

Supplementary Materials:

PRISM: PProgressive dependency maxIimization for Scale-invariant image Matching

Anonymous Authors

1 SUMMARY

In this supplementary material, we provide more details and experiment results about our proposed image matching method: PProgressive dependency maxIimization for Scale-invariant image Matching (PRISM). We first elaborate on the RoPE [13] and supervision of the Patch Pruning module mentioned in the main manuscript (Sec. 2). Then, we conduct more experiments to validate the effectiveness of our design choices and superior performance (Sec. 3). Finally, we give more qualitative results on MegaDepth [7], ScanNet [3] and InLoc [15] datasets (Sec. 4).

2 IMPLEMENTATION DETAILS

2.1 Details about Positional Encoding

We adopt the Rotary Position Embedding (RoPE) [13] to model the spatial positional relationships between patch features. Given the i th projected query $Q_i \in \mathbb{R}^d$ and n projected K-V pairs where $K_j \in \mathbb{R}^d$ and $V_j \in \mathbb{R}^d$, the attention message m_i can be defined as:

$$m_i = \sum_{j=1}^n \text{softmax}(a(Q_i, K_j)) V_j \quad (1)$$

The RoPE does not fully model the positional information of each input. Instead, it considers the relative distance between the current position and the position being attended to when computing attention score. In this case, the attention score between the query feature Q_i and j th key feature $K_j \in \mathbb{R}^d$ is defined as:

$$a_{ij} = Q_i^T R(\Delta x, \Delta y) K_j \quad (2)$$

Where $R(\cdot) \in \mathbb{R}^{d \times d}$ is the rotary encoding and $\Delta x = x_i - x_j$, $\Delta y = y_i - y_j$. x_i, y_i is the coordinates of Q_i , defined as the normalized central coordinate of grid corresponding to Q_i at $\frac{1}{8}$ resolution. x_j, y_j is the coordinates of K_j and defined in the similar way. To compute the rotary encoding, the feature space is first divided into $\frac{d}{2}$ 2D subspaces and each of them is rotated by an angle:

$$R(\Delta x, \Delta y) = \begin{pmatrix} R_1(\theta_1) & & & \\ & R_2(\theta_2) & & \\ & & \ddots & \\ & & & R_{\frac{d}{2}}(\theta_{\frac{d}{2}}) \end{pmatrix} \quad (3)$$

$$R_k(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}, \theta = b_k^T(\Delta x, \Delta y) \quad (4)$$

Where $R_k(\theta)$ is the rotary matrix for k th subspace and $b_k \in \mathbb{R}^2$ is a learned parameter to project the distance between features into frequencies. The whole process is shown in Figure 1.

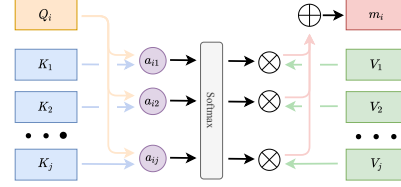


Figure 1: The process of attention mechanism. The \otimes means the product and the \oplus represents the summation. The RoPE considers the relative distance between query and key when computing the attention score a_{ij} .

resolution	Pose estimation AUC			VRAM (GiB)	time(ms)
	@5°	@10°	@20°	SP/HP	
1152 × 1152	60.0	74.9	85.1	13.5/11.0	209.1
960 × 960	58.3	73.0	83.6	9.9/6.6	153.3
832 × 832	56.4	71.6	82.7	7.7/4.5	119.7
640 × 640	52.9	68.5	80.2	5.4/3.2	99.4

Table 1: The Pose estimation AUC, VRAM and inference time under different test image resolutions. the SP means inference using Single-Precision and HP means Half-Precision.

2.2 Patch Pruning Supervision

Since the Patch Pruning is applied on coarse features at $\frac{1}{8}$ resolution, We use the central coordinates of grids at $\frac{1}{8}$ resolution to represent the coordinates of the coarse features. We project the central points in left image to the right image using its groundtruth depth map and camera pose, and take its nearest neighbor as a matching candidate. Then the central points in right image are projected to the left images in the same way. Based on the matching relationships in these two directions, we use MNN (Mutual Nearest Neighbors) filtering to obtain reasonable matches. The filtered matches can be used to indicate whether a feature can be matched or not and supervise the training.

