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%), DUSt3R and MASt3R 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 DUSt3R and MASt3R, 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

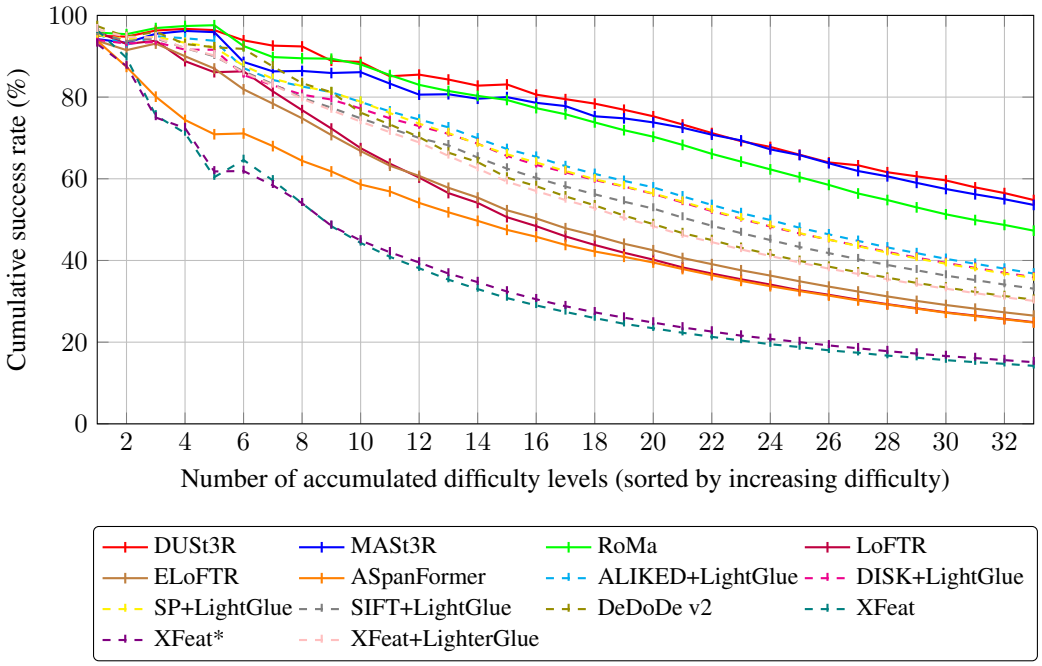


Figure 9. **Cumulative success rates across difficulty levels –** Methods are evaluated on increasingly difficult image pairs, sorted by the average success rate across all methods. Solid lines represent detector-free methods while dashed lines represent detector-based methods. The plot shows how performance 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
	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	Dataset
Number of boxes	1	3	5	9	15	14	8	7	4	9	9	12	3	33
Detector-based methods														
ALIKED+LightGlue [57]	53.4	**95.8**	**68.2**	38.0	**12.7**	**62.0**	**31.0**	**13.1**	1.6	**50.6**	**46.0**	**28.3**	2.0	**36.8**
DISK+LightGlue [49]	54.2	91.4	65.9	**38.7**	11.8	60.4	30.8	11.6	**2.4**	50.3	43.8	27.4	**2.7**	35.9
SP+LightGlue [12]	64.8	93.3	68.0	36.4	10.9	61.2	28.4	12.5	1.4	49.9	43.0	28.2	0.9	35.7
SIFT+LightGlue [33]	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**	93.3	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 [38]	85.4	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* [38]	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 [38]	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
Detector-free methods														
LoFTR [44]	**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 [52]	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	71.2	43.2	26.6	8.3	57.5	56.2	44.1	3.0	47.3
DUSt3R [51]	81.8	97.4	**90.8**	58.4	**30.4**	**73.3**	**57.9**	40.1	9.9	**67.4**	55.3	50.0	**35.2**	**54.8**
MASt3R [30]	52.0	97.5	89.6	**61.0**	28.4	71.2	52.3	**42.5**	**13.8**	53.5	**65.6**	**54.5**	14.1	53.6

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. Implementation Details

9.1. COLMAP Configuration

For our COLMAP reconstructions, we use the following configuration:

