SGAD: Semantic and Geometric-aware Descriptor for Local Feature Matching

Supplementary Material

A. Time Complexity Analysis

This section investigates the computational complexity of MESA [39] and SGAD. Given source and target images containing M and N areas, respectively, with each area resized to $H \times W$ pixels, our analysis focuses on the similarity computation stage.

A.1. MESA

MESA formulates area matching as a graph matching problem, where each area in the source and target images is represented as a graph node. In the first stage, activity maps are computed by applying self-attention and cross-attention on feature maps that have been downsampled to $\frac{1}{8}$ of the original resolution. The similarity of each paired area (R_A^i, R_B^j) is then determined as the product of their activity map expectations. The complexity is:

$$\mathcal{O}(L \times M \times N \times ((H' \times W')^2 \times D)), \tag{15}$$

where $H' = \frac{H}{8}$, $W' = \frac{W}{8}$, L is the number of attention layers, and D is the feature dimension.

To reduce this complexity, MESA employs Area Bayesian Network optimization to filter areas, reducing the number of areas to M' and N'. The complexity becomes:

$$\mathcal{O}(L \times M' \times N' \times ((H' \times W')^2 \times D)), \tag{16}$$

where $M' < M$ and $N' < N$.

The subsequent graph matching has a computational complexity of:

$$\mathcal{O}(M^2 + N^2). \tag{17}$$

A.2. SGAD

SGAD adopts a distinct strategy. For each area, it generates a compact descriptor by first pooling the corresponding DINOv2 features into a single vector. This initial descriptor is subsequently refined through N_{tr} layers of self-attention and cross-attention. The computational complexity is:

$$\mathcal{O}(N_{tr} \times M \times N \times D), \tag{18}$$

where D represents the feature dimension, which is independent of the image resolution.

Method	Step	Complexity
MESA [39]	Similarity Calculation	$\mathcal{O}(L \times M' \times N' \times (H' \times W')^2 \times D)$
	Graph Matching	$\mathcal{O}(M^2 + N^2)$
	Main Bottleneck	$\mathcal{O}(L \times M' \times N' \times (H' \times W')^2 \times D)$
SGAD	Similarity Calculation	$\mathcal{O}(N_{tr} \times M \times N \times D)$
	Descriptor Matching	$\mathcal{O}(M \times N)$
	Main Bottleneck	$\mathcal{O}(N_{tr} \times M \times N \times D)$

Table 7. Complexity analysis of SGAD and MESA [39].

After generating the confidence matrix $\mathcal{P}_{pr}$, SGAD uses the MNN algorithm to compute the matching results. The complexity is:

$$\mathcal{O}(M \times N). \tag{19}$$

A.3. Comparative Analysis

As shown in Tab. 7, the primary computational complexity of both SGAD and MESA lies in the similarity calculation module. MESA computes similarity node by node, requiring a complexity as described in Eq. (16), where the pixel-based activity map computation limits parallelism, and the cost remains tied to high-resolution image features. In contrast, SGAD's approach is independent of image resolution, which not only lowers the theoretical complexity but also enables more efficient hardware implementation, as discussed next.

In the matching stage, SGAD utilizes the Mutual Nearest Neighbor (MNN) algorithm, which operates directly on the dense confidence matrix $\mathcal{P}_{pr}$, The MNN algorithm benefits from efficient GPU parallelization due to its simplicity and the dense matrix structure, making it highly scalable for large-scale tasks. By contrast, the graph matching in MESA relies on iterative optimizations over sparsely connected graphs, which inherently limits its scalability and efficiency, especially when dealing with a large number of areas.

To validate the theoretical analysis, we measured the runtime performance of MESA and SGAD under varying numbers of areas (AreaNum). As shown in Tab. 8, the empirical results closely align with the theoretical predictions. When AreaNum increases from 11.18 to 30.89, the runtime of MESA increases by more than 6 times (from 49.92s to 311.63s). In contrast, when AreaNum increases from 15.13 to 36.44, SGAD exhibits only a minor runtime increase of approximately 36%

Method	Time(s)↓	AreaNum	AreaMatchesNum
MESA [39]	49.92	11.18	7.42
SGAD	**0.25**	15.13	10.44
MESA [39]	311.63	30.89	19.91
SGAD	**0.34**	36.44	21.54

Table 8. Runtime comparison for the area matching stage on the MegaDepth1500 benchmark. The results highlight the superior efficiency of SGAD compared to MESA [39].

