Rethinking Low-level Features for Interest Point Detection and Description

Changhao Wang, Guanwen Zhang✉, Zhengyun Cheng, and Wei Zhou

Northwestern Polytechnical University, Xi'an, China
guanwen.zh@.nwpu.edu.cn

Abstract. Although great efforts have been made for interest point detection and description, the current learning-based methods that use high-level features from the higher layers of Convolutional Neural Networks (CNN) do not completely outperform the conventional methods. On the one hand, interest points are semantically ill-defined and high-level features that emphasize semantic information are not adequate to describe interest points; On the other hand, the existing methods using low-level information usually perform detection on multi-level feature maps, which is time consuming for real time applications. To address these problems, we propose a Low-level descriptor-Aware Network (LANet) for interest point detection and description in self-supervised learning. Specifically, the proposed LANet exploits the low-level features for interest point description while using high-level features for interest point detection. Experimental results demonstrate that LANet achieves state-of-the-art performance on the homography estimation benchmark. Notably, the proposed LANet is a front-end feature learning framework that can be deployed in downstream tasks that require interest points with high-quality descriptors. (Code is available on https://github.com/wangch-g/lanet.)

1 Introduction

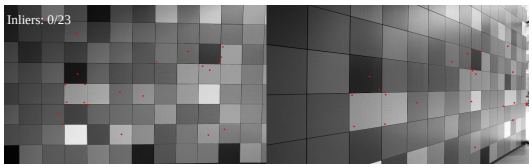
Interest point detection and description aims to propose reliable 2D points with representative descriptors for associating the 2D points projected from the same 3D point across images. It is an important task in many computer vision fileds, such as Camera Localiztion [28, 38], Structure-from-Motion (SfM) [23, 29], and Simultaneous Localization and Mapping (SLAM) [20, 21].

The conventional methods mainly use hand-crafted features for interest point detection and description. These methods focus on low-level features such as edges, gradients, and corners of high-contrast regions of the images, and try to design the features that can be detectable even under changes in image viewpoint, scale, noise, and illumination [2, 13, 14, 17, 26]. In recent years, deep learning has been extensively applied in computer vision fields, and learning-based methods for interest point detection and description are becoming increasingly prevalent. These methods expect to leverage the feature learning ability of the deep neural network and to extract the high-level features to outperform

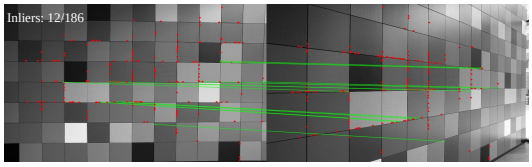
hand-crafted methods. In earlier literature, Convolutional Neural Networks (CNN) are used independently to learn the local descriptor based on the cropped image patches for detected points [9,18,22,32,40]. However, due to the weak capability of the existing detector and the patch-based descriptor, these methods easily produce inaccurate point locations and generate numerous wrong matches. Nowadays, researches perform the end-to-end manners to jointly learn detection and description based on the feature map extracted from the higher layers of CNN [6–8,19,24,37].

Although great efforts have been made for jointly learning detection and description, the learning-based methods fail to achieve significant improvements as expected as that in other computer vision tasks. The hand-crafted features still reveal high-quality description for interest points. Such as SIFT [17], its descriptors show distinctive and stable characteristics of interest points and are able to achieve remarkable performance [3]. Recently, some learning-based works proposed to leverage multi-level information for interest points detection and description [8,19]. These methods mainly focus on using the low-level and high-level features together as the multi-level features for detection and description. However, performing detection on multi-level feature maps is time consuming, and it is unfavourable for running interest points algorithm in real time for some downstream tasks. Besides, the interest points are semantically ill-defined and high-level features that emphasize semantic information are not adequate to describe interest points. Therefore, the low-level features tend to be appropriate for description, while the high-level features are effective for interest point detection. The two types of features exhibit complementary advantages relative to each other. From this perspective, one natural but less explored idea is to combine the advantages from both.

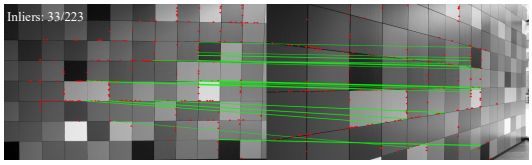
In this paper, we propose a Low-level descriptor-Aware Network (LANet) for interest point detection and description in self-supervised learning manner. The proposed LANet has a multi-task network architecture that consists of an interest point detection module, a low-level description-aware module, and a correspondence module. Similarly, the interest point detection module, following the previous works UnsuperPoint [6] and IO-Net [37], uses an encoder-decoder architecture to estimate locations and confidence scores of the interest points. Differently, we do not use a description decoder head after the encoder but a low-level description-aware module to directly exploit the low-level features from the lower layers of the encoder architecture for the learning of descriptors. Meanwhile, we introduce the learnable descriptor-aware scores to emphasize informative features of interest from different layers softly. Furthermore, to take full advantage of the pseudo-ground truth information, we introduce the correspondence module that takes the descriptor pairs to predict the correspondences between the detected points. The predicted correspondences is self-supervised by pseudo-ground truth labels for enhancing the learning of the interest point detection module and the low-level description-aware module. We evaluate the proposed LANet on the popular HPatches (Fig. 1), the experimental results demonstrate the proposed LANet is able to detect reliable interest points with high-quality descriptors



(a) SIFT + NN ($\epsilon = 1$)



(b) IO-Net + NN ($\epsilon = 1$)



(c) LANet (ours) + NN ($\epsilon = 1$)

Fig. 1. A challenging example. Compared with SIFT and the most recent self-supervised method IO-Net, our proposed LANet can obtain stronger descriptors for interest point matching and achieve the best performance. The results are obtained by the nearest neighbor (NN) matcher with the correct distance threshold $\epsilon = 1$. The inlier matches are linked in green while the mismatched points are marked in red.

that can be deployed in downstream tasks and outperforms the state-of-the-art methods on the homography estimation benchmark.