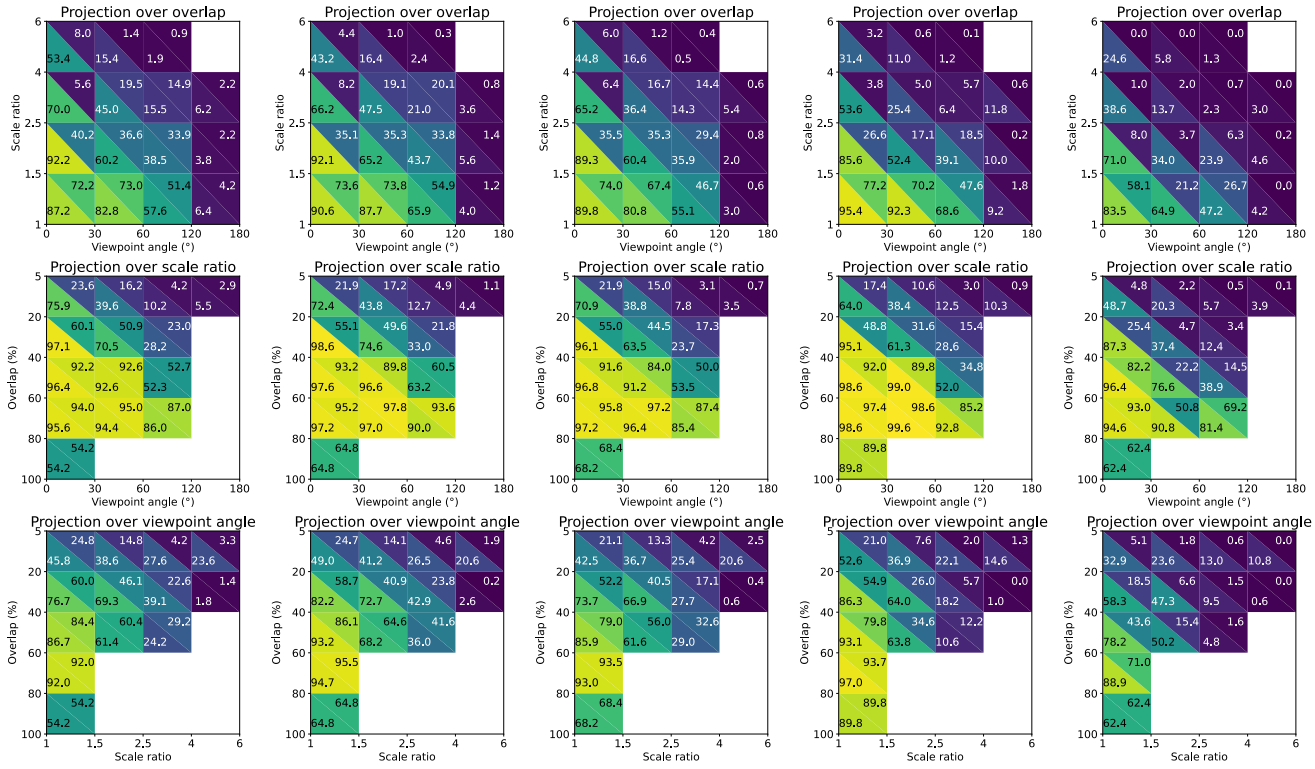
- Intrinsic parameters: We consider intrinsic parameters per camera per scene
- Camera model: Simple pinhole camera model, as images are already undistorted according to nuScenes documentation
- Matching: Exhaustive matching using SIFT descriptors