3 MORE EXPERIMENTS RESULTS

3.1 Impact of test image resolutions

We farther study the impact of test image resolutions to PRISM on the Pose estimation AUC and report the optimized VRAM usage (inference using Half-Precision), as shown in table 1. All results are based on one single A100 GPU. As the resolution is reduced, PRISM can still achieve competitive performance and the inference time and memory efficiency have both been greatly improved.

Methods	Pose estimation AUC@5° / AUC@10° / AUC@10°			
	Scale [1, 2)	Scale [2, 3)	Scale [3, 4)	Scale [4, inf)
Sparse	SIFT [9]+HardNet [10]	21.2 / 33.0 / 45.4	10.8 / 18.6 / 28.6	4.6 / 9.3 / 16.2
	DISK [16]	33.7 / 49.8 / 63.3	5.5 / 8.5 / 11.6	0.2 / 0.5 / 0.8
	R2D2 [11]	37.8 / 55.9 / 70.7	22.7 / 36.9 / 51.9	6.6 / 13.0 / 22.0
	SP [4]+SG [12]	50.4 / 67.6 / 80.0	39.4 / 57.78 / 72.3	19.7 / 35.2 / 52.0
Dense	LoFTR [14]	60.2 / 74.7 / 84.5	49.7 / 65.7 / 77.9	24.9 / 39.7 / 55.1
	ASpanFormer [1]	60.9 / 75.3 / 85.0	54.6 / 70.2 / 81.2	33.4 / 51.2 / 66.9
	AdaMatcher [6]	62.4 / 76.0 / 85.4	57.0 / 71.8 / 82.6	41.0 / 58.7 / 73.4
	PRISM (Ours)	66.6 / 79.2 / 87.2	61.9 / 75.9 / 85.2	43.5 / 60.7 / 74.9

Table 2: Relative pose estimation on the scale-split Megadepth test set.

θ_p	Pose estimation AUC		
	@5°	@10°	@20°
0.8	56.2	71.7	82.8
0.85	56.2	71.6	82.7
0.9	56.1	71.6	82.9
0.95	56.4	71.6	82.7

Table 3: The impact of Patch Pruning threshold θ_p on the MegaDepth dataset. The test image resolution is 832×832 .

3.2 Evaluating Pose Estimation Errors on a scale-split Megadepth test set

To further demonstrate the effectiveness of PRISM in handling scale discrepancy, we evaluate the pose estimation errors on a scale-split Megadepth test set. The image pairs are split by their scale ratio ranges in [1, 2), [2, 3), [3, 4) and [4, $+\infty$), following [2, 6]. All images are resized so that the longest dimension equals 840 (ASpanFormer [1] and our method (PRISM) use 832 due to the need for an image resolution divisible by 16). The pose error AUC at thresholds of 5°, 10°, and 20° are reported.

We compare with sparse methods [4, 9–12, 16] and dense methods [1, 6, 14], as shown in table 2. Our method outperforms all baselines under various relative scale ratios substantially. We attribute the top performance to the hierarchical design of SADPA, which can aggregate features across different scales. The Patch Pruning module also contributes to the accuracy by pruning irrelevant features.

3.3 Ablation on the Patch Pruning Threshold

This section analyzes the impact of the Patch Pruning threshold θ_p . We train several variants of PRISM using $\theta_p \in \{0.95, 0.9, 0.85, 0.8\}$ and test the pose estimation errors at 832×832 image resolution. Table 3 shows the results of relative pose estimation results for PRISM on MegaDepth. As shown in the table, the model with $\theta_p = 0.95$ achieved the best performance. Other models with different θ_p perform slightly worse.

3.4 Ablation on the gradual design of MPM

We conduct an ablation study to validate the gradual design of MPM. We train a variant of PRISM which only keeps the Patch Pruning module in the last MPM and removes the rest. This design

Method	Pose estimation AUC		
	@5°	@10°	@20°
only pruning at the last MPM	54.6	70.5	81.8
gradually pruning at every MPM (Full)	56.4	71.6	82.7

Table 4: Impact of gradual design on the MegaDepth dataset. The test image resolution is 832×832 .