2 Related work

Patch-based description. For interest point detection and description, both hand-crafted descriptors [2, 14, 17, 26] and early learning-based descriptors [9, 32, 40] are computed from the local patches around the detected points. ORB [26] and SIFT [17] are considered to be the most representative conventional methods, and they are widely used in practical 3D computer vision applications [20, 21]. With the progressive development of deep learning, the performance of CNN-based methods has been gradually improved. LF-Net [22] proposes to use the depth and relative camera pose cues as the supervisory signals to learn local descriptors from image patches. To leverage contextual information, ContextDesc [18] exploits the visual context from high-level image representation and the geometric context from the distribution of interest points for learning local descriptors.

Jointly learned detection and description. Recent years, numerous works have paid increased attention to simultaneously learning detectioin and description within a single CNN architecture [6–8, 19, 24, 37]. D2-Net [8] proposes to perform detection and description simultaneously via one CNN architecture, which tightly couples the learning of interest point locations and descriptors. To

learn low-level details, ASLFeat [19] uses a multilevel learning mechanism based on the backbone of D2-Net [8] for interest point detection and description. In addition, instead of using expensive ground truth supervision, SuperPoint [7] employs an encoder-decoder architecture that contains two decoder branches, a detection decoder, and a description decoder, to learn the detector and descriptors jointly with pseudo-ground truth labels that are generated from MagicPoint [7]. On the basis of SuperPoint, UnsuperPoint [6], a self-supervised learning framework for interest point matching, is trained by the siamese scheme to learn the scores, locations, and descriptors of interest points automatically with unlabeled images. Most recently, IO-Net [37] is proposed based on UnsuperPoint, which learns a detector and descriptors with supervision from the proposed outlier rejection loss.

Finding good matches. With detected points and descriptors, using nearest neighbor search based on the similarity of descriptors can obtain a set of matched points. However, building correspondences on the basis of descriptors simply may cause a mass of outliers [3, 27, 33, 41, 42]. To alleviate the problem of false correspondences, some studies, such as RANSAC [10] and GMS [4, 5], leverage geometric or statistical information to filter outliers. Recently, SuperGlue [27] proposes to address the matching problem with Graph Neural Network (GNN) in a learning-based manner. Inspired by SuperGlue, LoFTR [33], a detector-free approach, proposes to obtain high-quality matches with transformers [39]. Because our method is a front-end feature learning framework, it can be further enhanced by being embedded with a SuperGlue-like learnable matching algorithm.

3 Method

The proposed LANet consists of three substantial modules, *i.e.*, interest point detection module, low-level description-aware module, and correspondence module. An overview of the proposed LANet is depicted in Fig. 2.

3.1 Interest point detection module

The interest point detection module is constructed based on the backbone of UnsuperPoint [6] and IO-Net [37] with several modifications. As shown in Fig. 2, the interest point detection module has a location decoder, a score decoder, and a shared encoder. It aims to estimate locations, confidence scores, and descriptor-aware scores of the interest points. During the training stage, in a self-supervised learning manner [6], the interest point detection module disposes a source image I_S and a target image I_T with a siamese framework to predict interest points. Notably, the target image I_T is transformed from I_S by a random generated homography $\boldsymbol{H}_{S\to T}$, which allows us to train the proposed model without any human annotation.

Encoder. We use a convolutional structure as the encoder for extracting the features from the input images. The encoder consists of four convolutional blocks

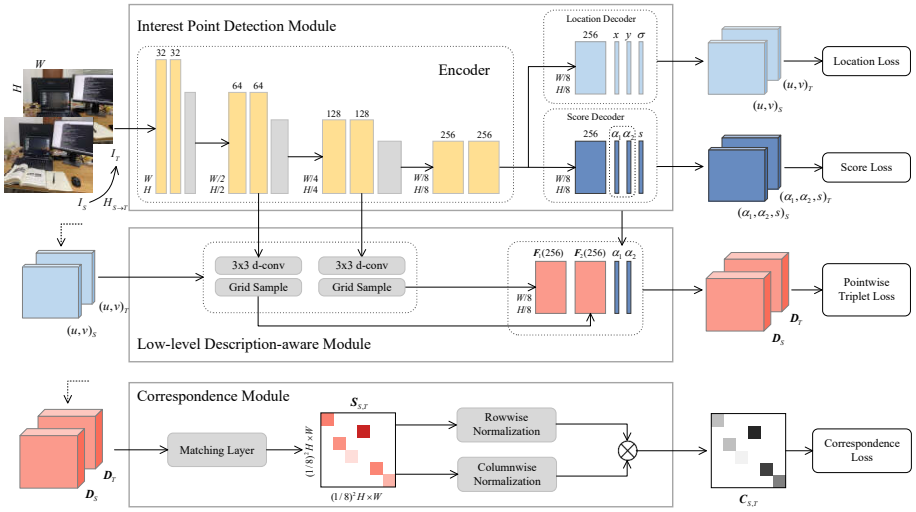


Fig. 2. Overview of LANet. The proposed LANet consists of an interest point detection module (Section 3.1), a low-level description-aware module (Section 3.2), and a correspondence module (Section 3.3). The interest point detection module is an encoder-decoder architecture that is used for estimating the point locations (u, v) , the confidence scores s, and the descriptor-aware scores α. The low-level description-aware module exploits the low-level information from the lower layers of the encoder for producing the descriptors $\boldsymbol{D}$ of the detected points. The correspondence module is introduced to improve the supervision of the first two modules by estimating the correspondence matrix $\boldsymbol{C}$ of the detected points between the input image-pair.

