Diff^{2}I2P: Differentiable Image-to-Point Cloud Registration with Diffusion Prior

Supplementary Material

Appendix

In this supplementary material, we first provide precise definitions of the evaluation metrics used in the paper (Sec. A). Next, we provide a detailed introduction and discussion of the related work (Sec. B). Then, we offer a detailed description of the training and test datasets (Sec. C), including their partitioning method. Additionally, we describe the network architecture and implementation details (Sec. D). We also conduct additional experiments (Sec. E) such as further metric measurements and runtime analysis. Finally, we present more visualization results on multiple datasets to illustrate the performance of the proposed method intuitively (Sec. F).

A. Evaluation Metrics

Inlier Ratio (IR): We follow [7] to compute the indicator inlier ratio. The Inlier Ratio for cross-modal registration measures the proportion of point-to-pixel correspondences $(\mathbf{x_i}, \mathbf{y_i}) \in \mathcal{C}$ that are within a certain residual threshold under the ground truth transformation $\bar{\mathcal{T}}_{\mathbf{I}}^{\mathbf{P}}$. Here, $\mathcal{C}$ denotes the estimated correspondence set between the 3D point set $\mathbf{I}$ and the image pixel set $\mathbf{P}$, and $\bar{\mathcal{T}}_{\mathbf{I}}^{\mathbf{P}}$ represents the ground truth transformation from $\mathbf{I}$ to $\mathbf{P}$. A correspondence pair is considered an inlier if the Euclidean norm of its residual is less than the threshold $\tau_1 = 10cm$. The Inlier Ratio for the cross-modal pair $\mathbf{I}$ and $\mathbf{P}$ is computed as:

$$\mathrm{IR}(\mathbf{I}, \mathbf{P}) = \frac{1}{|\mathcal{C}|} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{C}} \mathbb{I}\left[\|\bar{\mathcal{T}}_{\mathbf{I}}^{\mathbf{P}}(\mathcal{K}^{-1}(\mathbf{y}_i)) - \mathbf{x}_i\| < \tau_1\right], \tag{1}$$

where $\mathbb{I}[\cdot]$ is the indicator function that counts the number of correspondences with residuals less than the threshold τ_1 and $\mathcal{K}^{-1}$ is a function that unprojects a pixel to a 3D point.

Feature Matching Recall (FMR): The Feature Matching Recall is used to evaluate the result of feature matching by determining the fraction of cross-modal pairs where the Inlier Ratio exceeds a given threshold, $\tau_2 = 5\%$. This metric reflects the probability of accurately recovering the correct transformation using the estimated correspondence set $\mathcal{C}$, typically with the aid of a robust pose estimation algorithm such as RANSAC [2]. For a dataset $\mathcal{D}$ containing $|\mathcal{D}|$ cross-modal pairs, the Feature Matching Recall is defined as follows:

$$\mathrm{FMR}(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{I}, \mathbf{P}) \in \mathcal{D}} \mathbb{I}[IR(\mathbf{I}, \mathbf{P}) > \tau_2], \tag{2}$$

where $\mathbb{I}[\cdot]$ is the indicator function that counts the number of cross-modal pairs for which the Inlier Ratio exceeds the threshold τ_2. This metric provides insight into the overall robustness and accuracy of the feature matching process across the entire dataset.

Patch Inlier Ratio (PIR): PIR [5] represents the fraction of patch correspondences whose overlap ratios, under the ground-truth transformation $\bar{\mathcal{T}}_{\mathbf{I}}^{\mathbf{P}}$, are above 0.3. This metric reflects the quality of the estimated patch correspondences.

The overlap ratio between an image patch $\tilde{\mathbf{y}}_i$ and a point cloud patch (superpoint) $\tilde{\mathbf{x}}_i$ can be calculated in each modality. The definition of point-cloud-side overlap is:

$$\mathrm{Overlap}_{\mathbf{P}}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i) = \frac{1}{|\mathcal{X}_i|} \sum_{\mathbf{x}_i \in \mathcal{X}_i} \mathbb{I}[\min_{\mathbf{y}_i \in \mathcal{K}^{-1}(\mathcal{Y})} \|\mathbf{x}_i - \mathbf{y}_i\|_2 < \tau_3], \tag{3}$$

where $\mathcal{X}_i$ is the up-sampled points of the superpoint $\tilde{\mathbf{x}}_i$, $\tau_3 = 3.75$cm is the 3D distance threshold, and the definition of image-side overlap is:

$$\mathrm{Overlap}_{\mathbf{I}}(\tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i) = \frac{1}{|\mathcal{Y}_i|} \sum_{\mathbf{y}_i \in \mathcal{Y}} \mathbb{I}[\min_{\mathbf{x}_i \in \mathcal{K}(\mathcal{X})} \|\mathbf{x}_i - \mathbf{y}_i\|_2 < \tau_4], \tag{4}$$

where $\mathcal{Y}_i$ is the up-sampled pixels of the image patch $\tilde{\mathbf{y}}_i$, $\tau_4 = 8$ pixels is the 2D distance threshold. We take the smaller one of two overlaps and compute the PIR indicator by:

$$\mathrm{PIR} = \frac{1}{|\tilde{\mathcal{C}}|} \sum_{(\tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_i) \in \tilde{\mathcal{C}}} \mathbb{I}\left[\min\left(\mathrm{Overlap}_{\mathbf{I}}(\hat{\mathbf{x}}_i), \mathrm{Overlap}_{\mathbf{P}}(\hat{\mathbf{y}}_j)\right) > \tau_5\right], \tag{5}$$

where $\tilde{\mathcal{C}}$ denotes the estimated set of patch correspondences, $\mathbb{I}[\cdot]$ is the indicator function that returns 1 if the condition inside is true and 0 otherwise, and $\tau_5 = 0.3$ is the overlap threshold.

Registration Recall (RR): The Registration Recall is a metric used to evaluate the accuracy of cross-modal registration between a 3D point cloud and an image. It measures the fraction of image-point cloud pairs for which the Root Mean Square Error (RMSE) is below a certain threshold, denoted as $\tau_6 = 0.1$m. For a dataset $\mathcal{D}$ containing $|\mathcal{D}|$ pairs of image-point cloud pairs, the Registration Recall is defined as follows:

$$\mathrm{RR}(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{I}, \mathbf{P}) \in \mathcal{D}} \mathbb{I}[RMSE(\mathbf{I}, \mathbf{P}) < \tau_6], \tag{6}$$

where $\mathbb{I}[\cdot]$ is an indicator function that counts the number of image-point cloud pairs with an RMSE below the thresh-

old τ_6. The RMSE for each pair $(\mathbf{I}, \mathbf{P}) \in \mathcal{D}$ is calculated as:

$$\text{RMSE}(\mathbf{I}, \mathbf{P}) = \sqrt{\frac{1}{|\mathbf{P}|} \sum_{\mathbf{x_i} \in \mathbf{P}} \|\bar{\mathcal{T}}_{\mathbf{I}}^{\mathbf{P}}(\mathbf{x_i}) - \mathcal{T}_{\mathbf{I}}^{\mathbf{P}}(\mathbf{x_i})\|^2}, \quad (7)$$

where $\mathcal{T}_{\mathbf{I}}^{\mathbf{P}}$ represents the predicted transformation from $\mathbf{I}$ to $\mathbf{P}$, and $\bar{\mathcal{T}}_{\mathbf{I}}^{\mathbf{P}}$ denotes the ground truth transformation from $\mathbf{I}$ to $\mathbf{P}$. This metric provides an indication of the precision of the cross-modal registration process across the entire dataset.

B. Detailed Introduction and Discussion on Related Works

B.1. Baseline

2D3D-MATR. 2D3D-MATR [5] is a detection-free method for accurate and robust image-to-point cloud registration. It adopts a coarse-to-fine manner where it first forms a coarse correspondence set between downsampled patches of the input image and the point cloud, then extends them into dense correspondences within the patch. In coarse-level matching, a transformer facilitates contextual sharing between image and point cloud features. Then a multi-scale feature matching module is designed to match each point patch with its most suitable zoomed image patch to avoid the scale ambiguity problem. Finally, the PnP-RANSAC is applied to the fine-level dense correspondences to estimate the transformation. Though the design of the transformer block and the multi-scale matching improve the quality of the extracted correspondences and contribute to accurate 2D-3D registration, the registration process remains hindered by the inherent modality gap between images and point clouds. This gap often results in poor feature matching accuracy, ultimately leading to registration failures.

B.2. Closely Related Work

FreeReg. FreeReg [8] adopts a pretrained diffusion model with monocular depth estimators for cross-modality feature extraction. Specifically, it constructs two types of features for establishing correspondences: diffusion features and geometric features. The diffusion features are the intermediate representations of the depth-controlled diffusion model, which shows strong consistency across RGB images and depth maps. The geometric features capture distinct local geometric details on the RGB image and depth map using a monocular depth estimator. The combination of these two features enables accurate cross-modal correspondence estimation for registration. However, it still heavily relies on the explicit feature of the pretrained depth-controlled diffusion model, requiring manual selection of the feature layers. Additionally, its computational cost is significantly higher.

