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

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

[1]University of Oxford [2]National University of Defense Technology [3]Zhejiang University
[4]Inception Institute of Artificial Intelligence [5]University of Edinburgh [6]Durham University [7]Wuhan University

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 in combination with 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 will be available at [no-name-for-blind-review].*

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 [33], Structure-from-Motion [43], pose estimation [34], 3D reconstruction [24], and visual localization [41].

A typical pipeline of most existing methods is to first recover the 3D structure given an image sequence [23, 40], and subsequently perform matching between pixels and points based on the 2D to 3D reprojected features. These features will be homogeneous as the points in reconstructed 3D model inherit the descriptors from the corresponding pixels of the image sequence. However, this two-step procedure relies on accurate and dense 3D reconstruction, which itself relies on high-quality 2D images with sufficient overlap, something that is not always feasible to obtain, e.g., under challenging illumination. 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 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.

Existing conventional and learning-based approaches 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, i.e., descriptors from images cannot be directly used in the 3D space and vice versa. Some recent works [19, 38] have attempted to associate descriptors from different domains by mapping 2D and 3D inputs onto a shared latent space. However, they only 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. This is because 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).

To this end, we formulate a new task of *direct* 2D pixel and 3D point matching *without* any auxiliary steps (e.g., reconstruction). To tackle this challenging task, we propose a joint framework, named **P**ixel and **P**oint Network (P2-Net), which is able to simultaneously achieve effective feature description and detection between 2D and 3D views. Although similar attempts have been made in the 2D [17] or 3D domain [2] in isolation, jointly describing and detecting 2D and 3D keypoints is non-trivial. First, the *densities* of pixels and points are significantly different. Specifically, because of the sparsity of point clouds, fewer points than pixels represent the same local region. Under such circumstances, a point local feature can be mapped to (or from) many pixel features taken from pixels that are spatially close to the point. Second, the current art of detector designs [17, 30, 2] only focuses on penalizing confounding descriptors in a limited area, incurring sub-optimal matching results in practice. Last but not least, due to the large discrepancy between 2D and 3D data property, existing loss functions [17, 30, 2] for either 2D or 3D joint description and detection do not guarantee convergence in this new context. In this work, our contributions are as follows:

1. We propose a dual, fully-convolutional framework for simultaneous 2D and 3D local features description and detection to achieve direct pixel and point matching, without requiring any auxiliary reconstruction or reprojection steps.
2. We present an ultra-wide reception mechanism whilst extracting descriptors to tackle the intrinsic information variations between pixel and point local regions.
3. We design a novel loss based on a coarse-to-fine optimization strategy, which not only guarantees convergence whilst learning discriminative descriptors, but also provides explicit guidance for accurate detections.

To confirm the practicability of the proposed framework and the generalization ability of the new loss, we conduct thorough experiments on fine-grained image and point cloud matching, visual localization, image matching and point cloud registration tasks. To the best of our knowledge, we are the first to handle 2D and 3D local features description and detection for pixel and point level matching in a joint learning framework.

2. Related Work

2.1. 2D Local Features Description and Detection

Previous learning-based methods in 2D domain simply replaced the descriptor [49, 50, 29, 18, 37] or detector [42, 58, 4] with a learnable alternative. Recently, approaches to joint description and detection of 2D local features has attracted increased attention. LIFT [56] is the first, fully learning-based architecture to achieve this by rebuilding the main processing steps of SIFT with neural networks. Inspired by LIFT, SuperPoint [15] additionally tackles keypoint detection as a supervised task with labelled synthetic data before description, followed by being extended to an unsupervised version [12]. Differently, DELF [35] and LF-Net [36] 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 [17] designs a joint optimization framework based on non-maximal-suppression. To further encourage keypoints to be reliable and repeatable, R2D2 [39] proposes a listwise ranking loss based on differentiable average precision. Meanwhile, deformable convolution is introduced in ASLFeat [30] 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 [45, 59] instead extract a representation from multi-view images for 3D keypoint description. In contrast, 3dMatch [57] and PerfectMatch [22] 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 [13, 14] 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 [11]. For the detector learning, USIP [26] utilizes a probabilistic chamfer loss to detect and localize keypoints in an unsupervised manner. Motivated by this, 3DFeat-Net [55] 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 shed 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 [28]. Similarly, 3DTNet [53] learns discriminative 3D descriptors for 3D patches with auxiliary 2D features extracted from 2D patches. Recently, both 2D3DMatch-Net [19] and LCD [38] propose to learn descriptors that allow direct matching across 2D and 3D local patches for retrieval problems. However, all these methods are patch-based, which is impractical in real usage as discussed in Section 1. In contrast, we aim to extract per-point descriptors and detect keypoint locations in a single forward pass for efficient usage. To the best of our knowledge, we are the first learning approach to achieve pixel-point level 2D-3D matching.

