High-dimensional Convolutional Networks for Geometric Pattern Recognition Supplementary Materials

1. Line and Plane Detection

In this section, we provide detailed experimental setup of the linear pattern recognition experiments. For the line detection, we first sample random noise uniformly from a hypercube with side length $L = 1.5$. We also translate the hypercube randomly in the interval -2 to 2. We sample $d_n L^D$ points for noise where d_n is the noise density per unit volume and D is the dimension of the space. To generate points on a line, we first sample parameters of the line equation $t\mathbf{v} + \mathbf{c}$. The slope $\mathbf{v} \in \mathbb{R}^D$ is sampled uniformly from a unit hypercube centered at the origin and the bias (intercept) $\mathbf{c} \in \mathbb{R}^D$ is also sampled from the unit hypercube, but we perturb $\mathbf{c}$ differently during training and testing: during training, we add $\frac{1}{\sqrt{D}}$ to the first element of $\mathbf{c}$; during testing, we add $\frac{1}{\sqrt{D}}$ to the second and third elements of $\mathbf{c}$. It slightly changes the distribution of lines sampled during training and testing to simulate unseen data during testing. Next, we sample $d_l L\sqrt{D}$ points, proportional to the diagonal length of the D-dimensional hypercube, with Gaussian noise with a standard deviation of $0.001L$ where d_l is the line density.

As we increase the dimension of the problem, noise or outliers quickly dominate the dataset as the number of points from noise increases exponentially. To mitigate, we manually decrease the noise and line densities. At 16 dimension, we decrease the noise density from 100 to 10 and the line density from 100 to 50; at 24 dimension, we drop the noise density to 1 and the line density to 10; at 32 dimension, we further drop the noise density to 0.25 and the line density to 1.

We use batch size 2 for all baselines and ours to be able to load the data on a GPU, as the number of data points increases exponentially with the dimension. We use stochastic gradient descent with initial learning rate 1e-2 and decay the learning rate by the factor of 0.99 after every epoch. We train all networks for 40 epochs. For Qi *et al*. [1], we use the balanced cross entropy [3] to slightly boost its performance. We found that the balanced cross entropy was not effective for the other networks. We thus use the standard cross entropy to train all other networks.

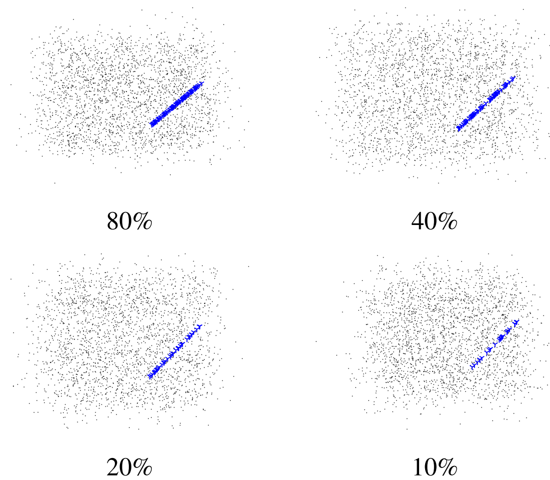


Figure 1: Examples of line detection in 8-dimensional data with different sample densities. Data projected to a 2-dimensional plane for visualization. Black dots are noise and blue marks are samples from a 8-dimensional line. Blue marks are 20 times larger than black dots for visualization.

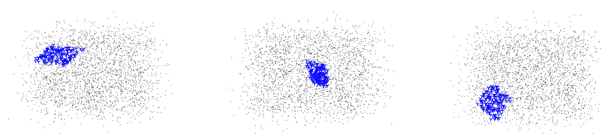


Figure 2: Examples of 8-dimensional rank-2 subspace dataset projected to a 2-dimensional plane for visualization. Black dots are noise and blue marks are samples from the rank-2 subspace.

