Supplementary Material
2D3D-MATR: 2D-3D Matching Transformer for Detection-free Registration between Images and Point Clouds

A. Implementation Details

We mainly compare to three baseline methods in the experiments: (1) FCGF-2D3D, a 2D-3D implementation of FCGF [4]; (2) P2-Net [17], a 2D-3D version of D2-Net [5] and D3Feat [1]; (3) Predator-2D3D, a 2D-3D version of Predator [8]. For FCGF-2D3D, we supervise the descriptors using circle loss [14] instead of the hardest-in-batch contrastive loss used in [4]. This model could be regarded as a simplified P2-Net without the detection branch. For P2-Net, as there is no official code released for P2-Net, we reimplement it from the scratch. We use the detection loss defined in [1] to supervise the detection scores because we find the model fails to converge on our benchmarks using the original detection loss in [17]. For Predator-2D3D, we find that it cannot predict reliable saliency scores in 2D-3D matching, so we only predict the overlapping scores and use them as probabilities to sample random keypoints. And we use transformer [16] instead of the graph network in [8] as we find transformer achieves better results. For the baseline methods, we sample 10000 2D keypoints and 1000 3D keypoints and extract correspondences between them using mutual nearest selection.

For fair comparison, we apply the same backbone networks in all the methods, *i.e.*, a 4-stage ResNet [7] with FPN [11] backbone for images and a 4-stage KPFCNN [15] backbone for point clouds. For the 2D backbone, the output channels of each stage are $\{128, 128, 256, 512\}$. For the 3D backbone, the output channels of each stage are $\{128, 256, 512, 1024\}$. The resolution of the input images is 480×640 and the resolution in the coarest level is 60×80. Following [13], we convert RGB images to *grayscale* before feeding them to the network. The point clouds are voxelized with an initial voxel size of 2.5cm and downsampled in each stage using grid subsampling as in [15]. The detailed architecture of our method is illustrated in Fig. 1. And we use the same training settings in all the methods. We use Adam [9] optimizer to train the networks. The networks are trained for 20 epochs and the batch size is 1. The initial learning rate is 10^{-4}, which is decayed by 0.05 every epoch. For all methods (including ours), 256 correspondences are randomly sampled to supervise the pixel (point) descriptors. To estimate the transformation, we use PnP-RANSAC implemented in OpenCV [3] with 5000 iterations and the distance tolerance of 8.0.

B. Metrics

Following [17], we mainly evaluate our method using 3 metrics: Inlier Ratio, Feature Matching Recall and Registration Recall.

Inlier Ratio (IR) measures the fraction of inliers among all putative pixel-point correspondences. Following [17], a correspondence is an inlier if their *3D distance* is below $\tau_1 = 5$cm under the ground-truth transformation $\mathbf{T}^*_{\mathbf{P}\to\mathbf{I}}$

$$\text{IR} = \frac{1}{|\mathcal{C}|} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{C}} [\![\| \mathbf{T}^*_{\mathbf{P}\to\mathbf{I}}(\mathbf{x}_i) - \mathcal{K}^{-1}(\mathbf{y}_i) \|_2 < \tau_1]\!], \quad (1)$$

where $[\![\cdot]\!]$ is the Iversion bracket, $\mathbf{x}_i \in \mathbf{P}$, $\mathbf{y}_i \in \mathbf{Q}$ ($\mathbf{Q}$ is the pixel coordinate matrix of $\mathbf{I}$), and $\mathcal{K}^{-1}$ is the function that unprojects a pixel to a 3D point according to its depth value.

Feature Matching Recall (FMR) is the fraction of image-point-cloud pairs whose IR is above $\tau_2 = 0.1$. FMR measures the potential success during the registration:

$$\text{FMR} = \frac{1}{M} \sum_{i=1}^{M} [\![\text{IR}_i > \tau_2]\!], \quad (2)$$

where M is the number of all image-point-cloud pairs.

