Handcrafted Outlier Detection Revisited –Supplement–

Luca Cavalli[1], Viktor Larsson[1], Martin Ralf Oswald[1], Torsten Sattler[2], Marc Pollefeys[1]

1: ETH Zurich, Switzerland
2: Chalmers University of Technology, Gothenburg, Sweden

1 Introduction

In this supplement we include additional experiments and implementation details that could not fit into the main paper. In particular:

- **Section 2**: we discuss the choice of LO-RANSAC [4] for the final robust pose estimation rather than OpenCV RANSAC.
- **Section 3**: we report extra metrics for the same experiments as in the main paper for better comparability with previous and future work. Moreover, we report inlier rate statistics for our experiments for all methods.
- **Section 4**: we report our experiments on the NGRANSAC [3] baseline, showing that the setup we used for our experiments achieves better pose estimation than the original paper's setup.
- **Section 5**: we report breakdowns of the runtime of our current implementation on different steps of the pipeline.
- **Section 6**: we report results on the Aachen Day-Night dataset [15,16].
- **Section 7**: we report several implementation details with the hyperparameters and keypoint preprocessing used in our experiments for better reproducibility.
- **Section 8**: we report additional qualitative results from our evaluation pipeline.

2 RANSAC implementation choice

In our evaluation pipeline we opted for an advanced RANSAC implementation, using the LO-RANSAC [4] implementation provided by COLMAP [17], rather than a standard implementation as the one found in OpenCV. The superiority in pose estimation performance of recent state-of-the-art RANSACs [1,4,13,20,11] compared to the first proposed algorithm [6] has been confirmed several times by each newly published method, by recent benchmarks [7], and tested again for our specific pipeline as shown in Table 1. This is not surprising as each new method builds on the initial idea from [6].

Table 1: **RANSAC Comparison**: we compare OpenCV RANSAC and LO-RANSAC on ratio-test filtered matches to further confirm the superiority of LORANSAC in our pipeline.

Method	TUM [18]			SUN3D [21]			YFCC100M [19]		
	AUC5	AUC10	AUC20	AUC5	AUC10	AUC20	AUC5	AUC10	AUC20
RT + LO-RANSAC [4]	**16.1**	**24.8**	**33.6**	**5.9**	**14.1**	**25.6**	**51.9**	**64.9**	**76.3**
RT + OpenCV RANSAC	9.2	16.3	23.7	2.1	5.2	10.2	19.4	31.2	43.1

3 Additional Metrics

Table 2 reports additional metrics for the experiments presented in the main paper for better comparability with previous and future work. In particular we add the two following metrics:

1. **mAPX**: mean average precision under X degrees, i.e. the rate (in percentages) of successful pose estimation considering as success a pose with maximum error lower than X degrees. This allows comparability of our results with the original OA-Net [23] paper.
2. **AUCX***: approximate Area Under the Curve below X degrees. As some previous works [3,10], including NGRANSAC [3], report AUC measures approximated as the cumulative area under a histogram with 5-degrees bins, we report the same for comparability with them.

The same observations drawn from the exact AUC are drawn from the extended metrics: our method greatly outperforms current state of the art both in outdoor and indoor scenarios.

Moreover, we report in Table 3 inlier statistics for our experiments on all methods. We collect recall, precision and F1 score with respect to ground truth inliers for all methods before the application of RANSAC. Note that all methods have been tuned for relative pose estimation performance, and inlier statistics may change substantially when methods are tuned for a different metric.

4 Neural Guided RANSAC Baseline

We observed that Neural Guided RANSAC (NGRANSAC) [3] performs better by re-fitting the essential matrix with LO-RANSAC [4] on its inlier set rather than directly using the essential matrix that is the output of the authors code. In Table 4, we report comparative results of using NGRANSAC as originally designed and with re-fitting. We test on three settings:

1. **Original**: this is the original NGRANSAC setup as suggested by the authors, reproducing the results in [3] for the SIFT+Ratio+NG-RANSAC (+SI) label on essential matrix estimation. As default in the sample code from the authors, we use a RANSAC threshold of 0.001 and run 25000 iterations. We then evaluate directly the essential matrix found by NGRANSAC.

