P2-Net: Joint Description and Detection of Local Features for Pixel and Point Matching

Bing Wang\textsuperscript{1}, Changhao Chen\textsuperscript{2}, Zhaopeng Cui\textsuperscript{3}, Jie Qin\textsuperscript{4}, Chris Xiaoxuan Lu\textsuperscript{5}, Zhengdi Yu\textsuperscript{6}, Peijun Zhao\textsuperscript{1}, Zhen Dong\textsuperscript{7}, Fan Zhu\textsuperscript{8}, Niki Trigoni\textsuperscript{1}, Andrew Markham\textsuperscript{1}

\textsuperscript{1}University of Oxford \textsuperscript{2}National University of Defense Technology \textsuperscript{3}Zhejiang University \textsuperscript{5}University of Edinburgh \textsuperscript{7}Wuhan University
\textsuperscript{4}Nanjing University of Aeronautics and Astronautics \textsuperscript{6}Durham \textsuperscript{8}IIAI

Abstract

Accurately describing and detecting 2D and 3D keypoints is crucial to establishing correspondences across images and point clouds. Despite a plethora of learning-based 2D or 3D local feature descriptors and detectors having been proposed, the derivation of a shared descriptor and joint keypoint detector that directly matches pixels and points remains under-explored by the community. This work takes the initiative to establish fine-grained correspondences between 2D images and 3D point clouds. In order to directly match pixels and points, a dual fully-convolutional framework is presented that maps 2D and 3D inputs into a shared latent representation space to simultaneously describe and detect keypoints. Furthermore, an ultra-wide reception mechanism and a novel loss function are designed to mitigate the intrinsic information variations between pixel and point local regions. Extensive experimental results demonstrate that our framework shows competitive performance in fine-grained matching between images and point clouds and achieves state-of-the-art results for the task of indoor visual localization. Our source code is available at \url{https://github.com/BingCS/P2-Net}.

1. Introduction

Establishing accurate pixel- and point-level matches across images and point clouds, respectively, is a fundamental computer vision task that is crucial for a multitude of applications, such as Simultaneous Localization And Mapping [34], Structure-from-Motion [44], pose estimation [35], 3D reconstruction [25], and visual localization [42].

A typical pipeline of most methods is to first recover the 3D structure given an image sequence [24, 41], and subsequently perform matching between pixels and points based on 2D to 3D reprojected features. Such features will be homogeneous as points in the reconstructed 3D model inherit the descriptors from corresponding pixels of the image sequence. However, this two-step procedure requires accurate 3D reconstruction, which is not always feasible to be achieved, e.g., under challenging illumination or large viewpoint changes. More critically, this approach treats RGB images as first-class citizens, and discounts the equivalence of sensors capable of directly capturing 3D point clouds, e.g., LIDAR, imaging RADAR and depth cameras. These factors motivate us to consider a unified approach to pixel and point matching, where an open question can be posed: how to directly establish correspondences between pixels in 2D images and points in 3D point clouds, and vice-versa? This is inherently challenging as 2D images capture scene appearance, whereas 3D point clouds encode structure.

To this end, we formulate a new task of direct 2D pixel and 3D point matching (cf. Fig. 1) without any auxiliary steps (e.g., reconstruction). This task is undoubtedly challenging for existing conventional and learning-based approaches, which fail to bridge the gap between 2D and 3D representations as separately extracted 2D and 3D local features are distinct and do not share a common embedding.
Some recent works [20, 39] attempt to associate descriptors from different domains by mapping 2D and 3D inputs onto a shared latent space. However, they construct patch-wise descriptors, leading to coarse-grained matching results only.

Even if fine-grained and accurate descriptors can be successfully obtained, direct pixel and point correspondences are still very difficult to establish. First, 2D and 3D keypoints are extracted based on distinct strategies - what leads to a good match in 2D (e.g., flat, visually distinct area such as a poster), does not necessarily correspond to what makes a strong match in 3D (e.g., a poorly illuminated corner of the room). Additionally, because of the sparsity of point clouds, the local feature for a point can be mapped to (or from) many pixel features taken from pixels that are spatially close to the point, increasing the matching ambiguity. Second, due to the large discrepancy between 2D and 3D data property and inflexible optimization manner, existing descriptor loss formulations [18, 31, 2] for either 2D or 3D local feature description do not guarantee convergence in this new context. Moreover, their detector designs only focus on penalizing the confounding descriptors from a safe region, incurring sub-optimal matching results in practice.