VP2P-Match. VP2P-Match primarily focuses on registration in outdoor scenes, with point clouds mainly captured by LiDAR, which differs from the benchmarks used by other baseline methods and our approach. VP2P-Match [11] propose to learn a structured cross-modality latent space to represent pixel features and 3D features via a differentiable probabilistic PnP solver. Specifically, it designs a triplet network to learn VoxelPoint-to-Pixel matching, where the 3D elements are represented using both voxels and points, enabling learning of the cross-modality latent space with pixels. The entire framework is trained end-to-end by applying supervision directly to the predicted pose distribution using a probabilistic PnP solver. Although using VoxelPoint for 3D feature extraction provides more descriptive local features, the registration still fails to bridge the modality gap. Note that VP2P-Match still follows the dense matching convention while leveraging the Monto Carlo strategy to approximate the KL divergence loss of the predicted pose distribution and ground truth pose distribution. It may require more computational overhead to achieve the differentiability, and the large search space of dense matching makes it prone to difficulty in finding the correct correspondences.

C. Datasets

We train and evaluate Diff^{2}I2P on two indoor datasets 7-Scenes [3] and RGB-D Scenes V2 [4], and compare it with the baselines. We also provide simple evaluation of Diff^{2}I2P on KITTI [6], which contains dynamic outdoor scenarios. The detailed information are as follows.

C.1 7-Scenes

The 7-Scenes dataset [3] contains RGB-D scans of seven indoor scenes: *Chess, Fire, Heads, Office, Pumpkin, Kitchen, and Stairs*. Each scene includes multiple sequences. We use preprocessed data from [5], with the preprocessing steps as follows. For each scene, we select 25 consecutive depth maps to generate a dense point cloud, which is then downsampled using a voxel size of 2.5 cm. After generating the point cloud data, we extracted each point cloud's first frame's corresponding image to form an image-point cloud pair. This process is repeated to generate the entire dataset. After data generation, a selection step is applied: each image is unprojected into 3D space to create a virtual point cloud, and its overlap ratio with the actual point cloud is calculated. Pairs with an overlap ratio below 50% are removed. The final dataset includes 2,304 test samples and 5,059 training samples, with the training data further split into 80% for training and 20% for validation. Since the images and depth maps in the 7-Scenes dataset are not calibrated, we follow [10] by rescaling the images by a factor of $\frac{585}{525}$ to achieve an approximate calibration.

C.2 RGB-D Scenes V2

The RGB-D Scenes V2 dataset [4] contains 14 indoor scenes, labeled from *Scene-1* to *Scene-14*. We follow the same data generation method as for 7-Scenes, but in this dataset, we remove image-point cloud pairs with an overlap ratio below 30%. Compared to 7-Scenes, RGB-D Scenes V2 has a smaller data volume, so we increase the proportion of training data accordingly. The dataset was randomly split into training, validation, and test sets, containing 1,978, 117, and 386 samples, respectively.

C.3 KITTI

The KITTI-DC dataset [6] contains dynamic image-point cloud pairs. The sparse point clouds are obtained with a 64-line LiDAR scan. The distance between image and point cloud pair is less than 10 meters. We follow RreeReg [8] Kitti benchmark for evaluation.

D. Implementation

D.1. Differentiable BPnP Solver

BPnP [1] efficiently derives accurate gradients of the PnP solver based on the Implicit Function Theorem [1] with excellent numerical stability, we employ it as the differentiable PnP solver in our pipeline. Following BPnP, we first construct the constraint function as $\mathbf{f}(\mathbf{x}, \mathbf{y}, \mathcal{T}', \mathbf{K}) = [f_1, f_2, ..., f_m]^T$, where $\mathbf{x}$ and $\mathbf{y}$ are the input correspondences, $\mathcal{T}' = [\mathbf{R}'|\mathbf{t}']$ is the predicted transformation, m is the number of its variables, and K is the camera intrinsic matrix. And for all $i \in \{1, ..., m\}$, f_i is defined as

$$f_i = \frac{\partial \sum_{i=1}^{n} \|\mathcal{K}(\mathbf{R}'\mathbf{x}_i + \mathbf{t}') - \mathbf{y}_i\|_2^2}{\partial \mathcal{T}_i'}. \tag{8}$$

Then given the output gradient $\nabla\mathbf{z}$, the input gradients $\nabla\mathbf{x}$ and $\nabla\mathbf{y}$ can be derived as

$$\nabla\mathbf{x} = \Big[-(\frac{\partial \mathbf{f}}{\partial \mathcal{T}'})^{-1} \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \Big]^T \nabla\mathbf{z}, \tag{9}$$

$$\nabla\mathbf{y} = \Big[-(\frac{\partial \mathbf{f}}{\partial \mathcal{T}'})^{-1} \frac{\partial \mathbf{f}}{\partial \mathbf{y}} \Big]^T \nabla\mathbf{z}. \tag{10}$$