3. Pixel and Point Matching

In this section, we introduce the proposed P2-Net framework for pixel and point matching, mainly consisting of

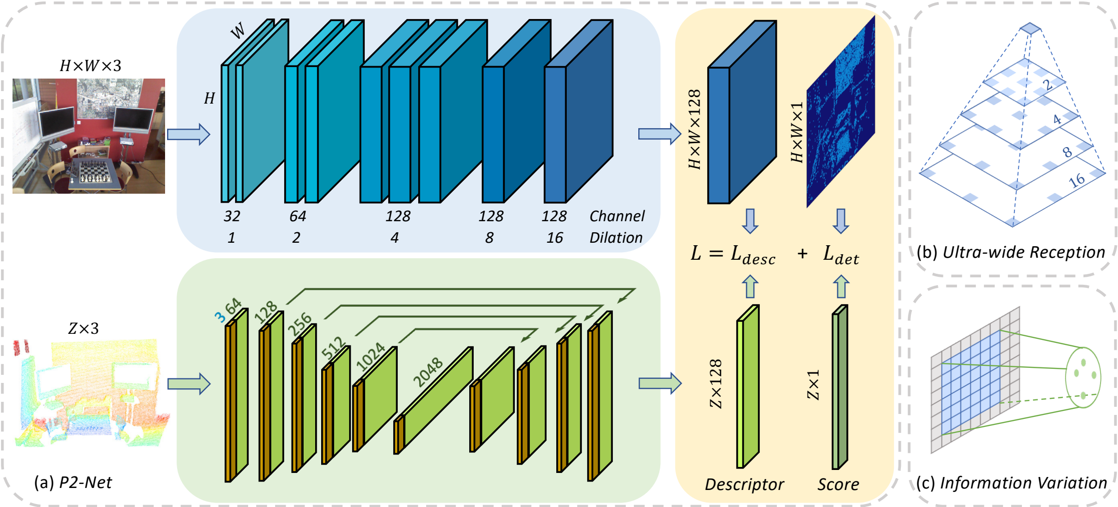

Figure 1: **An overview of the proposed P2-Net framework.** Our architecture is a two-branch fully convolutional network, which can be jointly optimized with a descriptor loss enforcing the similarity of corresponding representations as well as a detector loss encouraging higher scores for distinctive matches.

three parts, including feature extraction, feature description, and keypoint detection. To achieve this, we particularly present an ultra-wide reception mechanism to mitigate the intrinsic information variations of local regions between pixels and points, and novel losses for discriminative descriptors learning and accurate keypoints detection.

3.1. P2-Net Architecture

Feature Extraction As illustrated in Fig. 1 (a), two fully convolutional networks are exploited to separately perform feature extraction on images and point clouds. However, properly associating pixels with points through descriptors is non-trivial because of the intrinsic variation in information density (Fig. 1 (c)) between 2D and 3D local regions. Specifically, the local information represented by a point is typically larger than a pixel due to the sparsity of point clouds. To address the issue of association on asymmetrical embeddings and better capture the local geometry information, we design the 2D extractor based on an ultra-wide receptive field mechanism, shown in Fig. 1 (b). For computational efficiency, such a mechanism is achieved through nine 3×3 convolutional layers with progressively increasing dilation values, ranging from 1 to 16. Finally, a $128D$ feature map and a $1D$ score map at the input image resolution are generated. In a similar vein, we modify KPconv [48], a leading point-cloud network, to output a $128D$ feature vector and a score for each point.

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 associated with the pixel x_{hw} and point x_z can be denoted as d_{hw} and d_z, respectively,

$$\mathbf{d}_{hw} = F^I_{hw} \ \ ,\mathbf{d}_z = F^P_z, \mathbf{d} \in \mathbb{R}^C \ \ . \tag{1}$$

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 adjusted 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. In practice, the descriptors are L2-normalized to unit length for matching.

Keypoint Detection. Similar to [17, 30, 2], we define keypoints on 2D images based on the local maximum across the spatial and channel dimensions of feature maps. Given the dense feature map $F \in \mathbb{R}^{T\times C}$, there exist multiple detection maps D^c_T $(c = 1, ..., C)$, where $T = H \times W$ for images and $T = Z$ for point clouds:

$$D^c_T = F^{:c} \ \ , \ \ D^c_T \in \mathbb{R}^C \ \ , \tag{2}$$

in which, $F^{:c}$ denotes the detection map of channel c. The requirement for a pixel or point x_t to be detected is

$$\begin{aligned} x_t \text{ is a detection } &\Longleftrightarrow c = \arg\max_k \ D^k_t \ \text{ and} \\ &D^c_t \text{ is a local max in } D^c_R \ \ , \end{aligned} \tag{3}$$

where t represents the index hw for a pixel or z for a point. Intuitively, we firstly select the preeminent (i.e. the depth-wise max) channel for x_t, and then determine whether it is a local maximum among its spatial local neighboring area R, or on that particular response map D^c_R. We soften the above process to make it trainable by applying spatial and channel-wise scores for a pixel or points as follows:

$$\begin{aligned} \alpha^c_t &= \text{softplus}(D^c_t - \frac{1}{|\mathcal{N}_{x_t}|}\sum D^c_{t'}) \ \ , \\ \beta^c_t &= \text{softplus}(D^c_t - \frac{1}{C}\sum D^k_t) \ \ , \end{aligned} \tag{4}$$