To tackle all these challenges, we propose a dual fully-convolutional framework, named Pixel and Point Network (P2-Net), which is able to simultaneously achieve feature description and detection between 2D and 3D views. Furthermore, an ultra-wide reception mechanism is equipped when extracting descriptors to tackle the intrinsic information variations between pixel and point local regions. To optimize the network, we then design P2-Loss, consisting of two components: 1) a circle-guided descriptor loss in combination with a full sampling strategy, allowing to robustly learn distinctive descriptors by optimizing positive and negative matches in a self-paced manner; 2) a batch-hard detector loss, which additionally seeks for the repeatability of detections by encouraging the difference between the positive and globally hardest negative matches. Overall, our contributions are as follows:

1. We propose a joint learning framework with an ultra-wide reception mechanism for simultaneous 2D and 3D local features description and detection to achieve direct pixel and point matching.
2. We design a novel loss, composed of a circle-guided descriptor loss and a batch-hard detector loss, to robustly learn distinctive descriptors whilst explicitly guiding accurate detections for both pixels and points.
3. We conduct extensive experiments and ablation studies, demonstrating the practicability of the proposed framework and the generalization ability of the new loss, and providing the intuition behind our choices.

To the best of our knowledge, this is the first joint learning framework to handle 2D and 3D local features description and detection for direct pixel and point matching.

2. Related Work

2.1. 2D Local Features Description and Detection

Previous learning-based methods in 2D domain simply replaced the descriptor [50, 51, 30, 19, 38] or detector [43, 59, 4] with a learnable alternative. Recently, approaches to joint description and detection of 2D local features have attracted increased attention. LIFT [57] is the first, fully learning-based architecture to achieve this by re-building the main processing steps of SIFT with neural networks. Inspired by LIFT, SuperPoint [16] additionally tackles keypoint detection as a supervised task with labelled synthetic data before description, followed by being extended to an unsupervised version [13]. Differently, DELF [36] and LF-Net [37] exploit an attention mechanism and an asymmetric gradient back-propagation scheme, respectively, to enable unsupervised learning. Unlike previous research that separately learns the descriptor and detector, D2-Net [18] designs a joint optimization framework based on non-maximal-suppression. To further encourage keypoints to be reliable and repeatable, R2D2 [40] proposes a listwise ranking loss based on differentiable average precision. Meanwhile, deformable convolution is introduced in ASLFeat [31] for the same purpose.

2.2. 3D Local Features Description and Detection

Most prior work in the 3D domain has focused on the learning of descriptors. Instead of directly processing 3D data, early attempts [46, 60] instead extract a representation from multi-view images for 3D keypoint description. In contrast, 3DMatch [58] and PerfectMatch [23] construct descriptors by converting 3D patches into a voxel grid of truncated distance function values and smoothed density value representations, respectively. Ppf-Net and its extension [14, 15] directly operate on unordered point sets to describe 3D keypoints. However, such methods require point cloud patches as input, resulting in an efficiency problem. This constraint severely limits its practicability, especially when fine-grained applications are needed. Besides these, dense feature description with a fully convolutional setting is proposed in FCGF [12]. For the detector learning, USIP [27] utilizes a probabilistic chamfer loss to detect and localize keypoints in an unsupervised manner. Motivated by this, 3DFeat-Net [56] is the first attempt for 3D keypoints joint description and detection on point patches, which is then improved by D3Feat [2] to process full-frame point sets.

2.3. 2D-3D Local Features Description

Unlike the well-researched area of learning descriptors in either a single 2D or 3D domain, little attention has been paid on the learning of 2D-3D feature description. A 2D-3D descriptor is generated for object-level retrieval task by directly binding the hand-crafted 3D descriptor to a learned image descriptor [29]. Similarly, 3DTNet [54] learns distinctive 3D descriptors for 3D patches with auxil-
3. Pixel and Point Matching

In this section, we firstly introduce the architecture of the proposed P2-Net in detail, including joint feature description and keypoint detection [18]. Next, we present our designed P2-Loss, composed of a circle-guided descriptor loss and a batch-hard detector loss. Finally, implementation details for both training and testing stages are provided.

