MTLDesc: Looking Wider to Describe Better

Changwei Wang[1,4,*], **Rongtao Xu**[1,4,*], **Yuyang Zhang**[1,4],
Shibiao Xu[2,†], **Weiliang Meng**[1,3,4,†], **Bin Fan**[5], **Xiaopeng Zhang**[1,4]

[1]NLPR, Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, Beijing University of Posts and Telecommunications
[3]Zhejiang Lab; [4]School of Artificial Intelligence, University of Chinese Academy of Sciences
[5]School of Automation and Electrical Engineering, University of Science and Technology Beijing
{wangchangwei2019, xurongtao2019, yuyang.zhang, weiliang.meng, xiaopeng.zhang}@ia.ac.cn,
shibiaoxu@bupt.edu.cn, bin.fan@ieee.org

Abstract

Limited by the locality of convolutional neural networks, most existing local features description methods only learn local descriptors with local information and lack awareness of global and surrounding spatial context. In this work, we focus on making local descriptors "look wider to describe better" by learning local **Desc**riptors with **M**ore **T**han just **L**ocal information (**MTLDesc**). Specifically, we resort to context augmentation and spatial attention mechanisms to make our MTLDesc obtain non-local awareness. First, Adaptive Global Context Augmented Module and Diverse Local Context Augmented Module are proposed to construct robust local descriptors with context information from global to local. Second, Consistent Attention Weighted Triplet Loss is designed to integrate spatial attention awareness into both optimization and matching stages of local descriptors learning. Third, Local Features Detection with Feature Pyramid is given to obtain more stable and accurate keypoints localization. With the above innovations, the performance of our MTLDesc significantly surpasses the prior state-of-the-art local descriptors on HPatches, Aachen Day-Night localization and In-Loc indoor localization benchmarks. Our code is available at https://github.com/vignywang/MTLDesc.

Introduction

Local descriptors currently play a key role in various vision applications such as image matching, image retrieval, SfM, SLAM, and visual localization. With the industry's rapid development, these applications must deal with more complex and challenging scenarios (various conditions such as day, night, and seasons). As the local features detection and description are the critical for these applications, there is an urgent need to further boost their performance.

Geoffrey Hinton said that "local ambiguities have to be resolved by finding the best global interpretation" in his first paper (Hinton 1976). This idea still holds true in local descriptors learning. There are two main weaknesses for learning local descriptors only using limited local visual infor-
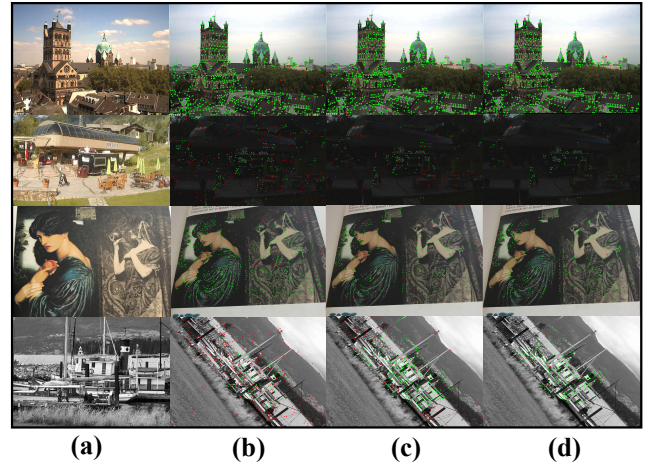

Figure 1: Matching results under illumination and viewpoint changes. (a): Matching images. (b): Baseline (SuperPoint). (c): MTLDesc w/ Context Augmentation but w/o Consistent Attention Weighting. (d): MTLDesc both w/ Context Augmentation and w/ Consistent Attention Weighting. Green dots: Correct matches. Red dots: Incorrect matches.

mation: i) Ambiguity regions with repetitive patterns (texture, color, shape, etc.) is difficult to be distinguished only by local information, as shown in the first row (trees) and the fourth row (river and ground) in Fig. 1 (**b**); ii) Local visual information becomes unreliable and indistinguishable for challenging scenes with large illumination and viewpoint changes, which will lead to massive incorrect matches (the second and fourth rows of Fig. 1 (**b**)). On this account, robust non-local context can be employed to better distinguish challenging local regions. We propose Context Augmentation and Consistent Attention Weighting to look wider for describing better, enabling our descriptors to gain awareness beyond the local region, in turn to effectively mitigate above weaknesses as shown in Fig. 1 (**d**).

CNN-based backbones like L2Net (Tian, Fan, and Wu 2017) and VGG (Simonyan and Zisserman 2014) are widely adopted by the local descriptors learning methods. Due to the inherent locality of CNN, features can only be extracted

in a limited receptive field. Although some of these methods (Luo et al. 2019, 2020; Tyszkiewicz, Fua, and Trulls 2020) can implicitly alleviate this problem by extracting features in a larger receptive field, they still only considered the context in a fixed patch-wise neighborhood. By contrast, our method further utilizes the context of the global and different receptive fields by the well-designed Adaptive Global Context Augmented Module (**AGCA**) and Diverse Local Context Augmented Module (**DLCA**).

In addition, the attention mechanism is also an effective way to use non-local information. When humans observe and describe images, they will quickly analyze spatial information and focus their attention on some key regions. Some image retrieval practices (Cao, Araujo, and Sim 2020) have indicated that the network can also obtain similar awareness with the attention mechanism. Based on the above inspiration, we design a new Consistent Attention Mechanism for both optimization and matching of local descriptors by a well-designed Consistent Attention Weighted Triplet Loss.