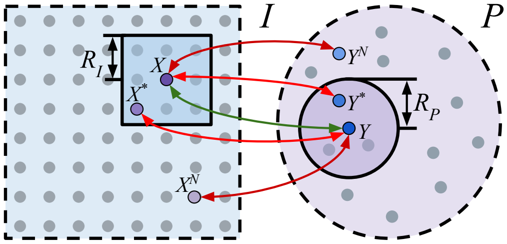

Figure 2: For each correspondence $X \leftrightarrow Y$, negative matches of X in P (Y^N) and of Y in I (X^N) are arbitrary samples lying outside R_P and R_I, respectively. X^* is the most confounding pixel of X for Y, and similarly for Y^*.

where α represents the score for spatial response while β denotes the channel-wise response. Next, in order to take both criteria into account, we maximize the product of both scores across all feature maps c to obtain a single score map:

$$\gamma_t = \max_c (\alpha_t^c \beta_t^c) \quad . \tag{5}$$

Finally, the soft detection score S_t at a pixel or point t is obtained by performing an image-level normalization:

$$S_t = \gamma_t \Big/ \sum \gamma_{t'} \quad . \tag{6}$$

3.2. Coarse-to-Fine Loss

To make the proposed network simultaneously describe and detect both 2D and 3D keypoints in a single forward pass, we design a coarse-to-fine loss $\mathcal{L}$ which can jointly optimize the description and detection objectives:

$$\mathcal{L} = \mathcal{L}_{desc} + \lambda \mathcal{L}_{det} \quad . \tag{7}$$

It consists of a circle-guided descriptor loss $\mathcal{L}_{desc}$ that provides relatively coarse supervision for all descriptors, a batch hard detector loss $\mathcal{L}_{det}$ that finely emphasizes on the most confounding ones, and a balance factor λ.

Circle-Guided Descriptor Loss. In the case of description, descriptors are expected to be distinctive to avoid incorrect match assignments. As shown in Fig. 2, given a pair of an image and a point cloud (I, P) and a correspondence $X \leftrightarrow Y$ between them (where $X \in I$, $Y \in P$), the descriptor loss seeks to maximize the positive similarity d_p of corresponding descriptors (d_X, d_Y), but to minimize the negative similarity d_n of all mismatched pairs (d_X, d_{Y^N}) and (d_Y, d_{X^N}). Under the cosine similarity metric, the *positive* similarity d_p and *negative* similarity d_n are defined as:

$$\begin{aligned} d_p &= d_X \cdot d_Y = \sum d_X^c d_Y^c \quad , \\ d_n &= \max(d_X \cdot d_{Y^N} \quad , \quad d_{X^N} \cdot d_Y) \quad . \end{aligned} \tag{8}$$

To extract descriptors with distinctiveness, both hard-triplet loss and hard-contrastive loss have been successfully introduced for 2D or 3D descriptor learning [17, 30, 2]:

$$\begin{aligned} \mathcal{L}_{triplet} &= [d_p - d_n - M]_+ \quad , \\ \mathcal{L}_{contrastive} &= [M_p - d_p]_+ + [d_n - M_n]_+ \quad . \end{aligned} \tag{9}$$

Please note that, they all pose an extra restriction for confounding points X^N and Y^N:

$$X^N = \arg\max_{Y^n \in P} (d_X \cdot d_{Y^n}) \quad \text{s.t.} \ \|Y^n - Y\|_2 > R_P \quad , \tag{10}$$

and similarly for Y^N. However, we found that such loss formulations, only focus on hard pairs and do not guarantee convergence in our context due to the large discrepancy between 2D and 3D data property.

To tackle this, we present a descriptor loss with a circular decision boundary [46]:

$$\begin{aligned} \mathcal{L}_{desc} = &\text{softplus}(\sum \exp(\zeta(\Delta_p - d_p^i)[O_p - d_p^i]_+) \\ &+ \sum \exp(\zeta(d_n^j - \Delta_n)[d_n^j - O_n]_+)) \quad , \end{aligned} \tag{11}$$

in which ζ represents a scale factor, O_p and O_n are the optimum for d_p^j and d_n^i respectively, Δ_n and Δ_p denote the between-class and within-class margins, respectively. Similar to [46], we reduce the hyper-parameters by introducing a relaxation margin and making O_p=$1+m$, O_n=$-m$, Δ_p=$1-m$, and Δ_n=m. Intuitively, our loss seeks to encourage the distinctiveness of descriptors by penalizing arbitrary confounding descriptors that may result in mismatching. Without the restriction in Eq. 10, our network can firstly optimize the negatives which are easy to recognize and then focus on harder ones. Moreover, such loss formulation has a circular decision boundary that can avoid ambiguous convergence [46]. With such improvements, the circle-guided descriptor loss can promote robust convergence status and learn distinctive 2D and 3D descriptors.

Batch Hard Detector Loss. For the case of detection, keypoints are expected to be distinctive and also repeatable regardless of whether the viewpoint or ambient illumination changes. To this end, we seek a loss formulation that encourages higher saliency for more discriminative correspondences. Existing detectors [17, 30, 2] still focus on enforcing discriminativeness between correspondences and hard mismatches defined in Eq. 10 and lack the supervision for globally confounding points, which typically leads to mismatching in practice. Moreover, the usage of ultra-wide reception mechanism in feature extraction and the circle-guided descriptor loss further bring two risks: **1)** the ultra-wide reception will guide spatially close pixels to possess increasingly similar representations; **2)** without the restriction in Eq. 10, our descriptor loss will pose less emphasis on optimizing the most confounding descriptors. Both of them will reduce the distinctiveness of keypoints and thus cause erroneous assignments.