3.1. P2-Net Architecture

Feature Description. The first step of our method is to obtain a 3D feature map \( F^I \in \mathbb{R}^{H \times W \times C} \) from image \( I \) and a 2D feature map \( F^P \in \mathbb{R}^{Z \times C} \) from point cloud \( P \), where \( H \times W \) is the spatial resolution of the image, \( Z \) is the number of points and \( C \) is the dimension of the descriptors. Thus, the descriptor \( d \) associated with the pixel \( X \) and point \( Y \) can be denoted as \( d^X \) and \( d^Y \), respectively,

\[
d^X = F^I_X, \quad d^Y = F^P_Y, \quad d \in \mathbb{R}^C.
\]

After being L2-normalized to unit length, these descriptors can be readily compared between images and point clouds to establish correspondences using the cosine similarity as a metric. During training, the descriptors will be optimized so that a pixel and point pair in the scene produces similar descriptors, even when the image or point cloud contains strong changes or noise. For clarity, we still use \( d \) to represent its normalized form in the following text.

Keypoint Detection. As illustrated in Fig. 2.B, we determine keypoints by performing a peakiness-softened non-local-maximum suppression [31] across the spatial and channel dimensions of a feature map. Given a feature map

![Figure 2: An overview of the proposed P2-Net framework.](image)

![Figure 3: To mitigate the intrinsic information variation (A) between 2D and 3D local regions, an ultra-wide reception mechanism (B) with progressively doubling dilation values, up to 16, is applied in the 2D branch of feature description.](image)
Definitions of pixel-point pairs. In a pair of image $I$ and point cloud $P$, $X \leftrightarrow Y$ is a correspondence (pixel $X \in I$ and point $Y \in P$). From the image perspective, $X \leftrightarrow Y_n$ demonstrates a negative match where $Y_n$ lies outside $R^P$ (the neighborhood of $Y$), denoting a negative point of $X$. $X \leftrightarrow Y^*$ represents the hardest negative match and $Y^*$ is the hardest negative point of $X$ in the whole point cloud space. The negative and the hardest negative matches in the perspective of a point cloud are the opposite.

During training, the above procedure is softened to be trainable and density-invariant using peakiness [40]:

$$\rho_t \text{ is a local max in } F_{t,c}^c,$$

in which $F_{t,c}^c$ represents the feature response at the position $t$ and channel $c$, $r_{pt}$ denotes the neighborhood of $\rho_t$.

During training, the above procedure is softened to be trainable and density-invariant using peakiness [40]:

$$\alpha_i^c = \text{softplus}(F_{t,c}^c - \frac{1}{|r_{pt}|} \sum_{\rho_{pt} \in r_{pt}} F_{t,c}^c),$$

$$\beta_i^c = \text{softplus}(F_{t,c}^c - \frac{1}{C} \sum_k F_{t,c}^k),$$

where $\alpha$ and $\beta$ are the spatial-wise and channel-wise detection scores, respectively. The final keypoint detection score of $\rho_t$ that takes both criteria into account is:

$$\xi_{pt} = \max_c (\alpha_i^c \beta_i^c).$$

During testing, pixels or points with top scores will be selected as keypoints for matching.

3.2. P2-Loss Formulation

To make the proposed network describe and detect 2D and 3D keypoints in a single forward pass, we design a novel loss that jointly optimizes the description and detection objectives for both pixels and points, named P2-Loss:

$$\mathcal{L}_{P2} = \mathcal{L}_{desc} + \lambda \mathcal{L}_{det}.$$  

It consists of a circle-guided descriptor loss $\mathcal{L}_{desc}$ that expects distinctive descriptors to avoid incorrect match assignments, a batch-hard detector loss $\mathcal{L}_{det}$ that encourages keypoints to be repeatable under viewpoint or illumination changes, and a balance factor $\lambda$ between them.

Circle-guided Descriptor Loss. To learn distinctive descriptors, various optimization strategies like hard-triplet and hard-contrastive losses [18, 31, 2] have been widely used in 2D or 3D domain. However, these formulations only focus on hard negative matches, and experimentally we found that they did not converge in our 2D-3D context. Inspired by the Circle Loss [47] using weighting factors and the circular decision boundary, we design a circle-guided descriptor loss with a full sampling strategy instead of only considering the hard negative matches, which allows self-paced optimization and avoids convergence ambiguity.