1.1. Analysis of Line Density

We analyze the performance of various methods on the line detection dataset. We conduct this experiment in 8-dimensional space and vary the sampling density of d_l. As we decrease the density from 100 to 10, the line becomes almost indistinguishable from noise (Fig. 1). We validate our approach in such challenging conditions and present the results in Tab. 1. Note that for all densities, our networks outperform all baseline methods by a large margin even for challenging cases where the density is extremely small.

Table 1: Line detection in 8-dimensional space for various line densities.

Data Statistics		Methods								
Line Density	SNR	Zaheer *et al.* [4] + BN + IN			Yi *et al.* [3]			Ours		
		MSE	F1	AP	MSE	F1	AP	MSE	F1	AP
100%	4.142%	0.192	0.487	0.692	0.132	0.530	0.845	2.07E-3	0.934	0.993
80%	3.940%	0.232	0.309	0.574	0.247	0.494	0.777	5.53E-3	0.883	0.991
40%	3.098%	0.275	0.207	0.216	0.248	0.329	0.653	0.026	0.795	0.979
20%	2.119%	0.314	0.155	0.295	0.332	0.151	0.200	0.168	0.384	0.832
10%	1.290%	0.517	0.095	0.132	0.509	0.096	0.112	0.331	0.132	0.523

Table 2: Pairwise registration with FPFH [2] on 3DMatch test scenes with 5cm downsampling. Translation Error (TE), Rotation Error (RE), Recall in percent. The pairwise registration is successful if TE $<$ 30cm and RE $<$ 15°. The time excludes the feature extraction.

		FPFH + FGR			FPFH + Yi *et al.* [3] + FGR			FPFH + Yi *et al.* [3] + RANSAC			FPFH + Ours + FGR			FPFH + Ours + RANSAC		
	SNR	TE	RE	Succ. Rate	TE	RE	Succ. Rate	TE	RE	Succ. Rate	TE	RE	Succ. Rate	TE	RE	Succ. Rate
Kitchen	4.90%	9.32	3.92	44.86	8.06	3.36	55.53	9.10	3.65	57.71	6.07	2.46	68.97	7.25	2.68	74.51
Home 1	7.50%	9.13	3.53	51.92	8.76	3.23	64.10	9.28	2.99	67.31	7.93	2.59	74.36	8.88	2.82	78.21
Home 2	6.65%	9.02	3.58	36.54	7.96	3.13	45.19	10.02	3.71	53.85	7.99	3.23	57.69	9.53	3.58	65.87
Hotel 1	5.22%	10.20	3.86	46.02	9.14	3.46	57.52	11.25	3.80	61.95	8.71	2.90	76.11	9.15	3.06	86.28
Hotel 2	4.75%	10.69	4.82	35.58	9.74	3.82	50.00	11.06	4.52	56.73	8.18	2.82	70.19	9.29	3.24	76.92
Hotel 3	5.20%	13.10	4.69	46.30	10.36	3.86	57.41	10.59	4.05	68.52	6.57	2.63	74.07	6.83	2.72	81.48
Study	3.83%	14.20	4.74	27.40	12.95	4.01	37.67	12.88	4.09	48.63	11.23	3.12	60.62	12.30	3.33	66.44
Lab	4.98%	9.33	3.60	46.75	7.51	3.26	49.35	8.85	2.94	50.65	6.45	2.04	50.65	9.63	2.84	63.64
Average		10.62	4.09	41.92	9.31	3.52	52.10	10.38	3.72	58.17	7.89	2.72	66.58	9.11	3.03	74.17

2. 3D Match Dataset Comparison

In this section, we present a detailed experimental setup and 5cm voxel-downsampling results of hyper surface detection experiments on the 3DMatch dataset. First, we train Yi *et al.* [3] on FPFH correspondences from 5cm voxel-downsampled point clouds, following the standard procedure [6]. We use the same hyperparameters, same data augmentation except that we use balanced cross entropy loss for Yi *et al.* [3] to boost its performance. For our convnets, we use the conventional cross entropy loss. We report the results in Tab. 2. Also, we present more visualizations of correspondences before and after processing the putative correspondences with our network in Fig. 3.