To address such problems, we design a new detector loss term that adopts the *hardest-in-batch* sampling strategy in [32] to explicitly provide the strictest guidance for the gradient of the scores:

$$L_{det} = \sum \frac{S_{X_i} S_{Y_i}}{\sum S_{X_j} S_{Y_j}} \left(\max\left(d_{X_i} \cdot d_{Y_i^*}, d_{X_i^*} \cdot d_{Y_i}\right) - d_{p_i}\right)$$
$$X_i^* = \arg\max_{Y_i^n \neq Y_i} \left(d_{X_i} \cdot d_{Y_i^n}\right), Y_i^* = \arg\max_{X_i^n \neq X_i} \left(d_{Y_i} \cdot d_{X_i^n}\right) \tag{12}$$

Intuitively, in order for the loss to be minimized, the most distinctive correspondences will get higher relative scores while mismatched pairs will be assigned lower scores. Different from existing detector loss formulations [17, 30, 2], we apply the *hardest-in-batch* strategy on the global area instead of only on a limited region, encouraging optimal distinctiveness. As such, we avoid the risks illustrated above by applying the strictest supervision on the most confounding pixels or points.

3.3. Implementation Details

Training. We implement our approach with PyTorch. During the training stage, 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, randomly sample 128 correspondences for each pair to optimize in each step. We use the relaxation margin $m = 0.2$, scale factor $\zeta = 10$, image neighbour radius $R_I = 12$ pixels, point cloud neighbour radius $R_P = 0.015$ m. In the training loss, we set the balance factor $\lambda = 1$. Finally, we train the network using the ADAM solver with an initial learning rate of 10^{-4}[1] with exponential decay.

Testing. During testing, we exploit the hard selection strategy demonstrated in Eq. 3 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. 6.

4. Experiments

We first demonstrate the effectiveness of proposed P2-Net framework 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 proposed 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 metrics.

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 manually annotated the 2D-3D correspondence labels on existing 3D datasets containing RGB-D scans[1]. Specifically, the 2D-3D correspondences of our dataset are generated on the 7Scenes dataset [20, 44], 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 [20, 44] 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 on fine-grained image and point cloud matching, five metrics widely used in previous image or point cloud matching tasks [30, 17, 3, 26, 57, 16, 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 its 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 (D2_Triplet) [17] and 3) P2-Net trained with the D3Feat loss (D3_Contrastive) [2]. Besides, to demonstrate the superiority of the 2D branch in P2-Net, we replace it with 4) the R2D2 feature extractor (R2D2_Based) [39] and 5) the ASL feature extractor (ASL_Based) [30]. Other training or testing settings are kept the same with the proposed architecture trained with our proposed loss (P2-Net) for a fair comparison.

As shown in Tab. 1, traditional descriptors fail to be matched, as hand-designed 2D and 3D descriptors are heterogeneous. Additionally, both D2_Triplet and D3_Contrastive loss formulations are not able to guarantee convergence on pixel and point matching task. However, when adopting our loss, R2D2_Based and ASL_Based 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. Overall, our proposed P2-Net performs consistently better regarding all evaluation metrics, outperforming all

[1] Please refer to the supplementary material for more details.