Given a correspondence $X \leftrightarrow Y$ between image $I$ and point cloud $P$ in Fig. 4, we can define a positive cosine similarity $s_p$ for corresponding descriptors $d_X$ and $d_Y$ as:

$$s_p = d_X d_Y = \sum_c d_X^c d_Y^c, \quad (6)$$

From the view of image, we fully sample negative pairs $X \leftrightarrow Y_n$ and define a negative cosine similarity set $s_n^c$ for all descriptor pairs $d_X$ and $d_Y$ as:

$$s_n^c = \left\{ d_X d_{Y_n^1}, \ldots, d_X d_{Y_n^m} \right\}, \quad \text{s.t.} \|Y_n^i - Y\|_2 > R^P, \quad (7)$$

where $Y_n^i$ denotes a negative sample of pixel $X$ lying outside $R^P$ which is the safe radius of point $Y$. The circle-guided descriptor loss of the image part is then derived as:

$$\mathcal{L}_{desc}^i = \frac{1}{|C|} \sum_i \log \left[ 1 + e^{\eta_p^i (1 - s_p^i)} \right] \sum_j e^{\eta_n^i (s_n^i - m)}, \quad (8)$$

in which $C$ is the set of correspondences between image $I$ and point cloud $P$ used for optimization in each step, $\eta_p^i = \zeta(1 + m - s_p^i)$ and $\eta_n^i = \zeta(s_n^i + m)$ represent weighting factors with a scale factor $\zeta$ that expect $s_p^i > 1 - m$ and $s_n^i < m$ in a self-paced manner. With that said, the margin $m$ controls the radius of the circular decision boundary at $(s_n^i - 0)^2 + (s_p^i - 1)^2 = 2m^2$. The reverse loss $\mathcal{L}_{desc}^p$ for point cloud $P$ is calculated in the same way for a total circle-guided descriptor loss $\mathcal{L}_{desc} = \frac{1}{2}(\mathcal{L}_{desc}^i + \mathcal{L}_{desc}^p)$.

Batch-hard Detector Loss. In the case of detection, keypoints should be sufficiently distinctive to be repeatably detected. Achieving this objective, however, faces two practical challenges: 1) the ultra-wide reception mechanism in feature detection may leave spatially close pixels possessing very similar descriptors; 2) the full sampling strategy in our descriptor loss is only effective to negative matches outside a safe region. Both of them will reduce the distinctiveness of keypoints and thus cause erroneous assignments. To this end, we design a batch-hard detector loss with applying hardest-in-batch strategy [33] on the whole image or point cloud space but not on a specific area, encouraging optimal distinctiveness and repeatability.

Similar to the hardest negative match $X \leftrightarrow Y^*$ in Fig. 4, $Y^*$ is determined by $\arg \max_{Y \neq Y^*} \|d_X(d_Y, d_{X})\|$ and denotes the hardest negative point of $X_i$ in the whole point cloud space. In extension to $X_i^*$, we can thus define the hardest negative similarity $s_n^i$ as $\max(d_X(d_Y, d_{Y})).$ Additionally, $\xi_X$ and $\xi_Y$ are the soft detection scores at pixel $X_i$ and point $Y_i$, respectively. With above definitions, we then formulize the batch-hard detector loss as:
\[ L_{det} = \sum_{i \in C} \frac{\xi_i}{\sum_{q \in C} \xi_q} \left( s^i_{n^*} - s^i_p \right), \]

Intuitively, such a detector loss seeks for higher detection scores for more discriminative correspondences. Specifically, \( L_{det} \) expects \( \xi_i \) and \( \xi_{q^*} \) to be high if \( s^i_{n^*} > s^i_p \). Moreover, the more discriminative correspondences, with a lower value of \( (s^i_{n^*} - s^i_p) \), are encouraged to possess higher relative detection scores and vice-versa.

### 3.3. Implementation Details

**Training.** We implement our approach with PyTorch. During the training, we use a batch size of 1 and all image-point cloud pairs with more than 128 pixel-point correspondences. For the sake of computational efficiency, \(|C|=128\) correspondences are randomly sampled from each pair to optimize in each step. We set the balance factor \( \lambda=1 \), the margin \( m=0.2 \), scale factor \( \zeta=10 \), image neighbour \( R_I=12 \) pixels, point cloud neighbour \( R_P=0.015 \) m. Finally, we train the network with the ADAM solver and use an initial learning rate of \( 10^{-4} \) with exponential decay\(^1\).

**Testing.** During testing, we exploit the hard selection strategy demonstrated in Eq. 2 rather than soft selection to mask detections that are spatially too close. Additionally, the SIFT-like edge elimination is applied for image keypoints detection. For evaluation, we select the top-K keypoints corresponding to the detection scores calculated in Eq. 4.