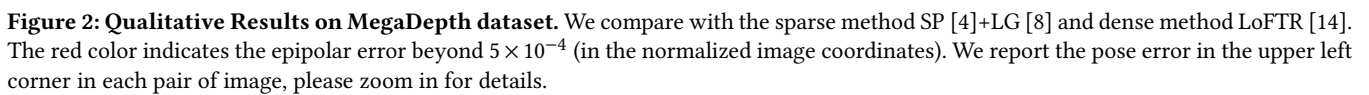
makes patch pruning degenerate into mere co-visible area prediction, only functioning during the Match Prediction phase, similar to existing works [5, 6]. This variant is trained using the same setting as the full model and tested on MegaDepth dataset at 832×832 image resolution. The Pose error AUC at thresholds of 5°, 10°, and 20° are reported. As shown in table 4, when only performing patch pruning at the last MPM, the pose estimation AUC declines noticeably. This validates the effectiveness of the gradual design of MPM.

4 MORE QUALITATIVE RESULTS

More qualitative comparisons between PRISM and LoFTR [14] and SP [4]+LG [8] are provided. Figure 2 shows the qualitative results on MegaDepth dataset and Figure 3 shows the qualitative results on ScanNet dataset. More qualitative results on the InLoc benchmark are shown in Figure 4.

REFERENCES

- [1] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David N. R. McKinnon, Yanghai Tsin, and Long Quan. 2022. ASpanFormer: Detector-Free Image Matching with Adaptive Span Transformer. In *European Conference on Computer Vision*. <https://api.semanticscholar.org/CorpusID:251929348>
- [2] Ying Chen, Dihe Huang, Shang Xu, Jianlin Liu, and Yong Liu. 2022. Guide Local Feature Matching by Overlap Estimation. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelfth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*. AAAI Press, 365–373. <https://doi.org/10.1609/AAAI.V36I1.19913>
- [3] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. 2017. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2432–2443. <https://doi.org/10.1109/CVPR.2017.261>
- [4] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2018. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 224–236.
- [5] Khang Truong Giang, Soohwan Song, and Sung-Guk Jo. 2022. TopicFM: Robust and Interpretable Feature Matching with Topic-assisted. *ArXiv abs/2207.00328* (2022). <https://api.semanticscholar.org/CorpusID:250243816>