9.2. Depth Map Alignment

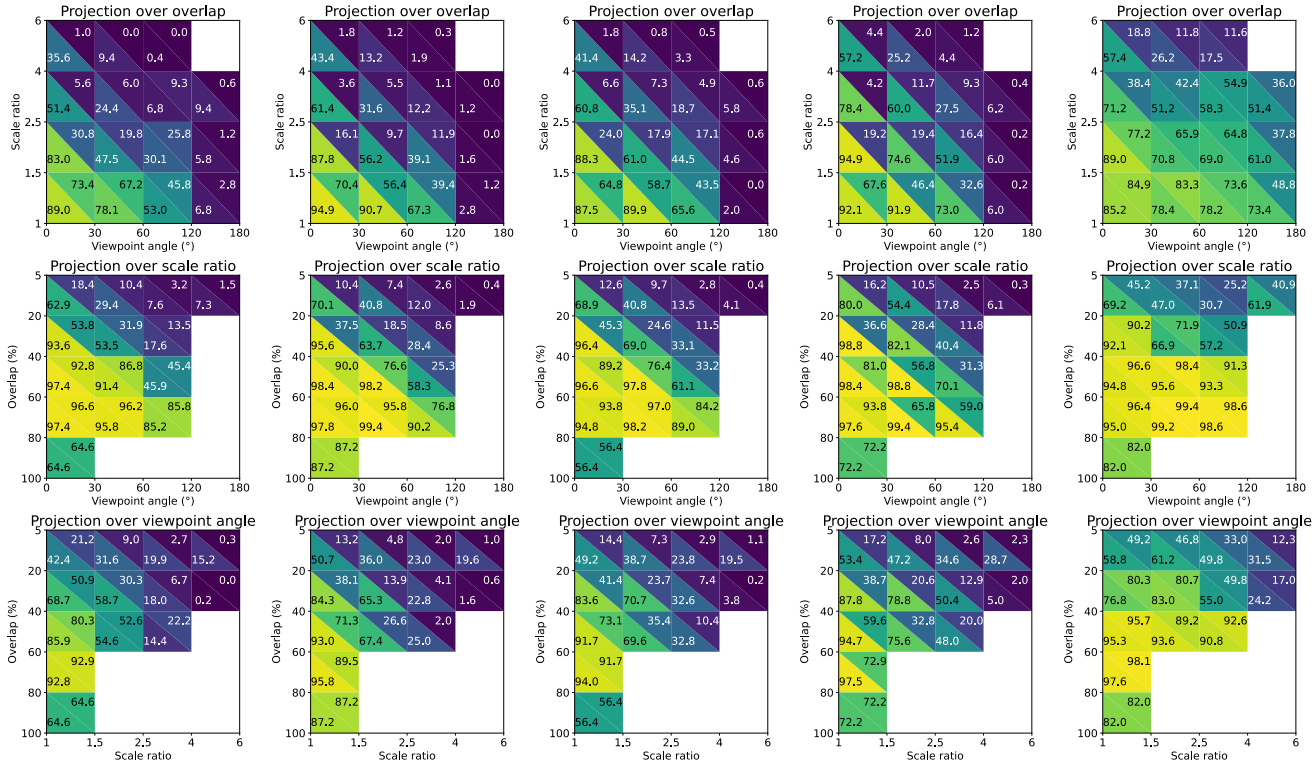
To ensure rigorous geometric criteria computation, we align Depth Anything V2 depth maps with COLMAP sparse reconstruction. This alignment is particularly important for accurate co-visibility estimation between views. We found that UniDepth is surprisingly accurate on nuScenes (as shown in Tab. 1 of https://arxiv.org/pdf/2410.02073), but the alignment with COLMAP provides additional validation of our geometric criteria computation.

9.3. Fundamental Matrix Evaluation

We also evaluated all methods using fundamental matrix estimation (OpenCV findFundamentalMat with



(a) DISK+LightGlue [49] (b) SP+LightGlue [12] (c) SIFT+LightGlue [33] (d) DeDode v2 [15] (e) XFeat* [38]



(f) XFeat+LighterGlue [38] (g) LoFTR [44] (h) ELoFTR [52] (i) ASpanFormer [9] (j) DUSt3R [51]

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.

MAGSAC++) to provide a comprehensive comparison. The results are shown in Tab. 4. While the ranking remains similar, all methods show significantly lower performance compared to essential matrix estimation, indicating that our image pairs are challenging and require the 5-point minimal solver. Interestingly, fundamental matrix estimation performs better for high-overlap cases (80-100%), where small translations make essential matrix estimation more sensitive to noise.