Furthermore, as the localization accuracy of keypoints also affects the results of local features matching, we propose a Local Features Detection with Feature Pyramid based on the classical scale-space (Lowe 2004a) and deep supervision (Lee et al. 2015) to obtain more stable and accurate keypoints localization. Besides, to meet the demand of practical applications, running speed should also be addressed. Therefore, the whole method is carefully designed to be as fast as possible. In summary, there are four main contributions in this paper: **1)** We devise AGCA and DLCA modules to aggregate effectively from global to local context information for local descriptors learning. **2)** We propose a novel Consistent Attention Weighted Triplet Loss to introduce spatial consistent attention awareness in both optimization and matching of local descriptors. **3)** We present the Local Features Detection with Feature Pyramid to improve the localization accuracy of keypoints. **4)** We provide a **real-time** solution for local features learning named MTLDesc, which achieves state-of-the-art on HPatches, Aachen-Day-Night and InLoc benchmarks.

Related Works

Local Descriptors Learning: Hand-crafted local descriptors are widely used in computer vision, and are comprehensively evaluated in (Mikolajczyk and Schmid 2005). Current deep learning based descriptors can roughly fall into two categories: patch-based descriptors and dense descriptors. Patch-based descriptors (Tian, Fan, and Wu 2017; Mishchuk et al. 2017; Tian et al. 2019; Luo et al. 2019) extract descriptors from corresponding patches of the detected keypoints (e.g. SIFT (Lowe 2004b)), while dense descriptors (DeTone, Malisiewicz, and Rabinovich 2018; Revaud et al. 2019; Dusmanu et al. 2019; Revaud et al. 2019; Luo et al. 2020; Wang et al. 2020b; Tyszkiewicz, Fua, and Trulls 2020) usually use a fully convolutional neural network (Long, Shelhamer, and Darrell 2015) to extract dense feature descriptors for the whole image in one forward pass. Dense descriptors have achieved good performance in image matching and long-term visual localization, showing great potential for practical applications. In contrast to these prior

works, we recommend the introduction of non-local information include both context and spatial attention awareness to local descriptors learning, aiming to make the local descriptors "look wider to describe better".

Context Awareness: Context awareness is essential for pixel-level tasks (Yu et al. 2018), but it has not been attracted widespread attention in local descriptors learning. ContextDesc (Luo et al. 2019) aggregates the cross-modality context for local descriptors, including visual context from a ResNet-50 (He et al. 2016) branch, and geometric context from keypoints distribution. Patch-based ContextDesc depends on a large network and needs to obtain the geometric context after additional keypoints detection calculation, so it consumes more computing time and memory space. Some CNN-based dense descriptors implicitly improve context awareness by increasing the receptive field. ASLFeat (Luo et al. 2020) proposes to use deformable convolution (Dai et al. 2017) to extract descriptors with shape context, MLIFeat (Zhang et al. 2020) utilizes hypercolumns (Hariharan et al. 2016) to fuse multi-level features, while DISK (Tyszkiewicz, Fua, and Trulls 2020) employs UNet-like backbone (Ronneberger, Fischer, and Brox 2015) to fuse multi-scale context information. However, all these descriptors only aggregate the context of a fixed receptive field and do not consider the global context. Recently, Visual Transformers (Wu et al. 2020) has shown the ability to aggregate global context in some computer vision applications (Carion et al. 2020; Wang et al. 2020a). In this work, we make the network get comprehensive context awareness from global to local. On the one hand, we use visual transformer and a learnable Gated Map to adaptively embed the global context and location information into local descriptors. On the other hand, we propose to flexibly learn local descriptors through surrounding contexts with different receptive fields.

Attention Mechanism: As a non-local awareness, spatial attention has been successfully applied to the learning of image-level global descriptors (Kalantidis, Mellina, and Osindero 2016; Noh et al. 2017; Cao, Araujo, and Sim 2020; Tolias, Jenicek, and Chum 2020) for image retrieval. In these methods, spatial attention is used as the weight of local descriptors, and global descriptors are derived from local descriptors through weighted summation. However, directly applying the local descriptors of these methods to image matching produces poor results, as reported by (Revaud et al. 2019; Dusmanu et al. 2019). This may be caused by the lack of supervision of local pixel correspondence. However, attention mechanisms in these methods are optimized with the supervision of image-level and it is not suitable for pixel correspondence supervision. In contrast, we proposes a special consistent attention mechanism to improve the optimization and matching of local descriptors for image matching.

Method

MTLDesc employs a fully convolutional network encoder as the **shared backbone** for both local features description and detection. The encoder consists of 3×3 convolutional layers, relu layers, and max-pooling layers. For a $h \times w$ image I, $C_1(h \times w)$, $C_2(h/2 \times w/2)$, $C_3(h/4 \times w/4)$, and $C_4(h/8 \times$
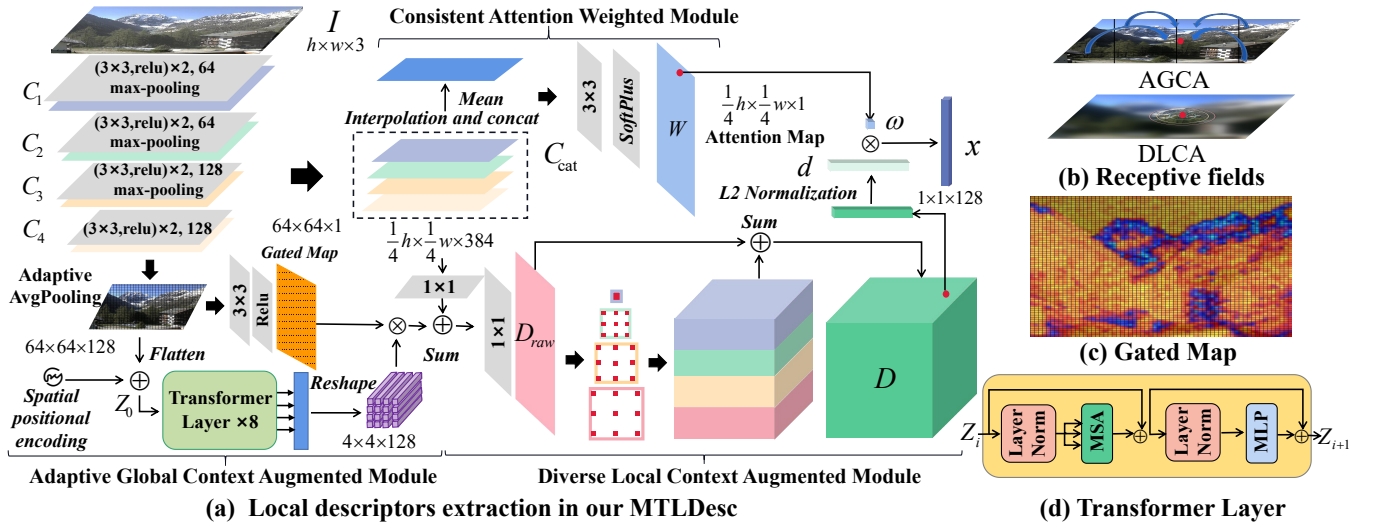
Figure 2: (a): Local descriptors extraction in our MTLDesc. (b): Receptive fields of proposed modules in (a). (c): Gated Map in (a). The values of the blue regions are 0 and the others are positive values. (d): Details of Transfomer Layer in (a).