# Scenes	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs
Feature Matching Recall							
SIFT + SIFT3D	Not Match						
D2_Triplet	Not Converge						
D3_Contrastive	Not Converge						
R2D2_Based	95.1	97.3	**100**	89.4	91.1	88.7	16.2
ASL_Based	95.3	96.0	**100**	34.3	41.6	47.5	11.9
P2[w/o Det]	93.0	97.0	99.1	73.8	61.5	43.8	15.0
P2[Mixed]	92.5	96.0	99.7	74.6	52.2	69.0	15.8
P2[D2_Det]	**100**	99.7	**100**	93.6	98.4	94.0	74.3
P2[D3_Det]	99.0	99.7	**100**	83.8	68.0	78.4	17.8
P2[Rand]	**100**	99.6	99.8	90.8	83.2	82.5	14.3
P2-Net	**100**	**100**	**100**	**97.3**	**98.5**	**96.3**	**88.8**
Registration Recall							
R2D2_Based	81.0	78.5	73.1	79.7	75.6	77.1	60.8
ASL_Based	70.5	66.0	63.4	52.9	41.6	48.0	38.2
P2[w/o Det]	68.0	64.5	53.8	59.6	48.4	56.1	42.3
P2[Mixed]	72.5	66.5	20.9	59.1	53.2	63.5	25.6
P2[D2_Det]	86.0	75.5	74.2	70.8	80.0	74.3	78.3
P2[D3_Det]	80.5	70.0	81.7	76.3	65.5	70.6	70.9
P2[Rand]	86.5	81.5	82.6	78.9	75.5	77.2	74.3
P2-Net	**87.0**	**82.4**	**84.5**	**83.4**	**88.7**	**82.7**	**82.6**
Keypoint Repeatability							
R2D2_Based	36.6	40.3	45.2	33.4	30.3	32.1	33.1
ASL_Based	18.7	19.2	33.8	13.8	12.9	15.5	11.9
P2[w/o Det]	17.4	17.8	37.0	18.2	16.0	15.7	17.7
P2[Mixed]	23.3	26.6	26.0	30.0	29.9	31.3	24.7
P2[D2_Det]	41.7	39.8	40.6	34.8	32.7	31.6	34.9
P2[D3_Det]	24.9	21.8	38.1	24.5	19.6	23.8	21.8
P2[Rand]	36.1	37.0	46.1	33.5	30.4	32.2	36.1
P2-Net	**50.4**	**47.1**	**50.2**	**38.0**	**45.2**	**38.3**	**48.1**
Recall							
R2D2_Based	28.5	26.7	24.7	25.0	24.6	26.4	16.0
ASL_Based	28.8	26.3	16.5	21.7	21.4	23.8	13.8
P2[w/o Det]	29.1	26.9	23.1	25.3	22.0	23.8	14.4
P2[Mixed]	30.1	26.2	25.2	24.5	24.1	26.9	15.1
P2[D2_Det]	30.3	28.9	26.1	27.0	**29.6**	28.7	17.7
P2[D3_Det]	31.8	31.1	26.4	26.6	25.6	27.5	17.1
P2[Rand]	31.4	30.8	25.7	29.5	28.0	30.6	17.6
P2-Net	**32.7**	**33.7**	**26.6**	**30.6**	**29.6**	**32.3**	**20.1**
Inlier Ratio							
R2D2_Based	65.5	66.5	69.8	54.0	54.5	55.3	38.5
ASL_Based	55.9	60.8	64.9	44.7	45.7	47.6	34.2
P2[w/o Det]	52.7	56.3	71.0	46.1	47.3	49.9	36.2
P2[Mixed]	51.5	55.2	67.4	52.1	50.1	56.7	35.1
P2[D2_Det]	68.2	72.2	74.9	58.0	**61.4**	59.3	42.9
P2[D3_Det]	61.1	64.6	75.4	51.3	47.6	51.8	37.9
P2[Rand]	58.5	61.4	76.2	53.2	50.0	53.4	40.4
P2-Net	**73.9**	**76.0**	**77.4**	**60.3**	60.8	**65.2**	**45.2**

Table 1: **Comparisons on the 7Scenes dataset [20, 44].** Evaluation metrics are reported within given thresholds.

competitive methods by a large margin on all scenes.

Comparisons on detectors. In order to demonstrate the importance of jointly learning the detector and descriptor, we report the results of: 1) the model trained without a detector but with randomly sampled keypoints (P2[w/o Det]); 2) the model trained without a detector but with SIFT and SIFT3D keypoints (P2[Mixed]). Furthermore, we also compare: 3) the model trained with the original D2-Net detector (P2[D2_Det]) [17], 4) the model trained with the D3Feat detector (P2[D3_Det])[2] and 5) P2-Net with randomly sampled keypoints (P2[Rand]) to indicate the superiority of our proposed detector.

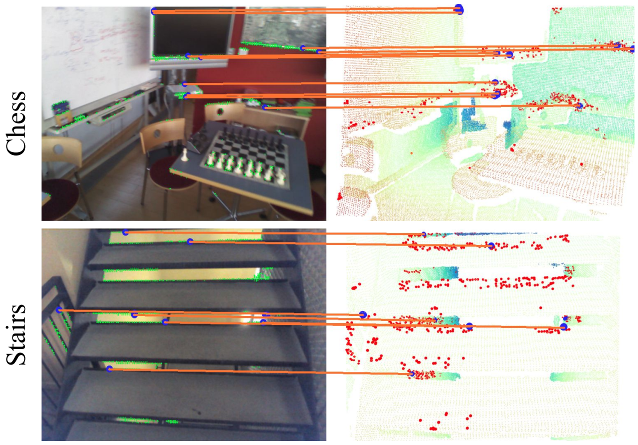

Figure 3: **Visualization on sampled scenes.** Detected pixels from images (left, green) and detected points from point cloud (right, red) are displayed on Chess and Stairs. Sampled matches are marked and connected (blue, orange).

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_Det] and P2[D3_Det]. Clearly, our P2-Net is able to maintain a competitive matching quality in terms of all evaluation metrics, if our loss is fully enabled. It is worth mentioning that, particularly in the scene of Stairs, P2-Net 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[2].

Qualitative results. Fig. 3 shows the top-1000 detected keypoints for images and point clouds from different scenes. For clarity, we randomly highlight some of good matches 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.

[2] Please refer to the supplementary material for additional results.

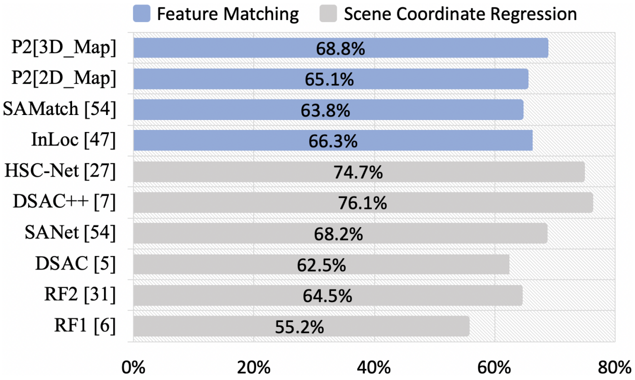

Figure 4: **Comparisons on visual localization.** Percentage of estimated camera poses falling within ($5cm$, $5°$).

object corners and edges) rather than the feature-less regions (e.g. floors, screens and tabletops), and thus show better consistency over environmental changes[2].