with 32, 64, 128, and 256 output channels. Each of the convolutional blocks contains two 3×3 convolutional layers followed by a batch normalization [11] and a ReLU activation function. The encoder uses three max-pooling layers with kernel size and stride of 2×2 among the four convolutional blocks to downsample the input image from $H \times W$ to $H/8 \times W/8$. We denote the pixels in the downsampled feature maps as cells as in [7]. Therefore, a 1×1 cell in the last feature map corresponds to 8×8 pixels in the raw image.

Location decoder. The location decoder contains two 3×3 convolutional layers with channels of 256 and 3 respectively. The location decoder takes the last feature map produced from the encoder as input and outputs a location map with 3 channels. The first two channels of a cell in the location map are denoted as (x, y), which indicates the normalized location of each interest point relative to the center of the corresponding 8×8 grid in the raw image. In addition, we denote the third channel of a cell in the location map by σ. σ is a learnable rectification factor for calculating the absolute pixel location of each interest point, which allows the predicted points across cell boundaries [37]. The pixel

location of each interest point is defined as:

$$(u, v) = (u', v') + \frac{\sigma(r-1)}{2}(x, y), \forall(x, y) \in [-1, 1], \tag{1}$$

where (u, v) and (u', v') are the pixel location and the cell center location of the corresponding interest point, respectively. In Eq. 1, r is the down sample ratio of the interest point detection module ($r = 8$ in this paper).

Score decoder. In the interest point detection module, the score decoder has the same structure as the location decoder. The score decoder outputs the score map with 3 channels. A cell in the first channel of the score map is the confidence score of the corresponding interest point. We normalize the confidence score to $[0, 1]$ and denote it by s. The score s is used for selecting the best K points that are most convinced for downstream tasks. Additionally, a cell in the last two channels of the score map is the descriptor-aware score α that is introduced to emphasize informative features of interest softly. We use Softmax to perform the channelwise normalization for the descriptor-aware scores.

3.2 Low-level description-aware module

From the core insights of this work, we do not learn descriptors with an extra description decoder but exploit the concrete low-level features directly from the lower layers of the backbone encoder. Specifically, as shown in Fig. 2, we take the output feature maps of the second and the third convolutional block of the encoder as the basic descriptor maps and employ two separate dilated 3×3 convolutional layers to increase the channel dimension of the basic descriptor maps to 256. The dilated 3×3 convolution has lager receptive filed with the dilation is set as 2, which is helpful for capturing more local information in the low-level feature maps. The descriptors, corresponding to the detected interest points from the basic descriptor maps, are warped by performing a differentiable grid sample operation [12, 15]. Then, the warped descriptors are L2-normalized and denoted by $\boldsymbol{F}$. The warped descriptors $\boldsymbol{F}$ from different convolutional blocks are aggregated with the descriptor-aware scores α to generate the final descriptors $\boldsymbol{D}$ by taking a weighted sum.

3.3 Correspondence module

The descriptors are used to correspond to the detected points based on matching algorithms for further applications. In recent years, several studies calculate the similarity matrix of descriptors between input pairs and use a differentiable matching layer to learn good correspondences [25, 27, 33].

Inspired by previous works, we introduce the correspondence module to predict the correspondence matrix by the descriptors of detected points between I_S and I_T, which is employed as an auxiliary task to supervise the training of the interest point detection module and the low-level description-aware module.

First, we calculate the similarity matrix $\boldsymbol{S}_{S,T}$ between the source descriptors $\boldsymbol{D}_S$ and the target descriptors $\boldsymbol{D}_T$ as:

$$\boldsymbol{S}_{S,T}^{i,j} =< \boldsymbol{D}_S^i, \boldsymbol{D}_T^j >,\ \boldsymbol{A} \in \mathbb{R}^{(1/8)^2 HW \times (1/8)^2 HW}, \tag{2}$$

where $< \cdot\ ,\ \cdot >$ is the inner product. With the similarity matrix $\boldsymbol{S}_{S,T}$ in size of $(1/8)^2 HW \times (1/8)^2 HW$, the correspondence module performs dual-softmax [25] to normalize the $\boldsymbol{S}_{S,T}$ in rowwise order and columnwise order respectively, and computes the correspondence matrix $\boldsymbol{C}_{S,T}$ as:

$$\boldsymbol{C}_{S,T}^{i,j} = \text{softmax}(\boldsymbol{S}_{S,T}^i)_i \cdot \text{softmax}(\boldsymbol{S}_{S,T}^j)_j,\ \boldsymbol{C} \in \mathbb{R}^{(1/8)^2 HW \times (1/8)^2 HW}. \tag{3}$$

The predicted correspondence matrix $\boldsymbol{C}_{S,T}$ reveals the interest point matching correlations between I_S and I_T. It is self-supervised by pseudo-ground truth matching labels as an auxiliary task to enhance the learning of the low-level description-aware module.