$w/8$) feature maps are obtained after four sequential sub-encoders. MTLDesc detects keypoints (**Fig. 5**) and extracts corresponding local descriptors (**Fig. 2**) at the same time, and the two parts use the same shared backbone network.

Local Descriptors with Non-local Information

I. Local Descriptors with Context Augmentation: We propose Adaptive Global Context Augmented Module and Diverse Local Context Augmented Module to implement context augmentation from global to local, as shown in Fig. 2 (a), while Fig. 2 (b) shows the difference between the two modules about the receptive field, which leads them to aggregate context from different perspectives.

(1) Adaptive Global Context Augmented Module. Different image regions usually have different degrees of demand for global context. For regions that are difficult to describe only with local information (e.g. weak or repeated textures regions), the global context can introduce more spatial information to make it more discriminative. But for regions with good distinguishable local information, directly adding the global context may bring some noise. Our Adaptive Global Context Augmented Module is designed to adaptively introduce global context for local descriptors. Specifically, we take the feature maps C_4 through adaptive average pooling to obtain a fixed-size feature map (64×64) as the input of the module. Compared with previous visual transformers, our method can effectively reduce the computational complexity and adapt to images of any size. Following (Wu et al. 2020), we perform tokenization by reshaping the input into a sequence of flattened 2D patches X_p, and each patch is of size 16×16. We map the vectorized patches X_p a latent 128-dimensional embedding space using a trainable linear projection. In order to make the local descriptors obtain the spatial position information relative to the global, we learn specific position embeddings which are added to the patch embeddings to retain positional information as follows: $Z_0 = [X_p^1 E; X_p^2 E; ...; X_p^N E] + E_{pos}$, where E is the

patch embedding projection and E_{pos} is the position embedding. After Z_0 passes through 8 transformer layers, the hidden features with global context are obtained. The structure of transformer layer is shown in Fig. 2 (d). Where MSA denotes Multihead Self-Attention and MLP denotes Multi-Layer Perceptron block (Wu et al. 2020). After reshaping, we can get the patch-wise descriptors with the global context. Another branch of the module predicts a gated map to mask regions that do not require a global context supplement. We implement the gating mechanism through the relu activation function, and the visualization result of gated map is shown in Fig. 2 (c). Finally, we merge the global context filtered by the gated map into local descriptors.

(2) Diverse Local Context Augmented Module. The surrounding context is also crucial for local descriptors learning. We design a simple and effective Diverse Local Context Augmented Module to extract diverse surrounding contexts. Unlike most previous CNN-based local descriptors which only deploy the top-layer feature maps to extract descriptors, we recommend using all feature maps derived from the backbone to construct descriptors. Specifically, the C_1, C_2, C_3, C_4 are interpolated to the same spatial size and then aggregated together to get C_{cat}. The size is set to $1/4$ of the input image I as it gives a good trade-off between accuracy and speed. It improves the utilization of feature maps and integrates information of different scales. To obtain diverse surrounding contexts, we decouple the descriptor into four 32-dimensional sub-descriptors and learn them in different description spaces respectively. This ensures that the sub-descriptors remain independent and diversity to contain more information. Specifically, we use 1×1 Conv and three 3×3 dilated Conv (Yu and Koltun 2015) with dilation rates of 6, 12 and 18 respectively to derive descriptors from different receptive fields. After being concatenated and added to D_{raw}, the final dense descriptor D is obtained. The surrounding context from different receptive fields further stimulates the representation ability of local descriptors.
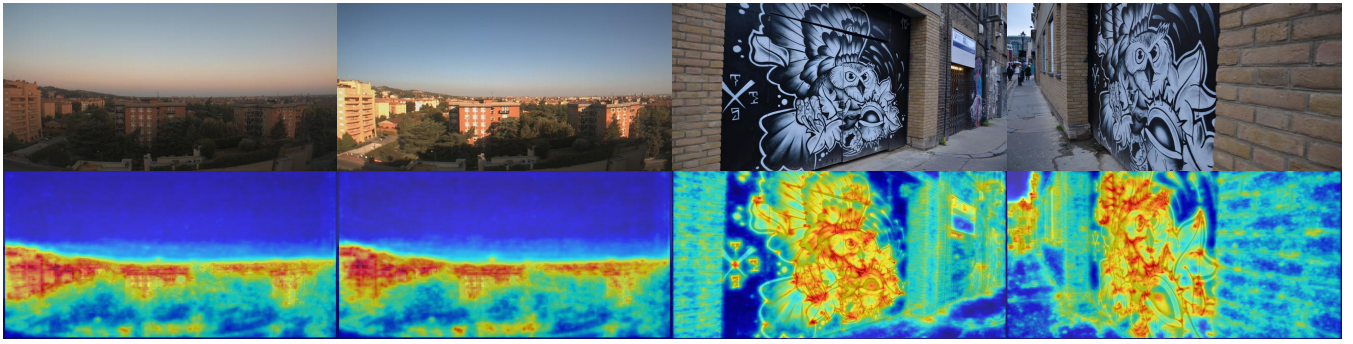
Figure 3: Images (first row) and Consistent Attention Maps (second row) under illumination or viewpoint changes. For Consistent Attention Maps, pixels close to red means higher attention score and close to blue means lower score. Meaningless regions (e.g. sky and ground) and regions with repetitive texture (e.g. trees and brick wall) are given low attention scores.