### 4. Experiments

We first demonstrate the effectiveness of proposed P2-Net on the direct pixel and point matching task, and then evaluate it on a downstream task, namely visual localization. Furthermore, we examine the generalization ability of our designed P2-Loss in single 2D and 3D domains, by comparing with the state-of-the-art methods in both image matching and point cloud registration tasks respectively. Finally, we investigate the effect of the loss selection.

#### 4.1. Image and Point Cloud Matching

To achieve fine-grained image and point cloud matching, a dataset of image and point cloud pairs annotated with pixel and point correspondences is required. To the best of our knowledge, there is no publicly available dataset with such correspondence labels. To address this issue, we annotated the 2D-3D correspondence labels\(^1\) on existing 3D datasets containing RGB-D scans. Specifically, the 2D-3D correspondences of our dataset are generated on the 7Scenes dataset [21, 45], consisting of seven indoor scenes with 46 RGB-D sequences recorded under various camera motion status and different conditions (e.g., motion blur), perceptual aliasing and textureless features in the room.

These conditions are widely known to be challenging for both image and point cloud matching.

#### 4.1.1 Evaluation on Feature Matching

We adopt the same data splitting strategy for the 7Scenes dataset as in [21, 45] to prepare the training and testing set. Specifically, 18 sequences are selected for testing, which contain partially overlapped image and point cloud pairs, and the ground-truth transformation matrices.

**Evaluation metrics.** To comprehensively evaluate the performance of our proposed P2-Net and P2-Loss on fine-grained image and point cloud matching, five metrics widely used in previous image or point cloud matching tasks [31, 18, 3, 27, 58, 17, 2] are adopted: 1) Feature Matching Recall, the percentage of image and point cloud pairs with the inlier ratio above a threshold (\( \tau_1 = 0.5 \)); 2) Inlier Ratio, the percentage of correct pixel-point matches over all possible matches, where a correct match is accepted if the distance between the pixel and point pair is below a threshold (\( \tau_2 = 4.5\)cm) under the ground truth transformation; 3) Keypoint Repeatability, the percentage of repeatable keypoints over all detected keypoints, where a keypoint in the image is considered repeatable if its distance to the nearest keypoint in the point cloud is less than a threshold (\( \tau_3 = 2\)cm) under the true transformation; 4) Recall, the percentage of correct matches over all ground truth matches; 5) Registration Recall, the percentage of image and point cloud pairs with the estimated transformation error smaller than a threshold (RMSE< 5cm)\(^1\).

**Comparisons on descriptors and networks.** To study the effects of descriptors, we report the results of 1) traditional SIFT and SIFT3D descriptors; 2) P2-Net trained with the D2-Net loss (P2[D2_Triplet]) [18] and 3) P2-Net trained with the D3Feat loss (P2[D3_Contrastive]) [2]. Besides, to demonstrate the superiority of the 2D branch in P2-Net, we replace it with 4) the R2D2 network (P2[R2D2]) [40] and 5) the ASL network (P2[ASL]) [31]. Other training or testing settings are the same with the proposed architecture trained with our proposed loss (P2[Full]) for a fair comparison. Among them, both P2[R2D2] and P2[Full] adopt L2-Net-style [50] 2D feature extractors but the latter is improved by our ultra-wide reception mechanism.

As shown in Tab. 1, traditional descriptors fail to be matched, as hand-designed 2D and 3D descriptors are heterogeneous. Both P2[D2_Triplet] and P2[D3_Contrastive] are not able to guarantee convergence on the pixel-point matching task. However, when adopting our loss, P2[R2D2] and P2[ASL] models not only converge but also present promising performance in most scenes, except the challenging Stairs scene, due to the intrinsic feature extractor limitation of R2D2 and ASL. Moreover, the comparison between P2[R2D2] and P2[Full] also demonstrates the effectiveness of the ultra-wide reception mechanism. Overall,
The following baselines are only able to localize queried images in 3D regression pipelines [6, 32, 5, 7, 55, 28]. Note that existing feature matching based [48, 55] and scene coordinate regression pipelines [6, 32] are directly and robustly associated, which is essential for real-world downstream applications (e.g., cross-domain information retrieval and localization tasks). Moreover, as our network is jointly trained with the detector, the association is able to bypass regions that cannot be accurately matched, such as the repetitive patterns. More specifically, our detectors mainly focus on the geometrically meaningful areas such as the repetitive patterns. More specifically, our detector is able to bypass regions that cannot be accurately matched, which is a desired property for reliable keypoints to possess.