3. 2D Correspondences

In this section, we present additional qualitative results on the YFCC100M test set in Fig. 4 and Fig. 5. Some failure cases are shown in Fig. 6.

References

[1] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, 2017. 1

[2] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (FPFH) for 3d registration. In *ICRA*, 2009. 2

[3] Kwang Moo Yi, Eduard Trulls, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to find good correspondences. In *CVPR*, 2018. 1, 2, 4, 5

[4] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan R Salakhutdinov, and Alexander J Smola. Deep sets. In *Advances in Neural Information Processing Systems*. 2017. 2

[5] Jiahui Zhang, Dawei Sun, Zixin Luo, Anbang Yao, Lei Zhou, Tianwei Shen, Yurong Chen, Long Quan, and Hongen Liao. Learning two-view correspondences and geometry using order-aware network. In *ICCV*, 2019. 4, 5, 6

[6] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *ECCV*, 2016. 2

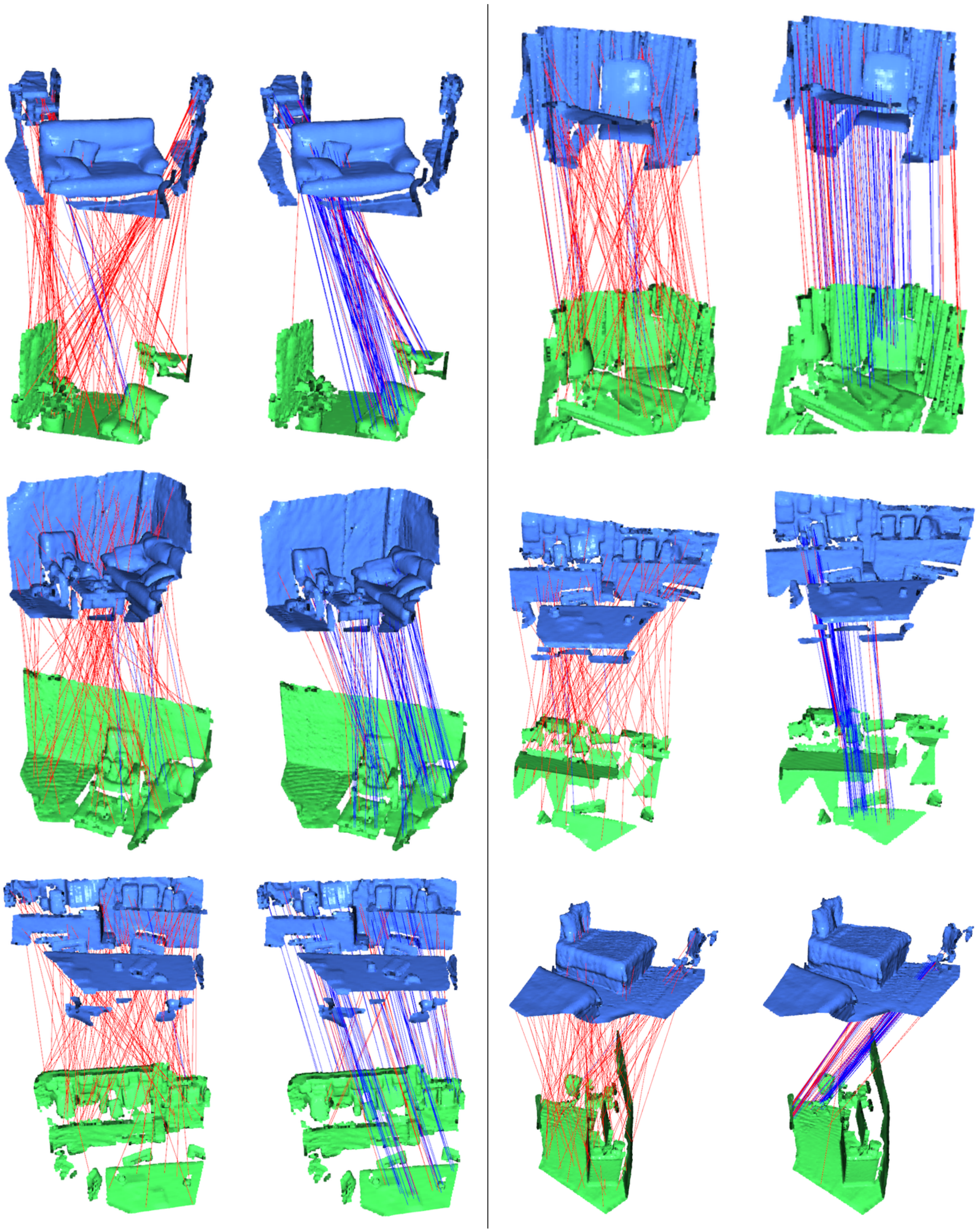


Figure 3: Visualization of color-coded correspondences before and after the hyper-surface recognition. For each pair, we visualize 100 random correspondences on the left and another 100 random correspondences after the hyper plane detection with probability > 0.5. Red lines are outlier correspondences and blue lines are inlier correpondences. The average inlier ratio is 1.76%.

