Online Invariance Selection for Local Feature Descriptors

Rémi Pautrat[1], Viktor Larsson[1], Martin R. Oswald[1], and Marc Pollefeys[1,2]

[1] Department of Computer Science, ETH Zurich [2] Microsoft

Abstract. To be invariant, or not to be invariant: that is the question formulated in this work about local descriptors. A limitation of current feature descriptors is the trade-off between generalization and discriminative power: more invariance means less informative descriptors. We propose to overcome this limitation with a disentanglement of invariance in local descriptors and with an online selection of the most appropriate invariance given the context. Our framework[1] consists in a joint learning of multiple local descriptors with different levels of invariance and of meta descriptors encoding the regional variations of an image. The similarity of these meta descriptors across images is used to select the right invariance when matching the local descriptors. Our approach, named Local Invariance Selection at Runtime for Descriptors (LISRD), enables descriptors to adapt to adverse changes in images, while remaining discriminative when invariance is not required. We demonstrate that our method can boost the performance of current descriptors and outperforms state-of-the-art descriptors in several matching tasks, when evaluated on challenging datasets with day-night illumination as well as viewpoint changes.

Keywords: Local descriptors, invariance, visual localization

1 Introduction

Sparse features detection and description is at the root of many computer vision tasks: Structure-from-Motion (SfM), Simultaneous Localization and Mapping (SLAM), image retrieval, tracking, etc. They offer a compact representation in terms of memory storage and allow for efficient image matching, and are thus well suited for large-scale applications [14,36,35]. These features should however be able to cope with real world conditions such as day-night changes [44], seasonal variations [34] and matching across large baselines [40].

To be able to do matching in extreme scenarios, the successive feature detectors and descriptors have become more and more invariant [23]. The Harris corner detector [12] was already invariant to rotations, but not to scale. The SIFT detector and descriptor [20] was one of the first to achieve invariance with respect to scale, rotation and uniform light changes. More recently, learned descriptors have been able to encode invariance without handcrafting it. On the one hand,

[1] `https://github.com/rpautrat/LISRD`

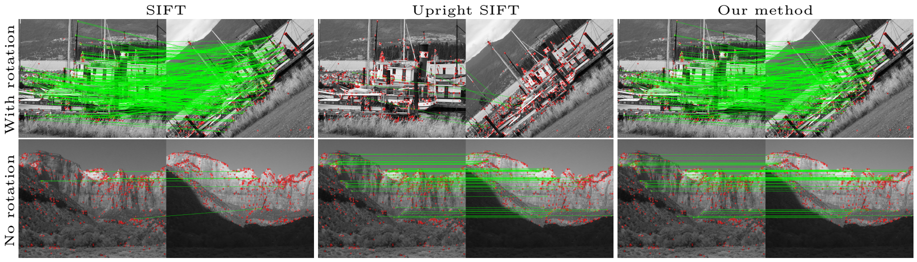

Fig. 1: **Importance of invariance among descriptors.** SIFT descriptors (left) perform well on rotated images (top), but are outperformed by Upright SIFT descriptors (middle) when no rotation is present (bottom). We propose a method (right) that automatically selects the proper invariance during matching time.

patch-based descriptors can become invariant to transforms when estimating the shape of the patch [43,29,25,10]. On the other hand, recent dense descriptors leverage the power of large convolutional neural networks (CNN) to become more general and invariant. Most of them are trained on images with many variations in the training set, either obtained through data augmentation [8], with large databases of challenging images [9,42] or with style transfer [31]. They can also directly encode the invariance in the network itself [19]. The general trend in descriptor learning is thus to capture as much invariance as possible.

While feature detectors should generally be invariant to be repeatable under different scenarios [44], the same is not necessarily true for descriptors [41]. There is a direct trade-off for descriptors between generalization and discriminative power. More invariance allows a better generalization, but produces descriptors that are less informative. Figure 1 shows that the rotation variant descriptor Upright SIFT performs better than its invariant counterpart SIFT when only small rotations are present in the data. We argue that the best level of invariance depends on the situation. As a consequence, this questions the recent trend of jointly learning detector and descriptor: they may have to be dissociated if one does not want the descriptor to be as invariant as the detector.