3.4 Optimization

The proposed LANet is optimized by four loss functions: the location loss L_{loc}, the score loss L_s, the pointwise triplet loss L_{tri}, and the correspondence loss L_{corr}. The final loss function L is balanced by trade-off parameters λ as:

$$L = \lambda_{loc} L_{loc} + \lambda_s L_s + \lambda_{tri} L_{tri} + \lambda_{corr} L_{corr}. \tag{4}$$

Location supervision. During the training stage, we can obtain the location of detected points in raw input images through the interest point detection module. Following the self-supervised learning scheme [6], we warp the source points (detected points in the source image I_S) to the target image with the known homography $\boldsymbol{H}_{S \to T}$ and find the nearest neighbor of the warped points among the target points (detected points in the target image I_T) using an L2-distance. A source point and its nearest neighbor in the target image are associated as a nearest point pair if the distance between them is less than a threshold ϵ. We denote a source point and its associated target point in a nearest point pair as p_S and p_T, respectively. The distance between a nearest point pair is defined as:

$$d(p_S, p_T) = ||\boldsymbol{H}_{S \to T}(p_S) - p_T||_2, \tag{5}$$

and the location loss can then be formulated as:

$$L_{loc} = \frac{1}{N} \sum_i^N d(p_S^i, p_T^i), \tag{6}$$

where N is the number of nearest point pairs.

Score supervision. Following [6, 37], the score loss L_s should increase the confidence score s of the relatively convinced points. In addition, it should jointly

keep the coherence of the confidence score s and the descriptor-aware score α between the nearest point pairs. Thus, we define the score loss as:

$$\begin{aligned} L_s = & \frac{1}{N}\sum_{i}^{N}[\frac{(s_S^i + s_T^i)}{2}(d(p_S^i, p_T^i) - \bar{d}) + (s_S^i - s_T^i)^2 \\ & + (\alpha_S^{1,i} - \alpha_T^{1,i})^2 + (\alpha_S^{2,i} - \alpha_T^{2,i})^2], \end{aligned} \tag{7}$$

where $\bar{d}$ is the average distance of the nearest point pairs. In Eq. 7, the first term is the confidence constraint that allows the nearest point pairs with closer distances to have higher confidence scores, the last three terms are the coherence constraints that ensure the consistency between the confidence score and the descriptor-aware scores of the nearest point pairs.

Description supervision. We use the pointwise triplet loss [30, 34, 36] for learning high-quality descriptors. In a triplet sample, the *anchor* and the *positive* are the descriptors of a source point and its matched target point, respectively. We denote the *anchor* and the *positive* as $\boldsymbol{D}_{p^i}$ and $\boldsymbol{D}_{p_+^i}$. The *negative* of the triplet is sampled among the descriptors of the unmatched target points with the hardest negative sample mining [36] and denoted by $\boldsymbol{D}_{p_-^i}$. The pointwise triplet loss is formulated as:

$$L_{tri} = \frac{1}{N}\sum_{i}^{N}[||\boldsymbol{D}_{p^i}, \boldsymbol{D}_{p_+^i}||_2 - ||\boldsymbol{D}_{p^i}, \boldsymbol{D}_{p_-^i}||_2 + \beta]_+, \tag{8}$$

where β is the distance margin for metric learning.

Correspondence supervision. We introduce the correspondence loss to optimize the predicted correspondence matrix $\boldsymbol{C}_{S,T}$. The correspondence loss is a negative log-likelihood loss that contains a positive term and a negative term, which is formulated as:

$$\begin{aligned} L_{corr} = & -[\frac{1}{|\boldsymbol{M}_{pos}|}\sum_{(i,j)\in\boldsymbol{M}_{pos}}\log \boldsymbol{C}_{S,T}^{i,j} \\ & + \gamma\cdot\frac{1}{|\boldsymbol{M}_{neg}|}\sum_{(i,j)\in\boldsymbol{M}_{neg}}\log(1-\boldsymbol{C}_{S,T}^{i,j})], \end{aligned} \tag{9}$$

where the $\boldsymbol{M}_{pos}$ and the $\boldsymbol{M}_{neg}$ are the sets of positive matches and negative matches respectively, γ is a hyperparameter used for balance the two terms. Notably, the operations in the correspondence module are all differentiable. Therefore, the gradient of the correspondence loss can be propagated back to the low-level description-aware module for auxiliary supervision.

3.5 Deployment

During the testing stage, only the interest point detection module and the low-level description-aware module of the proposed LANet are employed to predict

interest points with corresponding scores and descriptors. The scores are used to select the most convinced interest points for downstream applications. The proposed LANet is a novel feature extraction approach that provides interest points with high-quality descriptors, thus it can be introduced as a front-end solver embedded with existing matchers such as SuperGlue [27].

4 Experiments

4.1 Details

We use 118k images from COCO 2017 dataset [16] without any human annotation to train the proposed LANet. We perform spatial augmentation on the training dataset with scaling, rotation, and perspective transformation to generate the self-supervised signal for the siamese training scheme. The input images are resized to 240×320 and the augmentation settings are same as [37]. The proposed LANet is trained by Adam optimizer for 12 epochs with a batch size of 8. The learning rate is set to 3×10^{-4} and is reduced by a factor of 0.5 after 4 epochs and 8 epochs. The distance threshold ϵ between the point locations is set to 4.0 during the training process. The trade-off parameters of the loss function in Eq. 4 are set to $\lambda_{loc} = 1.0$, $\lambda_s = 1.0$, $\lambda_{\alpha-tri} = 4.0$, and $\lambda_{corr} = 0.5$, respectively. The pointwise triplet loss margin β in Eq. 8 is set to 1.0. The factor γ in Eq. 9 is set to 5×10^5.

