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

Minhao Li[1*]   Zheng Qin[2,1*]   Zhirui Gao[1]   Renjiao Yi[1]   Chenyang Zhu[1]   Yulan Guo[1,3]   Kai Xu[1†]

[1]National University of Defense Technology

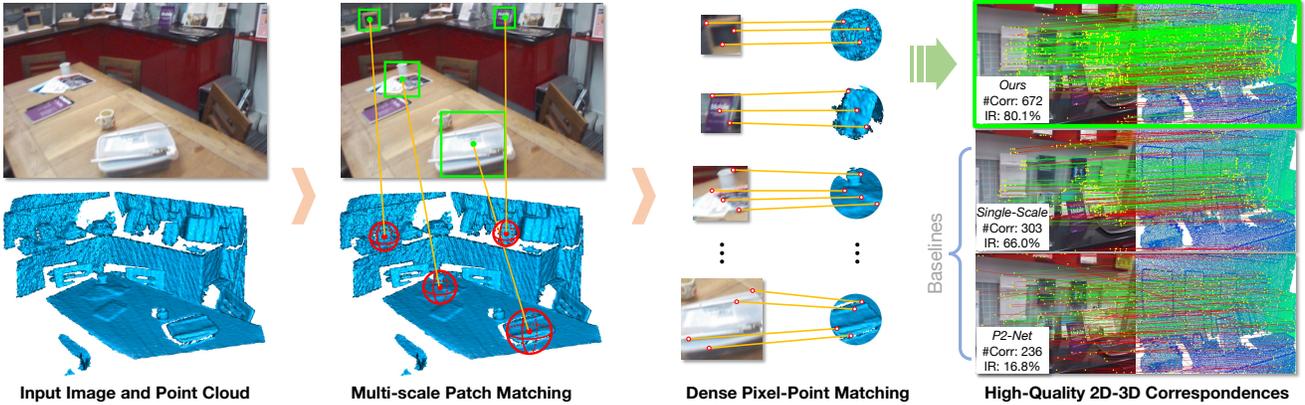[2]Defense Innovation Institute, Academy of Military Sciences   [3]Sun Yat-sen University

Figure 1: We propose 2D3D-MATR, a novel detection-free method for accurate inter-modality matching between images and point clouds. Our method adopts a coarse-to-fine pipeline where it first computes coarse correspondences between downsampled image patches and point patches and then extends them to form dense pixel-point correspondences within the patch region. A multi-scale sampling and matching scheme is devised to resolve the scale ambiguity in patch matching. Compared to detection-based P2-Net (bottom-right) and single-scale patch matching (middle-right), 2D3D-MATR (top-right) extracts significantly more accurate and dense 2D-3D correspondences. The inliers are in green and the outliers are in red.

Abstract

The commonly adopted detect-then-match approach to registration finds difficulties in the cross-modality cases due to the incompatible keypoint detection and inconsistent feature description. We propose, 2D3D-MATR, a detection-free method for accurate and robust registration between images and point clouds. Our method adopts a coarse-to-fine pipeline where it first computes coarse correspondences between downsampled patches of the input image and the point cloud and then extends them to form dense correspondences between pixels and points within the patch region. The coarse-level patch matching is based on transformer which jointly learns global contextual constraints with self-attention and cross-modality correlations with cross-attention. To resolve the scale ambiguity in patch matching, we construct a multi-scale pyramid for each image patch and learn to find for each point patch the best matching image patch at a proper resolution level. Ex-tensive experiments on two public benchmarks demonstrate that 2D3D-MATR outperforms the previous state-of-the-art P2-Net by around 20 percentage points on inlier ra-tio and over 10 points on registration recall. Our code and models are available at https://github.com/minhaolee/2D3DMATR.

1. Introduction

The inter-modality registration between images and point clouds finds applications in many computer vision tasks, e.g., 3D reconstruction, camera relocalization, SLAM and AR. It aims at estimating a rigid transformation that aligns a scene point cloud into the camera coordinates of an image capturing the same scene. The typical pipeline of 2D-3D registration is to first extract correspondences between pixels and points and then adopt robust pose estimators such as PnP-RANSAC [27, 17] to recover the alignment transfor-mation. Therefore, the accuracy of the putative correspon-dences is the crux of a successful registration.

Following the intra-modality correspondence methods for stereo images [13, 34, 46, 15] or point clouds [19, 10,

---
*Equal contribution.

†Corresponding author: kevin.kai.xu@gmail.com.

2, 22], 2D-3D matching methods [16, 39, 53] usually adopt a *detect-then-match* approach where 2D and 3D keypoints are first detected independently in the image and the point cloud, respectively, and then matched based on their associated descriptors. Such method, however, suffers from two difficulties. First, 2D and 3D keypoints are detected in different visual domains. While 2D keypoint detection is based on texture and color information, 3D detection is hinged on local geometry. This makes the detection of repeatable keypoints difficult. Second, 2D and 3D descriptors encode different visual information, which hampers extracting consistent descriptors for matching pixels and points. As a consequence, existing 2D-3D matching methods often lead to too low inlier ratio to be practically usable.

Recently, *detection-free* approach has received increasing attention in both stereo matching [43, 29, 56, 48] and point cloud registration [55, 40]. Saving the step of keypoint detection, it achieves high-quality correspondence with a *coarse-to-fine* pipeline: It first establishes coarse correspondences at the level of image or point patches and then refines them into fine-grained matching of pixels or points. This method has shown strong superiority over detection-based ones due to the exploitation of global contextual information at patch level. Such success, however, has not been attained for 2D-3D matching. This is because designing a coarse-level 2D-3D matching is non-trivial due to the scale ambiguity between image and point patches caused by perspective projection (see Fig. 1). On the one hand, the receptive fields for extracting 2D and 3D features could be misaligned, resulting in inconsistency between 2D and 3D features. On the other hand, there could be many pixels or points finding no counterpart on other side due to occlusion, leading to considerable ambiguity for fine-level matching.

We propose *2D3D-MATR*, the first, to our knowledge, detection-free method for accurate and robust 2D-3D registration via addressing the challenges above. Adapting the coarse-to-fine pipeline, our method first computes coarse correspondences between downsampled patches of the input image and the point cloud and then extends them to form dense correspondences between pixels and points within the patch regions. To achieve accurate feature alignment between image and point patches, we design a coarse-level matching module based on transformer [52] which jointly learns global contextual constraints with self-attention and cross-modality correlations with cross-attention.

Our key insight is that the feature misalignment between 2D and 3D due to projection can be resolved by *image-space multi-scale sampling and matching*, assuming that the area of local patches is small and the projection distortions is negligible. We construct a multi-scale pyramid for each image patch. During training, we find for each point patch the best matching image patch at a proper resolution level through computing the bilateral overlap between them

in the image space. During test, our model can automatically infer 2D-3D patch correspondences at a proper scale and produces dense correspondence in a high inlier ratio. Extensive experiments on the RGB-D Scenes V2 [24] and 7-Scenes [18] benchmarks demonstrate the efficacy of our method. In particular, 2D3D-MATR outperforms the previous state-of-the-art P2-Net [53] by at least 20 percentage points on inlier ratio and over 10 points on registration recall on the two benchmarks. Our contributions include:

- The first detection-free coarse-to-fine matching network for 2D-3D registration which first establishes coarse correspondences of patch level and then refines them into dense correspondences of pixel/point level.
- A transformer-based coarse matching module learning well-aligned 2D and 3D features with both global contextual constraints and cross-modality correlations.
- A multi-scale 2D-3D matching scheme that resolves 2D-3D feature misalignment through learning image-space multi-scale features and feature-scale selection.

2. Related Work

Stereo image registration. Traditional stereo image registration methods usually adopt a *detect-then-match* pipeline to extract correspondences. A set of sparse keypoints are first detected and described with hand-crafted [33, 44] or learning-based descriptors [41, 15, 13, 46, 34] from both sides, which are then matched based on feature similarity. Keypoints detection is ill-posed and detection-free methods [43, 42, 26] propose to bypass the keypoint detection step by computing a correlation matrix between all pairs of features. However, the all-pair correlation matrix requires huge computation, making the putative correspondences relatively coarse-grained. For this reason, [29, 56, 48] further propose to adopt a coarse-to-fine matching framework, which achieves accurate and efficient image matching.

Point cloud registration. Similar progress as in image registration has also been witnessed in point cloud registration. Early works leverage hand-crafted descriptors such as PPF [14] and FPFH [45] for keypoint detection. And recent learning-based descriptors [12, 11, 10, 19, 2, 22] achieve more robust and accurate matching results. To bypass the keypoint detection, CoFiNet [55] introduces the coarse-to-fine strategy to the matching of point clouds. And Geo-Transformer [40] further designs a transformation-invariant geometric structure embedding and achieves RANSAC-free point cloud registration. Moreover, there are also methods focusing on removing outlier correspondences [1, 9, 25], which act as an effective alternative of traditional robust estimators such as RANSAC [17].