9.4. Additional Baselines

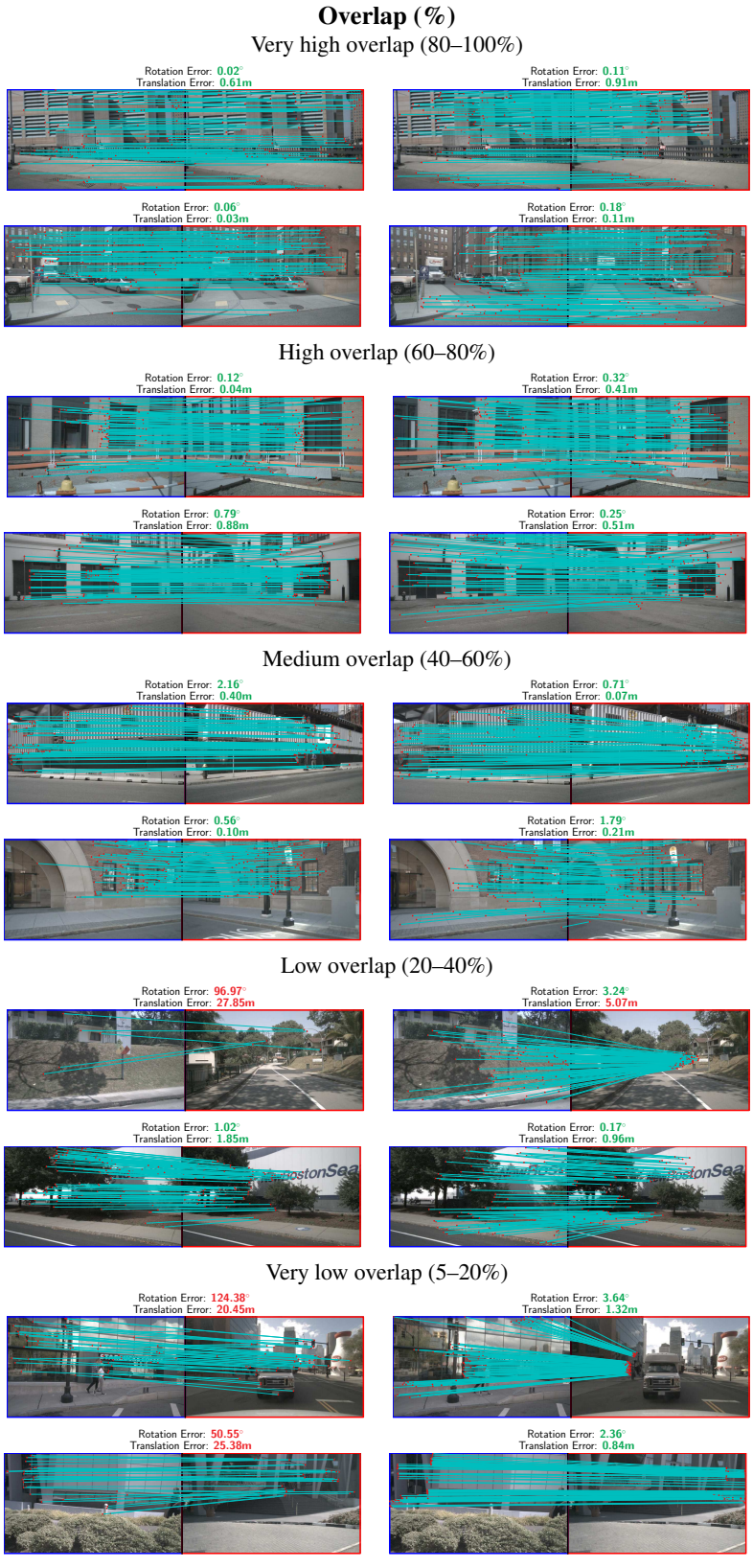
We also evaluated two additional baselines: rootSIFT and ORB [40], both with brute force matching. These classical methods provide valuable context. We also evaluated fundamental matrix estimation (OpenCV findFundamentalMat with MAGSAC++) to provide a comprehensive comparison. While the ranking remains similar, all methods show significantly lower performance with fundamental matrix estimation, indicating that our image pairs are challenging and require the 5-point minimal solver. Interestingly, fundamental matrix estimation performs better for high-overlap cases (80-100%), where small translations make essential matrix estimation more sensitive to noise. The results are shown in Tab. 4 alongside the other methods.