Pose AUC	MegaDepth1500 benchmark(image size 1200x1200)								
	832x832 (area size)			640x640 (area size)			480x480 (area size)		
	@5° ↑	@10° ↑	@20° ↑	@5° ↑	@10° ↑	@20° ↑	@5° ↑	@10° ↑	@20° ↑
LoFTR	61.49	75.47	85.27	61.49	75.47	85.27	61.49	75.47	85.27
SGAD+LoFTR	**66.24**	**78.40**	**86.75**	**65.10**	**77.90**	**86.44**	**65.12**	**77.86**	**86.56**
DKM	61.11	74.63	84.02	61.11	74.63	84.02	61.11	74.63	84.02
SGAD+DKM	**66.40**	**78.38**	**86.51**	**65.97**	**78.02**	**86.38**	**65.91**	**78.27**	**86.52**
ROMA	65.68	78.15	86.68	65.68	78.15	86.68	65.68	78.15	86.68
SGAD+ROMA	**68.43**	**80.35**	**88.26**	**68.12**	**80.24**	**88.14**	**67.17**	**79.07**	**87.32**

Table 9. Relative pose estimation results (%) on MegaDepth1500. Measured in AUC (higher is better). The baseline methods (LoFTR, DKM, ROMA) were evaluated on the full-resolution images, and their results are presented across all columns for direct comparison.

(from 0.25s to 0.34s), demonstrating its scalability and computational efficiency in handling large-scale tasks.

Ultimately, the significant performance gap observed in Tab. 8 stems from these fundamental architectural differences. The reliance of MESA on pixel-based activity maps and node-by-node matching leads to significant bottlenecks, particularly in high-density scenarios. By contrast, the area descriptor approach and dense matrix computations employed by SGAD drastically reduce complexity. Its ability to fully exploit GPU parallelization underscores its efficiency and suitability for large-scale tasks.

B. Effect of Area Size on Different Point Matchers

In this section, we analyze the impact of different area sizes on the performance of various point matchers on the MegaDepth1500 benchmark. Image size resized to 1200×1200. For the area sizes, we tested 832×832, 640×640, and 480×480. As shown in Tab. 9, SGAD significantly improves the performance of LoFTR, DKM, and Roma across multiple area sizes. This further demonstrates the effectiveness of SGAD in improving the performance of different point matchers.

C. Failure Cases

In Fig. 7, we illustrate SGAD's primary failure mode, which arises in challenging cases that combine high visual similarity with extreme geometric transformations. The red color indicates mismatched areas, while the yellow color highlights correctly matched ones.

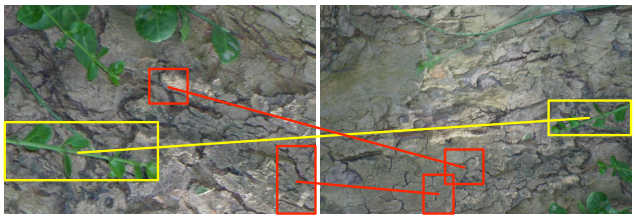

Figure 7. Failure cases of SGAD, demonstrating its vulnerability in challenging cases that combine high visual similarity and extreme geometric transformations. Mismatched areas are shown in red, correct matches in yellow.

This failure stems from a detrimental synergy between our model's architecture and its training data. Our architecture, prioritizing global context via DINOv2 and pooling, inherently sacrifices the fine-grained local features required to distinguish between visually similar areas. This architectural limitation becomes particularly critical because the training data (MegaDepth and ScanNet) lacks sufficient examples of extreme geometric transformations. Consequently, the model is not explicitly trained to be robust against such distortions. When confronted with them, it must rely more heavily on the very local details that the architecture has already discarded, leading to inevitable matching failures.

D. Additional Qualitative Results

D.1. Area Matching

This section provides a qualitative comparison of area matching between SGAD, MESA [39], and DMESA [38]. As shown in Fig. 8, SGAD consistently finds more content-consistent area matches across the MegaDepth and ScanNet datasets. This improved consistency establishes a stronger foundation for subsequent pixel-level matching.