Table 2: **Extended metrics results**: for better comparability with previous and future works we report additional metrics of our results. Starred Area under the Curve is computed as the area below an histogram with five-degrees bins.

Method	TUM [18]								
	AUC5	AUC10	AUC20	AUC5*	AUC10*	AUC20*	mAP5	mAP10	mAP20
Ours	**24.7**	**37.2**	**48.4**	**42.2**	**48.2**	**54.4**	**42.2**	**54.1**	**62.6**
OA-Net	20.9	32.2	43.3	36.3	42.3	48.6	36.3	48.3	56.6
NGR	19.4	29.6	38.7	33.4	38.9	44.1	33.4	44.4	50.6
GMS	19.6	30.5	41.3	34.4	40.2	47.0	34.4	46.0	55.1
RT (10k)	16.1	24.8	33.6	27.7	32.8	38.5	27.7	38.0	46.0
RT (100k)	17.3	26.6	36.2	29.3	34.6	41.1	29.3	39.9	49.1
	YFCC100M [19]								
Ours	**57.8**	**71.1**	**81.7**	**76.2**	**82.7**	**88.3**	**76.2**	**89.1**	**94.8**
OA-Net	53.5	66.0	76.7	70.6	77.2	82.9	70.6	83.8	89.4
NGR	53.8	66.7	77.7	71.4	78.1	84.0	71.4	84.7	90.9
GMS	52.3	65.0	76.0	69.5	76.2	82.2	69.5	83.0	89.1
RT (10k)	51.9	64.9	76.3	69.5	76.4	82.7	69.5	83.4	89.9
RT (100k)	53.2	66.3	77.5	71.1	78.0	84.0	71.1	84.9	90.9
	SUN3D [21]								
Ours	**7.6**	**18.2**	**33.2**	**18.9**	**28.1**	**40.2**	**18.9**	**37.3**	**56.2**
OA-Net	6.9	16.3	29.4	17.0	25.1	35.7	17.0	33.2	49.4
NGR	6.2	15.0	27.3	15.5	23.1	33.2	15.5	30.6	46.5
GMS	6.8	15.9	29.1	16.6	24.5	35.4	16.6	32.3	50.0
RT (10k)	5.9	14.1	25.6	14.9	21,7	31.1	14.9	28.5	43.5
RT (100k)	6.1	14.5	26.3	15.2	22.4	32.0	15.2	29.5	44.6

2. **Refitting-ST**: in this alternative we run the whole NGRANSAC pipeline exactly as in the *Original* case, with the same setup and threshold, and then use the inliers found to fit the essential matrix using the same LO-RANSAC setup used by all other methods. We found that the default threshold of 0.001, suggested originally for the method, is too strict and reduces the space for local optimization inside LO-RANSAC as the chosen inliers are already strictly agreeing on some model. Therefore, we found that larger thresholds were better for this setup.
3. **Refitting-LT**: this is the setup that we used and ran on all experiments reported in the main paper. We run the *Original* pipeline on essential matrix

Table 3: **Inlier statistics**: we report inlier statistics for all the competing methods in our experiments. Note that all methods have been tuned for relative pose estimation, so results may vary consistently with different tuning. Precision, recall and F1 score have been measured with respect to ground truth inliers *before* applying RANSAC.

Method	TUM [18]			SUN3D [21]			YFCC100M [19]		
	Recall	Precision	F	Recall	Precision	F	Recall	Precision	F
Ours	38.0	**72.6**	48.1	42.0	**61.4**	48.4	50.2	85.9	62.0
OA-Net	**44.5**	61.1	**49.7**	**47.7**	53.7	**48.7**	**60.2**	81.4	**68.1**
NGR	18.2	61.5	27.3	22.3	51.8	30.5	33.7	79.8	45.5
GMS	15.8	56.0	23.9	16.4	58.1	24.5	29.1	**88.7**	40.8
RT	20.3	32.9	24.7	24.5	31.2	26.9	34.4	56.6	42.1

Table 4: **Tuned NGRANSAC baseline setup**: we run experiments on YFCC100M to tune the setup for NGRANSAC within our evaluation pipeline. We report the metrics explained in Section 3.