Registration Recall (RR) is the fraction of correctly registered testing pairs. A pair of image and point cloud is regarded as correctly registered if the root mean square error (RMSE) between the point clouds transformed by the ground-truth and the predicted transformation $\mathbf{T}_{\mathbf{P}\to\mathbf{I}}$ is below $\tau_3 = 0.1$m:

$$\text{RMSE} = \sqrt{\frac{1}{|\mathbf{P}|} \sum_{\mathbf{p}_i \in \mathbf{P}} \| \mathbf{T}_{\mathbf{P}\to\mathbf{I}}(\mathbf{p}_i) - \mathbf{T}^*_{\mathbf{P}\to\mathbf{I}}(\mathbf{p}_i) \|_2^2}, \quad (3)$$

$$\text{RR} = \frac{1}{M} \sum_{i=1}^{M} [\![\text{RMSE}_i < \tau_3]\!]. \quad (4)$$

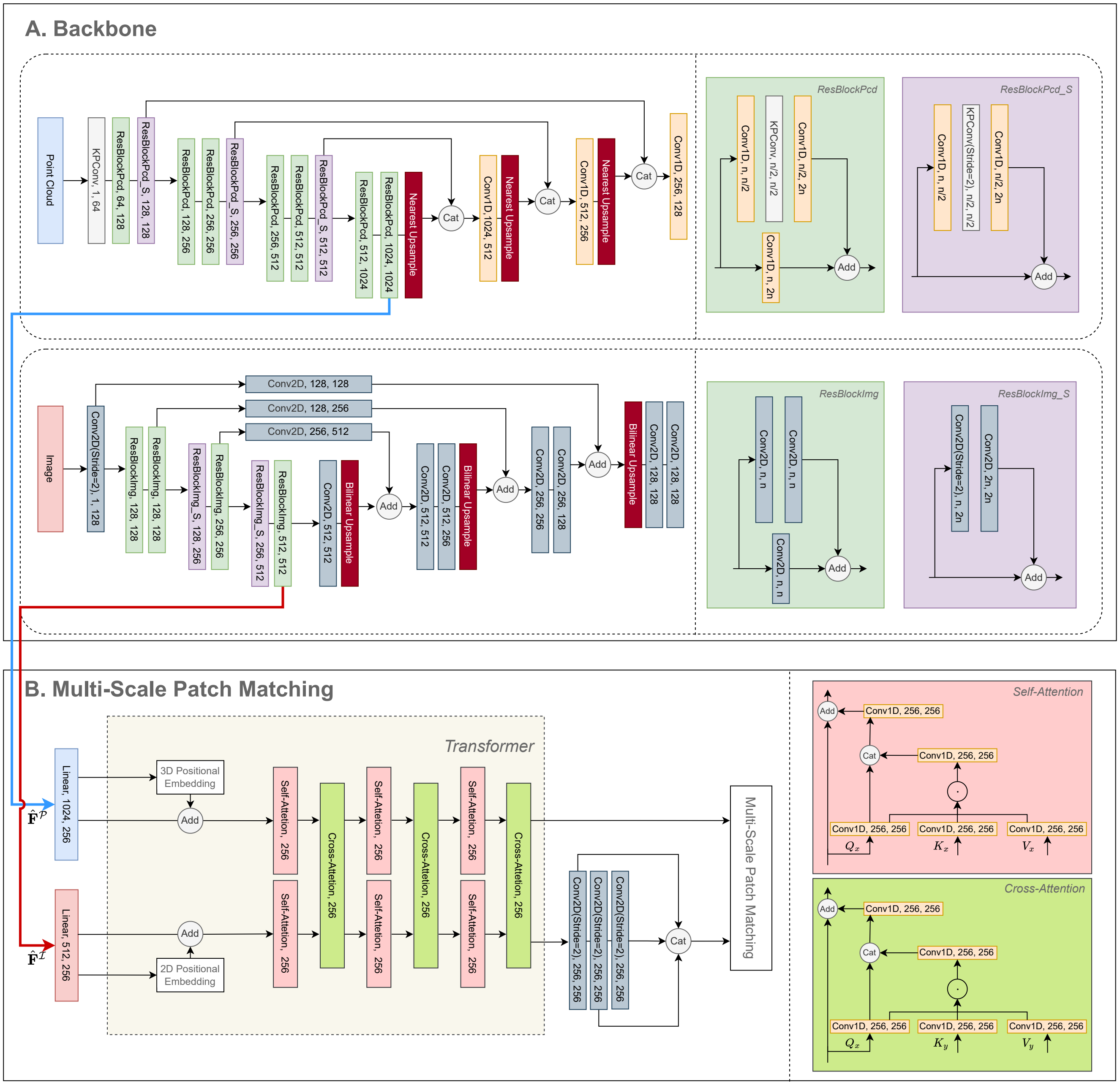


Figure 1: Network architecture.