D.2. Depth Densification

The sparse depth projected from the point cloud is densified using simple morphology operations like dilation. Specifically, we first invert the depth values below a threshold (0.1) to match a reference maximum depth value (15.0), enabling a more consistent interpolation. Then a diamond-shaped dilation with a size of 7×7 is utilized to fill empty areas while preserving significant depth features. After that, Hole closing is performed using erosion and dilation operations with smaller kernels (3×3 and 5×5) to clean up the depth map and remove noise or isolated points. Finally, a median blur and a Gaussian blur filter are applied with a kernel size of 5 to help smooth out the depth map. The depth map is inverted again to return to the original depth scale, ensuring that all valid depth values correspond to real-world distances.

Table 1. The comparison of RRE and RTE results between Diff^{2}I2P and 2D3D-MATR [5] on the 7-Scenes [3] dataset.

Method	RRE (m)	RTE (m)
2D3D-MATR [5]	3.053	0.072
Diff^{2}I2P (ours)	2.743	0.065

D.3. Loss Functions

Here we provide the detailed calculation of the circle loss. Following 2D3D-MATR [5], we define the general circle loss $\mathcal{L}_i$ of an anchor descriptor $\mathbf{d}_i$ as:

$$\mathcal{L}_i = \frac{1}{\delta} \log \left[1 + \sum_{\mathbf{d}_j \in \mathcal{D}_i^{\mathcal{P}}} e^{\beta_p^{i,j}\left(d_i^j - \Delta_p\right)} \cdot \sum_{\mathbf{d}_k \in \mathcal{D}_i^{\mathcal{N}}} e^{\beta_n^{i,k}\left(\Delta_n - d_i^k\right)} \right], \tag{11}$$

where $\mathcal{D}_i^{\mathcal{P}}$ and $\mathcal{D}_i^{\mathcal{N}}$ are the descriptors of its positive and negative pairs, d_i^j is the ℓ_2 feature distance, $\beta_p^{i,j} = \delta\lambda_p^{i,j}\left(d_i^j - \Delta_p\right)$ and $\beta_n^{i,k} = \delta\lambda_n^{i,k}\left(\Delta_n - d_i^k\right)$ are the individual weights for the positive and negative pairs, where $\lambda_p^{i,j}$ and $\lambda_n^{i,k}$ are the scaling factors for the positive and negative pairs.

We follow the hyperparameter configuration in [5]. On the coarse level, we generate the ground truth based on bilateral overlap. A patch pair is considered positive if the 2D and 3D overlap ratios between them are both at least 30%, and negative if both overlap ratios are below 20%. The overlap ratio between the 2D and 3D patches is used as λ_p, and λ_n is set to 1. On the fine level, a pixel-point pair is positive if the 3D distance is below 3.75 cm and the 2D distance is below 8 pixels, and negative if the 3D distance is above 10 cm or the 2D distance exceeds 12 pixels. The scaling factors are all set to 1. All other pairs are ignored during training on both levels as the safe region. The margins are set to $\Delta_p = 0.1$ and $\Delta_n = 1.4$.

E. Additional Experiments

E.1. Relative Rotation Error and Relative Translation Error

Relative Rotation Error (RRE) and Relative Translation Error (RTE) are commonly used to assess the alignment accuracy between two point clouds. In cross-modal registration tasks, these metrics can also evaluate the alignment between a point cloud and an image. Specifically, the input point cloud and the point cloud projected from the depth map corresponding to the image are treated as the two point clouds