Qualitative results. Fig. 1 shows the top-1000 detected keypoints for images and point clouds from different scenes. Detected pixels from images (left, green) and detected points from point cloud (right, red) are displayed on Chess and Stairs. For clarity, we randomly highlight some of good matches (blue, orange) to enable better demonstration of the correspondence relations. As can be seen, by our proposed descriptors, such detected pixels and points are directly and robustly associated, which is essential for real-world downstream applications (e.g., cross-domain information retrieval and localization tasks). Moreover, as our network is jointly trained with the detector, the association is able to bypass regions that cannot be accurately matched, such as the repetitive patterns. More specifically, our detectors mainly focus on the geometrically meaningful areas (e.g., object corners and edges) rather than the feature-less regions (e.g., floors, screens and tabletops), and thus show better consistency over environmental changes.

4.1.2 Application on Visual Localization
To further illustrate the practical usage of P2-Net, we perform a downstream task of visual localization [52, 28] on the 7Scenes dataset. The key localization challenge here lies in the fine-grained matching between pixels and points under significant motion blur, perceptual aliasing and textureless patterns. We evaluate our method against the 2D feature matching based [48, 55] and scene coordinate regression pipelines [6, 32]. Note that existing baselines are only able to localize queried images in 3D maps, while our method is not limited by this but can localize reverse queries from 3D to 2D as well. The following

Table 1: Comparisons on the 7Scenes dataset [21, 45].

<table>
<thead>
<tr>
<th># Scenes</th>
<th>Chess</th>
<th>Fire</th>
<th>Heads</th>
<th>Office</th>
<th>Pumpkin</th>
<th>Kitchen</th>
<th>Stairs</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFT + SIFT3D</td>
<td>Not Match</td>
<td>Not Converge</td>
<td>Not Converge</td>
<td>Not Converge</td>
<td>Not Converge</td>
<td>Not Converge</td>
<td>Not Converge</td>
</tr>
<tr>
<td>P2[D2_Triplet]</td>
<td>95.1</td>
<td>97.3</td>
<td>100</td>
<td>89.4</td>
<td>91.1</td>
<td>88.7</td>
<td>16.2</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
<td>95.3</td>
<td>96.0</td>
<td>100</td>
<td>93.4</td>
<td>94.6</td>
<td>47.5</td>
<td>11.9</td>
</tr>
<tr>
<td>P2[w/o Det]</td>
<td>93.0</td>
<td>97.0</td>
<td>99.1</td>
<td>73.8</td>
<td>61.5</td>
<td>43.8</td>
<td>15.0</td>
</tr>
<tr>
<td>P2[Mixed]</td>
<td>92.5</td>
<td>96.0</td>
<td>99.7</td>
<td>74.6</td>
<td>52.2</td>
<td>69.0</td>
<td>15.8</td>
</tr>
<tr>
<td>P2[D2_Triplet]</td>
<td>100</td>
<td>99.7</td>
<td>100</td>
<td>93.6</td>
<td>98.4</td>
<td>94.0</td>
<td>74.3</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
<td>99.0</td>
<td>99.7</td>
<td>100</td>
<td>83.8</td>
<td>68.0</td>
<td>78.4</td>
<td>17.8</td>
</tr>
<tr>
<td>P2[Rand]</td>
<td>100</td>
<td>99.6</td>
<td>99.8</td>
<td>90.8</td>
<td>83.2</td>
<td>82.5</td>
<td>14.3</td>
</tr>
<tr>
<td>P2[Full]</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>97.3</td>
<td>98.5</td>
<td>96.3</td>
<td>88.8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Registration Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>P2[D2_Triplet]</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
</tr>
<tr>
<td>P2[w/o Det]</td>
</tr>
<tr>
<td>P2[Mixed]</td>
</tr>
<tr>
<td>P2[D2_Triplet]</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
</tr>
<tr>
<td>P2[Rand]</td>
</tr>
<tr>
<td>P2[Full]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Keypoint Repeatability</th>
</tr>
</thead>
<tbody>
<tr>
<td>P2[D2_Triplet]</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
</tr>
<tr>
<td>P2[w/o Det]</td>
</tr>
<tr>
<td>P2[Mixed]</td>
</tr>
<tr>
<td>P2[D2_Triplet]</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
</tr>
<tr>
<td>P2[Rand]</td>
</tr>
<tr>
<td>P2[Full]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Inlier Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>P2[D2_Triplet]</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
</tr>
<tr>
<td>P2[w/o Det]</td>
</tr>
<tr>
<td>P2[Mixed]</td>
</tr>
<tr>
<td>P2[D2_Triplet]</td>
</tr>
<tr>
<td>P2[D3_Contrastive]</td>
</tr>
<tr>
<td>P2[Rand]</td>
</tr>
<tr>
<td>P2[Full]</td>
</tr>
</tbody>
</table>