Inter-modality registration. Compared to intra-modality matching problems, inter-modality matching between images and point clouds is more difficult. Based on how
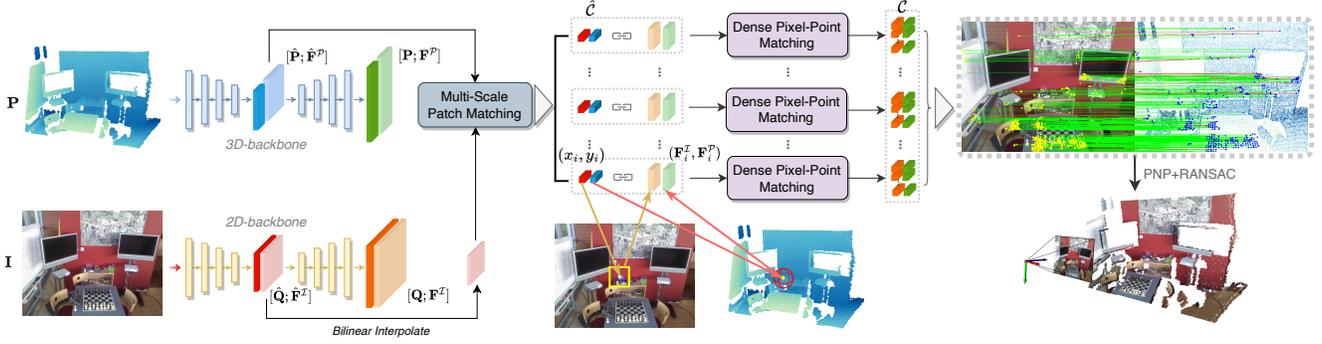
Figure 2: Overall pipeline of 2D3D-MATR. We first progressively downsample the input image $\mathbf{I}$ and the point cloud $\mathbf{P}$ and learn multi-scale 2D and 3D features. The 2D and 3D features $\hat{\mathbf{F}}^{\mathcal{I}}$ and $\hat{\mathbf{F}}^{\mathcal{P}}$ at the coarsest stage are used to extract coarse correspondences between the local patches of the image and the point cloud. A multi-scale patch matching module is devised to learn global contextual constraints and cross-modality correlations. Next, the patch correspondences are extended to dense pixel-point correspondences based on the high-resolution features $\mathbf{F}^{\mathcal{I}}$ and $\mathbf{F}^{\mathcal{P}}$. Finally, PnP-RANSAC is adopted to estimate the alignment transformation.

the correspondences are established, previous works can be classified into two categories. The first class focuses on visual localization in a known scene. The main idea of them is to predict the 3D coordinates of each image pixel with decision trees [47, 36, 37, 4, 51] or neural networks [3, 5, 6, 30, 31, 35, 54]. However, this class of methods lack generality to novel scenes. The second class follows the traditional detect-then-match pipeline [16, 39, 53], where keypoints are first detected from each modality and then matched with the associated descriptors. Compared to the first class, this class of methods have better generality theoretically. However, detecting repeatable inter-modality keypoints is much more difficult and unstable as keypoints are defined and described in different visual domains. For this reason, existing methods still suffer from low inlier ratio. In this work, we propose 2D3D-MATR to address these issues with two specific designs, *i.e.*, coarse-to-fine matching and transformer-based multi-scale patch matching.

3. Method

3.1. Overview

Given a image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ and a point cloud $\mathbf{P} \in \mathbb{R}^{N \times 3}$ of a scene, the goal of 2D-3D registration is to recover the alignment transformation $\mathbf{T}$ between them, which is composed of a 3D rotation $\mathbf{R} \in \mathcal{SO}(3)$ and a 3D translation $\mathbf{t} \in \mathbf{R}^3$. A traditional 2D-3D registration pipeline first extracts correspondences $\mathcal{C} = \{(\mathbf{x}_i, \mathbf{y}_i) \mid \mathbf{x}_i \in \mathbb{R}^3, \mathbf{y}_i \in \mathbb{R}^2\}$ between 3D points and 2D pixels, and then estimates the transformation by minimizing the 2D projection error:

$$\min_{\mathbf{R}, \mathbf{t}} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{C}} \|\mathcal{K}(\mathbf{R}\mathbf{x}_i + \mathbf{t}, \mathbf{K}) - \mathbf{y}_i\|^2, \qquad (1)$$

where $\mathbf{K}$ is the intrinsic matrix of the camera and $\mathcal{K}$ is the project function from 3D space to image plane. This prob-

lem can be effectively solved by PnP-RANSAC algorithm. However, the solution can be erroneous due to inaccurate correspondences.

In this work, we present a method to hierarchically extract inter-modality correspondences. We first adopt two respective backbones to learn features for the image and point cloud (Sec. 3.2). Next, we extract a set of coarse correspondences between the downsampled patches of the image and the point cloud (Sec. 3.3). At last, the patch correspondences are the refined to dense pixel-point correspondences on the fine level (Sec. 3.4). Fig. 2 illustrates the overall pipeline of our method.

3.2. Feature Extraction

Backbones. Given a pair of image and point cloud, two modality-specific encoder-decoder backbone networks are adopted for hierarchical feature extraction. For the image, we use a ResNet [20] with FPN [32] to generate multi-scale image features. The downsampled 2D features $\hat{\mathbf{F}}^{\mathcal{I}} \in \mathbb{R}^{\hat{H} \times \hat{W} \times \hat{C}}$ at the smallest resolution and $\mathbf{F}^{\mathcal{I}} \in \mathbb{R}^{H \times W \times C}$ at the original resolution are used for matching in coarse and fine levels. For simplicity, we denote the pixel coordinate matrices for $\hat{\mathbf{F}}^{\mathcal{I}}$ and $\mathbf{F}^{\mathcal{I}}$ as $\hat{\mathbf{Q}} \in \mathbb{R}^{\hat{H} \times \hat{W} \times 2}$ and $\mathbf{Q} \in \mathbb{R}^{H \times W \times 2}$, respectively. For the point cloud, we adopt KPFCNN [50] to learn 3D features following [2, 22, 55, 40]. Unlike images which have fixed resolutions, point clouds usually have inconsistent sizes and KPFCNN dynamically downsamples them via grid downsampling. We use the points $\hat{\mathbf{P}} \in \mathbf{R}^{\hat{N} \times 3}$ corresponding to the coarsest level and their associated features $\hat{\mathbf{F}}^{\mathcal{P}} \in \mathbb{R}^{\hat{N} \times \hat{C}}$ for coarse-level matching, while fine-level matching is conducted on the input point cloud $\mathbf{P}$ and the associated features $\mathbf{F}^{\mathcal{P}} \in \mathbb{R}^{N \times C}$.

Patch construction. To extract patch correspondences on the coarse level, we need first associate each downsampled
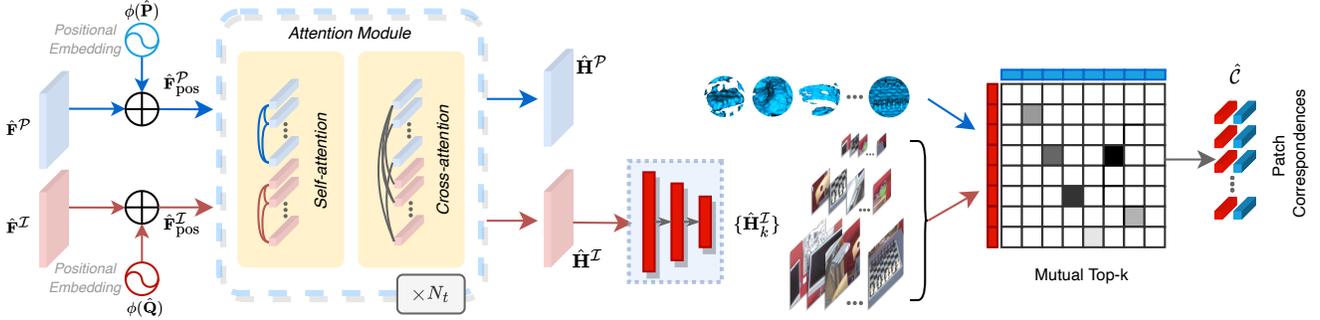
Figure 3: Multi-scale patch matching. Given the coarse 2D and 3D features, we first learn global contextual constraints with self-attention and cross-modality correlations with cross-attention. Then we adopt an image-space multi-scale sampling and matching strategy to extract patch correspondences which are better aligned in the image plane.

pixel (point) with an image (point) patch. For the image, we evenly divide $\mathbf{I}$ into $\hat{H} \times \hat{W}$ patches and each pixel in $\hat{\mathbf{F}}$ corresponds to an *image patch* of $\frac{H}{\hat{H}} \times \frac{W}{\hat{W}}$ pixels. For the point cloud, we use point-to-node partition [28] following [55, 40], which assigns each point in $\mathbf{P}$ to its nearest point in $\hat{\mathbf{P}}$ to compose the *point patches*.