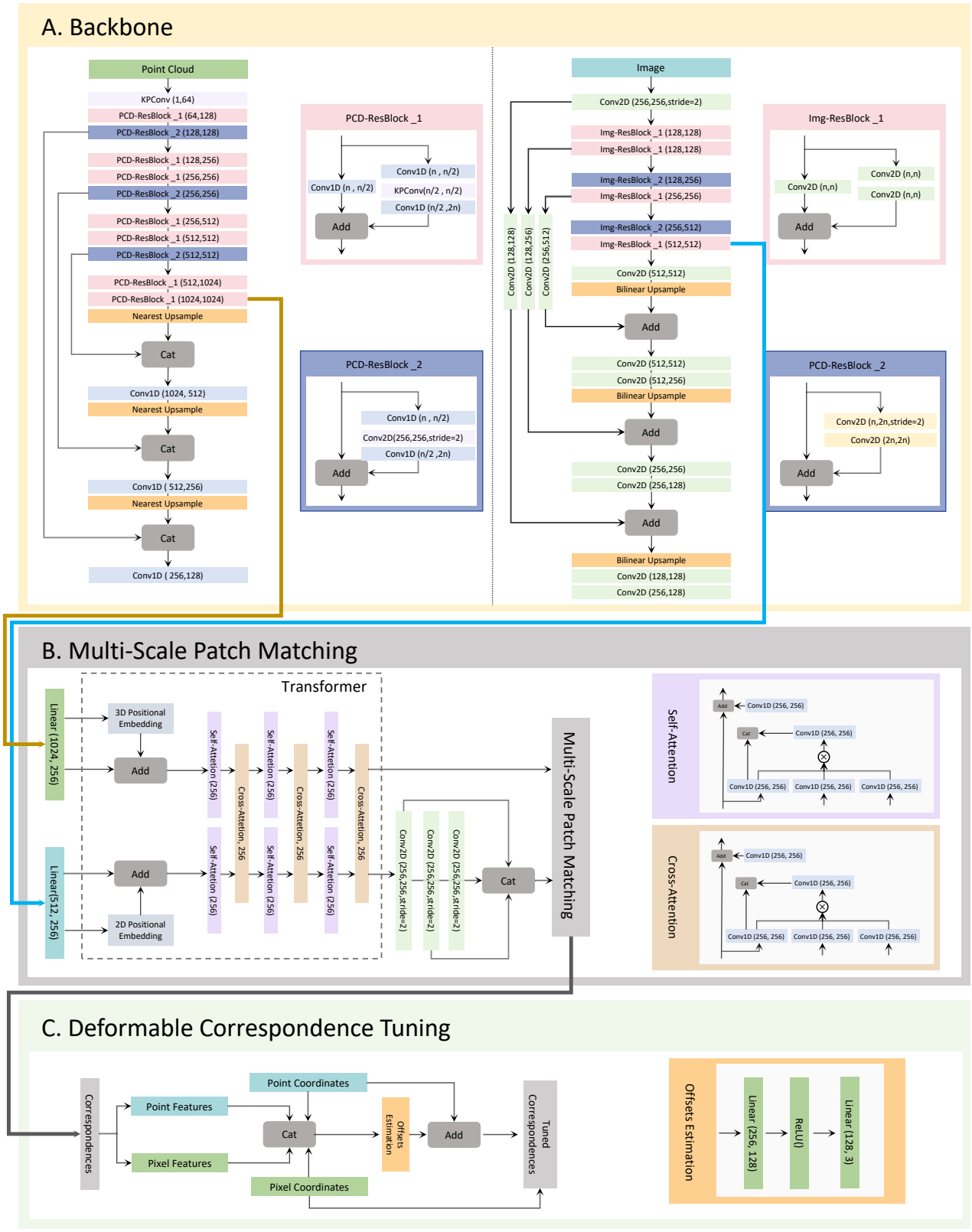


Figure 1. The network architecture of our proposed Diff^{2}I2P.

for evaluation, and their RRE and RTE are computed. Tab. 1 shows the RRE and RTE results of Diff^{2}I2P compared to the baseline, 2D3D-MATR, where the evaluation dataset is 7-Scenes [3].

Table 2. Registration under noise and sparse data conditions. Random shifts are sampled from a normal distribution $\mathcal{N}(0, 0.1)$ (m).

conditions	IR (%)↑	FMR (%)↑	RR (%)↑
(a) random shifts for all points	53.1	91.8	82.5
(b) randomize 1% points' coordinates	52.7	91.2	82.0
(c) random remove 10% points	53.0	91.8	81.8
(d) w/o additional conditions	53.2	92.1	83.0

As shown in the table, Diff^{2}I2P significantly outperforms the baseline method in both RRE and RTE metrics. This demonstrates that our method not only accurately estimates the transformation in most scenarios but also achieves high-quality results, ensuring tight alignment between the point cloud and the image.

E.2. Noise and sparse data conditions

To validate the robustness of Diff^{2}I2P under noisy scenarios and sparse data conditions, we simulate these conditions by randomizing point coordinates and removing a subset of points. As shown in Tab. 2, Diff^{2}I2P maintains stable performance in these simulated scenarios.

E.3. Outdoor and dynamic scenarios evaluation

Our method has been primarily evaluated in static scenarios. To further assess its performance in dynamic and outdoor environments, we conduct experiments on the KITTI dataset. As shown in Tab. 3, Diff^{2}I2P outperforms baselines across nearly all metrics while maintaining an inference speed comparable to that of the fastest method, 2D3D-MATR [5].