Method	YFCC100M [19]								
	AUC5	AUC10	AUC20	AUC5*	AUC10*	AUC20*	mAP5	mAP10	mAP20
Original	39.4	50.8	59.5	55.3	60.6	64.8	55.3	65.9	69.7
Refitting-ST	53.2	66.2	77.2	**71.5**	77.7	83.7	**71.5**	83.8	90.5
Refitting-LT	**53.8**	**66.7**	**77.7**	71.4	**78.1**	**84.0**	71.4	**84.7**	**90.9**

Table 5: **Datasets median statistics**. We report median number of keypoints detected per image and median number of matches found by our method on each dataset to aid interpretation of runtime breakdowns in Table 6.

Dataset	Median keypoints per image	Median matches found per image couple
YFCC100M [19]	6666	1559
TUM [18]	4125	561
SUN3D [21]	2595	474

estimation with 25000 iterations and a threshold of 0.01, and then fit the final essential matrix with LO-RANSAC on the inlier set found by NG-RANSAC. A higher threshold than the original allows more space for the local optimization to find better poses on average, as seen in Table 4.

5 Runtime Breakdown

We provide a runtime breakdown of our method on different datasets. While we always try to extract 8000 SIFT keypoints from every images, for small images with low texture it is common to find less, and the number of detected matches can vary accordingly, as shown in Table 5. These factors can influence runtimes of different sections of our algorithm. We report in Table 6 the runtime statistics of the following sections of our method:

1. **Init**: the initialization procedure, which includes loading the input keypoints to GPU and finding the nearest neighbors and ratio test scores.
2. **Seed selection**: the decision on which correspondences should be used as seed correspondences.
3. **Neighborhood selection**: the selection of neighborhood sets for each seed correspondence, and referencing each set to the local coordinate frame.
4. **Local RANSACs**: the parallel RANSACs on all local neighborhoods to fit affinities on multiple thresholds. For this run we use 128 RANSAC iterations and 4 threshold hypotheses.
5. **Match selection**: the compensation of the inlier ratio signals from RANSAC and the selection of the final correspondences to output.

Table 6: **Runtimes breakdown**: we split our method into sections to analyze the impact of each on runtime in different scenarios. Times are all measured in milliseconds per image pair. Bounds are obtained with 90% confidence.

Section	YFCC100M [19]				
	Mean	Median	Std. Dev.	Lower Bound	Upper Bound
Init	10.1	9.5	3.4	8.3	13.5
Seed selection	3.9	3.6	1.5	3.2	6.1
Neighborhood selection	1.9	1.7	0.8	1.5	2.0
Local RANSACs	25.2	22.1	11.47	16.45	36.6
Match selection	1.2	0.9	1.6	0.8	1.5
	TUM [18]				
	Mean	Median	Std. Dev.	Lower Bound	Upper Bound
Init	6.0	5.6	5.7	3.6	8.1
Seed selection	2.1	1.8	1.2	1.1	3.1
Neighborhood selection	1.8	1.6	0.9	1.5	2.1
Local RANSACs	23.8	22.1	9.0	15.7	35.9
Match selection	1.2	0.9	0.8	0.8	1.5
	SUN3D [21]				
	Mean	Median	Std. Dev.	Lower Bound	Upper Bound
Init	3.9	3.3	2.9	2.5	6.2
Seed selection	1.1	0.9	1.0	0.7	1.4
Neighborhood selection	1.8	1.5	1.2	1.4	2.0
Local RANSACs	22.0	19.1	8.7	14.6	32.5
Match selection	1.2	0.9	1.0	0.8	1.5

6 Aachen Day-Night Challenge

The Aachen Day-Night dataset [15,16] allows to measure pose accuracy achieved when trying to localize nighttime query images against a 3D model built from daytime images. We follow the setup of the Local Feature Challenge from the CVPR 2019 workshop on "Long-Term Visual Localization under Changing Conditions" and use the code provided by the organizers[1], but use our matching method rather than the default mutual nearest neighbor matching. Following [15], we report the percentage of nighttime queries localized within thresholds on their rotation and position errors with respect to the ground truth poses (three sets of thresholds are used: (0.5m, 2°) / (1m, 5°) / (5m, 10°)).

