

THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

P2-net: Joint description and detection of local features for pixel and point matching

Citation for published version:
Wang, B, Chen, C, Cui, Z, Qin, J, Lu, CX, Yu, Z, Zhao, P, Dong, Z, Zhu, F & Trigoni, N 2022, P2-net: Joint description and detection of local features for pixel and point matching. in *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2021 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, pp. 15984-15993, International Conference on Computer Vision 2021, 11/10/21. https://doi.org/10.1109/ICCV48922.2021.01570

Digital Object Identifier (DOI):
10.1109/ICCV48922.2021.01570

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Proceedings of the IEEE/CVF International Conference on Computer Vision

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

OPEN ACCESS

Download date: 29. Mar. 2024

This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

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[8], Niki Trigoni[1], Andrew Markham[1]

[1]University of Oxford [3]Zhejiang University [5]University of Edinburgh [7]Wuhan University
[2]National University of Defense Technology [4]Nanjing University of Aeronautics and Astronautics [6]Durham [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* `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.

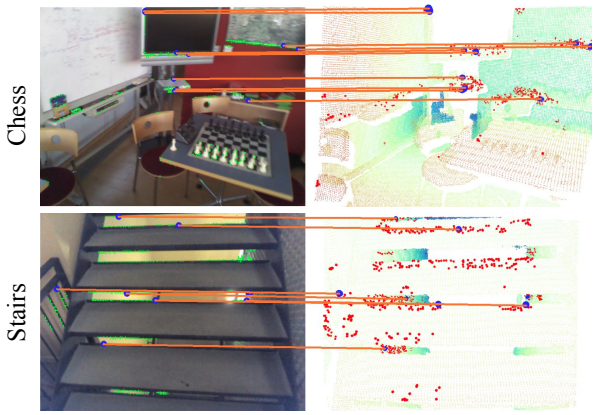


Figure 1: **Examples of 2D-3D matches obtained by the P2-Net.** The proposed method can directly establish correspondences across images and point clouds by the jointly learned feature description and detection.

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 **P**ixel and **P**oint 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 rebuilding 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-

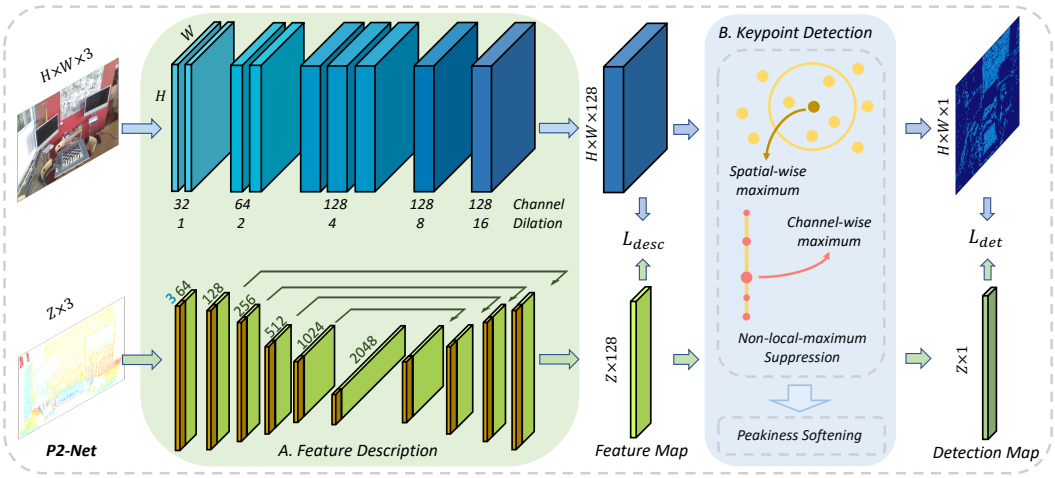


Figure 2: **An overview of the proposed P2-Net framework.** Our architecture is a two-branch fully convolutional network for the simultaneous 2D and 3D feature description (A) and keypoint detection (B). Such a network is jointly optimized with a descriptor loss $\mathcal{L}_{desc}$ enforcing the similarity of corresponding representations as well as a detector loss $\mathcal{L}_{det}$ encouraging higher detection scores for discriminative correspondences.

iary 2D features extracted from 2D patches. Recently, both 2D3DMatch-Net [20] and LCD [39] 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 are not applicable in real usages requiring high-resolution outputs. In contrast, we aim to extract per-point descriptors and detect keypoint locations in a single forward pass for efficient usage.