Table 1: Comparisons on the 7Scenes dataset [21, 45]. Evaluation metrics are reported within given thresholds.

Table 2: Evaluation results on the 7Scenes dataset.

As can be seen from Tab. 1, when a detector is not jointly trained with entire model, P2[w/o Det] shows the worst performance on all evaluation metrics and scenes. Such indicators are slightly improved by P2[Mixed] after introducing traditional detectors. Nevertheless, when the proposed detector is used, P2[Rand] achieves better results than P2[Mixed]. These results conclusively indicate that a joint learning with detector is also advantageous to strengthening the descriptor learning itself. Similar improvements can also be observed in both P2[D2_Triplet] and P2[D3_Det]. Clearly, our P2[Full] is able to maintain a competitive matching quality in terms of all evaluation metrics, if its loss is fully enabled. It is worth mentioning that, particularly in the scene of Stairs, P2[Full] is the only method that achieves outstanding matching performance on all metrics. In contrast, most of the other competing methods fail due to the highly repetitive texture in this challenging scenario. It indicates that the keypoints are robustly detected and matched even under challenging condition, which is a desired property for reliable keypoints to possess.

To further illustrate the practical usage of P2-Net, we perform a downstream task of visual localization [52, 28] on the 7Scenes dataset. The key localization challenge here lies in the fine-grained matching between pixels and points under significant motion blur, perceptual aliasing and textureless patterns. We evaluate our method against the 2D feature matching based [48, 55] and scene coordinate regression pipelines [6, 32, 5, 7, 55, 28]. Note that existing baselines are only able to localize queried images in 3D maps, while our method is not limited by this but can localize reverse queries from 3D to 2D as well. The following

2Please refer to the supplementary material for additional results.
experiments are conducted to show the uniqueness of our method: 1) recovering the camera pose of a query image in a given 3D map (P2[3D_Map]) and 2) recovering the pose of a query point cloud in a given 2D map (P2[2D_Map]).

**Evaluation protocols.** We follow the same evaluation pipeline used in [42, 48, 55]. This pipeline typically takes input as query images and a 3D point cloud submap (e.g., retrieved by NetVLAD [1]), and utilizes traditional hand-crafted or pre-trained deep descriptors to establish the matches between pixels and points. Such matches are then taken as the input of PnP with RANSAC [5] to recover the final camera pose. Here, we adopt the same setting in [55] to construct the 2D or 3D submaps that cover a range up to 49.6 cm. Recall that our goal is to evaluate the effects of matching quality for visual localization, we therefore assume the submap has been retrieved and focus more on comparing the distinctiveness of keypoints. During testing, we select the top 10,000 detected pixels and points to generate matches for camera pose estimation.

**Results.** We follow [48, 55] to evaluate models on \( \frac{1}{100} \) testing frames. The localization accuracy is measured in terms of percentage of predicted poses falling within (5cm, 5°). As shown in Fig. 5, when matching 2D features against 3D map, our P2[3D_Map] (68.8%), outperforms InLoc [48] and SAMatch [55] by 2.6% and 5%, respectively, where the conventional feature matching approach are used to localize query images. Moreover, our P2[3D_Map] presents better results than most of the scene coordinated based methods such as RF1 [6], RF2[32], DSAC [5] and SANet [55]. DSAC* [8] and HSC-Net [28] still show better performance than ours, because they are trained for individual scene specifically and use individual models for testing. In contrast, we directly use the single model trained from P2[Full] in Sec. 4.1.1, which is scene agnostic. In the unique application scenario that localizes 3D queries in a 2D map, our P2[2D_Map] also shows promising performance, reaching 65.1%. However, other baselines are not capable of realizing this inverse matching.