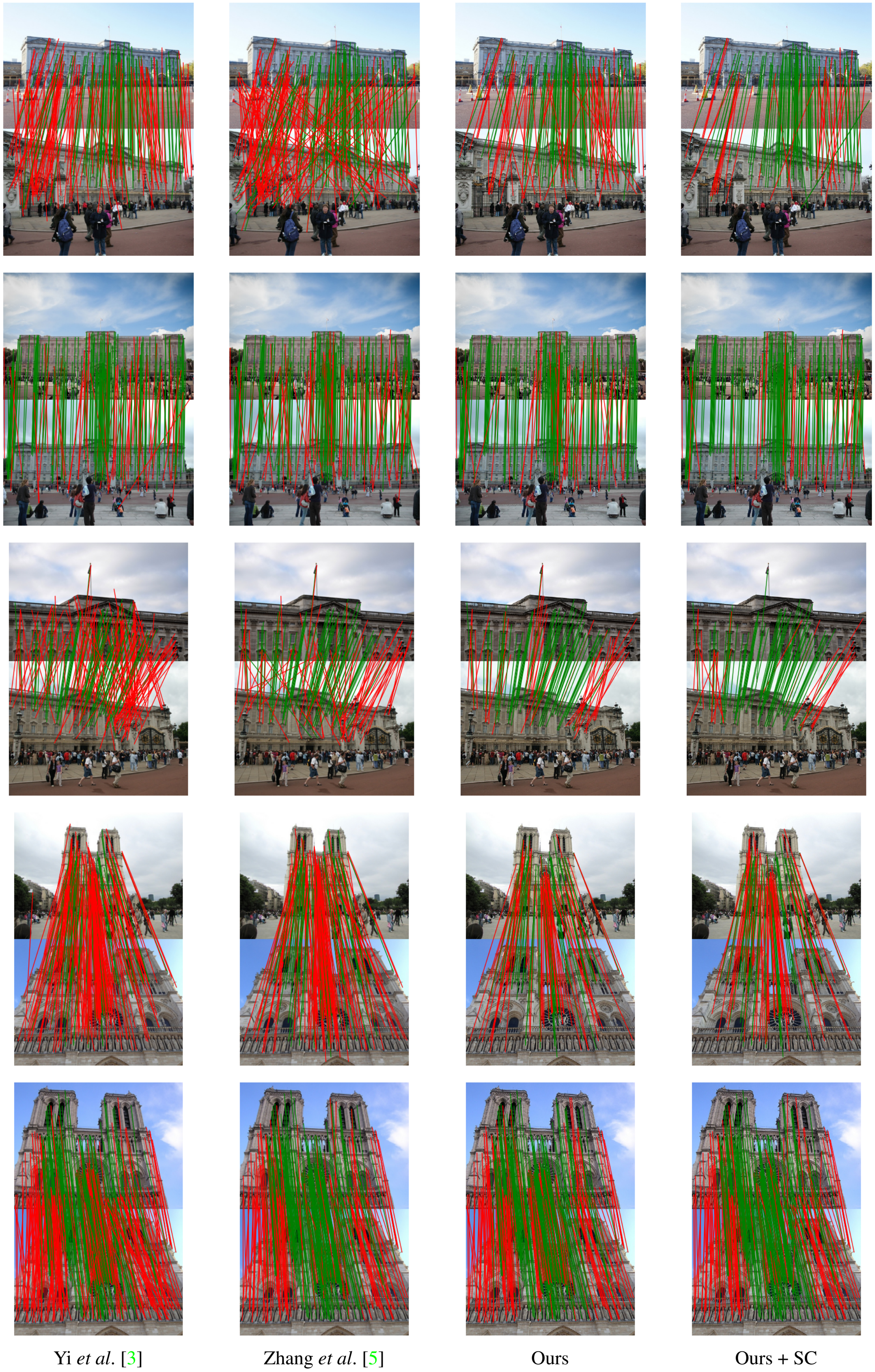