4.1.2 Application on Visual Localization

To further illustrate the practical usage of P2-Net, we perform a downstream task of visual localization [51, 27] 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 [47, 54] and scene coordinate regression pipelines [6, 31, 5, 7, 54, 27]. *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 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 [41, 47, 54]. 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 pixel and point. 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 [54] 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 previous works [47, 54] to evaluate models on 1 out of every 10 testing frames. The localization accuracy is measured in terms of percentage of predicted poses falling within the threshold of ($5cm$, $5°$). As shown in Fig. 4, when matching 2D features against 3D map, our method, P2[3D_Map] (68.8%), outperforms InLoc [47] and SAMatch [54] 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, i.e. RF1 [6], RF2[31], DSAC [5] and SANet [54]. DSAC++ [7] and HSC-Net [27] still show better performance than ours, because they are trained for individual scene specifically and therefore use individual models for testing. In contrast, we only use one single model trained in Sec. 4.1, which is agnostic to the scenes themselves. 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.

		D2-Net [17]		ASLFeat [30]	
		Triplet	**Our Loss**	Contrastive	**Our Loss**
HPatches Illum	HEstimation	0.818	**0.857**	**0.919**	0.915
	Precision	0.650	**0.664**	0.774	**0.787**
	Recall	**0.564**	0.560	0.696	**0.726**
HPatches View	HEstimation	0.553	**0.581**	0.542	**0.598**
	Precision	0.564	**0.576**	0.708	**0.740**
	Recall	0.382	**0.413**	0.583	**0.625**

Table 2: **Comparisons on the HPatches dataset [3].** HEstimation, Precision and Recall are calculated at the error threshold of 3 pixels.

4.2. Matching under Single Domains

In this experiment, we demonstrate how our novel proposed 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 [32, 15, 39, 29, 50, 37, 52]. Following D2-Net [17] and ASLFeat [30], 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. Particularly, we keep the same evaluation settings as the original papers for both training and testing.

Results on the HPatches. Here, three metrics 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 met-

	FCGF [11]			D3_Contrastive [2]			**D3_Our Loss**		
	Reg	FMR	IR	Reg	FMR	IR	Reg	FMR	Inlier
Kitchen	0.93			0.97	0.97	0.34	**0.98**	**0.99**	**0.46**
Home 1	0.91			0.90	0.99	0.45	**0.92**	**1.00**	**0.59**
Home 2	0.71			0.72	0.91	0.43	**0.73**	**0.93**	**0.55**
Hotel 1	0.91	\	\	0.95	0.98	0.39	**0.98**	**1.00**	**0.53**
Hotel 2	0.87			0.87	0.95	0.37	**0.91**	**0.97**	**0.49**
Hotel 3	0.69			0.80	0.96	0.47	**0.81**	**1.00**	**0.56**
StudyRoom	0.75			0.83	0.95	0.37	**0.86**	**0.96**	**0.56**
MIT Lab	0.80			0.69	0.92	0.42	**0.84**	**0.97**	**0.54**
Average	0.82	0.95	**0.54**	0.84	0.95	0.41	**0.88**	**0.98**	**0.54**

Table 3: **Comparisons on the 3DMatch dataset [57]**. Reg, FMR and IR are evaluated at the threshold of 0.2 m, 5% and 0.1 m, respectively.

rics. 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 coarse-to-fine 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 [57], a popular indoor dataset for point cloud matching and registration [25, 14, 22, 11, 10, 21, 9]. We follow the same evaluation protocols in [57] to prepare the training and testing data, 54 scenes for training and the remaining 8 scenes for testing. As D3Feat[2] is the only work which jointly detects and describes 3D local features, we replace its loss with ours for comparison. To better demonstrate the improvements, the results from FCGF [11] are also included.

Results on the 3DMatch. We report the performance on three evaluation metrics: 1) Registration Recall (Reg), 2) Inlier Ratio (IR), and 3) Feature Matching Recall (FMR). As illustrated in Tab. 3, when our loss is adopted, a 6% and a 3% improvements can be seen on Reg and FMR, respectively. In contrast, there is only 2% and 0% respective difference between FCGF and the original D3Feat. In particular, as for Inlier Ratio, our loss demonstrates better robustness, outperforming the original one by 13%, comparable to FCGF. Overall, our loss consistently achieves the best performance in terms of all metrics.

4.3. The Impact of Descriptor Loss

Finally, we come to analyse the impacts of loss choices on homogeneous (2D↔2D or 3D↔3D) and heterogeneous (2D↔3D) feature matching. From the detector loss formulation in Eq. 12, we can see that its optimization tightly depends on the descriptor. Therefore, we conduct a comprehensive study on three predominant metric learning losses for descriptor optimization and aim to answer: why is the circle-guided descriptor loss best suited for feature matching? To this end, we track the difference between the positive similarity d_p and the most negative similarity d_{n^*} ($\max(d_n)$) with various loss formulations and architectures.