While currently the majority of top-scoring methods are using specialized learned local features, we use our method for outlier rejection on upright RootSIFT [8] features. In Table 7 we report scores for the available baselines and with our method. We also run additional baselines using upright RootSIFT with simple filters such as mutual nearest neighbor and ratio-test for better comparability. We observe that our outlier rejection greatly improves localization performance on SIFT keypoints and descriptors, elevating its performance to comparable levels as the current state-of-the-art learned local features.

[1] `https://github.com/tsattler/visuallocalizationbenchmark/`

Table 7: **Aachen Day-Night:** percentage of nighttime queries localized within given accuracy measures of the ground truth poses

Method	(0.5m, 2°)	(1m, 5°)	(5m, 10°)
UprightRootSIFT (public baseline)	33.7	52.0	65.3
UprightRootSIFT + Mutual Nearest Neighbor	37.8	56.1	76.5
UprightRootSIFT + Ratio-Test (0.8)	41.8	57.1	75.5
UprightRootSIFT + Ours	**45.9**	**64.3**	**86.7**
UR2KID Scape Technologies[22]	46.9	67.3	88.8
D2-Net - single-scale [5]	45.9	68.4	88.8
R2D2 V2 20K [14]	46.9	66.3	88.8
Dense-ContextDesc10k_upright_OANet [9,23]	48.0	63.3	88.8
densecontextdesc10k_upright_mixedmatcher [9]	46.9	65.3	87.8

7 Implementation details and hyperparameters

As modern learning approaches run efficiently on highly parallel hardware, we also designed our algorithm to be extremely parallel to run efficiently on modern GPUs, and accordingly we provide a full implementation in PyTorch [12]. This allows a great speedup compared to CPU execution, although it still leaves a wide space for further low-level optimization.

In particular seed point selection is implemented as a local non-maximum suppression, where each correspondence is evaluated independently of the evaluation of the others. For affine fitting, random sampling methods such as RANSAC suit perfectly the needs as all different seed points as well as all iterations can be processed in parallel, including the sampling, minimal fitting, evaluation of the residuals and identification of the inliers.

Finally, the evaluation of the best threshold for each seed point and the final selection of the inliers is again fully independent for each seed point. Having no efficient closed form solution for the inlier counts compensation $\mathbb{E}_o\left[C^*_{i,t_k}\right]$ to be estimated from the distribution in Eq. (**??**), we observe that at runtime all of its parameters are fixed except for variables $\|\mathcal{N}_i\|$ and t_k. Therefore, we estimate this expectation offline by extensively sampling such distribution, and provide the method with the table of expectations for different entries of $\|\mathcal{N}_i\|$ and t_k to be used with virtually no runtime overhead.

All our experiments have been run using the same hyperparameter setup. We set the radius R for seed point selection to match a fixed ratio r_a between the area of the non-maximum suppression circle $R^2\pi$ and the area of the image wh. In particular, we set $R = \sqrt{\frac{wh}{\pi r_a}}$ with $r_a = 70$. For the purpose of collecting neighborhoods $\mathcal{N}_i$ for each seed point i, we use a radius λ times larger than R, with $\lambda = 6$. This ensures sufficient but controlled overlap between neighboring regions to be robust to errors in seed correspondences. We observed that the performance of our method saturates very early when increasing the number of RANSAC iterations, which are fixed to 128 iterations. When experimenting with SIFT keypoints, we set $t_\sigma = 1.5$ and $t_\alpha = 30°$. Code is available at https://github.com/cavalli1234/AdaLAM with our experimental hyperparameter setting by default.