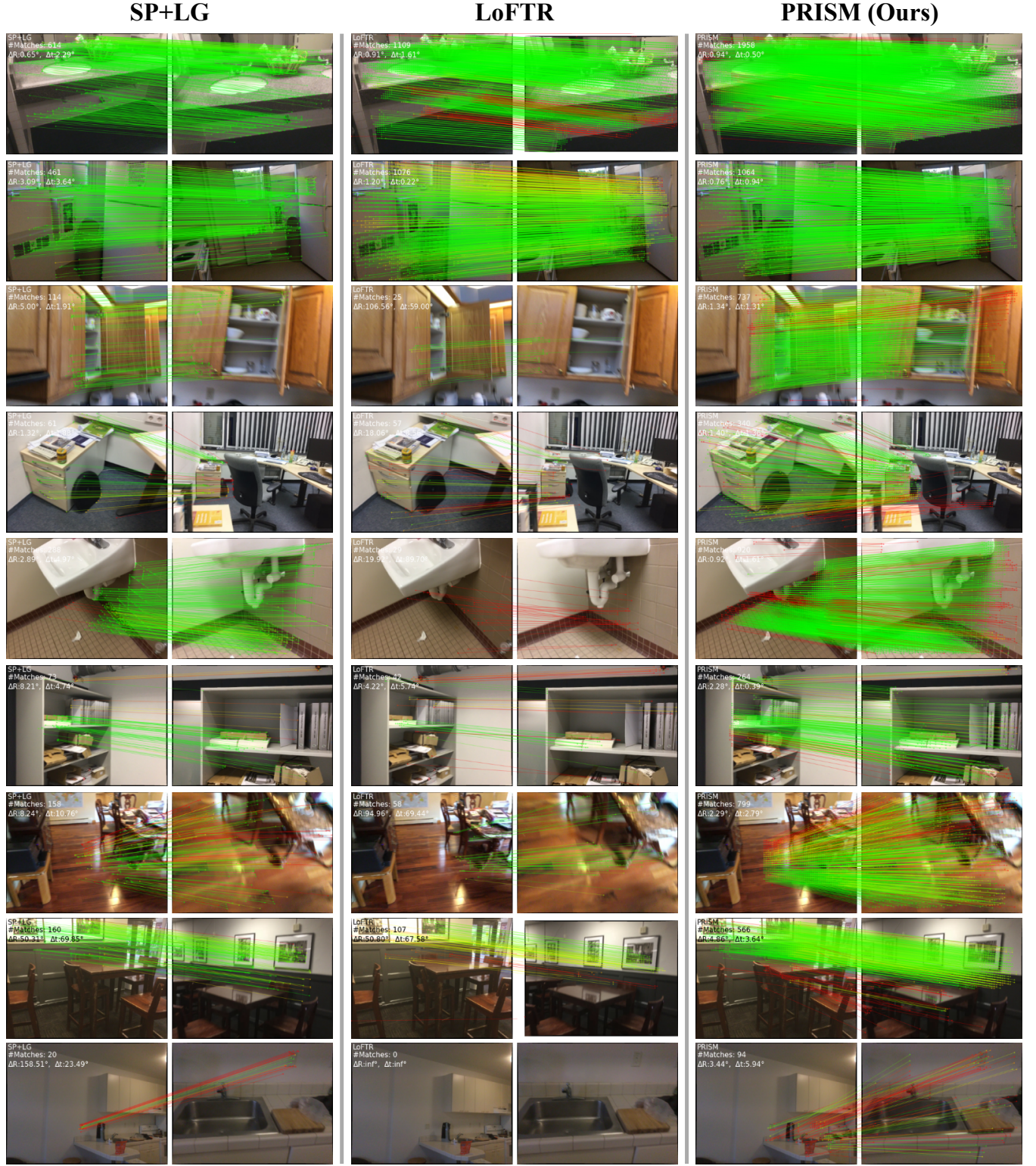


Figure 3: Qualitative Results on ScanNet dataset. We compare with the sparse method SP [4]+LG [8] and dense method LoFTR [14]. The red color indicates the epipolar error beyond 5×10^{-4} (in the normalized image coordinates). We report the pose error in the upper left corner in each pair of image, please zoom in for details.