4.2 Comparison

We evaluate the proposed LANet on HPatches dataset [1] which contains 116 scenes with dramatic changes in illumination or viewpoint. We use the metrics of Repeatability (Re), Localization Error (LE), Homography Estimation Accuracy with tolerance threshold $\epsilon = 1$ pixel (H-1), $\epsilon = 3$ pixels (H-3), $\epsilon = 5$ pixels (H-5), and Matching Score (MS) [7]. The evaluation metrics are measured with top P points selected according to the confidence scores. We report the results on testing images of 240×320 resolution ($P = 300$) and 480×640 resolution ($P = 1000$) in Table 1. Besides, we report the Mean Matching Accuracy (MMA) [8] with the error threshold of 3 pixels in Table 2.

Repeatability and localization error. The higher repeatability represents a higher probability that the same interest points can be detected in different images, and the lower localization error indicates that the pixel locations of detected points are more precise. The repeatability and the localization error are two basic metrics for evaluating the capability of an interest point detector. In the lower resolution settings, the proposed LANet achieves competitive performance with IO-Net and KP3D in repeatability. LANet has a lower localization error compared with that of IO-Net while approaching the performance of UnsuperPoint, SIFT, and KP3D. In the higher resolution settings, the proposed LANet achieves the second best performance both in repeatability and localization error.

Table 1. Comparisons on HPatches dataset with repeatability, localization error, homography estimation accuracy, and matching score.

Methods	240 × 320, 300 points						480 × 640, 1000 points					
	Re↑	LE↓	H-1↑	H-3↑	H-5↑	MS↑	Re↑	LE↓	H-1↑	H-3↑	H-5↑	MS↑
ORB [26]	0.532	1.429	0.131	0.422	0.540	0.218	0.525	1.430	0.286	0.607	0.710	0.204
SURF [2]	0.491	1.150	0.397	0.702	0.762	0.255	0.468	1.244	0.421	0.745	0.812	0.230
BRISK [14]	0.566	1.077	0.414	0.767	0.826	0.258	0.505	1.207	0.300	0.653	0.746	0.211
SIFT [17]	0.451	0.855	0.622	0.845	0.878	0.304	0.421	1.011	**0.602**	0.833	0.876	0.265
LF-Net(indoor) [22]	0.486	1.341	0.183	0.628	0.779	0.326	0.467	1.385	0.231	0.679	0.803	0.287
LF-Net(outdoor) [22]	0.538	1.084	0.347	0.728	0.831	0.296	0.523	1.183	0.400	0.745	0.834	0.241
SuperPoint [7]	0.631	1.109	0.491	0.833	0.893	0.318	0.593	1.212	0.509	0.834	0.900	0.281
UnsuperPoint [6]	0.645	0.832	0.579	0.855	0.903	0.424	0.612	0.991	0.493	0.843	0.905	0.383
IO-Net [37]	**0.686**	0.970	0.591	0.867	0.912	0.544	**0.684**	0.970	0.564	0.851	0.907	0.510
KP3D [35]	**0.686**	**0.799**	0.532	0.858	0.906	**0.578**	0.674	**0.886**	0.529	0.867	0.920	0.529
LANet (ours)	0.683	0.874	**0.662**	**0.910**	**0.941**	0.577	0.682	0.943	**0.602**	**0.874**	**0.924**	**0.543**

Table 2. Comparisons on HPatches dataset with mean matching accuracy (MMA).

MMA@3	SIFT [17]	D2Net(SS) [8]	D2Net(MS) [8]	SuperPoint [7]	ASLFeat [19]	**Ours**
Illumination	0.525	0.568	0.468	0.738	-	**0.822**
Viewpoint	0.540	0.354	0.385	**0.639**	-	0.620
Overall	0.533	0.457	0.425	0.686	**0.723**	0.717

Homography estimation accuracy. In practice, the correspondence between detected points is established based on the similarity of their descriptors. With the matched points, we can estimate the homography transformation between the input pairs with geometric constraint. In the experiments, we use the nearest neighbor matcher to associate the detected points across images and compute the homography matrix using OpenCV's *findHomography* method. As shown in Table 1, the proposed LANet surpasses the existing learning-based methods by a very large margin with different tolerance threshold in both lower and higher resolution settings.

Matching score. Matching score measures the probability of the correct correspondences over the points detected in shared viewpoints, which evaluates the general performance of the whole detection and description pipeline. Our method achieves almost the same highest matching score as KP3D in lower resolution setting, while it outperforms the other methods by a large margin in higher resolution setting as shown in Table 1.

Mean matching accuracy. Mean matching accuracy is the ratio of correct matches for each image pair. As shown in Table 2, the proposed LANet shows surprisingly effectiveness on illumination sequences and achieves competitive result on viewpoint sequences. Compared with ASLFeat, the proposed LANet performs on par with it on overall performance.

Table 3. Ablation study.

