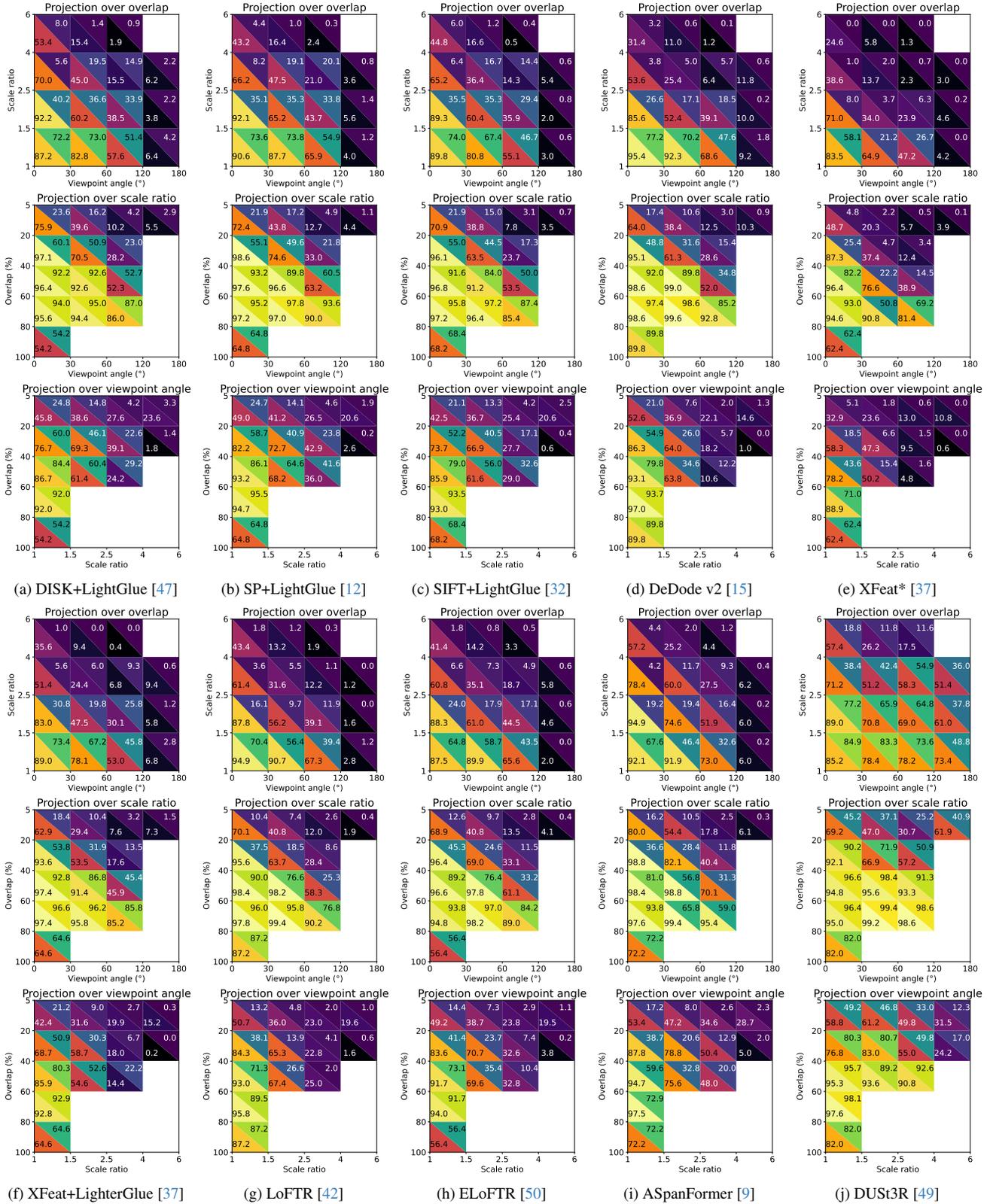


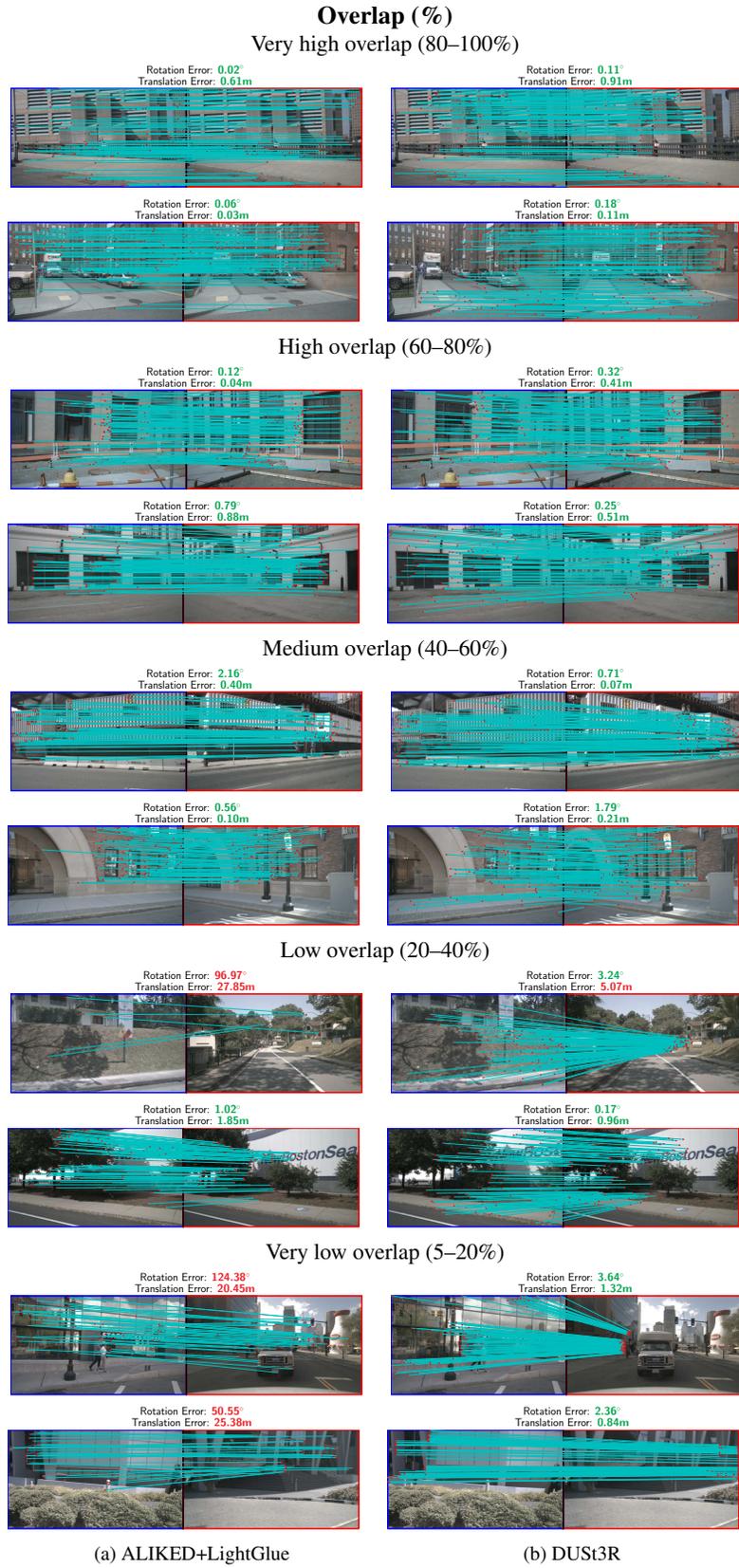
Figure 10. Performance analysis across geometric criteria – Results for other methods not in the main paper, similar than Fig. 8.