3.3. Multi-scale Patch Matching

Attention-based feature refinement. Given the downsampled image $(\hat{\mathbf{Q}}, \hat{\mathbf{F}}^{\mathcal{I}})$ and point cloud $(\hat{\mathbf{P}}, \hat{\mathbf{F}}^{\mathcal{P}})$, our goal in the coarse level is to extract patch correspondences that overlap with each other. However, inter-modality matching between 2D and 3D is non-trivial. On the one hand, 2D and 3D features are learned from different domains, leading to severe inconsistency between them. This problem is more serious in patch matching than point matching as patch features are learned from a large context, which aggravates the feature misalignment. Second, as noted in [48, 55, 40], coarse-level matching relaxes the matching criterion from the strict 3D distance into a much looser local texture-geometry similarity. This effectively eases the matching difficulty but requires more global context. For this reason, we devise a transformer-based [52] feature refinement module to learn global contextual constraints and cross-modality correlations.

Before feeding into transformer, we first augment the 2D and 3D features with their positional information:

$$\hat{\mathbf{F}}^{\mathcal{I}}_{\text{pos}} = \hat{\mathbf{F}}^{\mathcal{I}} + \phi(\hat{\mathbf{Q}}), \quad \hat{\mathbf{F}}^{\mathcal{P}}_{\text{pos}} = \hat{\mathbf{F}}^{\mathcal{P}} + \phi(\hat{\mathbf{P}}), \quad (2)$$

and $\phi(\cdot)$ is the Fourier embedding function [38]:

$$\phi(x) = \left[x, \sin(2^0 x), \cos(2^0 x), ..., \sin(2^{L-1} x), \cos(2^{L-1} x) \right], \quad (3)$$

where L is the length of the embedding. We then flatten the first two spatial dimensions of the 2D features for simplicity and use $\hat{\mathbf{F}}^{\mathcal{I}}_{\text{pos}}, \hat{\mathbf{F}}^{\mathcal{P}}_{\text{pos}}$ for future computation.

Afterwards, we leverage transformer to further refine the features in two modalities. Given anchor features $\mathbf{F}^{\mathcal{A}} \in$

$\mathbb{R}^{|\mathcal{A}| \times d}$ and memory features $\mathbf{F}^{\mathcal{M}} \in \mathbb{R}^{|\mathcal{M}| \times d}$, transformer models the pairwise correlations between them with attention mechanism to generate more discriminative features. Specifically, the two set of features are first projected as:

$$\mathbf{Q} = \mathbf{F}^{\mathcal{A}} \mathbf{W}^Q, \quad \mathbf{K} = \mathbf{F}^{\mathcal{M}} \mathbf{W}^K, \quad \mathbf{V} = \mathbf{F}^{\mathcal{M}} \mathbf{W}^V, \quad (4)$$

where $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^Q \in \mathbf{R}^{d \times d}$ are the projection weights for query, key and value. The attention features for the anchor set are then computed as:

$$\texttt{Attention}(\mathbf{F}^{\mathcal{A}}, \mathbf{F}^{\mathcal{M}}) = \texttt{Softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{d^{0.5}})\mathbf{V}. \quad (5)$$

And the attention features are further projected with a shallow MLP as the final output features.

We iteratively apply self-attention and cross-attention to refine the 2D and 3D features as shown in Fig. 3. In self-attention, we use the features from the same modality as both the anchor features and memory features for attention computation to encode intra-modality global contextual constraints. In cross-attention, we use the features from one modality as the anchor features and the other modality as the memory features to learn cross-modality correlations. By this means, we can obtain refined 2D and 3D features which are more discriminative and better aligned. The resultant features are denoted as $\hat{\mathbf{H}}^{\mathcal{I}} \in \mathbb{R}^{\hat{H} \times \hat{W} \times \hat{C}}$ and $\hat{\mathbf{H}}^{\mathcal{P}} \in \mathbb{R}^{|\hat{N}| \times \hat{C}}$ in 2D and 3D modalites, respectively.

Multi-scale matching. Due to the effect of perspective projection, the objects in images have the *scale ambiguity* problem, *i.e.*, an object looks larger if it lies close to the camera and smaller if far from the camera. However, the scale of an object in the point cloud remains unchanged and is agnostic to camera motion. As a result, the 2D and 3D patches could be seriously misaligned: a 3D patch could cover many 2D patches when the camera moves close, and vice versa. Fig. 4 illustrates the misalignment between 2D and 3D patches. This causes significant ambiguous objective during training: considering two nearby point patches with different physical properties, they could be supervised
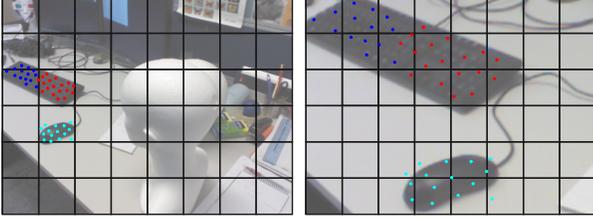
Figure 4: Scale misalignment between image patches and point patches due to perspective projection. **Left**: when the camera is far from the scene, the 3D patches on the keyboard are properly aligned with the 2D patches, and the 3D patch around the mouse is even slightly smaller than the matched 2D patch. **Right**: when camera move towards the scene, the 3D patches cover several 2D patches, leading to severe matching ambiguity.

to have similar features if covered by the same image patch. This is unexpected as it aggravates the feature misalignment and harms the distinctiveness of the features.

For this reason, we devise an image-space multi-scale sampling and matching strategy to alleviate the scale ambiguity between 2D and 3D patches. Technically, we first divide $\mathbf{I}$ into $\hat{H}_0 \times \hat{W}_0$ patches and then build a K-level patch pyramid for each image patch. At each pyramid level, the patch size is halved to generate a more fine-grained patch partition. The features of the patch pyramid is obtained by a lightweight K-stage CNN. We first downsample $\hat{\mathbf{H}}^{\mathcal{I}}$ to fit the finest patch pyramid level. The 2D features are then downsampled by a factor of 2 at each stage to match the resolutions of each patch pyramid level. For simplicity, the 2D patch features at the kth level are denoted as $\hat{\mathbf{H}}_k^{\mathcal{I}}$. At last, the multi-scale 2D patch features $\{\hat{\mathbf{H}}_k^{\mathcal{I}}\}$ and the 3D patch features $\hat{\mathbf{H}}^{\mathcal{P}}$ are normalized onto a unit hypersphere as the final features.

By leveraging the multi-scale matching strategy, for each 3D patch, we find the 2D patch that coincides the best with it on the image plane during training: the 3D patches far from the camera prefer small 2D patches in a later level, while the close ones are more likely to match with large 2D patches in a early level. Fig. 5 illustrates our multi-scale matching strategy, where our method provides 3D patches with better aligned 2D patches. This can effectively alleviate the matching ambiguity and reduce the difficulty in learning consistent 2D and 3D features. During inference, the putative patch correspondences $\hat{\mathcal{C}}$ are extracted with mutual top-k selection [40]:

$$(x_i, y_i) \text{ is matched} \Leftrightarrow \left(\hat{\mathbf{h}}_*^{\mathcal{I}}(x_i) \text{ is } k\text{NN of } \hat{\mathbf{h}}^{\mathcal{P}}(y_i)\right) \wedge \\ \left(\hat{\mathbf{h}}^{\mathcal{P}}(y_i) \text{ is } k\text{NN of } \hat{\mathbf{h}}_*^{\mathcal{I}}(x_i)\right) \quad (6)$$

3.4. Dense Pixel-Point Matching

After obtaining the patch correspondences, we further refine them to dense pixel-point correspondences. For each
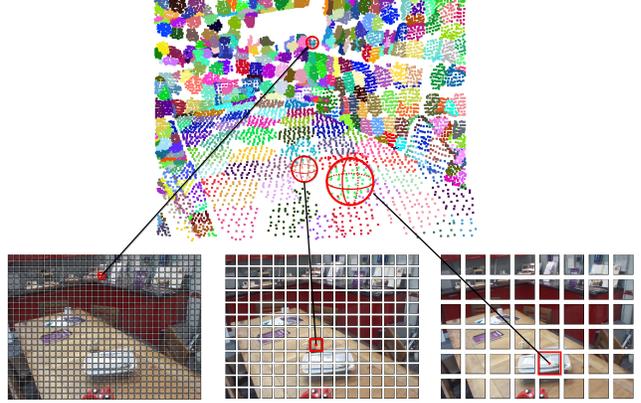

Figure 5: Multi-scale patch matching based on image-space patch pyramid with 3 levels. One matched patch pair is shown in each pyramid level. The 3D patches far from the camera are matched to small 2D patches in a later level, while the close ones are matched to large 2D patches in a early level.