We further report *Patch Inlier Ratio* (PIR) in the ablation studies to evaluate the accuracy of the patch matching following [12]. PIR is the fraction of patch correspondences whose overlap ratios under the ground-truth transformation are above 0.3. It reflects the quality of the putative patch correspondences. A pixel (point) is overlapped if there exists a point (pixel) such that their 3D distance is below a 3D threshold (*i.e.*, 3.75cm) and their 2D distance is below a 2D threshold (*i.e.*, 8 pixels). As a result, we obtain two overlap ratios, one on the image side and one on the point cloud side. Here we take the smaller one of them as the final overlap ratio between $\mathbf{I}$ and $\mathbf{P}$.

C. Data Preparation

As there is no off-the-shelf benchmarks for 2D-3D registration, we first build two challenging benchmarks based on RGB-D Scenes V2 [10] and 7Scenes [6] datasets.

Scene	Scene-11	Scene-12	Scene-13	Scene-14	Mean
Depth mean (m)	1.74	1.66	1.18	1.39	1.49
Depth std (m)	0.67	0.64	0.39	0.48	0.55
Depth range (m)	2.20	2.22	1.72	2.07	2.05

Table 1: Statistics on the testing set of RGB-D Scenes V2.

Scene	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs	Mean
Depth mean (m)	1.78	1.55	0.80	2.03	2.25	2.13	1.84	1.77
Depth std (m)	0.48	0.30	0.21	0.43	0.39	0.62	0.48	0.41
Depth range (m)	2.66	1.60	0.97	1.91	1.79	2.48	3.03	2.06

Table 2: Statistics on the testing set of 7-Scenes.

C.1. RGB-D Scenes V2

RGB-D Scenes V2 consists of RGB-D scans of 14 indoor scenes. We evaluate the generality to *unseen scenes* of our method and the baselines on this benchmark. For each scene, we fuse every 25 consecutive depth frames into a point cloud fragment, which is then voxelized with a voxel size of 2.5cm. The first RGB image of every 25 frames are sampled as the set of images. We then consider every pair of image and point cloud, and select those whose overlap ratios are at least 30%. The overlap is computed in the 3D space. The image are first unprojected into a point cloud according to the corresponding depth frame. Then a point is considered as overlapped if there exists a point in the other side which is closer than 3.75cm to it. The pairs from scenes 1-8 are used for training, scenes 9 and 10 for validation, and scenes 11-14 for testing. As last, we obtain a benchmark of 1748 training pairs, 236 for validation and 497 for testing. Tab. 1 shows the statistics on the testing set of our benchmark. In Scene-11 and Scene-12, the camera is further from the scene and the images have a larger range of depth. While in Scene-13 and Scene-14, the scene is much closer to the camera.

C.2. 7-Scenes

7-Scenes consists of RGB-D scans of 7 indoor scenes where each scene has multiple RGB-D sequences. We follow the data split in [6, 2, 17] to evaluate the generality to *unseen viewpoints* of our method and the baselines on this benchmark. For each squence, we follow the same method as in Appx. C.1 to prepare the point cloud fragments and the RGB image frames. Then, for each scene, we collect the all images and point cloud fragments in the training (testing) sequences, and select the image-point-cloud pairs from them whose overlap ratios are at least 50% as the training (testing) data. The training data are split by 80%/20% for training/validation. Note that as the RGB images and the depth images are not calibrated in 7-Scenes, we follow [18] to rescale the image by $\frac{585}{525}$ for an approximate calibration. Tab. 2 shows the statistics on the testing set of 7-Scenes. The distance between the camera and the scene significantly varies in different scenes. The camera is relatively far from the scene in *office*, *pumpkin* and *kitchen*, but is much closer in *heads*. As a result, the scale ambiguity is more significant in 7-Scenes.

Model	Scene-11	Scene-12	Scene-13	Scene-14	Mean
Inlier Ratio ↑					
$(24 \times 32, 12 \times 16, 6 \times 8)$	**32.8**	**34.4**	**39.2**	**23.3**	**32.4**
$(24 \times 32, 12 \times 16)$	32.9	34.4	35.3	21.6	31.1
(24×32)	31.7	33.3	27.3	16.8	27.3
Feature Matching Recall ↑					
$(24 \times 32, 12 \times 16, 6 \times 8)$	**98.6**	**98.0**	**88.7**	**77.9**	**90.8**
$(24 \times 32, 12 \times 16)$	97.2	98.0	86.6	77.0	89.7
(24×32)	97.2	97.1	85.6	75.7	88.9
Registration Recall ↑					
$(24 \times 32, 12 \times 16, 6 \times 8)$	**63.9**	**53.9**	**58.8**	**49.1**	**56.4**
$(24 \times 32, 12 \times 16)$	55.6	53.9	43.3	41.2	48.5
(24×32)	52.8	51.0	26.8	26.1	39.2

Table 3: Additional ablation studies on RGB-D Scenes V2. **Boldfaced** numbers highlight the best and the second best are underlined.