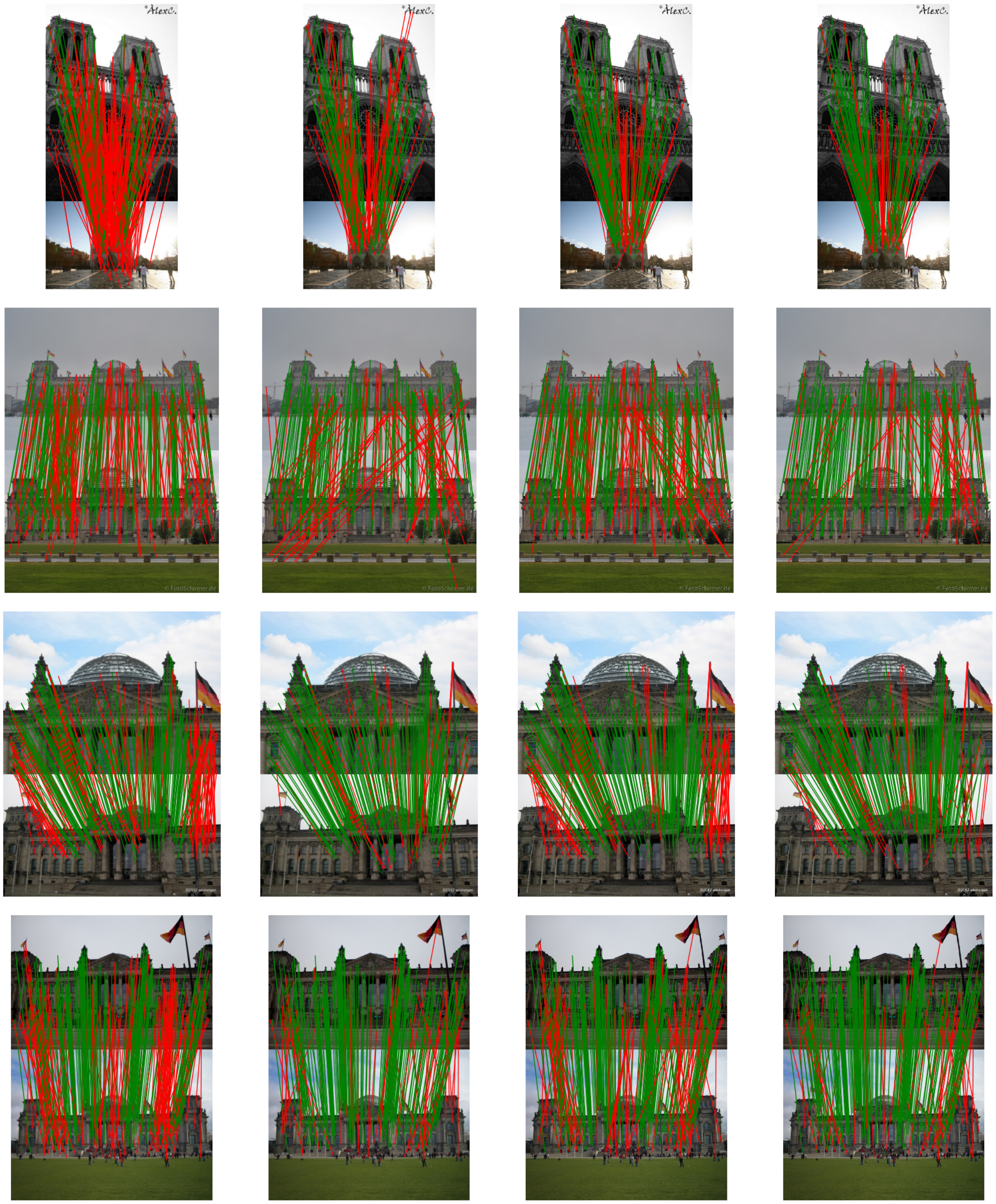