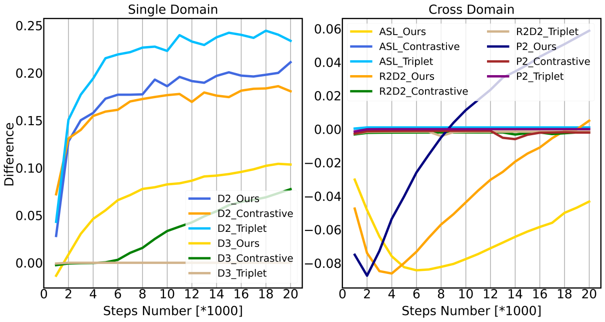

Figure 5: The difference between the positive similarity d_p and the most negative similarity d_{n^*} over time with different networks and losses. Left: single-domain matching; Right: cross-domain matching.

Fig. 5 (left) shows that, in single/homogeneous 2D or 3D domains, both D2-Net and D3Feat can gradually learn discriminative descriptors. D2-Net consistently ensures convergence, regardless of the choice of loss, while D3Feat fails when hard triplet loss is selected. This is consistent with the conclusion in [2]. In the cross-domain image and point cloud matching (Fig. 5 (right), we compare different losses and 2D feature extractors. This overwhelmingly demonstrates that neither hard triplet nor hard contrastive loss can converge in any framework (ASL, R2D2 or P2-Net). Both triplet and contrastive losses are inflexible, because the penalty strength for each similarity is restricted to be equal. Moreover, their decision boundaries are parallel to d_p=d_n, which causes ambiguous convergence [8, 32]. However, our loss enables all architectures to converge, showing promising trends towards learning distinctive descriptors. Thanks to the introduction of *circular decision boundary*, the proposed descriptor loss assigns different gradients to the similarities, promoting more robust convergence [46].

Interestingly, we can observe that the distinctiveness of descriptors initially is inverted for heterogeneous matching, unlike homogeneous matching. As pixel and point descriptors are initially disparate, their similarity can be extremely low for both positive and negative matches in the initial phase[3]. In such case, the gradients (ranging between $[0, 1]$) with respect to d_p and d_n almost approach 1 and 0 [46], respectively. Because of the sharp gradient difference, the loss minimization in network training will tend to overemphasize the optimization d_p while sacrificing the descriptor distinctiveness. 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.

[3] Please refer to the supplementary material for more analysis.

5. Conclusions

In this work, we propose a dual, fully-convolutional framework to simultaneously describe and detect 2D and 3D local features for direct matching between pixels and points. Considering the information density variation between images and point clouds, we firstly introduce an ultra-wide reception mechanism whilst extracting local features. Moreover, a coarse-to-fine loss function is designed to provide explicit guidance for the learning of distinctive descriptors and keypoints. Extensive experiments on pixel and point matching, visual localization, image matching and point cloud registration not only show the effectiveness and practicability of our proposed P2-Net but also demonstrate the generalization ability and superiority of our designed coarse-to-fine loss.

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*, 2016. 7

[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*, 2020. 2, 3, 4, 5, 6, 8

[3] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, 2017. 5, 7

[4] Axel Barroso-Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Key. net: Keypoint detection by handcrafted and learned cnn filters. In *ICCV*, 2019. 2

[5] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *CVPR*, 2017. 7

[6] 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*, 2016. 7

[7] Eric Brachmann and Carsten Rother. Learning less is more-6d camera localization via 3d surface regression. In *CVPR*, 2018. 7

[8] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005. 8

[9] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *CVPR*, 2020. 8

[10] Christopher Choy, Junha Lee, René Ranftl, Jaesik Park, and Vladlen Koltun. High-dimensional convolutional networks for geometric pattern recognition. In *CVPR*, 2020. 8

[11] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *ICCV*, 2019. 2, 8

[12] Peter Hviid Christiansen, Mikkel Fly Kragh, Yury Brodskiy, and Henrik Karstoft. Unsuperpoint: End-to-end unsupervised interest point detector and descriptor. *arXiv:1907.04011*, 2019. 2

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

[14] Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppfnet: Global context aware local features for robust 3d point matching. In *CVPR*, 2018. 2, 8

[15] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR Workshops*, 2018. 2, 7

[16] Zhen Dong, Fuxun Liang, Bisheng Yang, Yusheng Xu, Yufu Zang, Jianping Li, Yuan Wang, Wenxia Dai, Hongchao Fan, Juha Hyyppäb, et al. Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. *ISPRS Journal of Photogrammetry and Remote Sensing*, 163:327–342, 2020. 5

[17] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *CVPR*, 2019. 2, 3, 4, 5, 6, 7

[18] Patrick Ebel, Anastasiia Mishchuk, Kwang Moo Yi, Pascal Fua, and Eduard Trulls. Beyond cartesian representations for local descriptors. In *ICCV*, 2019. 2

[19] 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*, 2019. 1, 2

[20] Ben Glocker, Shahram Izadi, Jamie Shotton, and Antonio Criminisi. Real-time rgb-d camera relocalization. In *ISMAR*, 2013. 5, 6