II. Local Descriptors with Consistent Attention Weighting: To further overcome the limitation of keypoints description only based on local information, we make the network imitate humans to obtain the awareness and insight of spatial information with a consistent attention mechanism. We claim that the proposed Consistent Attention has **three types of properties**: *i)* The same regions in different images have consistent attention scores. *ii)* Representative regions are given higher attention scores, as these regions easily match to inliers while distinguishable to outliers. *iii)* Descriptors from regions with high attention scores are optimized first. We now explain how we design the module and loss for applying consistent attention to make the optimization and matching of local descriptors better.

Consistent Attention Weighting Module is shown in Fig 2 (a). Specifically, C_{cat} is averaged across the channel dimension, and the attention map W is predicted from this averaged feature map via the 3×3 Conv + SoftPlus.

Consistent Attention weighted Triplet Loss is designed to jointly optimize local descriptors and consistent attention. Considering an image pair (I, I'), the dense descriptors D, D' and attention maps W, W' of I, I' can be extracted by our MTLDesc, as shown in Fig. 2 (a). Given the sampled points set P of size N in I and their corresponding points P' in I' , the corresponding descriptors of P, P' are denoted as d_i and $d'_i, i \in 1 \dots N$. The corresponding score of the descriptor d_i on the attention map W is ω_i, so the attention weighted descriptor is defined as $x_i = \omega_i \cdot d_i$. For x_i, its *positive distance* $||x_i||^+$ is defined as:

$$||x_i||^+ = ||\omega_i \cdot d_i - \omega'_i \cdot d'_i||_2, \qquad (1)$$

and its *hardest negative distance* $||x_i||^-$ is defined as:

$$||x_i||^- = \min_{j \in 1 \dots N, j \neq i}(||\omega_i \cdot d_i - \omega'_j \cdot d'_j||_2). \qquad (2)$$

The Consistent Attention Weighted Triplet Loss $\mathcal{L}_{\text{Atrip}}$ can be defined as:

$$\mathcal{L}_{\text{Atrip}}(x) = \frac{e^{\omega/T}}{\sum_{i=1}^{N} e^{\omega_i/T}} \max(0, ||x||^+ - ||x||^- + 1), \quad (3)$$

where ω is the attention score corresponding to x, and T is a smoothing factor. T is used to adjust the effect of attention weight on loss. We will explore the impact of T in next

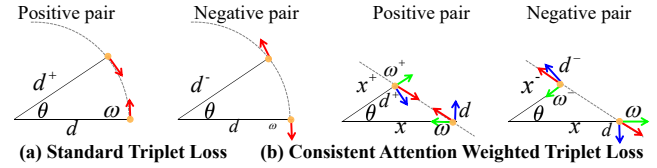(a) Standard Triplet Loss (b) Consistent Attention Weighted Triplet Loss

Figure 4: Example of the optimization direction of 2D descriptor. Red arrow: Gradient descent direction. Green arrow: Gradient component of consistent attention ω optimization. Blue arrows: Gradient component of descriptor d optimization.

section. The whole description loss is summed as:

$$\mathcal{L}_{\text{des}} = \sum_{i=1}^{N} \mathcal{L}_{\text{Atrip}}(x_i), \qquad (4)$$

(1) Consistent Attention in Optimization: First, we will discuss the difference between $\mathcal{L}_{\text{Atrip}}$ and standard triplet loss in the optimization direction. It is easy to validate the optimization of the L2 normalized descriptor's distance degenerates to angle's optimization (Tian et al. 2020), meaning that the common standard triplet loss only has the optimization of the angle component as shown in Fig. 4 (a). However, the optimization direction of our $\mathcal{L}_{\text{Atrip}}$ is decoupled into the component of the descriptor d (angle) optimization and the component of the consistent attention ω (weight) optimization in Fig. 4 (b). For descriptors, $\mathcal{L}_{\text{Atrip}}$ still optimizes the angle between them. For consistent attention, as shown in Fig. 4 (b), the attention scores of positive samples tend to convergent while the attention scores of negative samples tend to divergent. This trend will lead to the consistent distribution of attention scores (i.e., **property i)**) in corresponding regions of image pairs. The details are shown in Fig. 3.

Second, we will further explore the optimization goal of consistent attention based on the above discussion. As shown in Eq. 3, the consistent attention ω is affected by both triplet loss term $\max(0, ||x||^+ - ||x||^- + 1)$ which only provides consistency as mentioned before and softmax term $\frac{e^{\omega/T}}{\sum_{i=1}^{N} e^{\omega_i/T}}$ in optimization. To minimize the loss, the softmax term causes the network tends to give larger attention

ω to samples with smaller triplet loss term. These samples usually have large $||x||^-$ and small $||x||^+$, so they are more suitable for matching. To minimize the total loss, these two conditions need to be met together, meaning that consistent attention has **properties i)** and **ii)** as shown in Fig. 3.

Third, we will explore the role of attention score ω in the optimization of descriptor d by analyzing the gradient. Given weighted descriptor $x = \omega \cdot d$ and positive sample x^+, we can get the gradient of the positive distance to the descriptor d by partial derivative as following:

$$\frac{\partial ||x - x^+||_2}{\partial d} = \omega \cdot \frac{x - x^+}{||x - x^+||_2}. \quad (5)$$

In Eq. 5, it is obvious that the gradient of descriptors is weighted by the attention score. Moreover, as shown in Eq. 3, the gradient of d is also weighted by the softmax term in the whole loss. Note that softmax term is also proportional to w, so in summary, the sample with high attention score will contribute more gradients relatively (*i.e.,* **property iii)**). Obviously, not every sample's descriptor is worthy of equal optimization. Forcing learning descriptions on pixels that are not suitable as descriptors (e.g. sky, grass and waves) will bring noise and lead to sub-optimal results. Therefore, the local descriptor can be optimized more flexibly and selectively with the help of consistent attention weighting.