E.4. Comparison with Diff-Reg

Diff-Reg [9] serves as a baseline method that performs the diffusion denoising process at the correspondence set. Although it is not specifically designed for image-to-point cloud cross-modal registration, it can still be applied to this task. Here, we provide a detailed comparison with Diff-Reg. We continue to use 7Scenes as the primary benchmark dataset and report three key metrics: IR (Inlier Ratio), FMR (Feature Matching Recall), and RR (Registration Recall). We train Diff-Reg on 7-Scenes and select the best-performing checkpoint on the validation set for testing as shown in Fig 2.

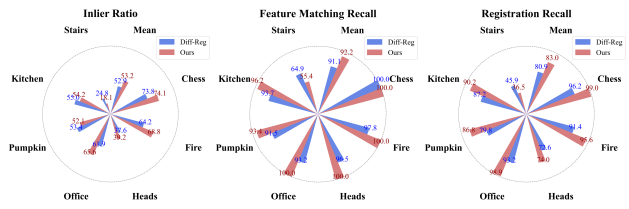


Figure 2. Detailed comparison with Diff-Reg.

Table 3. Comparison experiments with baselines on KITTI.

Method	FMR (%)↑	IR (%)↑	RRE (°)↓	RTE (m)↓	RR (%)↑	Time (s)↓
2D3D-MATR [5]	**99.7**	59.1	3.334	0.838	75.4	**0.061**
FreeReg [8]	**99.7**	58.3	5.987	2.414	70.5	8.763
Diff^{2}I2P (ours)	**99.7**	**62.9**	**2.836**	**0.773**	**82.2**	0.062

E.5. Detailed comparison with FreeReg

Owing to space constraints, we present a simple comparison with FreeReg in the main paper. Additional detailed results on 7-Scenes and RGB-D Scenes V2 are provided in Fig 3.

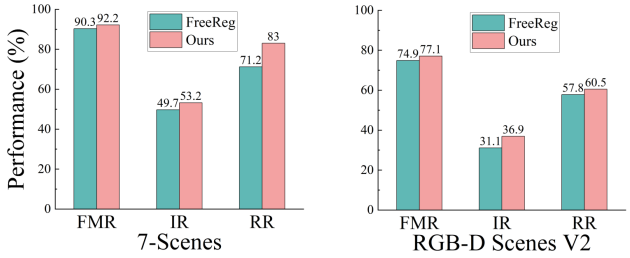


Figure 3. Detailed comparison with FreeReg.

F. Visualizations

We present additional qualitative results to compare Diff^{2}I2P with the baseline method, 2D3D-MATR [5]. For clarity, we visualize the correspondences extracted by both methods, selecting the top 500 correspondences with the highest feature matching scores. Fig. 4 illustrates the results on the 7-Scenes [3] dataset, while Fig. 5 highlights the results on the RGB-D Scenes V2 [4] dataset. The findings show that Diff^{2}I2P extracts more accurate correspondences, delivering robust and superior scene-agnostic registration performance.

References

[1] Bo Chen, Alvaro Parra, Jiewei Cao, Nan Li, and Tat-Jun Chin. End-to-end learnable geometric vision by backpropagating pnp optimization. In *CVPR*, pages 8100–8109, 2020. 3

[2] Martin A Fischler and Robert C Bolles. 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. 1

[3] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 173–179. IEEE, 2013. 2, 3, 4, 5

[4] Kevin Lai, Liefeng Bo, and Dieter Fox. Unsupervised feature learning for 3d scene labeling. In *IEEE international conference on robotics and automation*, pages 3050–3057. IEEE, 2014. 2, 3, 5

[5] Minhao Li, Zheng Qin, Zhirui Gao, Renjiao Yi, Chenyang Zhu, Yulan Guo, and Kai Xu. 2d3d-matr: 2d-3d matching

transformer for detection-free registration between images and point clouds. In *ICCV*, pages 14128–14138, 2023. 1, 2, 3, 5

[6] Jonas Uhrig, Nick Schneider, Lukas Schneider, Uwe Franke, Thomas Brox, and Andreas Geiger. Sparsity invariant cnns. In *2017 international conference on 3D Vision (3DV)*, pages 11–20. IEEE, 2017. 2, 3