(a) First view

(b) Second view

Figure 11. **Limitations in co-visibility estimation** – Our method for determining co-visible regions can be affected by dynamic objects in the scene. In these examples, different cars occupy the same space in two temporally separated views. On the top pair, the white car replaces the gray car, and part of both cars are marked as co-visible. On the bottom pair, the cars turning in both views are different, but marked as co-visible as well. This highlights a limitation in handling dynamic scene elements when computing co-visibility maps.



(a) ALIKED+LightGlue

(b) DUST3R

Figure 12. **Examples of image pairs with varying overlap** – For each overlap range, we show two random image pairs for the best methods in either detector-based (ALIKED+LightGlue on the left) or detector-free (DUST3R on the right) approaches.

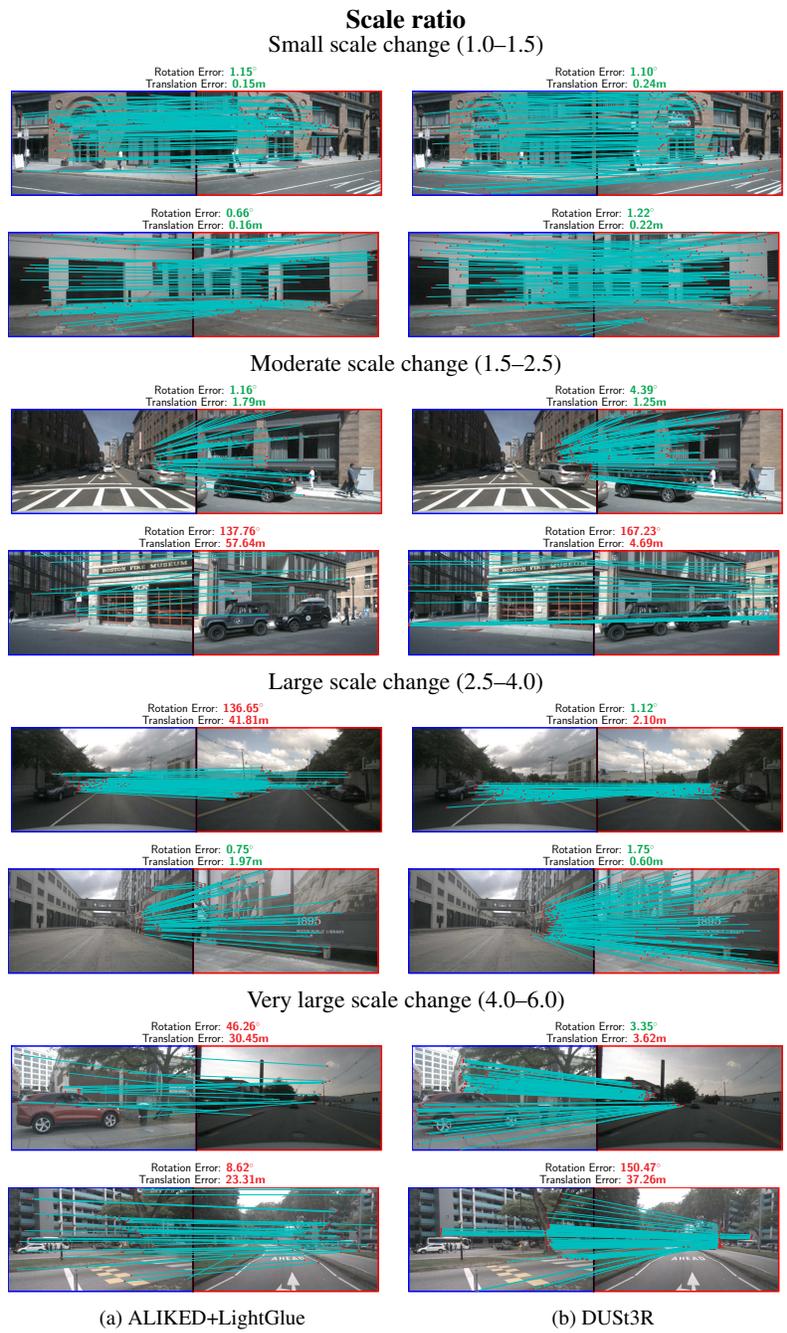


Figure 13. **Examples of image pairs with varying scale ratios** – For each scale ratio range, we show two random image pairs for the best methods in either detector-based (ALIKED+LightGlue on the left) or detector-free (DUST3R on the right) approaches.