(2) Consistent Attention in Matching: Interestingly, we find that attention-weighted local descriptors are also more suitable for matching. As shown in Fig. 3, the corresponding regions have similar consistent attention scores in different images, so consistent attention can be used as a prior information in the local descriptors matching. It is evident that regions with high attention scores are more likely to be successfully matched with the same high scores regions in another image. Thus, the weighted descriptor has a smaller matching space and lead to higher matching accuracy.

Local Features Detection with Feature Pyramid

We adopt pixel-wise classification to train keypoints detector with pseudo-keypoints supervision. Different from Super-Point, we recommend predicting keypoints at different scale spaces by proposed Local Features Detection with Feature Pyramid. Specifically, we set four detection headers to predict the keypoint heatmaps respectively as shown in Fig 5. In order to combine the predict results, we interpolate the predicted heatmaps of different scales to the image size $h \times w$. We set four learnable weights to fuse heatmaps of different scales to predict final keypoints and calculate the loss. Each detection header can receive direct supervision of detector loss, which can be considered as deep supervision (Lee et al. 2015). Our method also conforms to the famous scale-space theory (Lowe 2004a) which is accepted by many methods (Barroso-Laguna et al. 2019; Luo et al. 2020), with the difference from them that we directly predict keypoints through supervised learning without additional statistics and calculations. We use the weighted binary cross-entropy loss as the detector loss since there is an extreme imbalance in the number of keypoints and non-keypoints. Given the predicted keypoints heatmap $K \in \mathbb{R}^{h \times w}$ and pseudo-ground truth label $G \in \mathbb{R}^{h \times w}$, the detector loss is defined as:

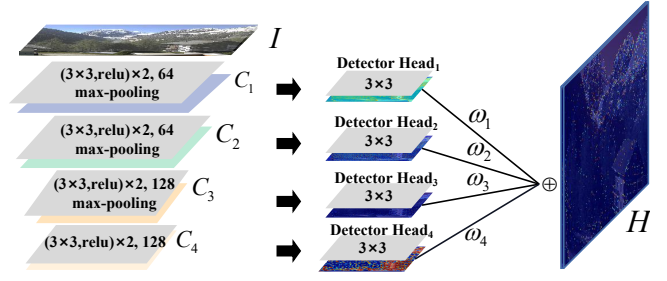$$\mathcal{L}_{\text{bce}}(k, g) = -\lambda g log(k) - (1 - g)log(1 - k), \quad (6)$$

Figure 5: Local Features Detection with Feature Pyramid.

$$\mathcal{L}_{\text{det}} = \frac{1}{hw} \sum_{u,v}^{h,w} \mathcal{L}_{\text{bce}}(K_{u,v}, G_{u,v}) \quad (7)$$

where the weight λ is empirically set to 200.

Training Strategy and Implementation Details

Data Preparation: We use MegaDepth (Li and Snavely 2018) to generate the training data with dense pixel-wise correspondences. MegaDepth dataset contains image pairs with known pose and depth information from 196 different scenes. Following the settings in D2-Net, we take 118 scenes from all scenes as the training set. We randomly select 100 image pairs from each scene, and intercept 400×400 image blocks from the original images for the training. Thus, we get $11,800$ image pairs with dense pixel correspondence. This part of the data contains real complex transformations, which are difficult to collect but closer to practical applications. Besides, we use random homography to synthesize more diverse image pairs to supplement the training data inspired by SuperPoint, further enriching the whole transformation types. In summary, our training data consists of $23,600$ image pairs in total. We compare our dataset settings with advanced methods in the appendix.

Keypoints Supervision with Distillation: We employ distillation to get the pseudo-keypoints ground truth directly from an off-the-shelf trained SuperPoint (DeTone, Malisiewicz, and Rabinovich 2018) (teacher model). To obtain reliable and more pseudo-labeling of keypoints, we use iterative homographic adaptation to obtain the probability map of keypoints heatmap.

Correspondences Supervision with Keypoints Heatmap Guidance: Obviously, not all locations are equally important. Forcing the network to train the descriptors in meaningless areas may lead to impaired performance. To get enough keypoints-specific and distributed diversely correspondences, we design a novel keypoints guided correspondences sampling method. Specifically, for each image pair I_1, I_2: 1) Predict keypoints heatmaps M_1, M_2 with a trained SuperPoint model. 2) Synthesize M_1' from M_2 based on the transformation between the image pair I_1, I_2. 3) Generate a compound keypoints heatmap by $M = M_1 + M_1'$ and divide this heatmap M into 40×40 grids. 4) Take the point with the largest score of each grid on M to obtain a candidate point set Q. 5) Apply the non-maximum suppression (NMS) to Q and select the top 400 points to construct the refined descriptor correspondences P.

Implementations: To optimize the keypoints detection and description jointly, the total loss is composed of detector loss $\mathcal{L}_{det}$ and descriptor loss $\mathcal{L}_{des}$, which is formulated as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{det}} + \mathcal{L}_{\text{des}}. \tag{8}$$

The Adam optimizer with poly learning rate policy is used to optimize the network, and the learning rate decays from 0.001. The training image size is set to 400×400 with the training batch size 12. The whole training process typically converges in 30 epochs and takes about 14 hours with a single NVIDIA Titan V GPU. During the testing, the detection threshold α is empirically set to 0.9 and non-maximum suppression (NMS) radius to 4 to balance the keypoints' number and reliability. Our method implemented by Pytorch[1] runs at **24 FPS** (real time) on 480×640 images with a single NVIDIA Titan V GPU.