D.2. Relative Pose Estimation

Qualitative results for relative pose estimation are presented for the MegaDepth (Figs. 9 and 10) and ScanNet (Figs. 11 and 12) datasets. Following the protocol of LoFTR [31], we report rotation and translation errors. Match precision is visualized by epipolar error, where red indicates errors exceeding the threshold (1×10^{-4} for MegaDepth and 5×10^{-4} for ScanNet). Across both datasets, our method consistently achieves more correct matches and lower pose errors, highlighting its robustness and accuracy under diverse conditions.

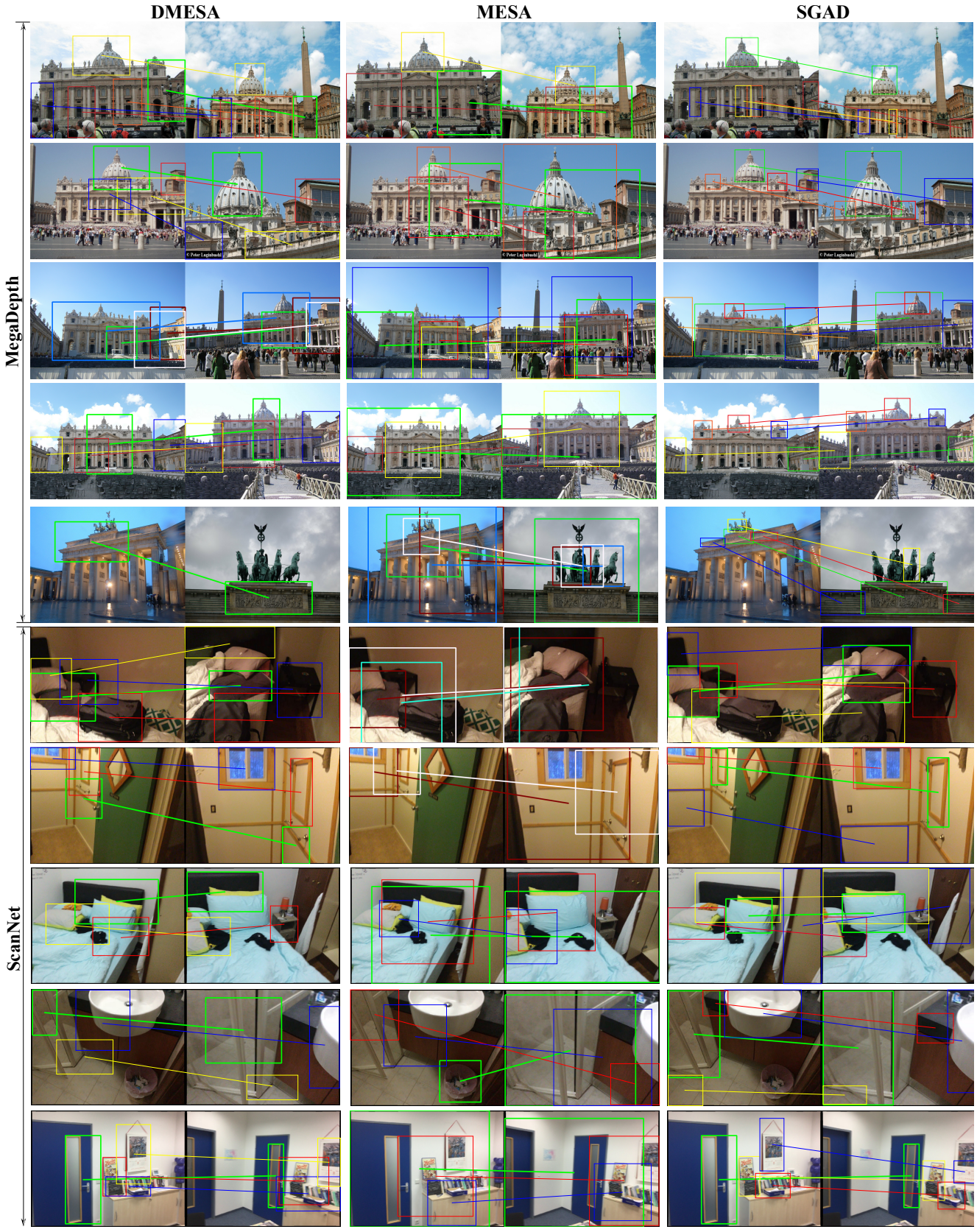

Figure 8. Qualitative comparison of area matching results on the MegaDepth and ScanNet datasets. Our method (SGAD) is compared against MESA [39] and DMESA [38]. The visualizations show that SGAD consistently identifies more semantically coherent area pairs, providing a better foundation for subsequent fine-grained matching.