In this work we focus on learning descriptors only and propose to select at runtime the right invariance given the context. Instead of learning a single generic descriptor, we compute several descriptors with different levels of invariance. We then propose a method to automatically select the most suitable invariance during matching. We achieve this by leveraging the local descriptors to learn meta descriptors that can encode global information about the variations present in the image. At matching time, the local descriptors distances are weighted by the similarity of these meta descriptors to produce a single descriptor distance. Matches based on this distance can then be filtered using standard heuristics such as ratio test or mutual nearest neighbor.

Overall, our method, named Local Invariance Selection at Runtime for Descriptors (LISRD - pronounced as lizard), brings flexibility and interpretability into the feature description. When some image variations are known to be limited for a given application, one may directly use the most discriminative descriptor among all our learned local descriptors. However, it is usually hard to make such an assumption about the inter-image variations, and LISRD can instead automatically select the best invariance independently for each local region. Hence we are able to distinguish between different levels of variations within the same image (e.g. if half of the image is in the shadow but not the other half) and we show that this can improve the matching capabilities in comparison to using a single descriptor. The meta descriptors formulation is also not restricted to our proposed learned local descriptors, but can be easily generalized to most keypoint detectors and descriptors, as shown in Figure 1 where it is applied to SIFT and Upright SIFT. Furthermore, the meta description only adds a small overhead to the current pipelines of keypoint detection and description in terms of runtime and memory consumption, which makes it suitable for real time applications. In summary, this work makes the following **contributions**:

- We show how to learn several local descriptors with multiple variance properties through a single network, in a similar spirit as in multi-task learning.
- We propose a light-weight meta descriptor approach to automatically select the best invariance of the local descriptors given the context.
- Our concept of meta descriptor and general approach of invariance selection can be easily transferred to most feature point detectors and descriptors, which we demonstrate for learned as well as traditional handcrafted descriptors.

2 Related work

Learned local feature descriptors. The recent progress in deep learning has enabled learned local descriptors to outperform the classical baselines by a large margin [8,9,21,31]. Following the classical approach, early works run a CNN on a small image region around the point of interest to get a patch descriptor [38,24,29]. The patch is not restricted to square areas, but can encode spatial transforms, such as affine [25] and polar [10] ones. The network is often optimized with a triplet loss using heuristics to extract positive and negative patches [3,22,11,39], or by directly maximizing the average precision (AP) [13]. Working on sparse features also gives the possibility to leverage both the visual context of the image and the spatial relationships between the keypoint locations [21]. More recently, descriptors extracted densely by CNN architectures from full images have shown both fast inference time and high performance on matching and retrieval tasks, and can jointly detect a heatmap of keypoints. Some works detect keypoints and describe them in parallel, such as SuperPoint [8] and R2D2 [31], with for the latter an additional reliability map keeping track of the most informative locations in the image. Another approach is to use the features of the network as dense descriptors to subsequently detect keypoints, based on those features [28,9,42]. DELF [28] selects the keypoints using a learned attention, D2-Net [9] retrieves the maximum responses of the descriptor feature map across all channels, while UR2KID [42] clusters the channels in different groups and extracts keypoints based on their L2 responses. Even though jointly estimating the keypoints and descriptors allows a faster prediction and yields descriptors that are more correlated to the keypoints, the consequence is that detector and descriptor will share the same invariance. Therefore, we choose to focus exclusively on descriptor learning in this work.

Invariance in feature descriptors. Selecting an online invariance for binary descriptors is the core idea of BOLD [2], where a subset of the binary tests is chosen at runtime for each image patch to maximize the invariance to small affine transformations. Similarly, the general trend of most recent learned methods is to obtain descriptors as invariant as possible to any image variations. LIFT [43] mimics SIFT to achieve rotation invariance by estimating the keypoints, their orientation and finally their descriptor. Invariance to specific geometric changes can be achieved through group convolutions [7] by clustering the different geometrical transformations into specific groups [19]. However, the usual strategy is to incorporate as much diversity in the training data as possible. Illumination invariance can for example be obtained by training on images with multiple lighting conditions [15]. Photometric and homographic data augmentations also increase robustness to illumination and viewpoint changes [8]. Similarly, R2D2 [31] improves the robustness to day-night changes by synthesizing night images with style transfer and also to viewpoint changes by leveraging flow between close-by images [30]. Methods like D2-Net [9] and UR2KID [42] leverage a large database of images with multiple conditions and non planar viewpoint changes thanks to SfM data [17]. In this work, we adopt a mixture of the previously mentioned methods, namely the same synthesized night images as in [31], homographic augmentation, and training on datasets with multiple illumination changes [27].