Experiments

Comparisons on Image Matching

Dataset and Metrics: We use the popular HPatches benchmark (Balntas et al. 2017) for ablation studies and comparisons. Following previous methods, we use 108 sequences with viewpoint or illumination variations after excluding high-resolution sequences from 116 available sequences. The entire benchmark includes 56 sequences with changes in viewpoint and 52 sequences with changes in illumination. We use three standard metrics for evaluation: 1) Mean matching accuracy (*MMA*) is the average percentage of correct matches in image pairs under different matching error thresholds. 2) Match score (*M.S.*) is the ratio of the correct match to the total number of keypoints estimated in the shared view, following the definition in (Revaud et al. 2019). 3) Accuracy of homography (*HA*) is used to compare the estimated homography of image pairs with its corresponding ground truth.

Comprehensive Ablation Studies: Ablation studies are reported in Tab. 1. Our proposed data construction method is used to re-implement the SuperPoint named *our impl.* as a stronger baseline with higher *MMA* and *M.S.* scores compared to the SuperPoint *orig.* After applying proposed *FP Keypoints*, all metrics have been significantly improved, due to more accurate keypoints localization. Almost all metrics get steady improvement after incrementally applying AGCA and DLCA. Note that the Gated Map in AGCA can further improve the performance, especially for M.S. score. In addition, we also compared some related operations that can implicitly improve context awareness including DCN in (Luo et al. 2020), Hypercolumns in (Zhang et al. 2020), UNet-like backbone in (Tyszkiewicz, Fua, and Trulls 2020) and ASPP in (Chen et al. 2017). Our proposed Context Augmentation is superior to these alternative context augmentation operations. Furthermore, all metrics still have a significant improvement with only using consistent attention in optimization (*CA in Optimization*) and not using consistent attention in matching. This indicates that our proposed Consistent Attention Weighted Triplet Loss can independently improve the performance of the descriptors sub-

---

[1] We also implemented our method by using Mindspore (https://www.mindspore.cn/) and observed similar performance.

HPatches dataset @3				
Method	**Config**	**MMA %**	**M.S. %**	**HA %**
Baseline	*SuperPoint orig.*	64.44	42.41	72.59
	SuperPoint our impl.	67.51	43.54	71.48
	+ FP KeyPoints	**69.88**	**45.68**	**73.01**
+ Related Comparison Methods	*+ DCN*	70.56	44.19	72.14
	+ Hypercolumns	70.35	45.89	73.56
	+ ASPP	71.03	46.28	73.89
	+ UNet	71.15	45.78	73.34
+ Context Augmentation	*+ AGCA w/o Gated Map*	71.33	45.35	74.32
	+ AGCA	72.28	47.07	75.13
	+ DLCA	71.25	46.72	74.59
	+ AGCA & DLCA	**73.14**	**47.35**	**75.43**
+ Consistent Attention Weighting	*+ CA in Optimization Only*	74.94	**50.67**	77.03
	+ CA in Optimization and Matching (MTLDesc)	**78.66**	47.16	75.92
Current SOTA	*DISK (2K)*	76.09	44.36	68.14

Table 1: We report metrics at a 3px error threshold for different variants. FP Keypoints means Local features Detection with Feature Pyramid. AGCA means Adaptive Global Context Augmented Module. DLCA means Diverse Local Context Augmented Module. CA means Consistent Attention.
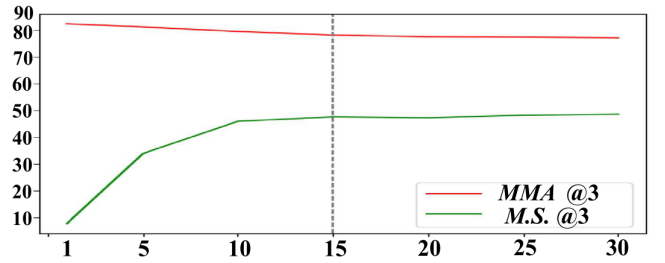

Figure 6: Different T settings are evaluated in HPatches.

stantially. Different from the above settings, the consistent attention weighted descriptors ($CA\ in\ Matching$) are also used for matching to obtain an *MMA* score of **78.66**. Our MTLDesc significantly exceeds the current state-of-the-art DISK (Tyszkiewicz, Fua, and Trulls 2020) on all metrics.

Impact of Smoothing Factor T: The T is used to adjust the effect of consistent attention on the loss function. When T becomes larger, the *weight* $\frac{e^{\omega/T}}{\sum_{i=1}^{N} e^{\omega_i/T}}$ in Eq. 3 is smoothed and the *weights* of different samples are closer. On the contrary, the *weights* will be concentrated on some better samples (easier to optimize) with a smaller T. It will cause the network to only optimize better samples and the optimization of other normal samples is undermined. In the evaluation, these non-optimized normal samples lead to fewer possible matches (including correct and incorrect matches) and more correct matches in possible matches. The increase in the proportion of correct matches in possible matches will lead to a higher *MMA*. However, the reduced number of possible matches also contains some correct matches, which leads to a lower *M.S.* score. To obtain both accurate and dense local features matching results, it is necessary to bal-
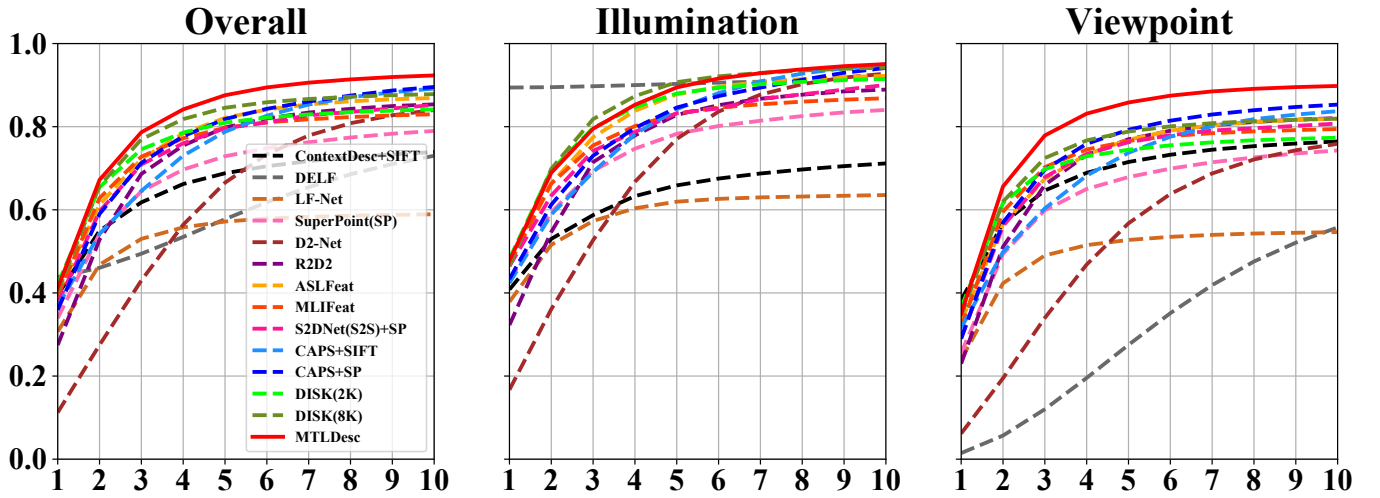
Figure 7: Comparisons on HPatches with different thresholds Mean Matching Accuracy (MMA).