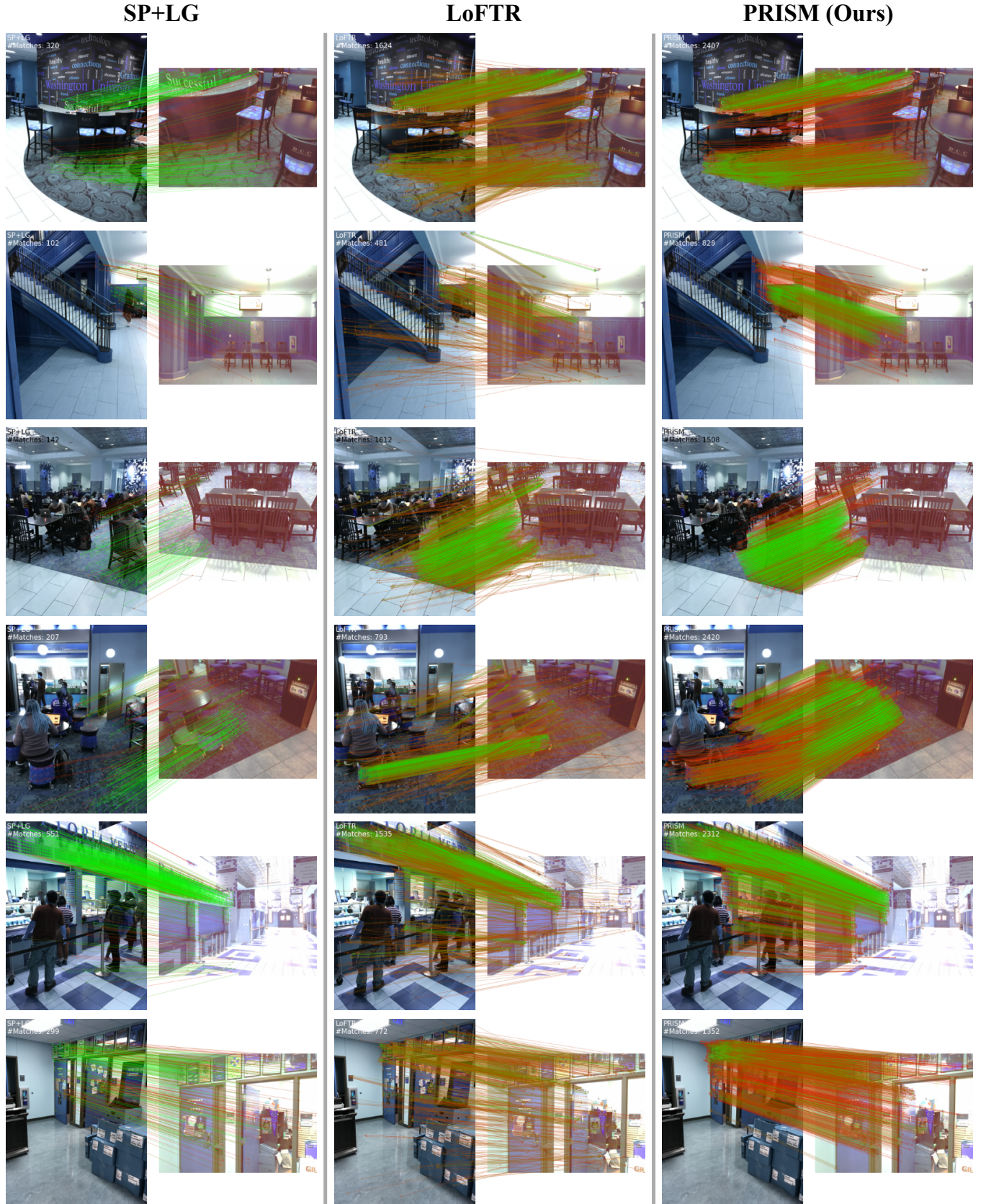


Figure 4: Qualitative Results on InLoc dataset. We compare with the sparse method SP [4]+LG [8] and dense method LoFTR [14]. We color the matches with matching scores since there is no ground-truth pose available. The green indicates higher score and red for the opposite.

- [6] Dihe Huang, Ying Chen, Shang Xu, Yong Liu, Wen-Qi Wu, Yikang Ding, Chengjie Wang, and Fan Tang. 2022. Adaptive Assignment for Geometry Aware Local Feature Matching. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), 5425–5434. <https://api.semanticscholar.org/CorpusID:250626872>
- [7] Zhengqi Li and Noah Snavely. 2018. MegaDepth: Learning Single-View Depth Prediction From Internet Photos. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2041–2050. <https://doi.org/10.1109/CVPR.2018.00218>
- [8] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. 2023. LightGlue: Local Feature Matching at Light Speed. *ArXiv abs/2306.13643* (2023). <https://api.semanticscholar.org/CorpusID:259243929>
- [9] David G Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision* 60 (2004), 91–110.
- [10] Anastasya Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. 2017. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 4826–4837. <https://proceedings.neurips.cc/paper/2017/hash/831caa1b600f852b7844499430ecac17-Abstract.html>
- [11] Jérôme Revaud, Philippe Weinzaepfel, César Roberto de Souza, Noé Pion, Gabriela Csurka, Yohann Cabon, and M. Humenberger. 2019. R2D2: Repeatable and Reliable Detector and Descriptor. *ArXiv abs/1906.06195* (2019). <https://api.semanticscholar.org/CorpusID:189927786>
- [12] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2019. SuperGlue: Learning Feature Matching With Graph Neural Networks. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), 4937–4946. <https://api.semanticscholar.org/CorpusID:208291327>
- [13] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yinfeng Liu. 2021. RoFormer: Enhanced Transformer with Rotary Position Embedding. *ArXiv abs/2104.09864* (2021). <https://api.semanticscholar.org/CorpusID:233307138>
- [14] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. 2021. LoFTR: Detector-Free Local Feature Matching with Transformers. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 8918–8927. <https://api.semanticscholar.org/CorpusID:232478646>
- [15] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomás Pajdla, and Akihiko Torii. 2021. InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 4 (2021), 1293–1307. <https://doi.org/10.1109/TPAMI.2019.2952114>
- [16] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. 2020. DISK: Learning local features with policy gradient. *Advances in Neural Information Processing Systems* 33 (2020), 14254–14265.