D. Additional Experiments

D.1. Additional Ablation Studies

In Tab. 3, we further progressively ablate the patch pyramid and report the detailed results on each scene. Note that here all the models are both trained and tested with the corresponding resolution levels, while we albate each pyramid level only in the inference in Tab. 3 of the main paper.

For *Inlier Ratio*, three models achieves comparable results on the first two scenes, but the models with multi-scale patch pyramid performs considerably better than the single-scale one on Scene-13 and Scene-14. As discussed in Tab. 1, the camera is closer to the scene in Scene-13 and Scene-14, which could cause severe inconsistency between the image patchs and the point patches. By leveraging the patch pyramid, the scale ambiguity is alleviated such that more accurate correspondences are obtained.

For *Registration Recall*, more significant improvements are also obtained in the last two scenes. Note that although the three models achieve similar inlier ratios in Scene-11, the multi-scale patch pyramid provide more thoroughly-distributed correpondences, which contributes more accurate registration.

D.2. Additional Evaluations on 7-Scenes

We further present the evaluation results on 7-Scenes [6] following the settings in [17]. We fuse a point cloud fragment with 5 consecutive depth frames. During training, we construct 5 training pairs between the fused point cloud and the corresponding RGB images. During testing, we only

Model	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs	Mean
Inlier Ratio ↑								
FCGF-2D3D [4]	59.2	58.5	67.5	54.4	45.0	51.6	33.5	52.8
P2-Net [17]	60.9	66.9	66.1	55.8	57.0	56.1	42.4	57.9
Predator-2D3D [8]	75.3	71.6	**82.1**	56.1	55.3	57.2	57.7	65.0
2D3D-MATR (*ours*)	**84.1**	**79.2**	76.5	**73.6**	**71.8**	**78.0**	**69.1**	**76.0**
Feature Matching Recall ↑								
FCGF-2D3D [4]	81.8	81.0	91.0	67.5	41.7	52.3	10.5	60.8
P2-Net [17]	82.5	93.0	89.5	70.6	76.2	64.6	22.5	71.3
Predator-2D3D [8]	98.8	94.0	**100.0**	66.5	69.0	61.5	69.0	79.8
2D3D-MATR (*ours*)	**100.0**	**96.5**	99.0	**99.0**	**92.0**	**99.5**	**99.0**	**97.9**
Registration Recall ↑								
FCGF-2D3D [4]	99.8	**98.0**	98.0	97.0	89.2	96.7	94.5	96.2
P2-Net [17]	99.8	**98.0**	96.0	98.1	91.7	97.2	93.0	96.3
Predator-2D3D [8]	99.6	92.5	**99.0**	96.5	82.0	95.5	87.0	93.2
2D3D-MATR (*ours*)	**100.0**	**98.0**	98.5	**98.5**	**95.0**	**100.0**	**98.0**	**98.3**

Table 4: Evaluation results on 7Scenes following the experiment settings in [17]. **Boldfaced** numbers highlight the best and the second best are underlined.

use the last RGB frame to construct 1 testing pair for each point cloud fragment. The RGB images are rescaled as described in Appx. C.2. As a result, we obtain 23500 training pairs, 2500 validation pairs, and 3400 testing pairs. All the models are trained from scratch in the experiments. Compared to our benchmark in the main paper, this setting is more easier due to small transformation and high overlap ratio. Note that the thresholds for the metrics in this setting are $\tau_1 = 4.5$cm, $\tau_2 = 50\%$ and $\tau_3 = 5$cm following [17].

The results are shown in Tab. 4. For *Inlier Ratio*, 2D3D-MATR outperforms the baseline methods by a large margin, especially on the last four harder scenes. This further contributes to significant improvements on *Feature Matching Recall*, where our method surpasses the second best Predator-2D3D by 18 pp. For *Registration Recall*, the performance tends to be saturated in most scenes. Nonetheless, 2D3D-MATR still achieves the best results, especially on *pumpkin* and *stairs*. These results have demonstrated the efficacy of our method.

D.3. Qualitative Results

We provide more qualitative comparisons of P2-Net [17] and 2D3D-MATR on 7-Scenes (Fig. 2) and RGB-D Scenes V2 (Fig. 3). It is observed that the correspondences from our method are much denser and more accurate those from P2-Net. Moreover, 2D3D-MATR extracts correspondences from both near and far regions, showing strong robustness to scale variance.