ance *M.S.* and *MMA* by adjusting the value of parameter T. As observed in Fig. 6, the best balance between *M.S.* and *MMA* is achieved when T is set to 15.

Comparisons with Advanced Local Descriptors: In Fig. 7, we compare our MTLDesc with advanced local descriptors (Ono et al. 2018; Luo et al. 2019; Noh et al. 2017; DeTone, Malisiewicz, and Rabinovich 2018; Dusmanu et al. 2019; Revaud et al. 2019; Luo et al. 2020; Zhang et al. 2020; Germain, Bourmaud, and Lepetit 2020; Wang et al. 2020b; Tyszkiewicz, Fua, and Trulls 2020) on HPatches benchmark. All methods use the optimal configuration and results reported in their papers, while MTLDesc notably outperforms these methods in all thresholds under overall *MMA*. Although the performance of the recent DISK is closest to our method, we should note that when DISK uses **2 K keypoints** (equivalent to our 1.5 K), the performance is much lower than our method. It is worth mentioning that DISK does not exceed our method even if using **unfair 8 K keypoints**.

Comparisons on Visual Localization

We resort to Aachen Day-Night v1.1 (Sattler et al. 2012) and InLoc indoor visual localization (Taira et al. 2018) benchmarks to further demonstrate the effectiveness of our MTLDesc. For a fair comparison, all methods use the same image matching pairs provided by the benchmarks and the same evaluation pipelines except for local features. The number of maximum local features of all methods is limited to 20 K as reported in the previous methods. For Aachen, our evaluation is performed via a localization pipeline based on COLMAP (Schonberger and Frahm 2016). For InLoc, we evaluate all methods based on HLOC (Sarlin et al. 2019). See supplementary material for more details. The comparison results are shown in Tab 2. Our MTLDesc evidently outperforms other local descriptors under the tolerance ($0.25m, 2°$ and $0.5m, 5°$) and achieves competitive performance under the tolerance ($5m, 10°$ or $1m, 10°$) on both Aachen outdoor and InLoc indoor benchmarks, validating the effectiveness of our MTLDesc for visual localization task, especially under high-precision requirements.

Aachen Day-Night v1.1 Benchmark					
Method	**Dim**	**Features**	**Correctly localized queries**		
			0.25m,2°	*0.5m,5°*	*5m,10°*
ROOT-SIFT	128	11 K	53.4	62.3	72.3
DSP-SIFT	128	11 K	40.3	47.6	51.3
SuperPoint	256	7 K	68.1	85.9	94.8
D2Net	512	14 K	67.0	86.4	97.4
R2D2	128	10 K	70.7	85.3	96.9
ASLFeat	128	10 K	71.2	85.9	96.9
CAPS + SuperPoint	256	7 K	71.2	86.4	**97.9**
DISK	128	10 K	72.8	86.4	97.4
Our MTLDesc	128	7 K	**74.3**	**86.9**	96.9

InLoc Benchmark					
Method	**Localized queries**(%, 0.25m/0.5m/1.0m)				
	DUC1			**DUC2**	
SuperPoint	39.9	55.6	67.2	37.4	57.3	70.2
D2Net	39.9	57.6	67.2	36.6	53.4	61.8
R2D2	36.4	57.1	73.7	44.3	60.3	68.7
ASLFeat	36.4	56.1	66.7	36.6	55.7	61.1
CAPS + SuperPoint	32.8	53.0	64.6	32.8	58.8	64.1
DISK	38.9	59.1	67.7	37.4	57.3	64.1
Our MTLDesc	**41.9**	**61.6**	72.2	**45.0**	**61.1**	70.2

Table 2: We report the percentage of successfully located images within three error thresholds.

Conclusion

In this work, we propose a novel method named **MTLDesc** to cope with the local features detection and description simultaneously. In order to make our descriptor "look wider to describe better", the Context Augmentation and Consistent Attention Weighting is designed to give descriptors a context awareness beyond the local region, while the Local Features Detection with Feature Pyramid is presented to obtain accurate and reliable keypoints localization. We have conducted thorough experiments on the standard HPatches, Aachen and InLoc benchmark, and validate our MTLDesc can achieve state-of-the-art performance among local descriptors.

Acknowledgements

References

Balntas, V.; Lenc, K.; Vedaldi, A.; and Mikolajczyk, K. 2017. HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5173–5182.