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,

$$d_X = F^I_X, \;\; d_Y = F^P_Y, d \in \mathbb{R}^C \;\; . \tag{1}$$

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.

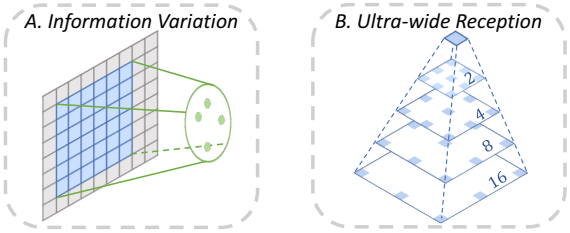


Figure 3: To mitigate the intrinsic information variation (A) between 2D and 3D local regions, an ultra-wide reception mechanism (B) with progressively doubling dilation values, up to 16, is applied in the 2D branch of feature description.

As shown in Fig. 2.A, two fully convolutional networks are exploited to separately perform feature description on images and point clouds. However, properly associating pixels with points through descriptors is non-trivial because of the intrinsic variation in information density between 2D and 3D local regions (Fig. 3.A). Specifically, the local information extracted for 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 reception mechanism, shown in Fig. 3.B. For computational efficiency, such a mechanism is achieved through nine 3×3 convolutional layers with progressively doubling dilation values, from 1 to 16. Finally, a $H\times W\times 128$ feature map is generated and then its corresponding $H\times W\times 1$ detection map can be computed. In a similar vein, we modify KPconv [49] to output a 128D descriptor and a score for each point in the input point cloud.

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

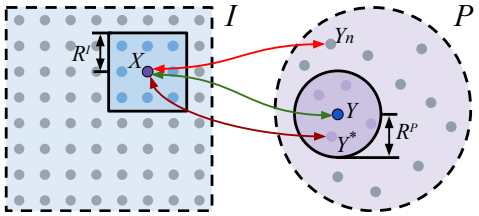


Figure 4: **Definitions of pixel-point pairs.** In a pair of image I and point cloud P, $\boldsymbol{X}\leftrightarrow\boldsymbol{Y}$ is a correspondence (pixel $X\in I$ and point $Y\in P$). From the image perspective, $\boldsymbol{X}\leftrightarrow\boldsymbol{Y_n}$ demonstrates a negative match where Y_n lies outside R^P (the neighborhood of Y), denoting a negative point of X. $\boldsymbol{X}\leftrightarrow\boldsymbol{Y^*}$ 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.

$F \in \mathbb{R}^{T\times C}$, where $T{=}H{\times}W$ for images and $T{=}Z$ for point clouds. The requirement for a pixel or point ρ_t to be detected by a non-local-maximum suppression is

$$\begin{aligned}\rho_t \text{ is a detection } \Longleftrightarrow F_t^c \text{ is a local max in } F^c_{\mathcal{R}_{\rho_t}}\\ \text{with } c=\arg\max_k F_t^k \ ,\end{aligned} \tag{2}$$

in which F_t^c represents the feature response at the position t and channel c. $\mathcal{R}_{\rho_t}$ denotes the neighborhood of ρ_t.

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

$$\begin{aligned}\alpha_t^c &= \text{softplus}(F_t^c - \frac{1}{|\mathcal{R}_{\rho_t}|}\sum_{\rho_{t'}\in\mathcal{R}_{\rho_t}} F_{t'}^c) \ ,\\ \beta_t^c &= \text{softplus}(F_t^c - \frac{1}{C}\sum_k F_t^k) \ ,\end{aligned} \tag{3}$$

where α and β are the spatial-wise and channel-wise detection scores, respectively. The final keypoint detection score of ρ_t that takes both criteria into account is:

$$\xi_{\rho_t} = \max_c(\alpha_t^c\beta_t^c) \ . \tag{4}$$

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**:

$$\mathcal{L}_{P2} = \mathcal{L}_{desc} + \lambda\mathcal{L}_{det} \ . \tag{5}$$

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 λ 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 $\boldsymbol{X}\leftrightarrow\boldsymbol{Y}$ 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:

$$s_p = d_X d_Y = \sum_c d_X^c d_Y^c, \tag{6}$$

From the view of image, we fully sample negative pairs $\boldsymbol{X}\leftrightarrow\boldsymbol{Y_n}$ and define a negative cosine similarity set s_n^I for all negative descriptor pairs d_X and $d_{Y_n^{1,\cdots,j}}$ as:

$$s_n^I = \left\{d_X d_{Y_n^1}, \cdots, d_X d_{Y_n^j}\right\}, \ \text{s.t. } \|Y_n^j - Y\|_2 > R^P, \tag{7}$$

where Y_n^j 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:

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

in which $\mathcal{C}$ 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 $\eta_n^j{=}\zeta(s_n^{I_i^j}{+}m)$ represent weighting factors with a scale factor ζ that expect $s_p^i{>}1{-}m$ and $s_n^{I_i^j}{<}m$ in a self-paced manner. With that said, the margin m controls the radius of the circular decision boundary at $(s_n^{I_i^j}{-}0)^2{+}(s_p^i{-}1)^2{=}2m^2$. 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 description 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 $\boldsymbol{X}\leftrightarrow\boldsymbol{Y^*}$ in Fig. 4, Y_i^* is determined by $\arg\max_{Y_i^l\neq Y_i}(d_{Y_i^l}d_{X_i})$ 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 $\max(d_{X_i}d_{Y_i^*}, d_{Y_i}d_{X_i^*})$. Additionally, ξ_{X_i} and ξ_{Y_i} are the soft detection scores at pixel X_i and point Y_i, respectively. With above definitions, we then formualte the batch-hard detector loss as:

$$\mathcal{L}_{det} = \sum_{i \in \mathcal{C}} \frac{\xi_{X_i} \xi_{Y_i}}{\sum_{q \in \mathcal{C}} \xi_{X_q} \xi_{Y_q}} (s^i_{n^*} - s^i_p), \tag{9}$$

Intuitively, such a detector loss seeks for higher detection scores for more discriminative correspondences. Specifically, L_{det} expects ξ_{X_i} and ξ_{Y_i} 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, $|\mathcal{C}|$=128 correspondences are randomly sampled from each pair to optimize in each step. We set the balance factor λ=1, the margin m=0.2, scale factor ζ=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.

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

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 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 (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,

# Scenes	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs
Feature Matching Recall							
SIFT + SIFT3D	Not Match						
P2[D2_Triplet]	Not Converge						
P2[D3_Contrastive]	Not Converge						
P2[R2D2]	95.1	97.3	**100**	89.4	91.1	88.7	16.2
P2[ASL]	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[Full]	**100**	**100**	**100**	**97.3**	**98.5**	**96.3**	**88.8**
Registration Recall							
P2[R2D2]	81.0	78.5	73.1	79.7	75.6	77.1	60.8
P2[ASL]	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[Full]	**87.0**	**82.4**	**84.5**	**83.4**	**88.7**	**82.7**	**82.6**
Keypoint Repeatability							
P2[R2D2]	36.6	40.3	45.2	33.4	30.3	32.1	33.1
P2[ASL]	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[Full]	**50.4**	**47.1**	**50.2**	**38.0**	**45.2**	**38.3**	**48.1**
Recall							
P2[R2D2]	28.5	26.7	24.7	25.0	24.6	26.4	16.0
P2[ASL]	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[Full]	**32.7**	**33.7**	**26.6**	**30.6**	**29.6**	**32.3**	**20.1**
Inlier Ratio							
P2[R2D2]	65.5	66.5	69.8	54.0	54.5	55.3	38.5
P2[ASL]	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[Full]	**73.9**	**76.0**	**77.4**	**60.3**	60.8	**65.2**	**45.2**

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

our P2[Full] performs consistently better regarding all evaluation metrics, outperforming all 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 P2-Net trained with our circle-guided descriptor loss and :1) without a detector but with randomly sampled keypoints during inference (P2[w/o Det]); 2) without a detector but with conventional SIFT and SIFT3D keypoints (P2[Mixed]); 3) with the original D2-Net detector (P2[D2_Det]) [18]; 4) with the D3Feat detector (P2[D3_Det]) [2]; 5) with our batch-hard detector loss but using randomly sampled keypoints for testing (P2[Rand]) to indicate the superiority of our proposed detector.

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[Full] 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[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[2].