Method	σ	LD-α	CL	240 × 320, 300 points					
				Re↑	LE↓	H-1↑	H-3↑	H-5↑	MS↑
Baseline	-	-	-	0.633	1.049	0.486	0.812	0.893	0.525
LANet	✓	-	-	0.685	0.874	0.545	0.862	0.914	0.580
	✓	✓	-	0.682	0.861	0.653	0.898	0.922	0.572
	✓	✓	✓	0.683	0.874	0.662	0.910	0.941	0.577

4.3 Ablation study

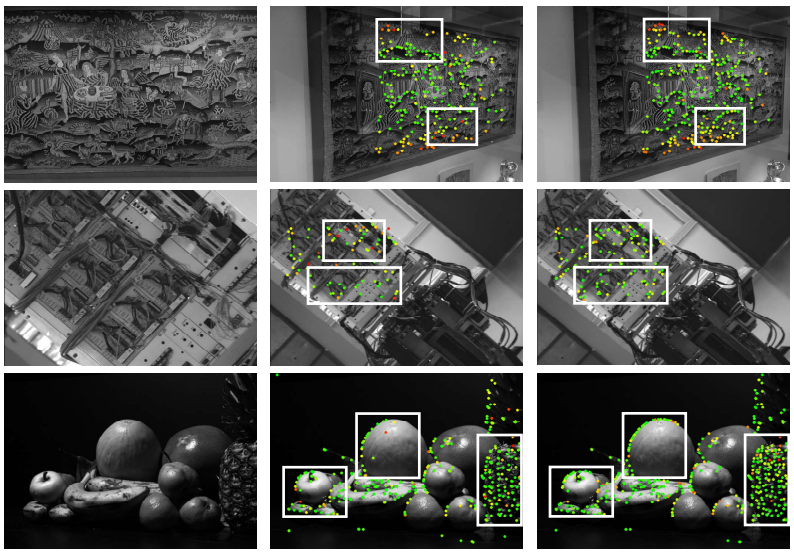
In this section, we perform an ablation study on HPatches dataset to demonstrate the effectiveness of each module in our proposed method. The results are summarized in Table 3.

Baseline. We use the encoder-decoder architecture of the interest point detection module as the baseline. The baseline outputs the descriptors by a 3×3 convolutional layer that is connected after the 4th convolutional block of the backbone encoder. Besides, the rectification factor σ is fixed to 1 during the training process and the descriptor-aware score α is removed.

Ablation on the learned $\boldsymbol{\sigma}$. The learnable rectification factor σ enables the detection decoder to predict the locations across cell borders as described in [37]. Different from [37], we optimize the rectification factor σ with the predicted locations during the training stage rather than setting the σ as a default hyperparameter. Compared with the baseline, using the learnable rectification factor σ improves the overall performance of the detection and description obviously.

Ablation on the low-level descriptor-aware module. The low-level description-aware module (LD-α) learns descriptors from the lower convolution layers and the learned descriptors are weighted sum by the descriptor-aware score α. As shown in Table 3, compared with the results of second row, the low-level description-aware module boosts the homography estimation accuracy with +10.8% in H-1, +3.6% in H-3, and +0.8% in H-5 respectively, while only having slight influence on the repeatability, the localization error and the matching score. In Fig. 3, we visualize the interest points matching results that are obtained with and without the low-level descriptor-aware module. Compared with Fig. 3(b), Fig. 3(c) shows that using low-level features for description can learn stronger descriptors to acquire more precise and denser matches for interest point matching.

Ablation on the correspondence loss. The correspondence loss (CL) is introduced as an auxiliary supervision to enhance the learning of descriptors. With the correspondence loss, the proposed LANet achieves further improvement on homography estimation accuracy and achieves the best overall performance.



(a) (b) (c)

Fig. 3. Interest point matching with and without the low-level descriptor-aware module. (a) Source images. (b) Target images (processed using baseline + σ). (c) Target images (processed using baseline + σ + LD-α). The matched points are marked in different colors with the localization error.

4.4 Validation on low-level feature learning

In this section, we conduct further study on learning descriptors from different layers individually to validate the effectiveness of low-level features. We use the interest point detection module of the proposed LANet as the basic network for the experiments. We extract the learned features from different convolutional blocks for evaluation. Meanwhile, we deepen the network by adding a descriptor decoder with upsample blocks for further comparisons. Each upsample block consists of an upsampling layer [31] and a basic convolutional block. Notably, the output layer is a 3×3 convolution to reshape the descriptors to the same size of 256 dimensions. The experimental results are summarized in Table 4.

Comparisons within the backbone encoder. From Table 4 we can see that the learned descriptors from the deeper layers have better performance on repeatability and matching score. However, the learned descriptors from different layers do not affect the localization error obviously. As for homography estimation accuracy, the learned descriptors from the lowest three convolutional blocks significantly boost the accuracy of H-1, H-3, and H-5 compared with that of the deeper layer. The learned descriptors from the convolutional block 2 achieve the best performance on H-1 (0.669), H-3 (0.895), and H-5 (0.934) at the same time.

Backbone encoder vs. description decoder. With a deeper structure and higher-resolution feature map, the descriptors extracted from the upsam-

Table 4. Validation on low-level features extracted from different layers.