### 4.2. Matching under Single Domains

<table>
<thead>
<tr>
<th>Method</th>
<th>SP [16]</th>
<th>D2-Net [18]</th>
<th>ASLFeat [31]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>HEstimation</td>
<td>Pre</td>
<td>Rec</td>
</tr>
<tr>
<td></td>
<td>0.877</td>
<td>0.681</td>
<td>0.553</td>
</tr>
<tr>
<td>Illum Precision</td>
<td>0.629</td>
<td>0.650</td>
<td>0.664</td>
</tr>
<tr>
<td>Recall</td>
<td>0.365</td>
<td>0.564</td>
<td>0.560</td>
</tr>
<tr>
<td>View HEstimation</td>
<td>0.651</td>
<td>0.553</td>
<td>0.581</td>
</tr>
<tr>
<td>Precision</td>
<td>0.595</td>
<td>0.564</td>
<td>0.576</td>
</tr>
<tr>
<td>Recall</td>
<td>0.446</td>
<td>0.382</td>
<td>0.413</td>
</tr>
</tbody>
</table>

Table 2: Comparisons on HPatches [3]. HEstimation, Precision and Recall are calculated at the threshold of 3 pixels. The best score among methods is underlined and the better one between losses is in bold.

In this experiment, we demonstrate how our novel proposed P2-Loss formulation can greatly improve the performance of state-of-the-art 2D and 3D matching networks.

#### 4.2.1 Image Matching

In the image matching experiment, we use the HPatches dataset [3], which has been widely adopted to evaluate the quality of image matching [33, 16, 40, 30, 51, 38, 53]. Following D2-Net [18] and ASLFeat [31], we exclude 8 high-resolution sequences, leaving 52 and 56 sequences with illumination or viewpoint variations, respectively. For a precise reproduction, we directly use the open source code of two state-of-the-art joint description and detection of local features methods, ASLFeat and D2-Net, replacing their losses with ours. SuperPoint (SP) [16] is also a powerful approach to image matching. However, it resorts to interest point pre-training and self-labelling that need synthetic shapes and homographic adaptation, which are very difficult to be directly adopted with our loss. Despite this, we still report the 2D matching results by SuperPoint in Tab. 2 to better present the enhancements on other baselines. Particularly, we keep the same evaluation settings as the original papers for both training and testing.

**Results on the HPatches.** Here, three metrics [38] are used: 1) Homography estimation (HEstimation), the percentage of correct homography estimation between an image pair; 2) Precision, the ratio of correct matches over possible matches; 3) Recall, the percentage of correct predicted matches over all ground truth matches. As illustrated in Tab. 2, when using our loss, clear improvements (up to 3.9%) under illumination variations can be seen in almost all metrics. The only exception happens for D2-Net on Recall and ASLFeat on HEstimation where our loss is only negligibly inferior. On the other side, the performance gain from our method can be observed on all metrics under view variations. This gain ranges from 1.2% to 5.6%. Our proposed optimization strategy shows more significant improvements under view changes than illumination changes.

#### 4.2.2 Point Cloud Registration

In terms of 3D domain, we use the 3DMatch [58], a popular indoor dataset for point cloud matching and registration.
In this work, we propose P2-Net, a dual and fully-convolutional framework in combination with an ultra-wide reception mechanism to jointly describe and detect 2D and 3D local features for direct matching between pixels and points. Moreover, a novel loss function, P2-Loss that consists of a circle-guided descriptor loss and a batch-hard detector loss, is designed to explicitly guide the network to learn distinctive descriptors and detect repeatable keypoints. P2-Net reveals the promising trends towards learning distinctive descriptors. As d_p increases, our loss reduces its gradient and thus enforces a gradually strengthened penalty on d_n, encouraging the distinctiveness between d_p and d_n.

5. Conclusions

In this work, we propose P2-Net, a dual and fully-convolutional framework in combination with an ultra-wide reception mechanism to jointly describe and detect 2D and 3D local features for direct matching between pixels and points. Moreover, a novel loss function, P2-Loss that consists of a circle-guided descriptor loss and a batch-hard detector loss, is designed to explicitly guide the network to learn distinctive descriptors and detect repeatable keypoints for both pixels and points. Extensive experiments on pixel and point matching, visual localization, image matching and point cloud registration not only show the effectiveness and practicability of our P2-Net but also demonstrate the generalization ability and superiority of our P2-Loss.

---

3 Please refer to the supplementary material for more analysis.
References