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[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 [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

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

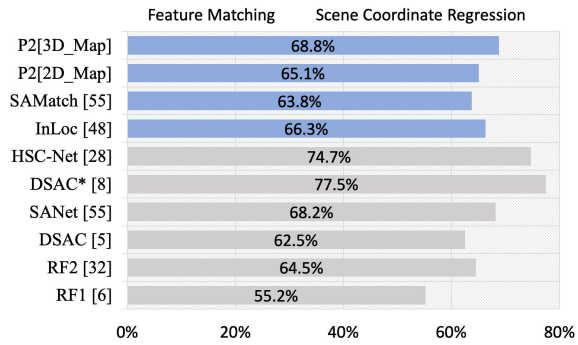


Figure 5: **Comparisons on visual localization.** Percentage of estimated camera poses falling within ($5cm$, $5°$).

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 $\frac{1}{10}$ 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. 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

		SP [16]	D2-Net [18] Triplet	D2-Net [18] **Ours**	ASLFeat [31] Contra	ASLFeat [31] **Ours**
Illum	HEstimation	0.877	0.818	**0.857**	**0.919**	0.915
	Precision	0.629	0.650	**0.664**	0.774	**0.787**
	Recall	0.565	**0.564**	0.560	0.696	**0.726**
View	HEstimation	0.651	0.553	**0.581**	0.542	**0.598**
	Precision	0.595	0.564	**0.576**	0.708	**0.740**
	Recall	0.446	0.382	**0.413**	0.583	**0.625**

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

	FCGF [12]			D3Feat [2]			**D3_Ours**		
	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 3DMatch [58]**. Reg, FMR and IR are evaluated at the threshold of 0.2 m, 5% and 0.1 m.

[26, 15, 23, 12, 11, 22, 10]. We follow the same evaluation protocols in [58] 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 [12] 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 P2-Loss is adopted (D3_Ours), a 4% and a 3% improvements (compared with D3Feat) can be seen on Reg and FMR, respectively. In contrast, there is only 2% and 0% respective difference between FCGF and D3Feat. In particular, as for Inlier Ratio, our loss demonstrates better robustness, outperforming D3Feat by 13%, comparable to FCGF. Overall, P2-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. 9, 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.

Fig. 6 (left) shows that, in single/homogeneous 2D or 3D domains, both D2-Net and D3Feat can gradually learn distinctive 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. 6 (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 [9, 33]. 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 [47].

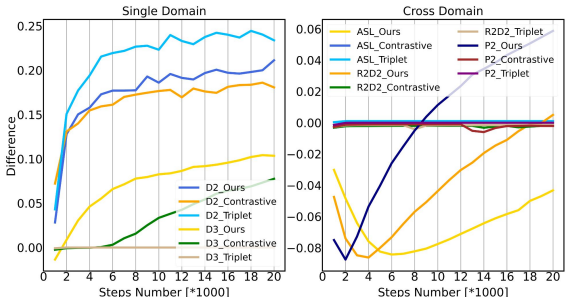


Figure 6: The difference between the positive similarity d_p and the most negative similarity d_{n^*} over time with different networks and losses. Left: single-domain matching; Right: cross-domain matching.

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 *Abs*(gradients), ranging between $[0, 1]$, with respect to d_p and d_n almost approach 1 and 0 [47], respectively. Because of the sharp gradient difference, the loss minimization in network training will tend to overemphasize the optimization of 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.

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

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

[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.

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

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

[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.

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

[8] Eric Brachmann and Carsten Rother. Visual camera relocalization from rgb and rgb-d images using dsac. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

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

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

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

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

[13] 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.

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

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

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

[17] 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.

[18] 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.

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

[20] 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.

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

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

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

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

[25] 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.

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

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

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

[29] 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.

[30] 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.

[31] 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.

[32] 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.

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

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

[35] Jogendra Nath Kundu, Aditya Ganeshan, and R Venkatesh Babu. Object pose estimation from monoc-

ular image using multi-view keypoint correspondence. In *ECCV*, 2018.

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

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

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

[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*, 2020.

[40] 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.

[41] 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.

[42] 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.

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

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

[45] 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.

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

[47] 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.

[48] 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.

[49] 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.

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

[51] 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.

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

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

[54] 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.

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

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

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

[58] 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.

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

[60] 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.