$(x_i, y_i) \in \hat{\mathcal{C}}$, we collect the fine-level 2D and 3D features of its local pixels and points from $\mathbf{F}^{\mathcal{I}}$ and $\mathbf{F}^{\mathcal{P}}$, denoted as $\mathbf{F}_i^{\mathcal{I}}$ and $\mathbf{F}_i^{\mathcal{P}}$. For computational efficiency, we uniformly sample $1/4$ of the pixels in each 2D patch. Following Sec. 3.3, we normalize $\mathbf{F}_i^{\mathcal{I}}$ and $\mathbf{F}_i^{\mathcal{P}}$ to unit-length vectors and match the pixels and points with mutual top-k selection as the *local dense correspondences* of (x_i, y_i). We do not adopt a specific matching layer such as Sinkhorn [46, 55, 40] here as the 2D patches in large scales could have enormous pixels (*e.g.*, 1600 pixels in our experiments), which causes unacceptable computational cost. On the contrary, mutual top-k selection is very efficient and still achieves promising performance. At last, we gather the local correspondences of all (x_i, y_i) from $\hat{\mathcal{C}}$ as the final dense pixel-point correspondences. Note that as the 2D patches from different scales can overlap with each other, we explicitly remove the repeated correspondences from the final correspondences.

3.5. Loss Functions

Our model is trained in a metric learning fashion. On the coarse level, we adopt a scaled circle loss [49, 40] to supervise the patch features. On the fine level, another standard circle loss [49] is used to supervise the dense pixel and point features. The overall loss is then computed as $\mathcal{L}_{\text{all}} = \mathcal{L}_{\text{coarse}} + \lambda \mathcal{L}_{\text{fine}}$, where $\lambda = 1$ is a balance factor.

Compared to contrastive loss [8] and triplet loss [21], circle loss [49] has a circular decision boundary which facilitates convergence. Given an anchor descriptor $\mathbf{d}_i$, the descriptors of its positive and negative pairs are denoted as $\mathcal{D}_i^{\mathcal{P}}$ and $\mathcal{D}_i^{\mathcal{N}}$. The general circle loss on $\mathbf{d}_i$ is computed as:

$$\mathcal{L}_i = \frac{1}{\gamma} \log\Big[1 + \sum_{\mathbf{d}_j \in \mathcal{D}_i^{\mathcal{P}}} e^{\beta_p^{i,j}(d_i^j - \Delta_p)} \cdot \sum_{\mathbf{d}_k \in \mathcal{D}_i^{\mathcal{N}}} e^{\beta_n^{i,k}(\Delta_n - d_i^k)}\Big], \quad (7)$$

where d_i^j is the ℓ_2 feature distance, $\beta_p^{i,j} = \gamma\lambda_p^{i,j}(d_i^j - \Delta_p)$ and $\beta_n^{i,k} = \gamma\lambda_n^{i,k}(\Delta_n - d_i^k)$ 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.

On the coarse level, we generate the ground truth based on the bilateral overlap. A patch pair is regarded as positive if the 2D and 3D overlap ratios between them are both at least 30%, and as negative if both the overlap ratios are below 20%. Please refer to Sec. 4.1 for more details. The overlap ratio between the 2D and 3D patches are used as λ_p, and λ_n is set to 1. On the fine level, a pixel-point pair is positive with the 3D distance is below 3.75cm and the 2D distance is below 8 pixels, while being negative with a 3D distance above 10cm or a 2D distance above 12 pixels. The scaling factors are all 1. We ignore all other pairs on both levels during training as the safe region. The margins are set to $\Delta_p = 0.1$ and $\Delta_n = 1.4$ following [22, 40].

4. Experiments

As there is no public 2D-3D registration benchmark, we build two challenging benchmarks based on RGB-D Scenes V2 [24] (Sec. 4.2) and 7Scenes [18] (Sec. 4.3) datasets, and evaluate the efficacy of 2D3D-MATR on them. Extensive ablation studies are provided to study the influence of different design choices (Sec. 4.4).

4.1. Implementation Details

Network architecture. We adopt a 4-stage ResNet [20] with FPN as the image backbone network, where the output channels of each stage are $\{128, 128, 256, 512\}$. The resolution of the input images is 480×640 and is downsampled to 60×80 in the coarsest level. For the 3D backbone, we use a 4-stage KPFCNN [50] where the output channels of each stage are $\{128, 256, 512, 1024\}$. The point clouds are voxelized with an initial voxel size of 2.5cm which is doubled at each stage. In the coarse level, we resize the 2D features to 24×32 before feeding them to the transformer for computational efficiency. All the transformer layers have 256 features channels with 4 attention heads and ReLU activation. In the patch pyramid, we use $H_0 = 6$ and $W_0 = 8$ in the coarsest level and build $K = 3$ pyramid levels, *i.e.*, $\{6 \times 8, 12 \times 16, 24 \times 32\}$. In the fine level, we project both the 2D and 3D features to 128-d for feature matching.

Metrics. We mainly evaluate the models with 3 metrics: (1) *Inlier Ratio* (IR), the ratio of pixel-point matches whose 3D distance is below a certain threshold (*i.e.*, 5cm) over all putative matches. (2) *Feature Matching Recall* (FMR), the ratio of image-point-cloud pairs whose inlier ratio is above a certain threshold (*i.e.*, 10%). (3) *Registration Recall* (RR), the ratio of image-point-cloud pairs whose RMSE is below a certain threshold (*i.e.*, 10cm).

Baselines. We mainly compare to 3 keypoint detection-based baseline methods: (1) FCGF2D3D, a 2D-3D imple-

Model	Scene-11	Scene-12	Scene-13	Scene-14	Mean
Mean depth (m)	1.74	1.66	1.18	1.39	1.49
Inlier Ratio ↑					
FCGF-2D3D [10]	6.8	8.5	11.8	5.4	8.1
P2-Net [53]	9.7	12.8	17.0	<u>9.3</u>	12.2
Predator-2D3D [22]	<u>17.7</u>	<u>19.4</u>	<u>17.2</u>	8.4	<u>15.7</u>
2D3D-MATR (*ours*)	**32.8**	**34.4**	**39.2**	**23.3**	**32.4**
Feature Matching Recall ↑					
FCGF-2D3D [10]	11.1	30.4	51.5	15.5	27.1
P2-Net [53]	48.6	65.7	<u>82.5</u>	<u>41.6</u>	59.6
Predator-2D3D [22]	<u>86.1</u>	<u>89.2</u>	63.9	24.3	<u>65.9</u>
2D3D-MATR (*ours*)	**98.6**	**98.0**	**88.7**	**77.9**	**90.8**
Registration Recall ↑					
FCGF-2D3D [10]	26.4	41.2	37.1	16.8	30.4
P2-Net [53]	40.3	40.2	<u>41.2</u>	<u>31.9</u>	<u>38.4</u>
Predator-2D3D [22]	<u>44.4</u>	<u>41.2</u>	21.6	13.7	30.2
2D3D-MATR (*ours*)	**63.9**	**53.9**	**58.8**	**49.1**	**56.4**

Table 1: Evaluation results on RGB-D Scenes V2. **Bold-faced** numbers highlight the best and the second best are <u>underlined</u>.

mentation of FCGF [10] which samples random keypoints from the image and the point cloud. (2) Predator2D3D, a 2D-3D implementation of Predator [22] which leverages a graph network to learn the saliency of each pixel (point) for sampling keypoints. (3) P2-Net [53], a 2D-3D correspondence method which detects locally salient pixels (points) in the feature space. Note that albeit successful in point cloud registration, we find that Predator-2D3D fails to predict reliable saliency scores in 2D-3D scenarios. To this end, we ignore the saliency scores in Predator-2D3D and randomly sample keypoints according to the predicted overlap scores. For fair comparison, we use the same backbones for all the methods. Please refer to Appx. A for more details.

4.2. Evaluations on RGB-D Scenes V2

Dataset. *RGB-D Scenes V2* [24] contains 11427 RGB-D frames from 14 indoor scenes. For each scene, we fuse a point cloud fragment with every 25 consecutive depth frames and sample a RGB image every 25 frames. We select the image-point-cloud pairs with an overlap ratio of at least 30%. 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.