[21] Zan Gojcic, Caifa Zhou, Jan D Wegner, Leonidas J Guibas, and Tolga Birdal. Learning multiview 3d point cloud registration. In *CVPR*, 2020. 8

[22] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *CVPR*, 2019. 2, 8

[23] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 1

[24] Jared Heinly, Johannes L Schonberger, Enrique Dunn, and Jan-Michael Frahm. Reconstructing the world* in six days*(as captured by the yahoo 100 million image dataset). In *CVPR*, 2015. 1

[25] Marc Khoury, Qian-Yi Zhou, and Vladlen Koltun. Learning compact geometric features. In *ICCV*, 2017. 8

[26] Jiaxin Li and Gim Hee Lee. Usip: Unsupervised stable interest point detection from 3d point clouds. In *ICCV*, 2019. 2, 5

[27] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. Hierarchical scene coordinate classification and regression for visual localization. In *CVPR*, 2020. 7

[28] Yangyan Li, Hao Su, Charles Ruizhongtai Qi, Noa Fish, Daniel Cohen-Or, and Leonidas J Guibas. Joint embeddings of shapes and images via cnn image purification. *ACM transactions on graphics*, 34(6):1–12, 2015. 2

[29] Yuan Liu, Zehong Shen, Zhixuan Lin, Sida Peng, Hujun Bao, and Xiaowei Zhou. Gift: Learning transformation-invariant dense visual descriptors via group cnns. In *NeurIPS*, 2019. 2, 7

[30] 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*, 2020. 2, 3, 4, 5, 7

[31] 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*, 2017. 7

[32] Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *NeurIPS*, 2017. 4, 7, 8

[33] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al. Fastslam: A factored solution to the simultaneous localization and mapping problem. *AAAI*, 593598, 2002. 1

[34] Jogendra Nath Kundu, Aditya Ganeshan, and R Venkatesh Babu. Object pose estimation from monocular image using multi-view keypoint correspondence. In *ECCV*, 2018. 1

[35] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *ICCV*, 2017. 2

[36] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: learning local features from images. In *NeurIPS*, 2018. 2

[37] Rémi Pautrat, Viktor Larsson, Martin R Oswald, and Marc Pollefeys. Online invariance selection for local feature descriptors. *arXiv:2007.08988*, 2020. 2, 7

[38] 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*, 2020. 1, 2

[39] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yohann Cabon, and Martin Humenberger. R2d2: Repeatable and reliable detector and descriptor. *arXiv:1906.06195*, 2019. 2, 5, 7

[40] Renato F Salas-Moreno, Richard A Newcombe, Hauke Strasdat, Paul HJ Kelly, and Andrew J Davison. Slam++: Simultaneous localisation and mapping at the level of objects. In *CVPR*, 2013. 1

[41] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & effective prioritized matching for large-scale image-based localization. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1744–1756, 2016. 1, 7

[42] Nikolay Savinov, Akihito Seki, Lubor Ladicky, Torsten Sattler, and Marc Pollefeys. Quad-networks: unsupervised learning to rank for interest point detection. In *CVPR*, 2017. 2

[43] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1

[44] 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*, 2013. 5, 6

[45] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *ICCV*, 2015. 2

[46] 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*, 2020. 4, 8

[47] Hajime Taira, Masatoshi Okutomi, Torsten Sattler, Mircea Cimpoi, Marc Pollefeys, Josef Sivic, Tomas Pajdla, and Akihiko Torii. Inloc: Indoor visual localization with dense matching and view synthesis. In *CVPR*, 2018. 7

[48] 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*, 2019. 3

[49] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *CVPR*, 2017. 2

[50] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *CVPR*, 2019. 2, 7

[51] Bing Wang, Changhao Chen, Chris Xiaoxuan Lu, Peijun Zhao, Niki Trigoni, and Andrew Markham. Atloc: Attention guided camera localization. In *AAAI*, 2020. 7

[52] Olivia Wiles, Sebastien Ehrhardt, and Andrew Zisserman. D2d: Learning to find good correspondences for image matching and manipulation. *arXiv:2007.08480*, 2020. 7

[53] Xiaoxia Xing, Yinghao Cai, Tao Lu, Shaojun Cai, Yiping Yang, and Dayong Wen. 3dtnet: Learning local features using 2d and 3d cues. In *3DV*, 2018. 2

[54] Luwei Yang, Ziqian Bai, Chengzhou Tang, Honghua Li, Yasutaka Furukawa, and Ping Tan. Sanet: Scene agnostic network for camera localization. In *ICCV*, 2019. 7

[55] Zi Jian Yew and Gim Hee Lee. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In *ECCV*, 2018. 2

[56] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *ECCV*, 2016. 2

[57] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *CVPR*, 2017. 2, 5, 8

[58] Linguang Zhang and Szymon Rusinkiewicz. Learning to detect features in texture images. In *CVPR*, 2018. 2

[59] Lei Zhou, Siyu Zhu, Zixin Luo, Tianwei Shen, Runze Zhang, Mingmin Zhen, Tian Fang, and Long Quan. Learning and matching multi-view descriptors for registration of point clouds. In *ECCV*, 2018. 2