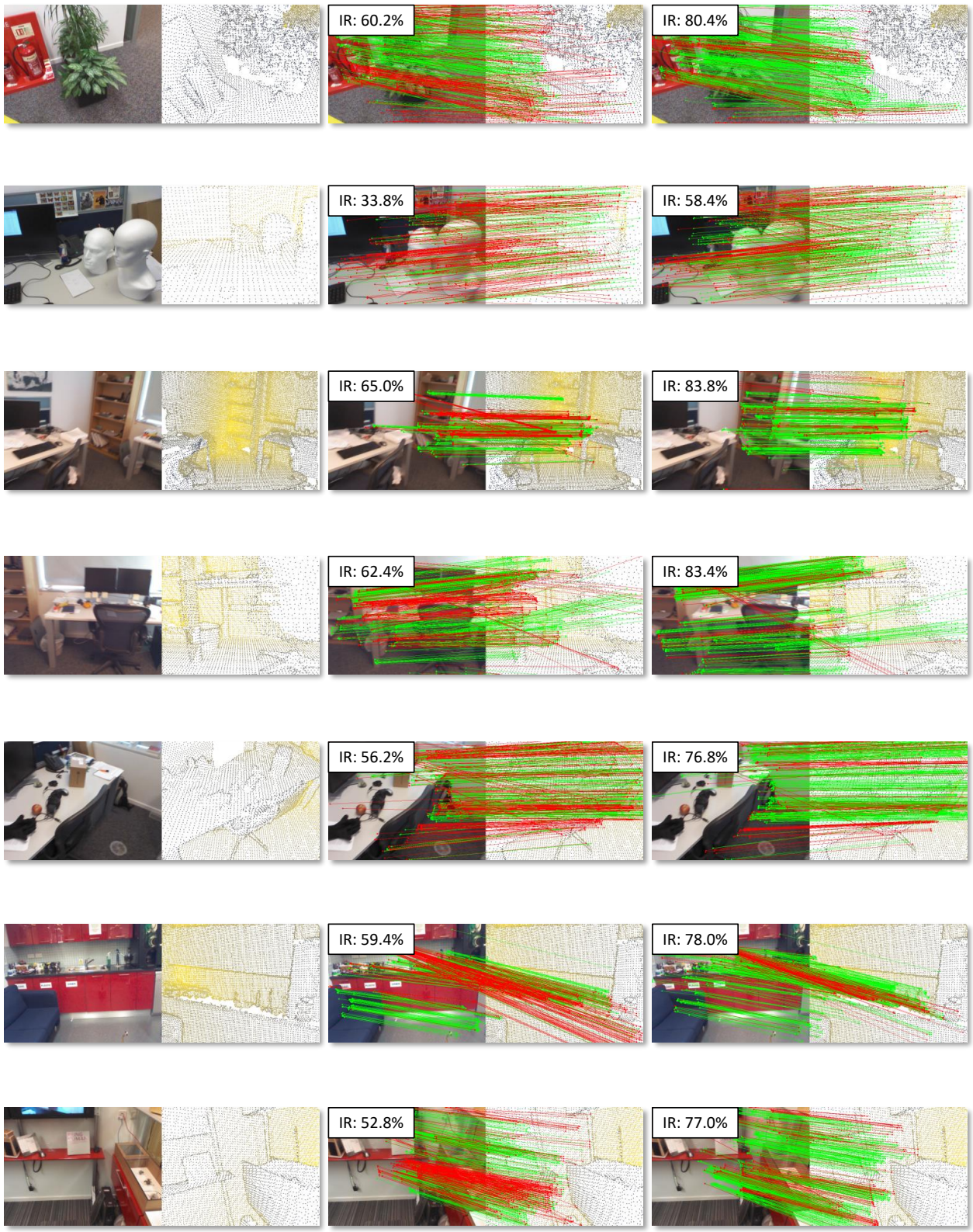
[7] Bing Wang, Changhao Chen, Zhaopeng Cui, Jie Qin, Chris Xiaoxuan Lu, Zhengdi Yu, Peijun Zhao, Zhen Dong, Fan Zhu, Niki Trigoni, et al. P2-net: Joint description and detection of local features for pixel and point matching. In *ICCV*, pages 16004–16013, 2021. 1

[8] Haiping Wang, Yuan Liu, Bing Wang, Yujing Sun, Zhen Dong, Wenping Wang, and Bisheng Yang. Freereg: Image-to-point cloud registration leveraging pretrained diffusion models and monocular depth estimators. *arXiv preprint arXiv:2310.03420*, 2023. 2, 3, 5

[9] Qianliang Wu, Haobo Jiang, Lei Luo, Jun Li, Yaqing Ding, Jin Xie, and Jian Yang. Diff-reg: Diffusion model in doubly stochastic matrix space for registration problem. In *ECCV*, pages 160–178. Springer, 2024. 5