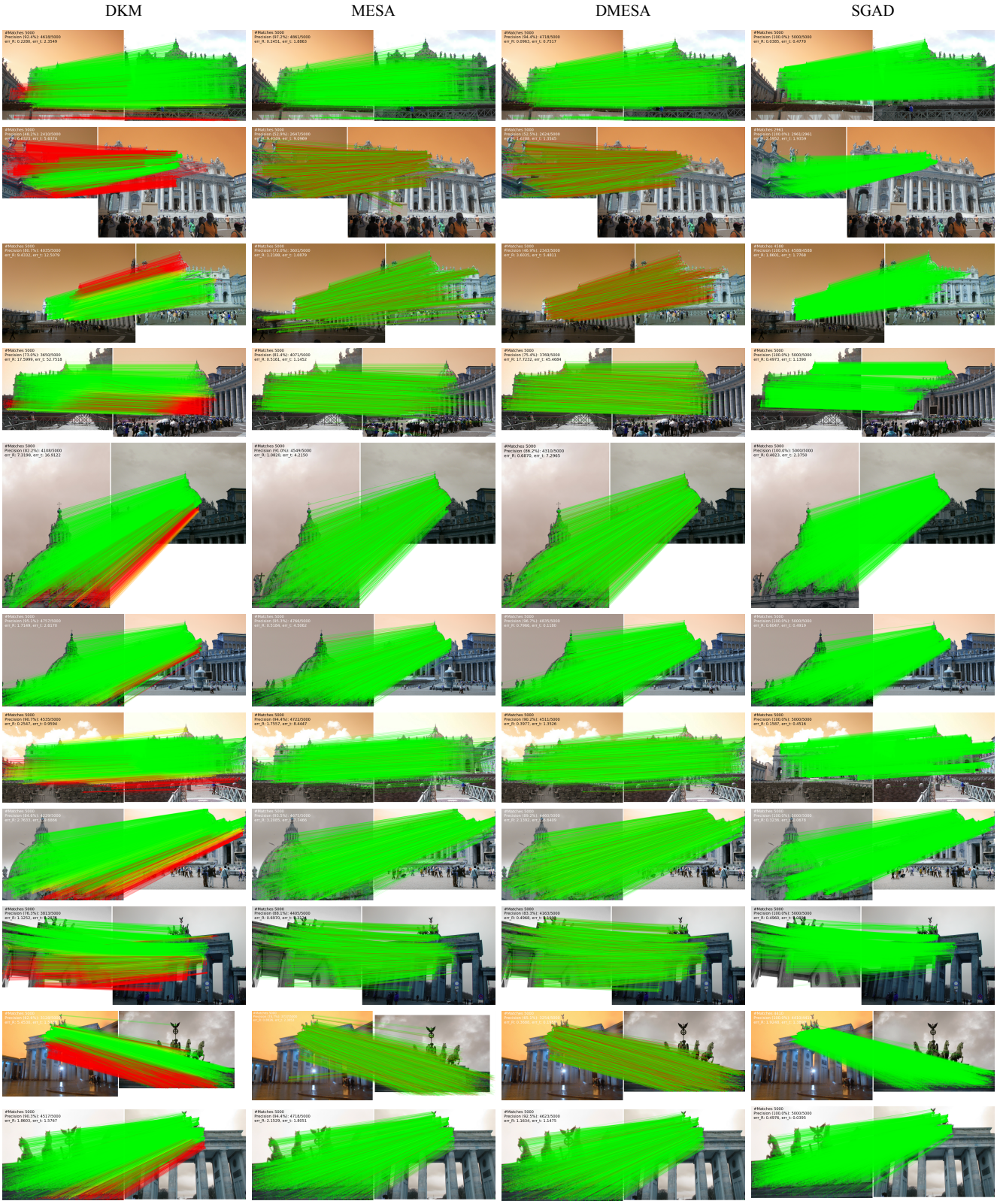

Figure 9. Qualitative comparison of matching methods on the MegaDepth dataset. Our method (SGAD) is compared against MESA [39], DMESA [38], and DKM [10]. To ensure a fair comparison of the upstream area matchers, SGAD, MESA, and DMESA all use DKM as the downstream point matcher, while DKM is also evaluated as a standalone baseline. Matches with an epipolar error greater than 1×10^{-4} are highlighted in red. The results show SGAD leads to more correct final matches and lower pose errors.