Output layer in description		Depth	Output resolution	240 × 320, 300 points					
				Re↑	LE↓	H-1↑	H-3↑	H-5↑	MS↑
Backbone encoder	conv block 1	2	240 × 320	0.669	0.804	0.641	0.872	0.900	0.403
	conv block 2	4	120 × 160	0.679	0.880	0.669	0.895	0.934	0.506
	conv block 3	6	60 × 80	0.685	0.842	0.624	0.878	0.921	0.571
	conv block 4	8	30 × 40	0.685	0.874	0.545	0.862	0.914	0.580
Description decoder	upsample block 5	10	60 × 80	0.692	0.866	0.566	0.866	0.928	0.583
	upsample block 6	12	120 × 160	0.685	0.857	0.562	0.872	0.922	0.571
	upsample block 7	14	240 × 320	0.682	0.862	0.553	0.866	0.916	0.559

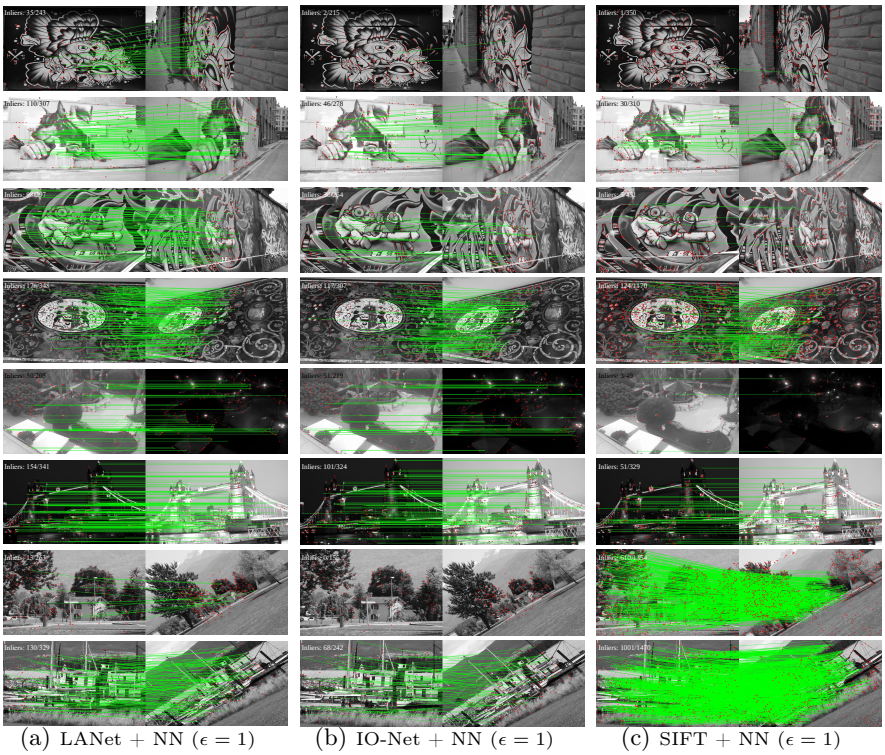
ple block 5 increase the homography estimation accuracy on H-1, H-3, and H-5 compared with that extracted from the convolutional block 4. However, compared with that of upsample block 5, using the deeper network could not further improve the performance. By using the feature map in the same resolution, the learned descriptors from the convolutional blocks 2 and 3 outperform those extracted from the deeper upsample blocks 5 and 6 by a large margin on homography estimation accuracy.

Observations. Based on the above comparisons, we can see that the low-level features can effectively improve the distinguishability of descriptors and significantly increase the homography estimation accuracy. Meanwhile, the low-level features have barely effects on the point repeatability and the localization error. It should be noted, although the descriptors extracted from the convolutional block 2 achieve the best performance on homography estimation accuracy, they do have an inferior matching score. Therefore, it is a promising solution to combine the low-level features extracted from the convolutional blocks 2 and 3 to balance the overall performance.

4.5 Further analysis

Fig. 4 shows a few qualitative matching results comparing the SIFT, IO-Net, and the proposed LANet on HPatches dataset. The results are obtained by using the nearest neighbor (NN) matcher with the correct distance threshold $\epsilon = 1$. The proposed LANet outperforms IO-Net and SIFT obviously in perspective scenes (Fig. 4 Row 1 to 4) and is on par with IO-Net in illumination scenes (Fig. 4 Row 5 and 6).

As shown in Row 7 and 8 of Fig. 4, when the images are rotated dramatically, the proposed LANet easily fails in the interest points detection and description. Whereas, the SIFT provides rotation invariant features and has the best performance compared with IO-Net and the proposed LANet. Using the largely rotated images during the training process could help to improve the performance of the proposed LANet in such cases. However, it will affect overall performance since most of the images in the dataset have moderate rotation. One promising solution is to introduce the smooth penalty function during the optimization.



(a) LANet + NN ($\epsilon = 1$) (b) IO-Net + NN ($\epsilon = 1$) (c) SIFT + NN ($\epsilon = 1$)

Fig. 4. Matching results visualization. The examples of perspective scenes are shown in row 1 - 4 and the examples of illumination scenes are shown in row 5 and 6. We also show some inferior examples obtained in the largely rotated cases in row 7 and 8.

5 Conclusion

In this paper, we proposed a Low-level descriptor-Aware Network (LANet) for interest points detection and description in self-supervised scheme, which exploits the low-level features to learn adequate descriptors for interest points. Besides, we introduce a correspondence loss as an auxiliary supervision to enhance the learning of interest point descriptor. We show that using low-level features for description while using high-level features for detection is a feasible solution for interest point matching. Extensive experimental results demonstrate that the proposed LANet using low-level features can improve the distinguishability of interest point descriptor and outperform the popularly used methods.

Acknowledgements This work was supported in part by the National Key R&D Program of China (2018AAA0102801 and 2018AAA0102803), and in part of the National Natural Science Foundation of China (61772424, 61702418, and 61602383).