Figure 4: Visualization of image correspondenses on YFCC100M test dataset. Correspondences are colored as green if their symmetric epipolar distance is lower than 10^{-4}.



Yi *et al*. [3] Zhang *et al*. [5] Ours Ours + SC

Figure 5: Visualization of image correspondenses on YFCC100M test dataset. Correspondences are colored as green if their symmetric epipolar distance is lower than 10^{-4}.

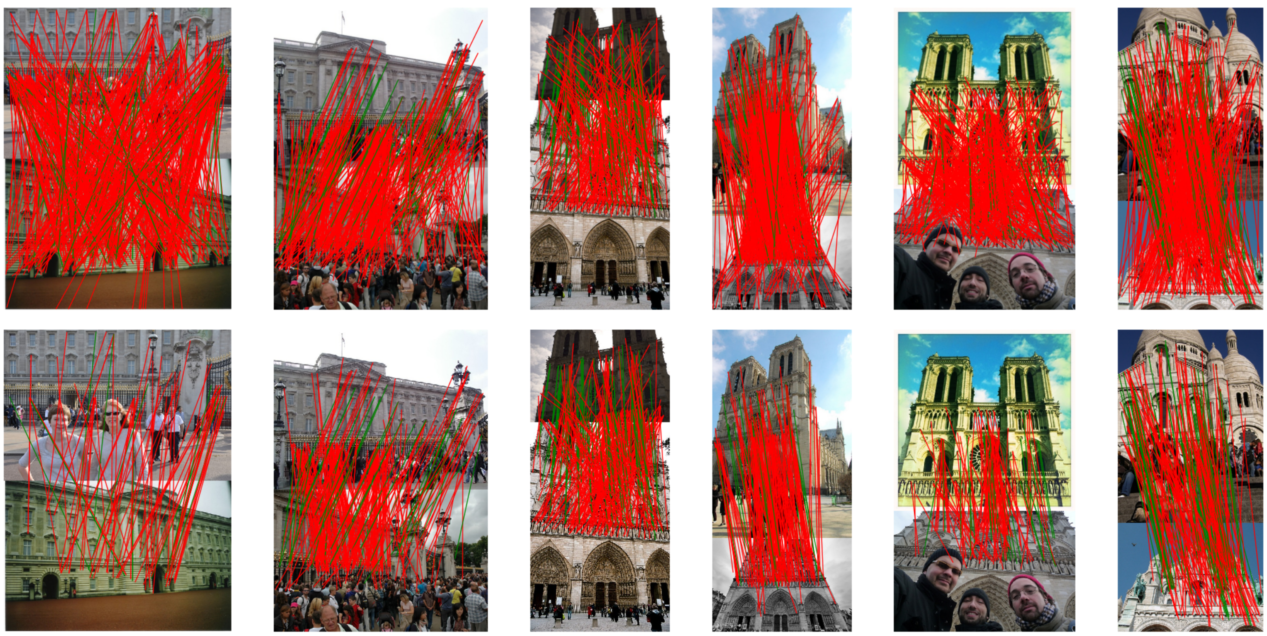


Figure 6: Visualization of correspondence filtering failure cases with Zhang *et al*. [5] (top) and Ours + SC (bottom)