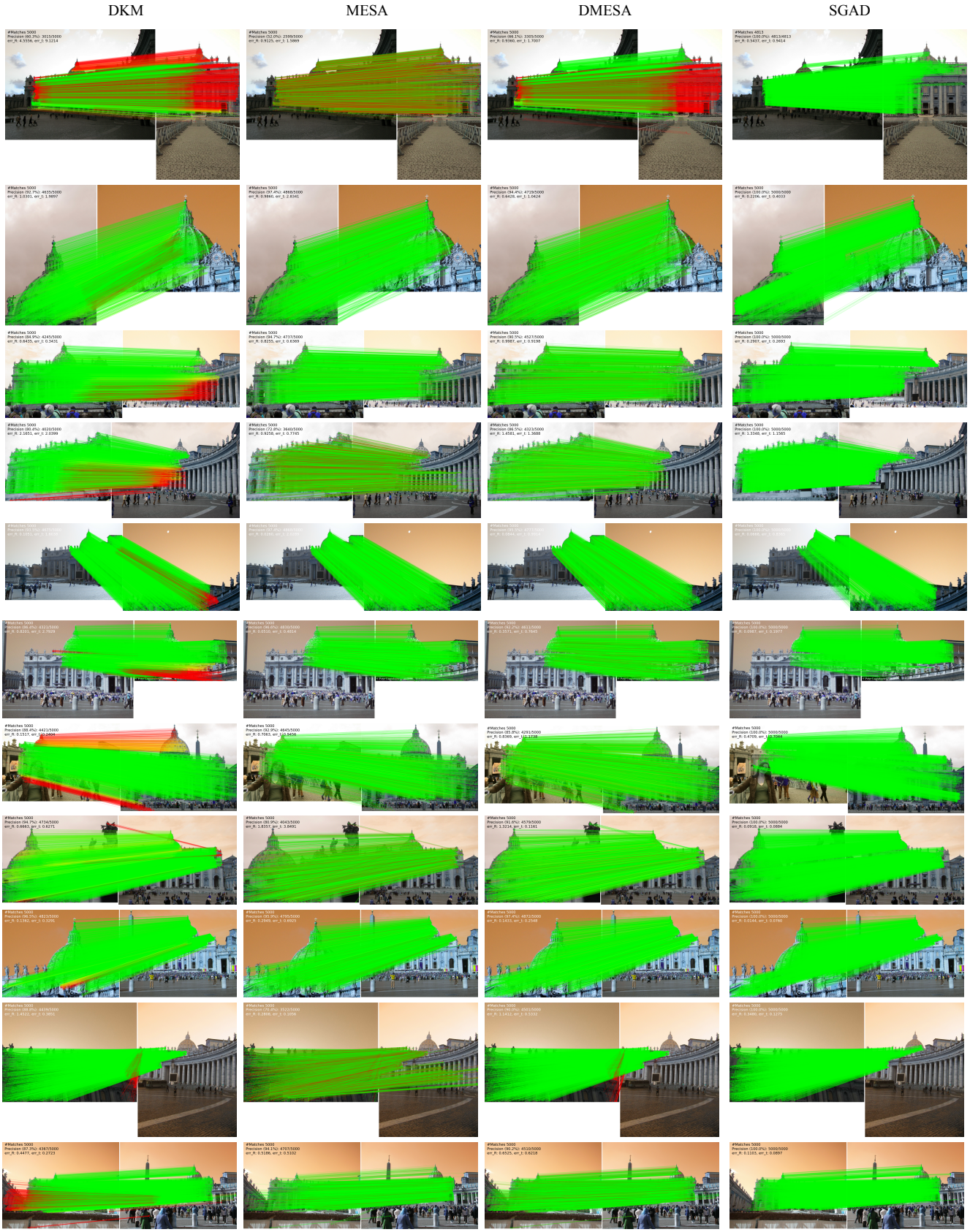

Figure 10. Qualitative comparison of matching methods on the MegaDepth dataset (continued).