References

[1] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *CVPR*, pages 6359–6367, 2020. 1

[2] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *CVPR*, pages 6684–6692, 2017. 3

[3] G. Bradski. The opencv library. *Dr. Dobb's Journal of Software Tools*, 2000. 1

[4] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *ICCV*, pages 8958–8966, 2019. 1, 4

[5] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *CVPR*, pages 8092–8101, 2019. 1

[6] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *ISMAR*, pages 173–179. IEEE, 2013. 2, 3

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1

[8] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *CVPR*, pages 4267–4276, 2021. 1, 4

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 1

[10] Kevin Lai, Liefeng Bo, and Dieter Fox. Unsupervised feature learning for 3d scene labeling. In *ICRA*, pages 3050–3057. IEEE, 2014. 2

[11] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 2117–2125, 2017. 1

[12] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *CVPR*, pages 11143–11152, 2022. 2

[13] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *CVPR*, pages 8922–8931, 2021. 1

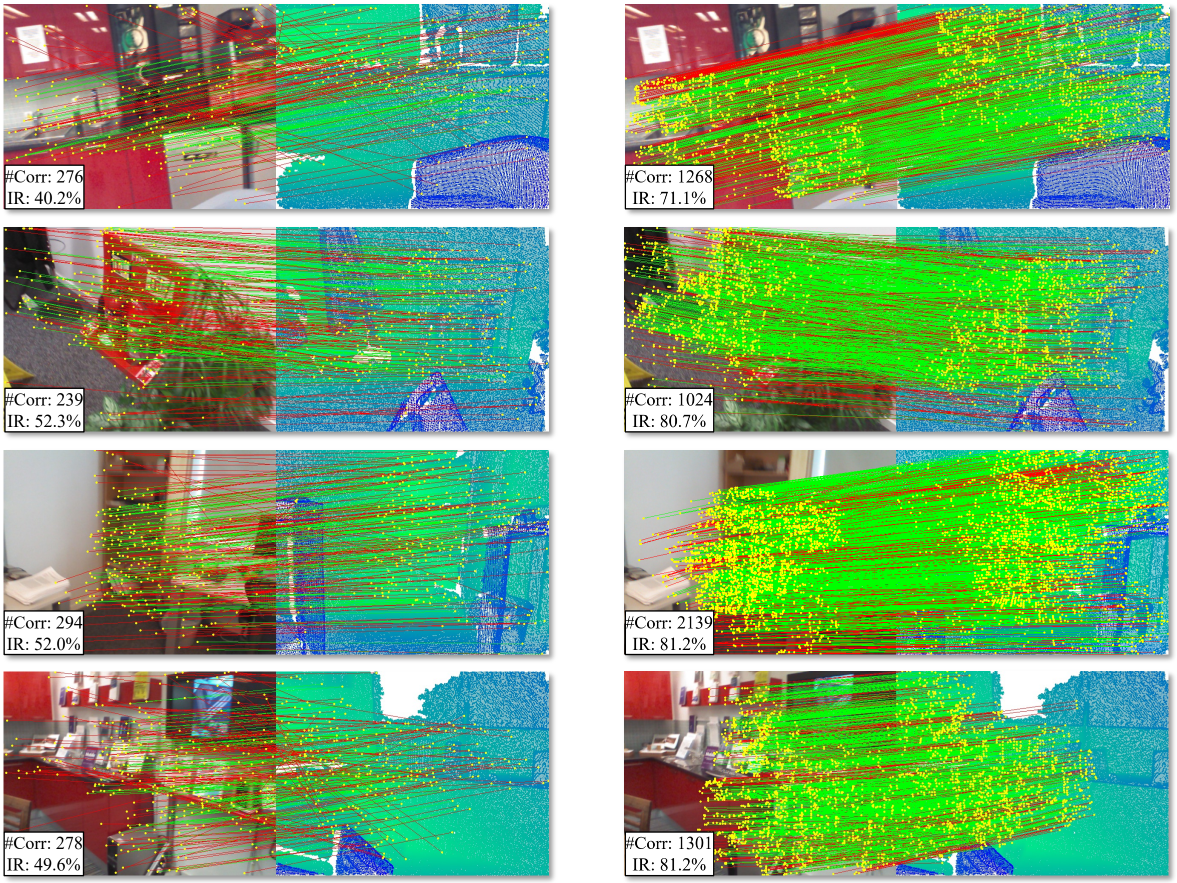
[14] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *CVPR*, pages 6398–6407, 2020. 1