Quantative results. We first compare our method to the baselines on RGB-D Scenes V2 in Tab. 1. For *Inlier Ratio*, P2-Net outperforms FCGF-2D3D benefiting from the feature saliency-based keypoint detection. However, it still suffers from low inlier ratio. And albeit achieving better inlier ratio on the first two scenes, Predator-2D3D performs worse in the later two scenes where the camera is closer to the scene. On the contrary, thanks to the coarse-to-fine matching pipeline and the multi-scale patch pyramid, our 2D3D-MATR significantly improves the inlier ratio by 20

Model	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs	Mean
Mean depth (m)	1.78	1.55	0.80	2.03	2.25	2.13	1.84	1.77
Inlier Ratio ↑								
FCGF-2D3D [10]	34.2	32.8	14.8	26.0	23.3	22.5	6.0	22.8
P2-Net [53]	55.2	46.7	13.0	36.2	32.0	32.8	5.8	**31.7**
Predator-2D3D [22]	34.7	33.8	16.6	25.9	23.1	22.2	7.5	23.4
2D3D-MATR (*ours*)	**72.1**	**66.0**	**31.3**	**60.7**	**50.2**	**52.5**	**18.1**	**50.1**
Feature Matching Recall ↑								
FCGF-2D3D [10]	99.7	98.2	69.9	97.1	83.0	87.7	16.2	78.8
P2-Net [53]	100.0	99.3	58.9	99.1	87.2	92.2	16.2	79.0
Predator-2D3D [22]	91.3	95.1	76.7	88.6	79.2	80.6	31.1	77.5
2D3D-MATR (*ours*)	100.0	99.6	98.6	100.0	92.4	95.9	58.1	92.1
Registration Recall ↑								
FCGF-2D3D [10]	89.5	79.7	19.2	85.9	69.4	79.0	6.8	61.4
P2-Net [53]	96.9	86.5	20.5	91.7	75.3	85.2	4.1	65.7
Predator-2D3D [22]	69.6	60.7	17.8	62.9	56.2	62.6	9.5	48.5
2D3D-MATR (*ours*)	96.9	90.7	52.1	95.5	80.9	86.1	28.4	75.8

Table 2: Evaluation results on 7Scenes. **Boldfaced** numbers highlight the best and the second best are underlined.

Model	PIR	IR	FMR	RR
(a.1) 2D3D-MATR (*full*)	48.5	**32.4**	**90.8**	**56.4**
(a.2) 2D3D-MATR w/o coarse-to-fine	-	11.2	52.2	34.6
(a.1) 2D3D-MATR (*full*)	48.5	**32.4**	90.8	**56.4**
(b.2) 2D3D-MATR w/o self-attention	45.9	29.0	**91.8**	44.0
(b.3) 2D3D-MATR w/o cross-attention	50.4	29.3	89.1	47.7
(b.4) 2D3D-MATR w/o attention	37.0	23.1	87.0	42.3
(c.1) 2D3D-MATR (*full*)	48.5	**32.4**	90.8	**56.4**
(c.2) 2D3D-MATR w/ (24 × 32)	37.7	29.2	88.3	36.9
(c.3) 2D3D-MATR w/ (12 × 16)	44.2	29.9	89.2	51.7
(c.4) 2D3D-MATR w/ (6 × 8)	41.7	23.6	87.7	50.2
(c.5) 2D3D-MATR w/ (24 × 32, 12 × 16)	46.1	32.2	90.5	54.5
(c.6) 2D3D-MATR w/ (24 × 32, 6 × 8)	42.3	31.6	90.0	51.3
(c.7) 2D3D-MATR w/ (12 × 16, 6 × 8)	**49.8**	30.9	90.1	54.2

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

percentage points (pp). And this advantage further contributes to much higher *Feature Matching Recall*, where our method surpasses the second best P2-Net by over 24 pp.

For the most important *Registration Recall*, P2-Net achieves the best results among the three detection-based baselines. And our method outperforms P2-Net by 18 pp on registration recall thanks to the more accurate correspondences. These results have demonstrated the strong generality of our method to unseen scenes.

4.3. Evaluations on 7Scenes

Dataset. *7-Scenes* [18] consists of 46 RGB-D sequences from 7 indoor scenes. We use the same method as above to prepare the image-point-cloud pairs from each scene and preserve the pairs that share at least 50% overlap. We follow the official sequence split to generate the training, validation and testing data, which makes 4048 training pairs, 1011 validation pairs and 2304 testing pairs. Note that compared to the benchmark used in [53], we provide a more challenging one with richer viewpoint changes and smaller overlap ratios. For the evaluation results under the setting of [53], please refer to Appx. D.

Quantative results. In contrast with Sec. 4.2, we evaluate the generality to unseen viewpoints in known scenes on 7-Scenes. The results are demonstrated in Tab. 2. For *Inlier Ratio*, our method outperforms the second best P2-Net by over 18 pp. For *Feature Matching Recall*, 2D3D-MATR achieves an average improvement of 13.1 pp. And our method surpasses the baseline methods by at least 10 pp on *Registration Recall*. More surprisingly, Predator-2D3D performs the worst on 7-Scenes. As the image-point-cloud pairs in 7-Scenes commonly share more overlap, we assume that explicitly predicting the overlap scores contributes to little benefit but harms the distinctiveness of the learned feature representations.

Compared to RGB-D Scenes V2, 7-Scenes have more significant scale variations across different scenes. Nevertheles, our method still outperforms the baseline methods by a large margin, demonstrating the strong robustness of 2D3D-MATR to scale variance. It is noteworthy that 2D3D-MATR achieves more significant improvements on the two hard scenes, *i.e.*, *heads* and *stairs*. On the one hand, the camera is much closer to the scene surfaces in *heads* than in other scenes. This causes great difficulty to extract accurate correspondences as a small error in 3D space could be amplified on the image plane. On the other hand, *stairs* contains numerous repeated patterns which is hard to distinguish. Thanks to our multi-scale patch pyramid and coarse-to-fine matching strategy, our method can better handle these hard cases.

Qualitative results. Fig. 6 visualizes the extracted correspondences from P2-Net and 2D3D-MATR. We also show the single-scale version of 2D3D-MATR where 24 × 32 image patches are used. Our method extracts more accurate and more thoroughly distributed correspondences over the whole scene, which is crucial for successful registration. The last two rows shows two difficult cases from *heads* and *stairs*. In the 3rd row, P2-Net fails to detect reliable keypoints and thus suffers from low inlier ratio. Due to a near placement of the camera, the single-scale version of 2D3D-MATR can only extract the correspondences in the distant background areas. On the contrary, benefiting from multi-scale patch pyramid, full 2D3D-MATR extracts much more accurate correspondences distributed over the whole scene. And the 4th row contains repeated patterns distributed from near to far. P2-Net detects keypoints near the boundaries but fails to match them correctly. Benefiting from the global contextual contraints and cross-modality correlations learned from the transformer module, 2D3D-MATR extracts more accurate correspondences from the stairs. Please refer to Appx. D for more visualizations.

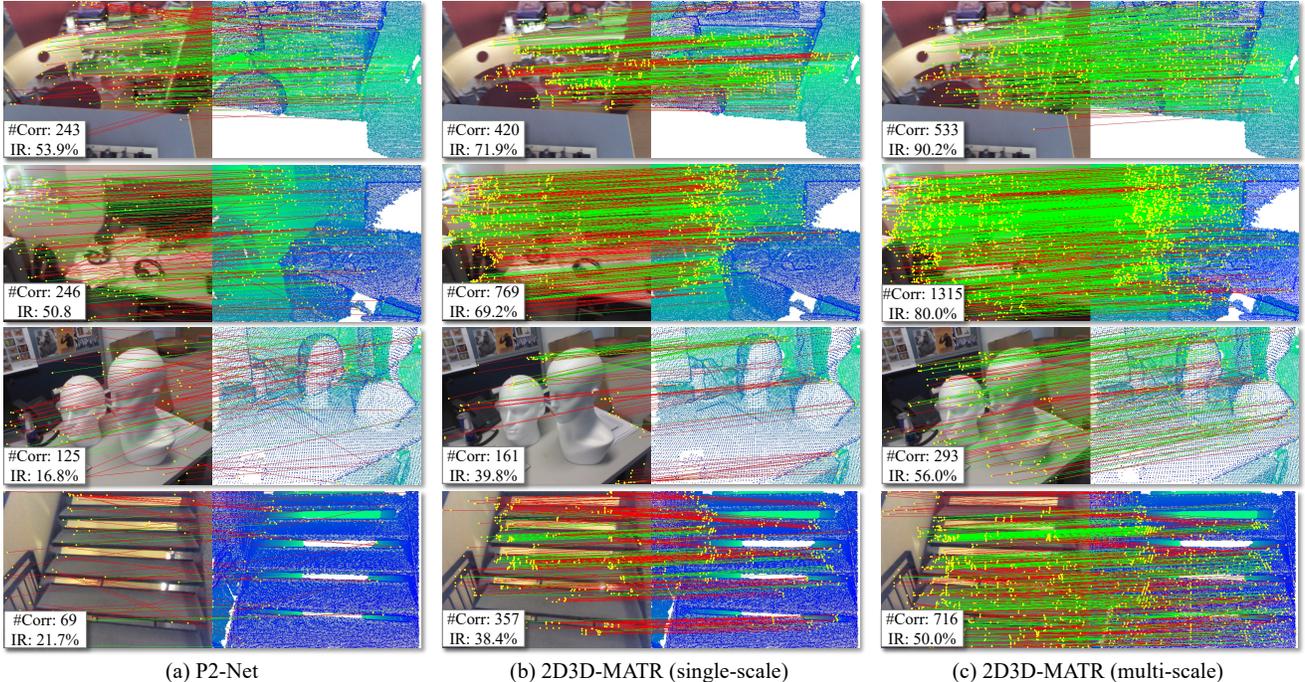| (a) P2-Net | (b) 2D3D-MATR (single-scale) | (c) 2D3D-MATR (multi-scale) |

Figure 6: Comparisons of correspondences on 7-Scenes. Our method extracts more accurate and more thoroughly distributed correspondences over the whole scene. And it extracts accurate correspondences from repeated patterns (see the 4th row).