Moreover, only for our method we preprocess the input set to drop duplicates of keypoints with exactly overlapping locations, even with different descriptors. These are usually produced by SIFT when the dominant orientation is ambiguous in a DoG peak. We observed that this preprocessing step generally degrades performance, as it solves the orientation ambiguity by randomly dropping one alternative. However, our method appears to benefit from such preprocessing which makes local inlier distributions to behave more closely to our modeling. Improvements to fully exploit the input set without the need for this preprocessing step are left for future work.

8 Qualitative results

We report in Figures 1, 2 and 3 additional qualitative comparative results from our evaluation pipeline. For better visualization of our method's results, we disabled the hard minimum of 20 output correspondences that are normally added from the best according to ratio-test score when our criterion cannot validate enough matches.

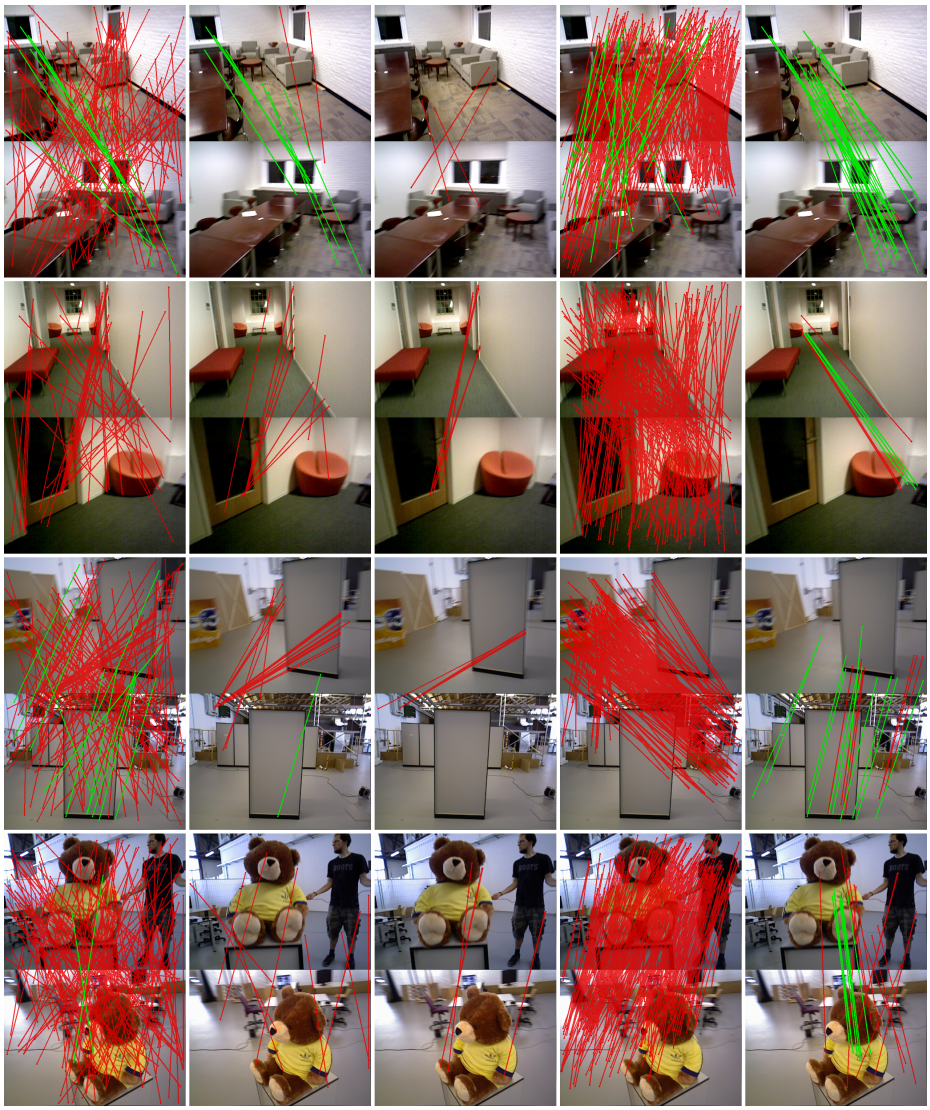

Fig. 1: **Success cases** from our experiments. Matches agreeing with ground truth epipolar geometry are shown in green, others are in red. Examples include cases with low texture, non-planar object, minimal image overlap.