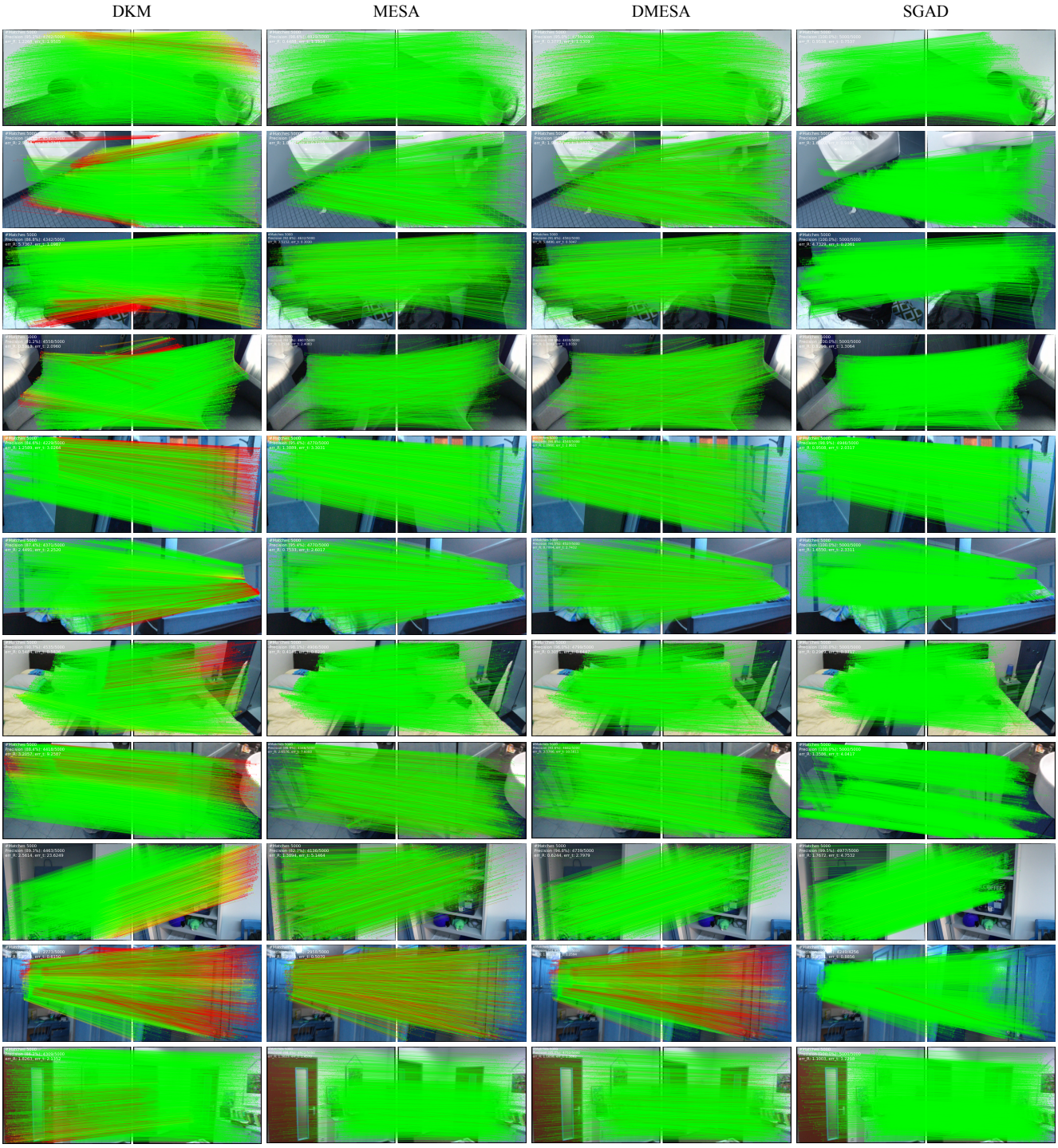

Figure 11. Qualitative comparison of matching methods on the ScanNet dataset. Our method (SGAD) is compared against MESA [39], DMESA [38], and DKM [10]. To ensure a fair comparison of the upstream area matchers, SGAD, MESA, and DMESA all use DKM as the downstream point matcher, while DKM is also evaluated as a standalone baseline. Matches with an epipolar error greater than 5×10^{-4} are highlighted in red. The results show SGAD leads to more correct final matches and lower pose errors.