4.4. Ablation Studies

We further conduct extensive ablation studies to investigate the efficacy of our designs on RGB-D Scenes V2. Following [40], we report another metric *Patch Inlier Ratio* (PIR), the ratio of patch correspondences whose overlap ratios are above a certain threshold (*i.e.*, 0.3), to evaluate the performance on the coarse level.

Coarse-to-fine matching. First, we ablate the coarse-level matching step in our pipeline and match randomly sampled keypoints from both sides as correspondences. In this model, we apply the attention-based feature refinement module between the encoders and the decoders. As shown in Tab. 3(a), the performance drops significantly without the coarse-to-fine matching pipeline. Compared to strict pixel-point matching, patch matching is more robust and reliable as more context could be leveraged. This effectively reduces the searching space during matching, and facilitates extracting accurate correspondences.

Feature refinement module. Next, we study the influence of the attention-based feature refinement in Tab. 3(b). We first remove the self-attention modules and the cross-attention modules in the network. The model without self-attention suffers from more serious performance degradation, which means global context plays a more important role than cross-modal aggregation in 2D-3D registration. We then completely remove all attention modules, which further degrades the performance.

Multi-scale patch pyramid. At last, we evaluate the effici-

cay of the multi-scale patch pyramid in Tab. 3(c). We progressively ablate each resolution level from our full model and evaluate the performance. Obviously, the models with one single resolution perform worse than the multi-scale models, demonstrating the effectiveness of our design. And note that the inlier ratios of the models with small resolution are lower. This is because the image patches in these models are larger and thus leads to more matching ambiguity.

5. Conclusion

We have presented 2D3D-MATR to hierarchically extract pixel-point correspondences for inter-modality registration between images and point clouds. Benefiting from a coarse-to-fine matching pipeline, our method bypasses the need of keypoint detection across two modalities. We further construct a multi-scale patch pyramid to alleviate the scale ambiguity during patch matching. These designs significantly improve the quality of the extracted correspondences and contribute to accurate 2D-3D registration. A potential limitation of our method is that it still relies on RANSAC for successful registration. In the future, we would like to extend our method for RANSAC-free inter-modality registration.

A. Implementation Details

We mainly compare to three baseline methods in the experiments: (1) FCGF-2D3D, a 2D-3D implementation of FCGF [10]; (2) P2-Net [53], a 2D-3D version of D2-Net [15] and D3Feat [2]; (3) Predator-2D3D, a 2D-3D version of Predator [22]. For FCGF-2D3D, we supervise the descriptors using circle loss [49] instead of the hardest-in-batch contrastive loss used in [10]. 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 [2] to supervise the detection scores because we find the model fails to converge on our benchmarks using the original detection loss in [53]. 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 [52] instead of the graph network in [22] 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 [20] with FPN [32] backbone for images and a 4-stage KPFCNN [50] 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 [48], 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 [50]. The detailed architecture of our method is illustrated in Fig. 7. And we use the same training settings in all the methods. We use Adam [23] 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 [7] with 5000 iterations and the distance tolerance of 8.0.

B. Metrics

Following [53], 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 [53], a correspondence is an inlier if their *3D distance* is below

$\tau_1 = 5$cm under the ground-truth transformation $\mathbf{T}^*_{\mathbf{P} \to \mathbf{I}}$

$$\text{IR} = \frac{1}{|\mathcal{C}|} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{C}} \llbracket \| \mathbf{T}^*_{\mathbf{P} \to \mathbf{I}}(\mathbf{x}_i) - \mathcal{K}^{-1}(\mathbf{y}_i) \|_2 < \tau_1 \rrbracket, \quad (8)$$

where $\llbracket \cdot \rrbracket$ 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:

$$\text{FMR} = \frac{1}{M} \sum_{i=1}^{M} \llbracket \text{IR}_i > \tau_2 \rrbracket, \quad (9)$$

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:

$$\text{RMSE} = \sqrt{\frac{1}{|\mathbf{P}|} \sum_{\mathbf{p}_i \in \mathbf{P}} \| \mathbf{T}_{\mathbf{P} \to \mathbf{I}}(\mathbf{p}_i) - \mathbf{T}^*_{\mathbf{P} \to \mathbf{I}}(\mathbf{p}_i) \|_2^2}, \quad (10)$$

$$\text{RR} = \frac{1}{M} \sum_{i=1}^{M} \llbracket \text{RMSE}_i < \tau_3 \rrbracket. \quad (11)$$

We further report *Patch Inlier Ratio* (PIR) in the ablation studies to evaluate the accuracy of the patch matching following [40]. 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 [24] and 7Scenes [18] datasets.

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
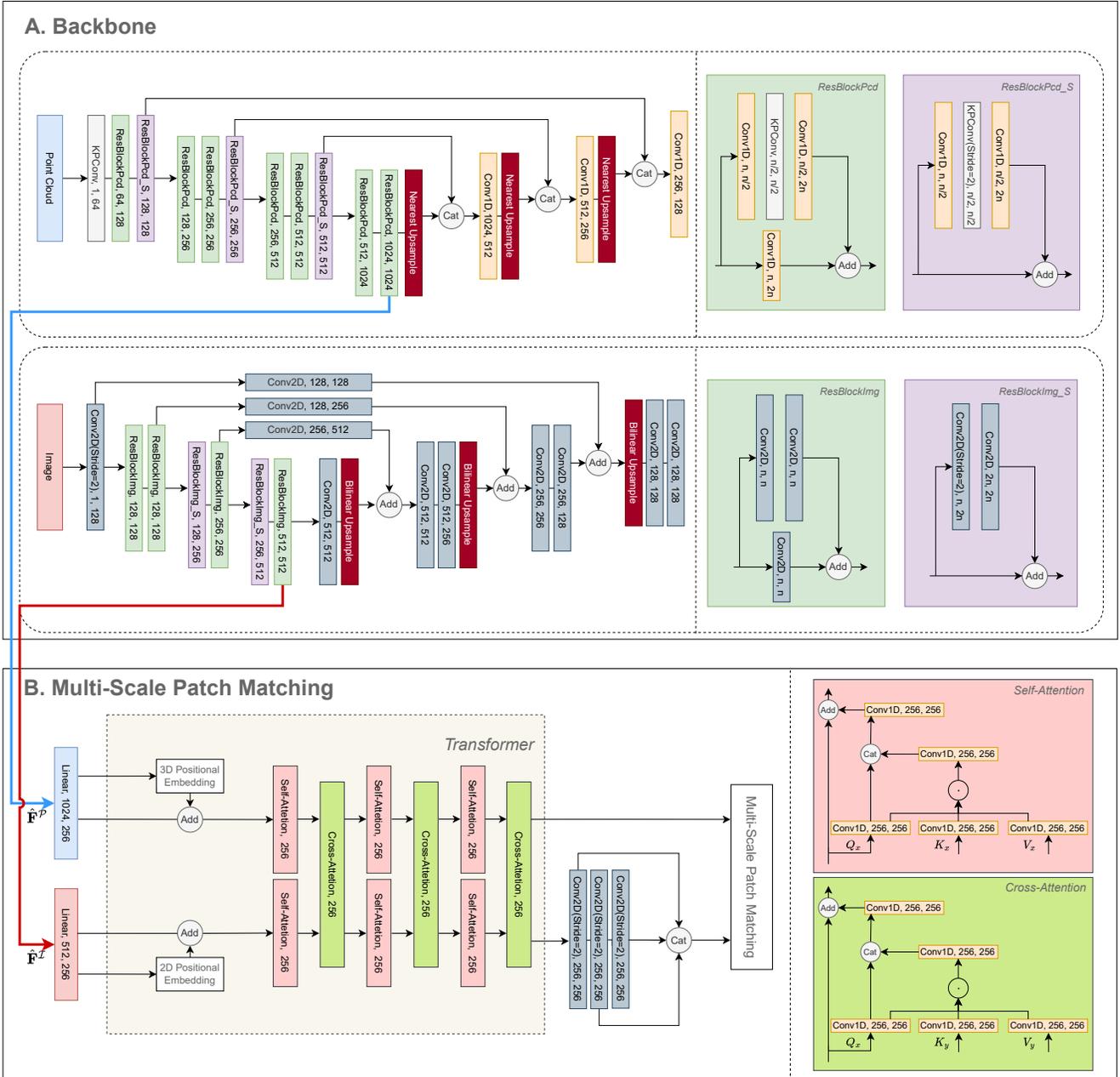
Figure 7: Network architecture.