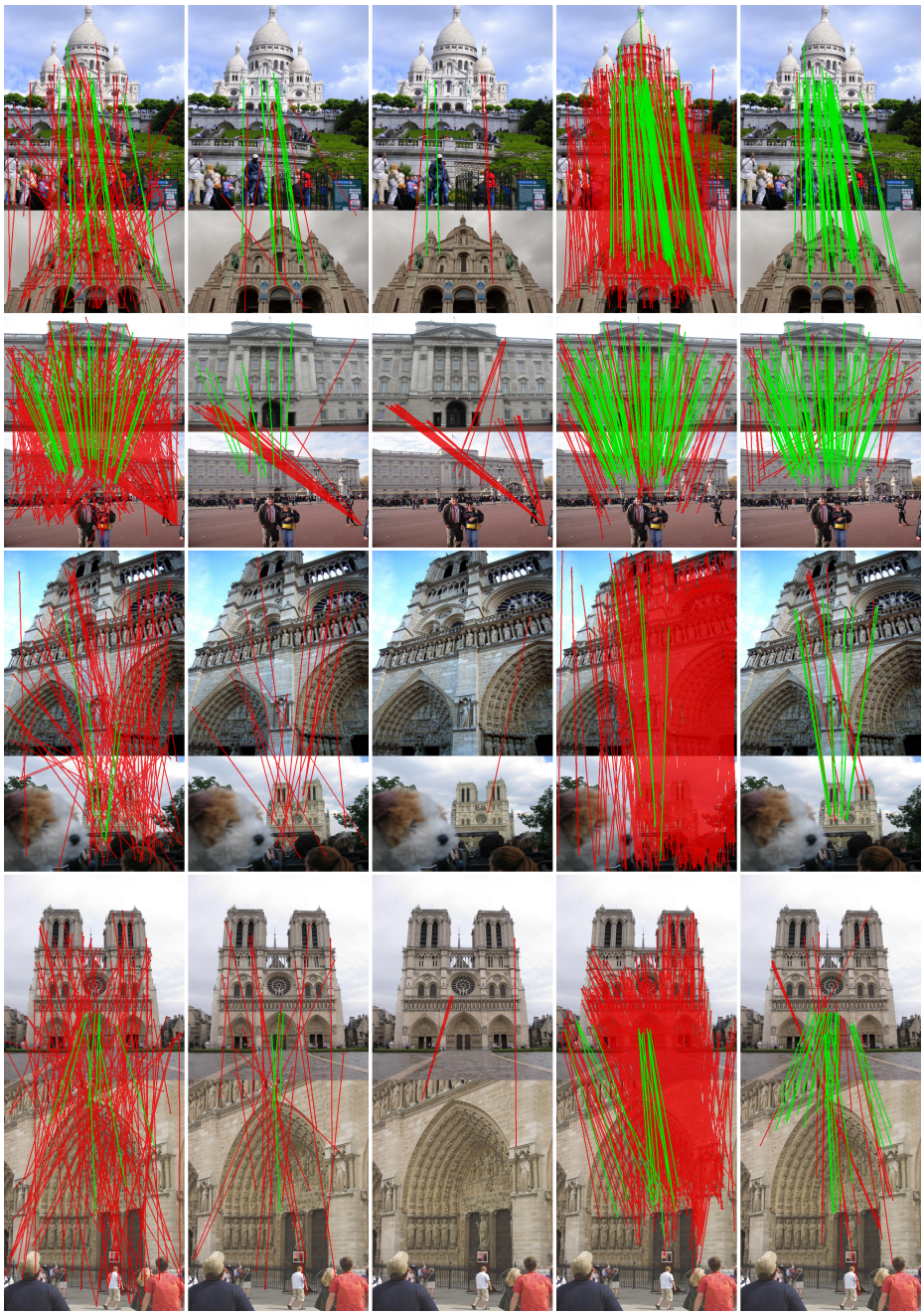

Fig. 2: **Success cases** from our experiments. Matches agreeing with ground truth epipolar geometry are shown in green, others are in red. Examples include cases with repeated structures, strong scale changes and perspective deformations.