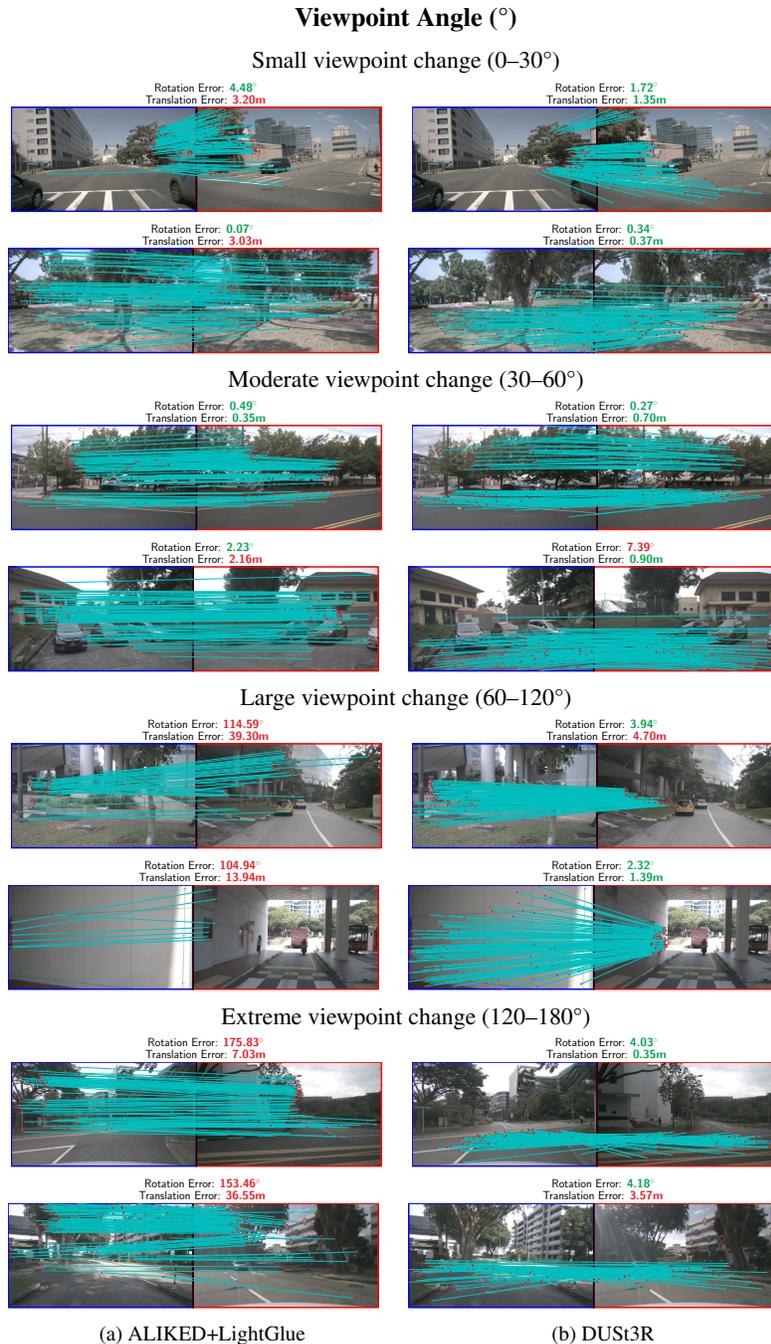


Figure 14. **Examples of image pairs with varying viewpoint angles** – For each viewpoint angle range, we show two random image pairs for the best methods in either detector-based (ALIKED+LightGlue on the left) or detector-free (DUST3R on the right) approaches.