10. 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.

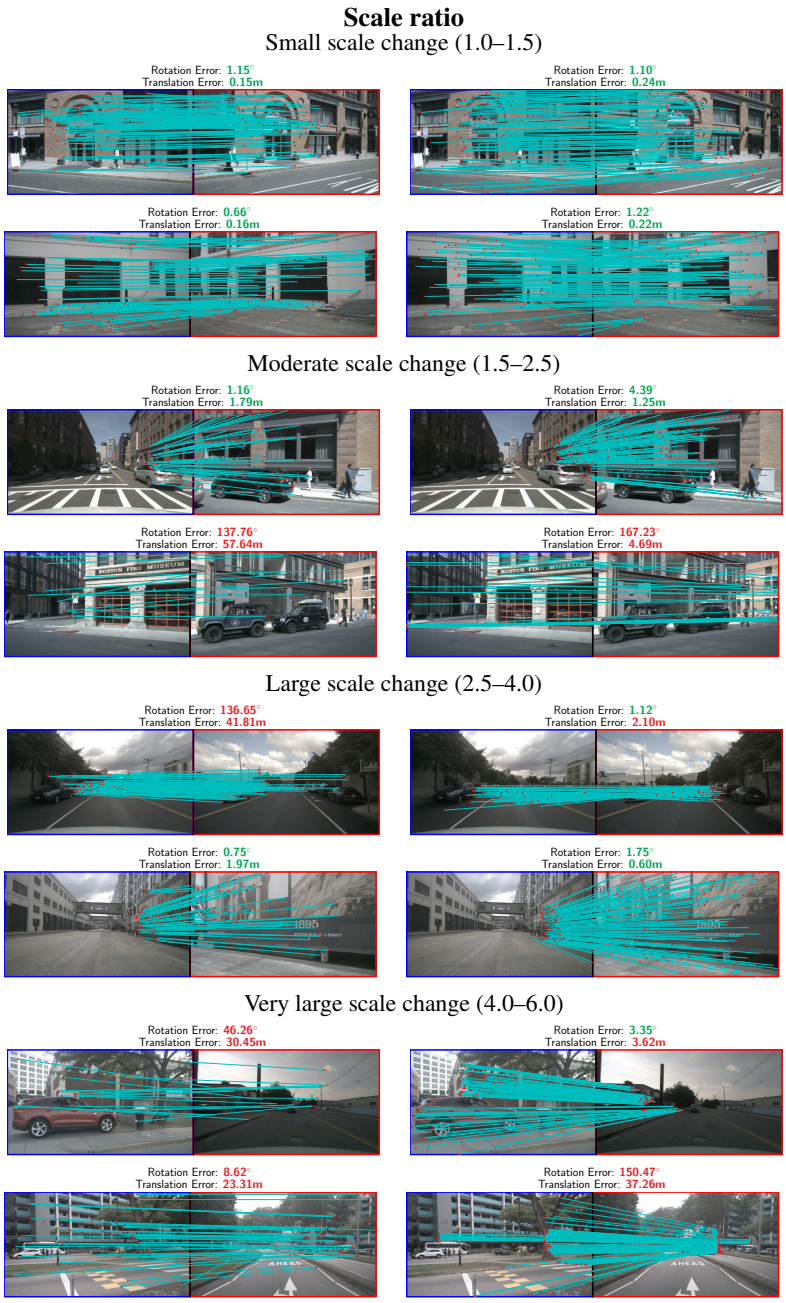
Table 4. **Essential vs. Fundamental Matrix –** Comparison of pose estimation using essential vs fundamental matrix estimation at 5° / 2m threshold.