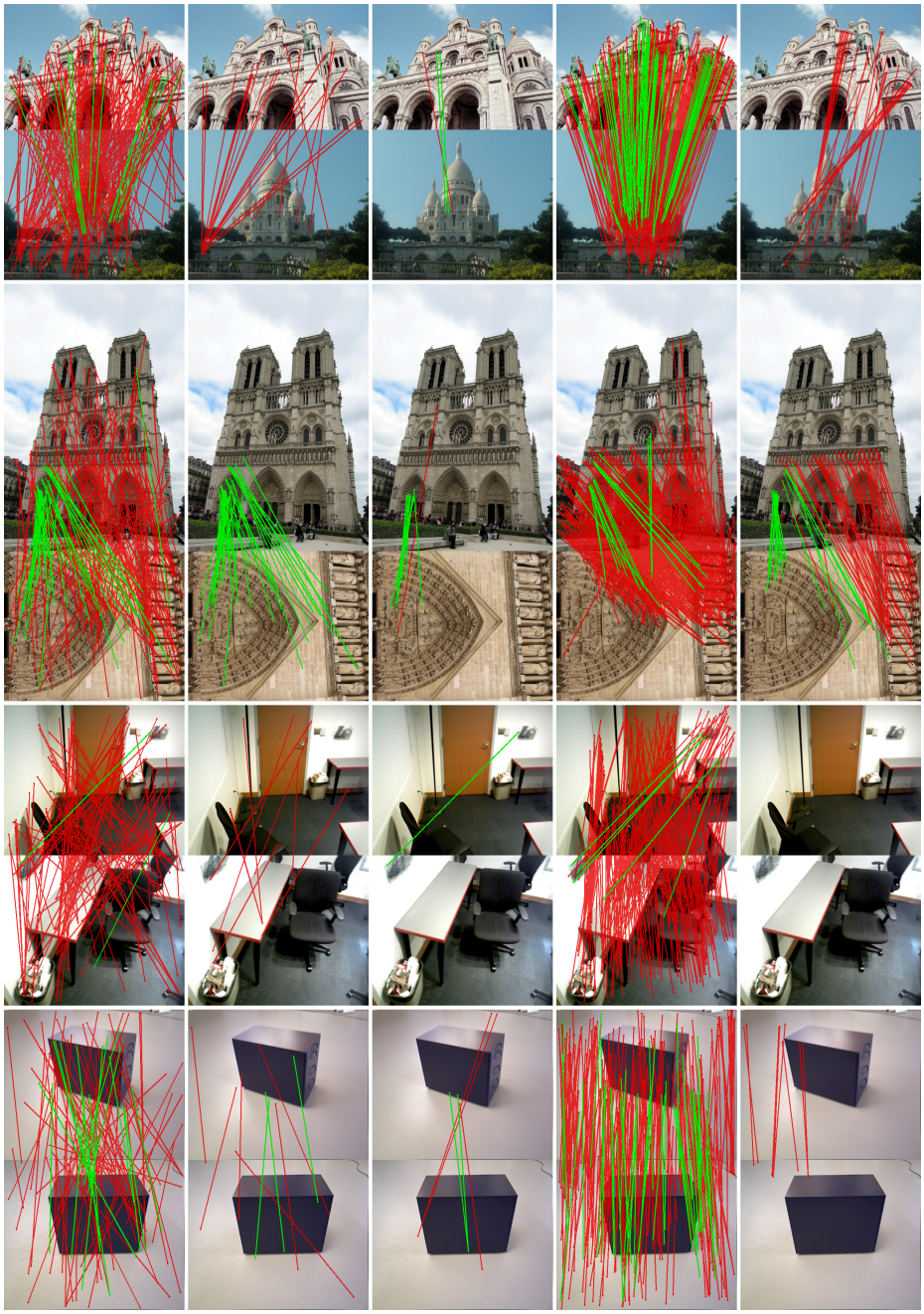

Fig. 3: **Failure cases** from our experiments. Matches agreeing with ground truth epipolar geometry are shown in green, others are in red. Examples include strong scale changes with repeated structures, limited image overlap, and textureless surfaces.