References

1. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: CVPR (2017)
2. Bay, H., Ess, A., Tuytelaars, T., Gool, L.V.: Speeded-up robust features (SURF). Comput. Vis. Image Underst. **110**(3), 346–359 (2008)
3. Bhowmik, A., Gumhold, S., Rother, C., Brachmann, E.: Reinforced feature points: Optimizing feature detection and description for a high-level task. In: CVPR (2020)
4. Bian, J., Lin, W., Liu, Y., Zhang, L., Yeung, S., Cheng, M., Reid, I.: GMS: grid-based motion statistics for fast, ultra-robust feature correspondence. Int. J. Comput. Vis. **128**(6), 1580–1593 (2020)
5. Bian, J., Lin, W., Matsushita, Y., Yeung, S., Nguyen, T., Cheng, M.: GMS: grid-based motion statistics for fast, ultra-robust feature correspondence. In: CVPR (2017)
6. Christiansen, P.H., Kragh, M.F., Brodskiy, Y., Karstoft, H.: Unsuperpoint: End-to-end unsupervised interest point detector and descriptor. arXiv: 1907.04011 (2019)
7. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: CVPR Workshops (2018)
8. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-net: A trainable CNN for joint description and detection of local features. In: CVPR (2019)
9. Ebel, P., Trulls, E., Yi, K.M., Fua, P., Mishchuk, A.: Beyond cartesian representations for local descriptors. In: ICCV (2019)
10. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6), 381–395 (1981)
11. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: ICML (2015)
12. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. In: NeurIPS (2015)
13. Ke, Y., Sukthankar, R.: PCA-SIFT: A more distinctive representation for local image descriptors. In: CVPR (2004)
14. Leutenegger, S., Chli, M., Siegwart, R.: BRISK: binary robust invariant scalable keypoints. In: ICCV (2011)
15. Li, X., You, A., Zhu, Z., Zhao, H., Yang, M., Yang, K., Tan, S., Tong, Y.: Semantic flow for fast and accurate scene parsing. In: ECCV (2020)
16. Lin, T., Maire, M., Belongie, S.J., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In: ECCV (2014)
17. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
18. Luo, Z., Shen, T., Zhou, L., Zhang, J., Yao, Y., Li, S., Fang, T., Quan, L.: Contextdesc: Local descriptor augmentation with cross-modality context. In: CVPR (2019)
19. Luo, Z., Zhou, L., Bai, X., Chen, H., Zhang, J., Yao, Y., Li, S., Fang, T., Quan, L.: Aslfeat: Learning local features of accurate shape and localization. In: CVPR (2020)
20. Mur-Artal, R., Montiel, J.M.M., Tardós, J.D.: ORB-SLAM: A versatile and accurate monocular SLAM system. IEEE Trans. Robotics **31**(5), 1147–1163 (2015)
21. Mur-Artal, R., Tardós, J.D.: ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. IEEE Trans. Robotics **33**(5), 1255–1262 (2017)

22. Ono, Y., Trulls, E., Fua, P., Yi, K.M.: Lf-net: Learning local features from images. In: NeurIPS (2018)
23. Pittaluga, F., Koppal, S.J., Kang, S.B., Sinha, S.N.: Revealing scenes by inverting structure from motion reconstructions. In: CVPR (2019)
24. Revaud, J., de Souza, C.R., Humenberger, M., Weinzaepfel, P.: R2D2: reliable and repeatable detector and descriptor. In: NeurIPS (2019)
25. Rocco, I., Cimpoi, M., Arandjelović, R., Torii, A., Pajdla, T., Sivic, J.: Neighbourhood consensus networks. In: NeurIPS (2018)
26. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: ORB: an efficient alternative to SIFT or SURF. In: ICCV (2011)
27. Sarlin, P., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: CVPR (2020)
28. Sarlin, P., Unagar, A., Larsson, M., Germain, H., Toft, C., Larsson, V., Pollefeys, M., Lepetit, V., Hammarstrand, L., Kahl, F., Sattler, T.: Back to the feature: Learning robust camera localization from pixels to pose. In: CVPR (2021)
29. Schönberger, J.L., Frahm, J.: Structure-from-motion revisited. In: CVPR (2016)
30. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: CVPR (2015)
31. Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: CVPR (2016)
32. Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P., Moreno-Noguer, F.: Discriminative learning of deep convolutional feature point descriptors. In: ICCV (2015)
33. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: Loftr: Detector-free local feature matching with transformers. In: CVPR (2021)
34. Sun, Y., Xu, Q., Li, Y., Zhang, C., Li, Y., Wang, S., Sun, J.: Perceive where to focus: Learning visibility-aware part-level features for partial person re-identification. In: CVPR (2019)
35. Tang, J., Ambrus, R., Guizilini, V., Pillai, S., Kim, H., Jensfelt, P., Gaidon, A.: Self-supervised 3d keypoint learning for ego-motion estimation. In: CoRL (2020)
36. Tang, J., Folkesson, J., Jensfelt, P.: Geometric correspondence network for camera motion estimation. IEEE Robotics Autom. Lett. **3**(2), 1010–1017 (2018)
37. Tang, J., Kim, H., Guizilini, V., Pillai, S., Ambrus, R.: Neural outlier rejection for self-supervised keypoint learning. In: ICLR (2020)
38. Tang, S., Tang, C., Huang, R., Zhu, S., Tan, P.: Learning camera localization via dense scene matching. In: CVPR (2021)
39. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
40. Yi, K.M., Trulls, E., Lepetit, V., Fua, P.: LIFT: learned invariant feature transform. In: ECCV (2016)
41. Yi, K.M., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., Fua, P.: Learning to find good correspondences. In: CVPR (2018)
42. Zhao, C., Cao, Z., Li, C., Li, X., Yang, J.: Nm-net: Mining reliable neighbors for robust feature correspondences. In: CVPR (2019)