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. 4 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.

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 4: 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 5: Statistics on the testing set of 7-Scenes.

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 [18, 3, 53] 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 [54] to rescale the image by $\frac{585}{525}$ for an approximate calibration. Tab. 5 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.

D. Additional Experiments

D.1. Additional Ablation Studies

Patch pyramid. In Tab. 6, 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. 4, the camera is closer to the scene in Scene-13 and Scene-14, which could cause severe inconsistency between the image patches and the point patches. By leveraging the patch pyramid, the scale ambiguity is alleviated such that more accurate correspondences are obtained.

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

Model	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs	Mean
Inlier Ratio ↑								
FCGF-2D3D [10]	59.2	58.5	67.5	54.4	45.0	51.6	33.5	52.8
P2-Net [53]	60.9	66.9	66.1	55.8	57.0	56.1	42.4	57.9
Predator-2D3D [22]	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 [10]	81.8	81.0	91.0	67.5	41.7	52.3	10.5	60.8
P2-Net [53]	82.5	93.0	89.5	70.6	76.2	64.6	22.5	71.3
Predator-2D3D [22]	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 [10]	99.8	98.0	98.0	97.0	89.2	96.7	94.5	96.2
P2-Net [53]	99.8	98.0	96.0	98.1	91.7	97.2	93.0	96.3
Predator-2D3D [22]	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 7: Evaluation results on 7Scenes following the experiment settings in [53]. **Boldfaced** numbers highlight the best and the second best are underlined.

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.

Mutual top-k selection. We replace the mutual top-k selection in the point matching module with the non-mutual version on RGB-D Scenes V2, which achieves 31.7% IR (0.7 pp↓), 91.6% FMR (0.8 pp↑) and 50.8% RR (5.6 pp↓). We also note that the model with non-mutual top-k selection still beats all the baselines, demonstrating the effectiveness of our method.

D.2. Additional Evaluations on 7-Scenes

We further present the evaluation results on 7-Scenes [18] following the settings in [53]. 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 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 $\tau_1 = 4.5\text{cm}$, $\tau_2 = 50\%$ and $\tau_3 = 5\text{cm}$ following [53].

The results are shown in Tab. 7. 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 [53] and 2D3D-MATR on 7-Scenes (Fig. 8) and RGB-D Scenes V2 (Fig. 9). 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.

E. Limitations

Albeit achieving the new state-of-the-art preformance, 2D3D-MATR could have the following three limitations.

First, despite of higher inlier ratio, our method still rely on PnP-RANSAC to estimate the alignment transformation. Compared to point cloud registration, the 2D errors of the 2D-3D correspondences are sensitive to the camera pose. For instance, given two points which are 5cm away from each other in the 3D space, their distance in the image plane could be merely 2 pixels if they are far from the camera but up to tens of pixels if they are close to the camera. For this reason, it is more difficult to achieve accurate registration and thus PnP-RANSAC is still necessary.

Second, we find that the generality of 2D-3D matching to novel scenes is not as good as that of image matching or point cloud matching. This can be observed by comparing the results on RGB-D Scenes V2 and 7-Scenes, where the former is worse. We assume the reason is that intermodality matching is more difficult than intra-modality matching as one need project the visual information from different domains to a common feature space.

Third, the uniform patch partition strategy in our method is relatively simple and coarse. Although we design a multiscale patch pyramid mechanism to handle scale ambiguity, the patches are still not perfectly aligned in most cases. This could cause difficulty in learning consistent features for the patches, and increase redundancy in the fine-level matching. A possible solution is to leverage semantic information to extract patches, which we will leave as future work.

References

[1] Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. Pointdsc: Robust point cloud registration using deep spatial consistency. In *CVPR*, pages 15859–15869, 2021. 2

[2] 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. 2, 3, 9

[3] 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, 11

[4] Eric Brachmann, Frank Michel, Alexander Krull, Michael Ying Yang, Stefan Gumhold, et al. Uncertainty-driven 6d pose estimation of objects and scenes from a single rgb image. In *CVPR*, pages 3364–3372, 2016. 3

[5] Eric Brachmann and Carsten Rother. Learning less is more-6d camera localization via 3d surface regression. In *CVPR*, pages 4654–4662, 2018. 3

[6] Eric Brachmann and Carsten Rother. Neural-guided ransac: Learning where to sample model hypotheses. In *CVPR*, pages 4322–4331, 2019. 3

[7] G. Bradski. The opencv library. *Dr. Dobb's Journal of Software Tools*, 2000. 9

[8] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, volume 1, pages 539–546. IEEE, 2005. 5

[9] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *CVPR*, pages 2514–2523, 2020. 2

[10] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *ICCV*, pages 8958–8966, 2019. 2, 6, 7, 9, 11

[11] Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In *ECCV*, pages 602–618, 2018. 2

[12] Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppfnet: Global context aware local features for robust 3d point matching. In *CVPR*, pages 195–205, 2018. 2

[13] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, pages 224–236, 2018. 1, 2

[14] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *CVPR*, pages 998–1005. Ieee, 2010. 2
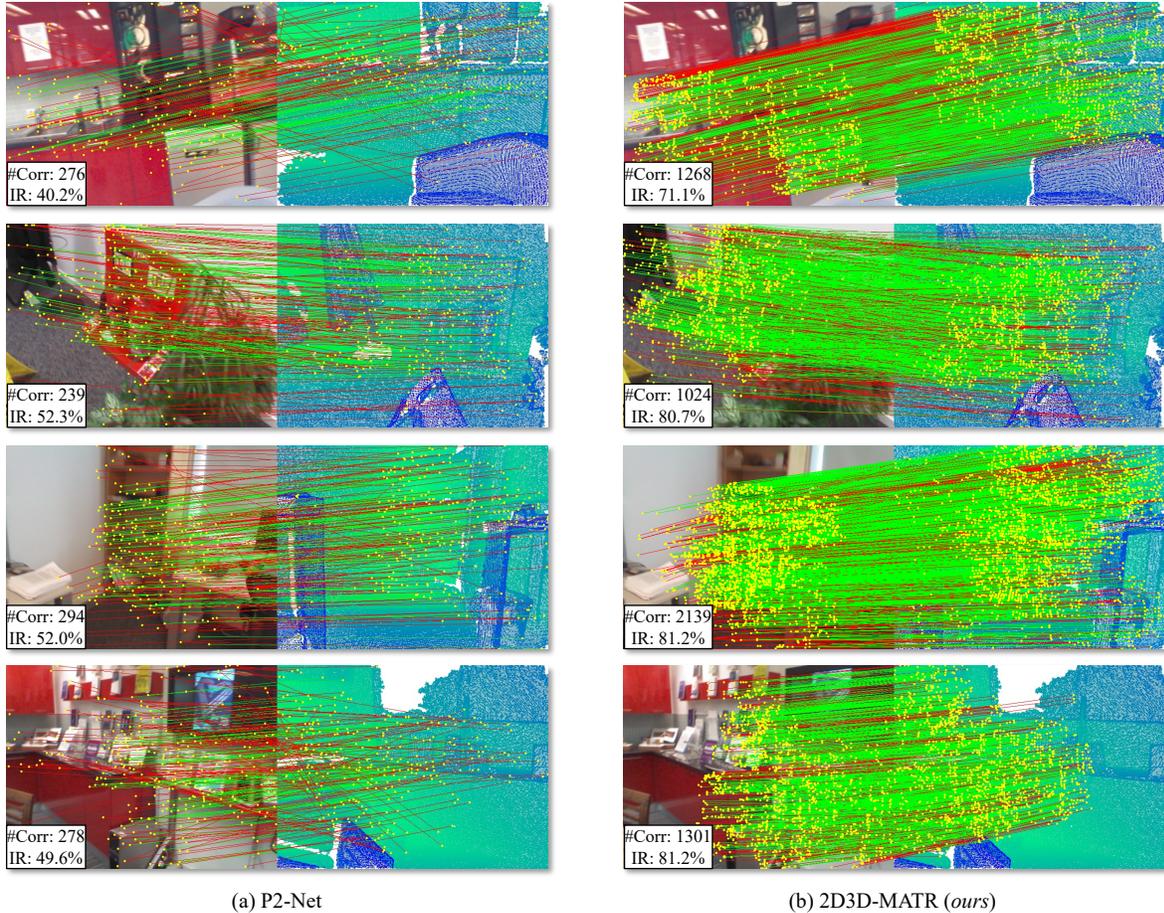
(a) P2-Net	(b) 2D3D-MATR (*ours*)

Figure 8: Comparisons of extracted correspondences on 7-Scenes.