References

1. Barath, D., Matas, J., Noskova, J.: Magsac: marginalizing sample consensus. In: Computer Vision and Pattern Recognition (CVPR) (2019)
2. Bian, J., Lin, W.Y., Matsushita, Y., Yeung, S.K., Nguyen, T.D., Cheng, M.M.: Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence. In: Computer Vision and Pattern Recognition (CVPR) (2017)
3. Brachmann, E., Rother, C.: Neural-guided ransac: Learning where to sample model hypotheses. In: International Conference on Computer Vision (ICCV) (2019)
4. Chum, O., Matas, J., Kittler, J.: Locally optimized ransac. In: Joint Pattern Recognition Symposium. pp. 236–243. Springer (2003)
5. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-net: A trainable cnn for joint description and detection of local features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8092–8101 (2019)
6. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6), 381–395 (1981)
7. Jin, Y., Mishkin, D., Mishchuk, A., Matas, J., Fua, P., Yi, K.M., Trulls, E.: Image matching across wide baselines: From paper to practice. arXiv preprint arXiv:2003.01587 (2020)
8. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV) **60**(2), 91–110 (2004)
9. Luo, Z., Shen, T., Zhou, L., Zhang, J., Yao, Y., Li, S., Fang, T., Quan, L.: Contextdesc: Local descriptor augmentation with cross-modality context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2527–2536 (2019)
10. Moo Yi, K., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., Fua, P.: Learning to find good correspondences. In: Computer Vision and Pattern Recognition (CVPR) (2018)
11. Ni, K., Jin, H., Dellaert, F.: Groupsac: Efficient consensus in the presence of groupings. In: International Conference on Computer Vision (ICCV) (2009)
12. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. In: NIPS 2017 Workshop on Autodiff (2017), `https://openreview.net/forum?id=BJJsrmfCZ`
13. Raguram, R., Chum, O., Pollefeys, M., Matas, J., Frahm, J.M.: Usac: a universal framework for random sample consensus. Trans. Pattern Analysis and Machine Intelligence (PAMI) **35**(8), 2022–2038 (2012)
14. Revaud, J., De Souza, C., Humenberger, M., Weinzaepfel, P.: R2d2: Reliable and repeatable detector and descriptor. In: Neural Information Processing Systems (NIPS). pp. 12405–12415 (2019)
15. Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., Safari, D., Okutomi, M., Pollefeys, M., Sivic, J., et al.: Benchmarking 6dof outdoor visual localization in changing conditions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8601–8610 (2018)
16. Sattler, T., Weyand, T., Leibe, B., Kobbelt, L.: Image retrieval for image-based localization revisited. In: BMVC. vol. 1, p. 4 (2012)
17. Schönberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Computer Vision and Pattern Recognition (CVPR) (2016)

18. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of rgb-d slam systems. In: International Conference on Intelligent Robots and Systems (IROS) (2012)
19. Thomee, B., Shamma, D.A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Borth, D., Li, L.J.: Yfcc100m: The new data in multimedia research. Communications of the ACM **59**(2), 64–73 (2016)
20. Torr, P.H., Nasuto, S.J., Bishop, J.M.: Napsac: High noise, high dimensional robust estimation-it's in the bag. British Machine Vision Conference (BMVC) (2002)
21. Xiao, J., Owens, A., Torralba, A.: Sun3d: A database of big spaces reconstructed using sfm and object labels. In: International Conference on Computer Vision (ICCV) (2013)
22. Yang, T.Y., Nguyen, D.K., Heijnen, H., Balntas, V.: Ur2kid: Unifying retrieval, keypoint detection, and keypoint description without local correspondence supervision. arXiv preprint arXiv:2001.07252 (2020)
23. Zhang, J., Sun, D., Luo, Z., Yao, A., Zhou, L., Shen, T., Chen, Y., Quan, L., Liao, H.: Learning two-view correspondences and geometry using order-aware network. International Conference on Computer Vision (ICCV) (2019)