[15] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *ICCV*, pages 6411–6420, 2019. 1

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017. 1

[17] Bing Wang, Changhao Chen, Zhaopeng Cui, Jie Qin, Chris Xiaoxuan Lu, Zhengdi Yu, Peijun Zhao, Zhen Dong, Fan Zhu, Niki Trigoni, et al. P2-net: Joint description and detection of local features for pixel and point matching. In *ICCV*, pages 16004–16013, 2021. 1, 3, 4

[18] Luwei Yang, Ziqian Bai, Chengzhou Tang, Honghua Li, Yasutaka Furukawa, and Ping Tan. Sanet: Scene agnostic net-

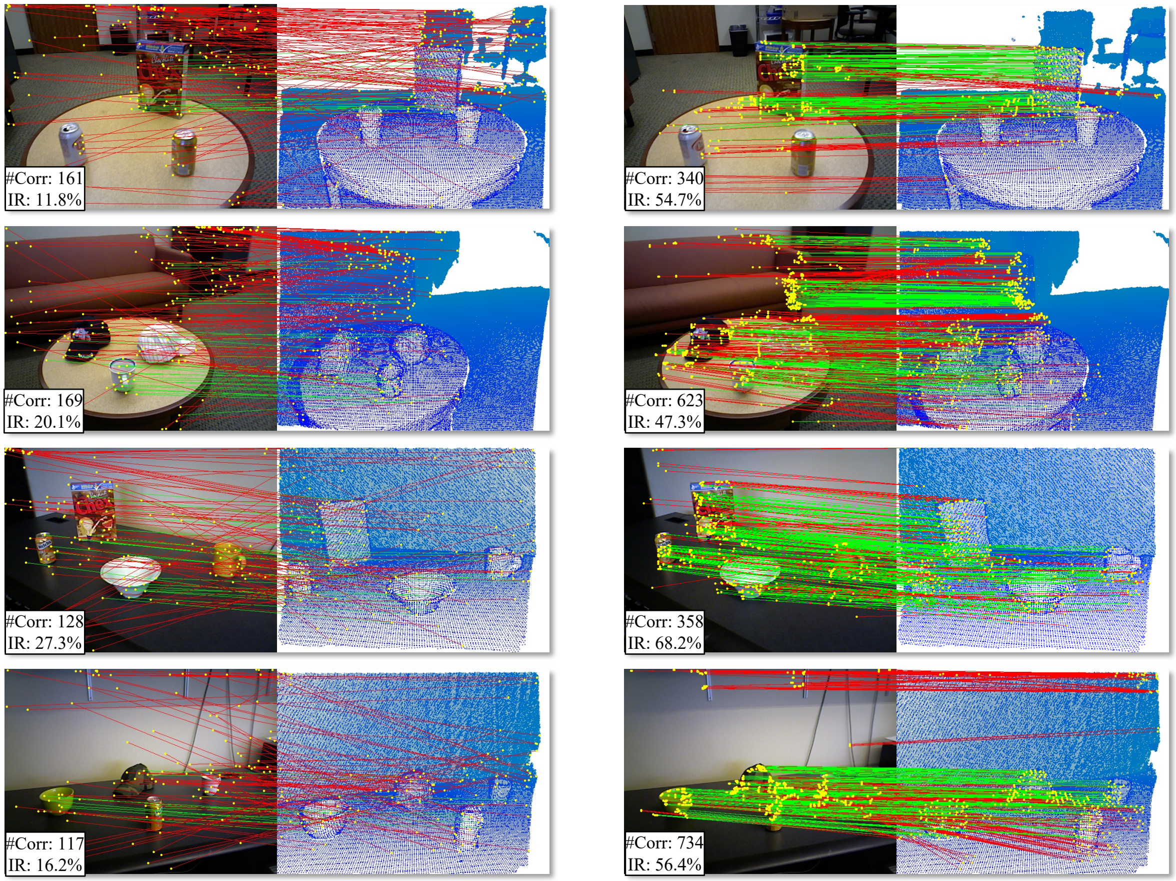


(a) P2-Net

(b) 2D3D-MATR (*ours*)

Figure 2: Comparisons of extracted correspondences on 7-Scenes.

work for camera localization. In *ICCV*, pages 42–51, 2019.
3



(a) P2-Net

(b) 2D3D-MATR (*ours*)

Figure 3: Comparisons of extracted correspondences on RGB-D Scenes V2.