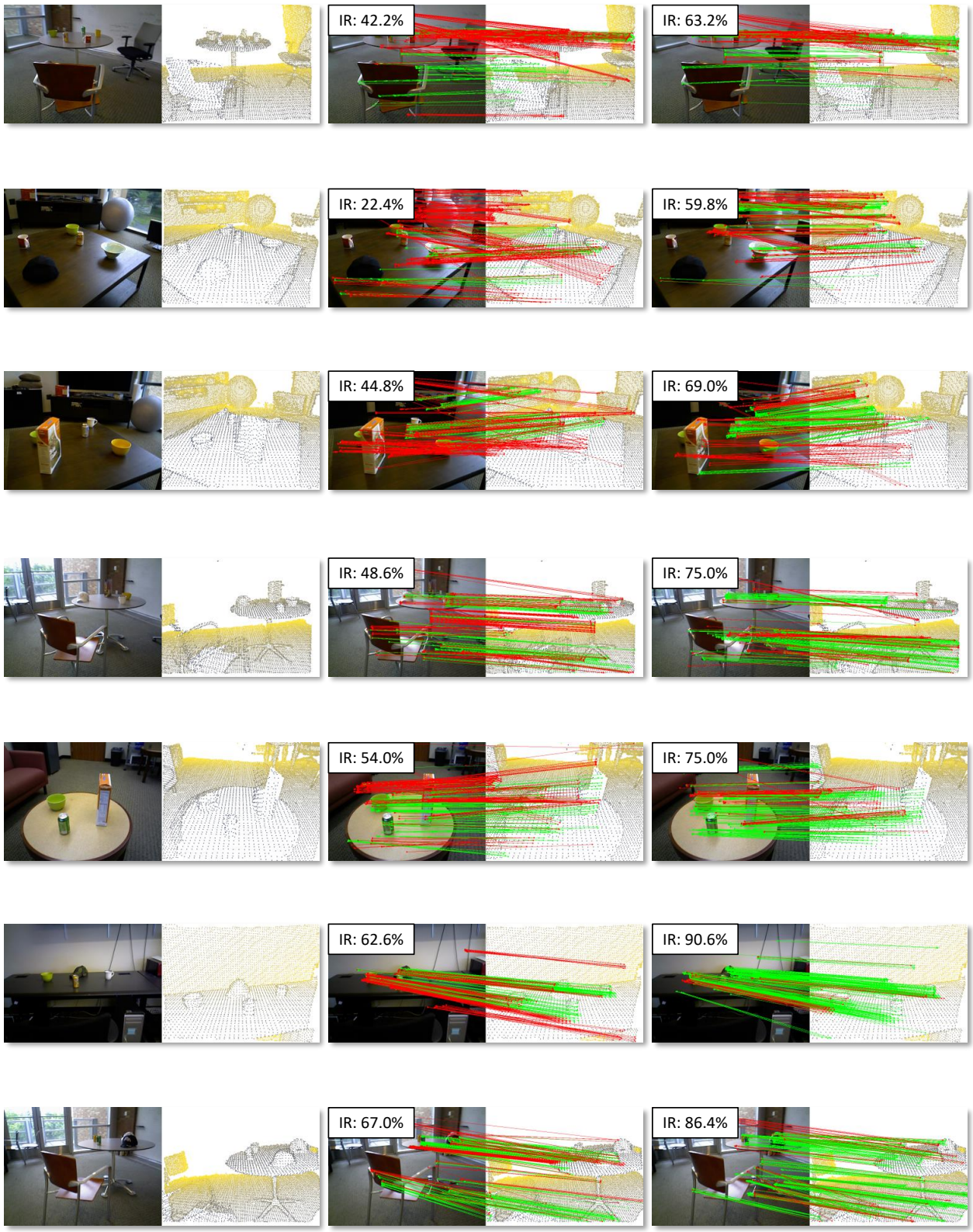
[10] Luwei Yang, Ziqian Bai, Chengzhou Tang, Honghua Li, Yasutaka Furukawa, and Ping Tan. Sanet: Scene agnostic network for camera localization. In *ICCV*, pages 42–51, 2019. 2

[11] Junsheng Zhou, Baorui Ma, Wenyuan Zhang, Yi Fang, Yu-Shen Liu, and Zhizhong Han. Differentiable registration of images and lidar point clouds with voxelpoint-to-pixel matching. *NeurIPS*, 36, 2024. 2



(a) Input Image & Point Cloud (b) 2D3D-MATR (c) Ours

Figure 4. Correspondence visualizations on 7-scenes



(a) Input Image & Point Cloud (b) 2D3D-MATR (c) Ours

Figure 5. Correspondence visualizations on RGB-D Scenes V2.