[15] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Polle-feys, 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, 2, 9

[16] Mengdan Feng, Sixing Hu, Marcelo H Ang, and Gim Hee Lee. 2d3d-matchnet: Learning to match keypoints across 2d image and 3d point cloud. In *ICRA*, pages 4790–4796. IEEE, 2019. 2, 3

[17] 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, 2

[18] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *ISMAR*, pages 173–179. IEEE, 2013. 2, 6, 7, 9, 11

[19] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *CVPR*, pages 5545–5554, 2019. 2

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3, 6, 9

[21] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. *arXiv preprint arXiv:1412.6622*, 2014. 5

[22] 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. 2, 3, 6, 7, 9, 11

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

[24] Kevin Lai, Liefeng Bo, and Dieter Fox. Unsupervised feature learning for 3d scene labeling. In *ICRA*, pages 3050–3057. IEEE, 2014. 2, 6, 9

[25] Junha Lee, Seungwook Kim, Minsu Cho, and Jaesik Park. Deep hough voting for robust global registration. In *ICCV*, pages 15994–16003, 2021. 2

[26] Jae Yong Lee, Joseph DeGol, Victor Fragoso, and Sudipta N Sinha. Patchmatch-based neighborhood consensus for semantic correspondence. In *CVPR*, pages 13153–13163, 2021. 2

[27] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *IJCV*, 81(2):155–166, 2009. 1

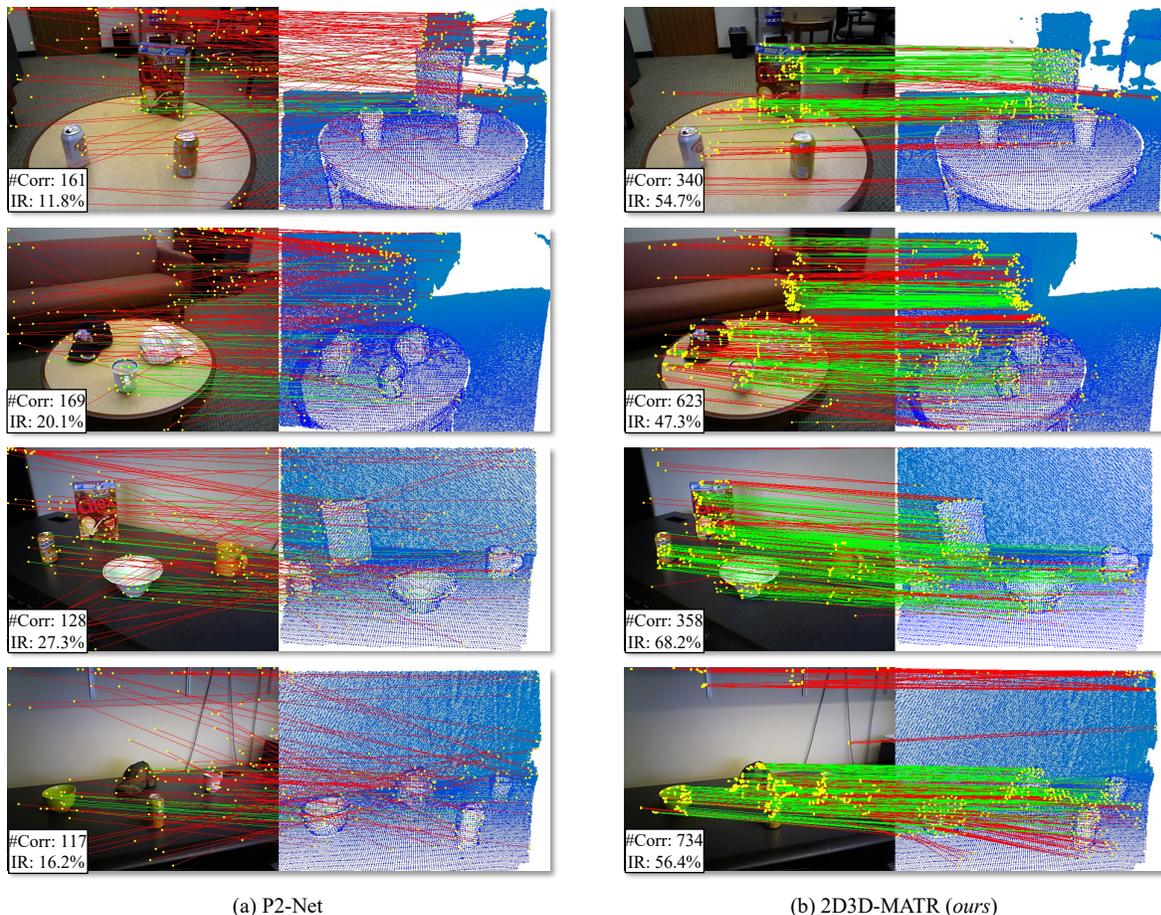|	
(a) P2-Net	(b) 2D3D-MATR (*ours*)

Figure 9: Comparisons of extracted correspondences on RGB-D Scenes V2.

[28] Jiaxin Li, Ben M Chen, and Gim Hee Lee. So-net: Self-organizing network for point cloud analysis. In *CVPR*, pages 9397–9406, 2018. 4

[29] Xinghui Li, Kai Han, Shuda Li, and Victor Prisacariu. Dual-resolution correspondence networks. *NeurIPS*, 33:17346–17357, 2020. 2

[30] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. Hierarchical scene coordinate classification and regression for visual localization. In *CVPR*, pages 11983–11992, 2020. 3

[31] Xiaotian Li, Juha Ylioinas, and Juho Kannala. Full-frame scene coordinate regression for image-based localization. *arXiv preprint arXiv:1802.03237*, 2018. 3

[32] 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. 3, 9

[33] David G Lowe. Object recognition from local scale-invariant features. In *ICCV*, volume 2, pages 1150–1157. Ieee, 1999. 2

[34] Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. Aslfeat: Learning local features of accurate shape and localization. In *CVPR*, pages 6589–6598, 2020. 1, 2

[35] Daniela Massiceti, Alexander Krull, Eric Brachmann, Carsten Rother, and Philip HS Torr. Random forests versus neural networks—what's best for camera localization? In *ICRA*, pages 5118–5125. IEEE, 2017. 3

[36] Lili Meng, Jianhui Chen, Frederick Tung, James J Little, Julien Valentin, and Clarence W de Silva. Backtracking regression forests for accurate camera relocalization. In *IROS*, pages 6886–6893. IEEE, 2017. 3

[37] Lili Meng, Frederick Tung, James J Little, Julien Valentin, and Clarence W de Silva. Exploiting points and lines in regression forests for rgb-d camera relocalization. In *IROS*, pages 6827–6834. IEEE, 2018. 3

[38] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, pages 405–421, 2020. 4

[39] Quang-Hieu Pham, Mikaela Angelina Uy, Binh-Son Hua, Duc Thanh Nguyen, Gemma Roig, and Sai-Kit Yeung. Lcd: Learned cross-domain descriptors for 2d-3d matching. In *AAAI*, volume 34, pages 11856–11864, 2020. 2, 3

[40] 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, 3, 4, 5, 6, 8, 9

[41] Jerome Revaud, Philippe Weinzaepfel, César De Souza, and Martin Humenberger. R2d2: repeatable and reliable detector and descriptor. In *NeurIPS*, pages 12414–12424, 2019. 2

[42] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. Efficient neighbourhood consensus networks via submanifold sparse convolutions. In *ECCV*, pages 605–621. Springer, 2020. 2

[43] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. *NeurIPS*, 31, 2018. 2

[44] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, pages 2564–2571. Ieee, 2011. 2

[45] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *ICRA*, pages 3212–3217. IEEE, 2009. 2

[46] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, pages 4938–4947, 2020. 1, 2, 5

[47] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *CVPR*, pages 2930–2937, 2013. 3

[48] 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. 2, 4, 9

[49] 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. 5, 9

[50] 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. 3, 6, 9

[51] Julien Valentin, Matthias Nießner, Jamie Shotton, Andrew Fitzgibbon, Shahram Izadi, and Philip HS Torr. Exploiting uncertainty in regression forests for accurate camera relocalization. In *CVPR*, pages 4400–4408, 2015. 3

[52] 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. 2, 4, 9

[53] 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. 2, 3, 6, 7, 9, 11, 12

[54] 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. 3, 11

[55] Hao Yu, Fu Li, Mahdi Saleh, Benjamin Busam, and Slobodan Ilic. Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. *NeurIPS*, 34:23872–23884, 2021. 2, 3, 4, 5

[56] Qunjie Zhou, Torsten Sattler, and Laura Leal-Taixe. Patch2pix: Epipolar-guided pixel-level correspondences. In *CVPR*, pages 4669–4678, 2021. 2