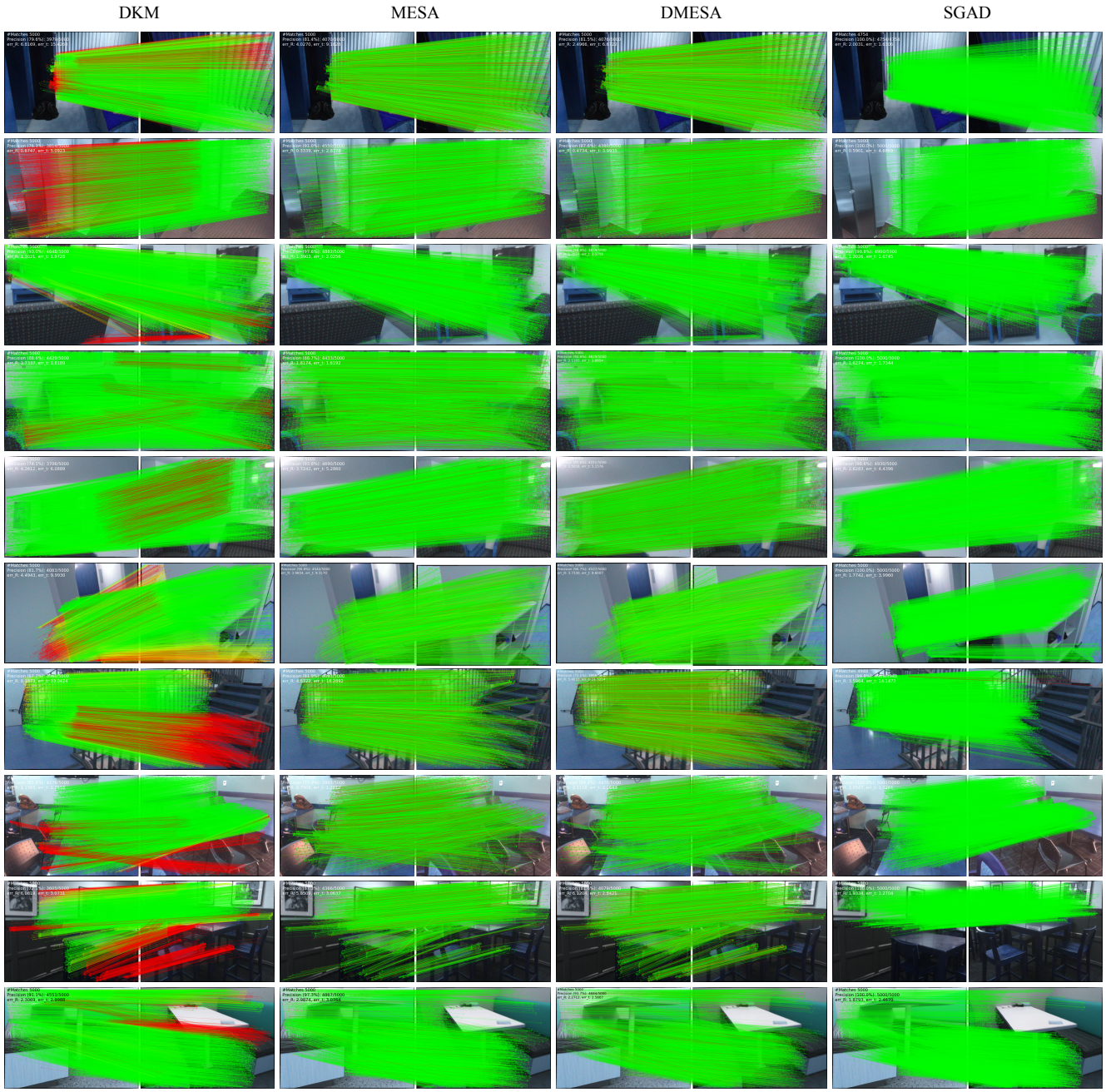

Figure 12. Qualitative comparison of matching methods on the ScanNet dataset (continued).