Barroso-Laguna, A.; Riba, E.; Ponsa, D.; and Mikolajczyk, K. 2019. Key. net: Keypoint detection by handcrafted and learned cnn filters. In *Proceedings of the IEEE International Conference on Computer Vision*, 5836–5844.

Cao, B.; Araujo, A.; and Sim, J. 2020. Unifying deep local and global features for image search. In *European Conference on Computer Vision*, 726–743. Springer.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European Conference on Computer Vision*, 213–229. Springer.

Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4): 834–848.

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 764–773.

DeTone, D.; Malisiewicz, T.; and Rabinovich, A. 2018. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 224–236.

Dusmanu, M.; Rocco, I.; Pajdla, T.; Pollefeys, M.; Sivic, J.; Torii, A.; and Sattler, T. 2019. D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In *CVPR 2019*.

Germain, H.; Bourmaud, G.; and Lepetit, V. 2020. S2DNet: Learning Image Features for Accurate Sparse-to-Dense Matching. In *European Conference on Computer Vision*, 626–643. Springer.

Hariharan, B.; Arbelaez, P.; Girshick, R.; and Malik, J. 2016. Object instance segmentation and fine-grained localization using hypercolumns. *IEEE transactions on pattern analysis and machine intelligence*, 39(4): 627–639.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hinton, G. 1976. Using relaxation to find a puppet. In *Proceedings of the 2nd Summer Conference on Artificial Intelligence and Simulation of Behaviour*, 148–157.

Kalantidis, Y.; Mellina, C.; and Osindero, S. 2016. Cross-dimensional weighting for aggregated deep convolutional features. In *European conference on computer vision*, 685–701. Springer.

Lee, C.-Y.; Xie, S.; Gallagher, P.; Zhang, Z.; and Tu, Z. 2015. Deeply-supervised nets. In *Artificial intelligence and statistics*, 562–570. PMLR.

Li, Z.; and Snavely, N. 2018. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2041–2050.

Long, J.; Shelhamer, E.; and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3431–3440.

Lowe, D. G. 2004a. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2): 91–110.

Lowe, D. G. 2004b. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2): 91–110.

Luo, Z.; Shen, T.; Zhou, L.; Zhang, J.; Yao, Y.; Li, S.; Fang, T.; and Quan, L. 2019. Contextdesc: Local descriptor augmentation with cross-modality context. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2527–2536.

Luo, Z.; Zhou, L.; Bai, X.; Chen, H.; Zhang, J.; Yao, Y.; Li, S.; Fang, T.; and Quan, L. 2020. Aslfeat: Learning local features of accurate shape and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6589–6598.

Mikolajczyk, K.; and Schmid, C. 2005. A performance evaluation of local descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 27(10): 1615–1630.

Mishchuk, A.; Mishkin, D.; Radenovic, F.; and Matas, J. 2017. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, 4826–4837.

Noh, H.; Araujo, A.; Sim, J.; Weyand, T.; and Han, B. 2017. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE international conference on computer vision*, 3456–3465.

Ono, Y.; Trulls, E.; Fua, P.; and Yi, K. M. 2018. LF-Net: learning local features from images. In *Advances in neural information processing systems*, 6234–6244.

Revaud, J.; De Souza, C.; Humenberger, M.; and Weinzaepfel, P. 2019. R2D2: Reliable and Repeatable Detector and Descriptor. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In

International Conference on Medical image computing and computer-assisted intervention, 234–241. Springer.

Sarlin, P.-E.; Cadena, C.; Siegwart, R.; and Dymczyk, M. 2019. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12716–12725.

Sattler, T.; Weyand, T.; Leibe, B.; and Kobbelt, L. 2012. Image Retrieval for Image-Based Localization Revisited. In *BMVC*.

Schonberger, J. L.; and Frahm, J.-M. 2016. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4104–4113.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Taira, H.; Okutomi, M.; Sattler, T.; Cimpoi, M.; Pollefeys, M.; Sivic, J.; Pajdla, T.; and Torii, A. 2018. InLoc: Indoor visual localization with dense matching and view synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7199–7209.

Tian, Y.; Barroso Laguna, A.; Ng, T.; Balntas, V.; and Mikolajczyk, K. 2020. HyNet: Learning Local Descriptor with Hybrid Similarity Measure and Triplet Loss. *Advances in Neural Information Processing Systems*, 33.

Tian, Y.; Fan, B.; and Wu, F. 2017. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 661–669.

Tian, Y.; Yu, X.; Fan, B.; Wu, F.; Heijnen, H.; and Balntas, V. 2019. SOSNet: Second order similarity regularization for local descriptor learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 11016–11025.

Tolias, G.; Jenicek, T.; and Chum, O. 2020. Learning and aggregating deep local descriptors for instance-level recognition. In *European Conference on Computer Vision*, 460–477. Springer.

Tyszkiewicz, M.; Fua, P.; and Trulls, E. 2020. DISK: Learning local features with policy gradient. *Advances in Neural Information Processing Systems*, 33.

Wang, H.; Zhu, Y.; Green, B.; Adam, H.; Yuille, A.; and Chen, L.-C. 2020a. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European Conference on Computer Vision*, 108–126. Springer.

Wang, Q.; Zhou, X.; Hariharan, B.; and Snavely, N. 2020b. Learning Feature Descriptors using Camera Pose Supervision. In *Proc. European Conference on Computer Vision (ECCV)*.

Wu, B.; Xu, C.; Dai, X.; Wan, A.; Zhang, P.; Tomizuka, M.; Keutzer, K.; and Vajda, P. 2020. Visual transformers: Token-based image representation and processing for computer vision. *arXiv preprint arXiv:2006.03677*.

Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; and Sang, N. 2018. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, 325–341.

Yu, F.; and Koltun, V. 2015. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Zhang, Y.; Wang, J.; Xu, S.; Liu, X.; and Zhang, X. 2020. MLIFeat: Multi-level information fusion based deep local features. In *Proceedings of the Asian Conference on Computer Vision*.