	Essential		Fundamental	
Method	Med.R	Succ(%)	Med.R	Succ(%)
Detector-based methods				
ALIKED+LG	**5**	**36.8**	6	**12.4**
DISK+LG	**5**	35.9	**5**	12.3
SP+LG	6	35.7	10	11.8
SIFT+LG	7	33.1	6	12.0
DeDoDe v2	9	30.4	9	10.5
XFeat	15	14.2	11	8.4
XFeat*	14	15.1	13	8.8
XFeat+LG	9	30.1	10	11.0
rootSIFT	13	15.5	13	8.8
ORB	16	8.3	14	6.5
Detector-free methods				
LoFTR	11	24.9	12	10.0
ELoFTR	10	26.6	11	11.0
ASpanFormer	10	24.8	4	12.5
RoMa	3	47.3	3	14.7
DUSt3R	**2**	**54.8**	**2**	**16.8**
MASt3R	**2**	53.6	**2**	16.5



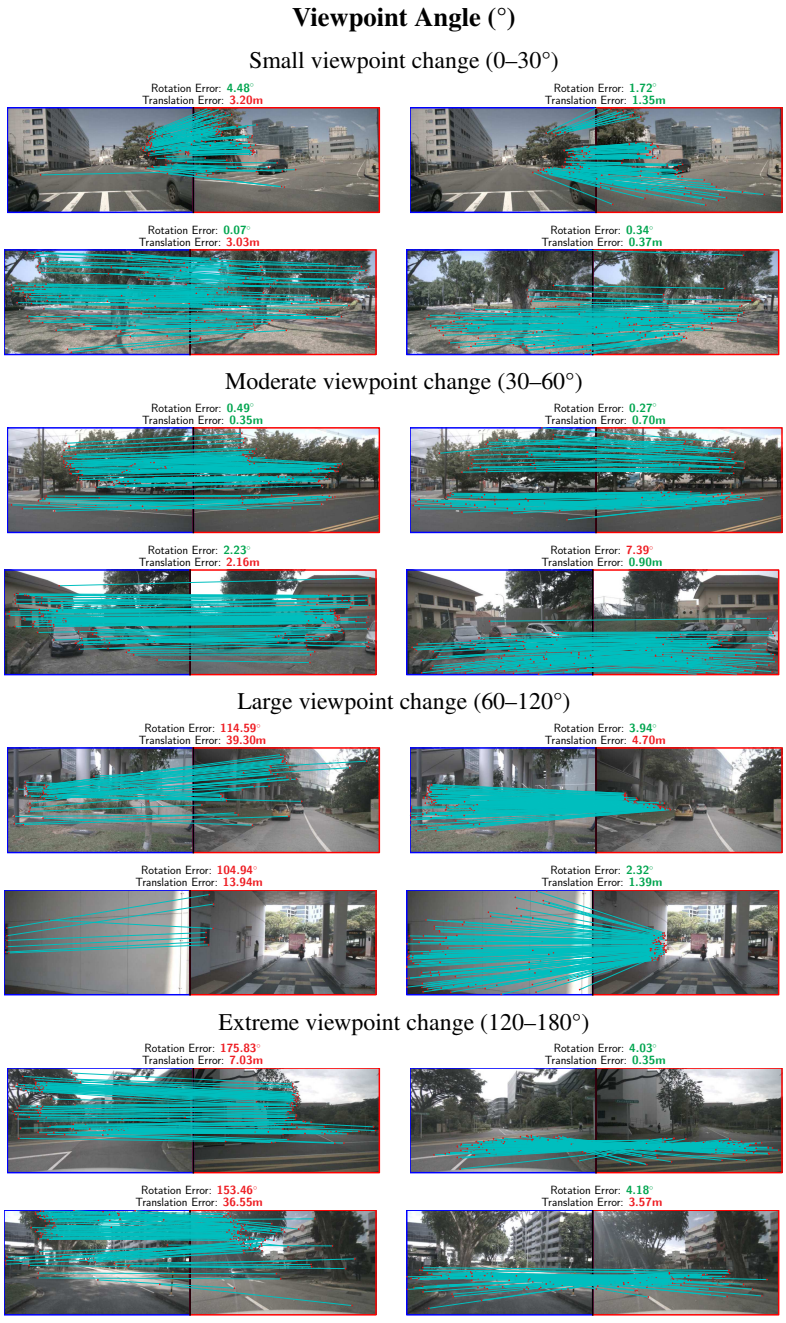
(a) ALIKED+LightGlue (b) DUSt3R

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.



(a) ALIKED+LightGlue (b) DUSt3R

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.



(a) ALIKED+LightGlue (b) DUSt3R

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.