Multi-task learning in description and matching tasks. Using a single network to achieve multiple and related tasks in feature description and matching is not new. Jointly learning the detector and descriptor [8,9,31] is already multi-task learning that makes the descriptors more discriminative at the predicted keypoint locations. HF-Net [32] unifies the detection of feature points, local and also global descriptors for image retrieval using multi-task distillation with a teacher network. Methods such as SuperGlue [33] and ContextDesc [21] can leverage both visual and geometric context in their descriptors in order to get a more consistent matching between images. UR2KID [42] bypasses the need of keypoint supervision during training and directly optimizes the descriptors jointly for local matching and image retrieval. In our approach, multiple descriptors are also learned in parallel, but instead of differing in their scope, they differ in their level of invariance. Furthermore, unlike previous hierarchical global-to-local approaches, our method relies on local descriptors first and leverages global information only to refine the local matching.

3 Learning the best invariance for local descriptors

Our approach to select the most relevant variance for local feature descriptors consists in two steps. First, we design a network to learn several dense descriptors, each with a different type of invariance (see Section 3.1). Second, we propose a strategy in Section 3.2 to determine the best invariance to use when matching the local descriptors. Figure 2 provides an overview of the full architecture.

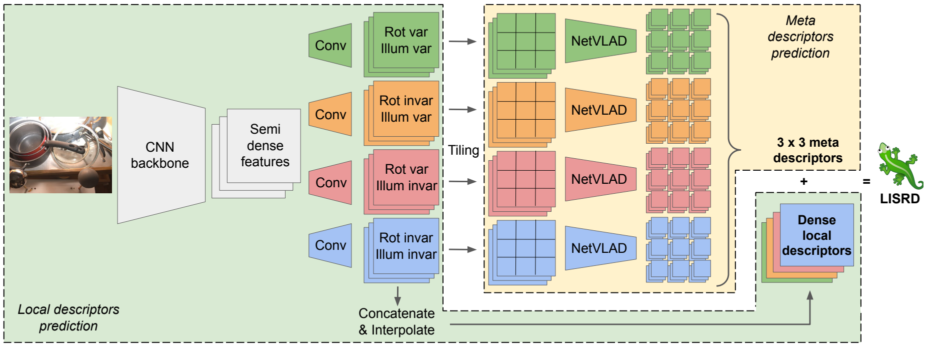

Fig. 2: **Overview of our network architecture.** Our network computes four local dense descriptors with diverse invariances and aggregates them through a NetVLAD layer [1] to obtain a regional description of the variations of the image.

3.1 Disentangling invariance for local descriptors

Many properties of an image have an influence on descriptors, but disentangling all of them would be intractable. We focus here on two factors known to have a large impact on descriptors performance: rotation and illumination. Our framework can however be generalized to other kinds of variations, for instance scaling. Since each of the two factors can either be variant or invariant, there are four possible combinations of variance with respect to illumination and rotation. We show in the following that the variant versions of descriptors are more discriminative since they are more specialized, while the invariant ones are trading the discriminative power for better generalization capabilities.

Network architecture. Our network is inspired by SuperPoint [8], with slight modifications. It takes RGB images as input, computes semi-dense features with a shared backbone of convolutions and is then divided into 4 heads predicting a semi-dense descriptor each, one per combination of variance, as shown in Figure 2. Since most computations are redundant between the 4 local descriptors, the shared backbone reduces the number of weights in the network and offers an inference time competitive with the current learned descriptors.

Dataset preparation. The training dataset is composed of triplets of images. The first one, the *anchor image* I^A, is taken from a large database of real images. The *variant image* I^V is a warped version of the anchor by a homography without rotation and with equal illumination to train variant descriptors. Finally, the *invariant image* I^I used for invariant descriptors is also related to the anchor by a homography, but its orientation and illumination can differ from the anchor.

Training losses. The local descriptors are trained using variants of the margin triplet ranking loss [5,24], depending on whether the descriptor should be invariant or not to the variations present in I^I. The dense descriptors are first sampled on selected keypoints of the images, they are L2-normalized and the losses are computed on the resulting set of feature descriptors. Since we focus on descriptors only, we use SIFT keypoints during training to propagate the gradient in informative areas of the image only. Any kind of keypoint can be used at inference time nonetheless, as demonstrated in Section 4.5.

Formally, given two images I^a and I^b related by a homography $\mathcal{H}$ and n keypoints $\mathbf{x}^a_{1..n}$ in image I^a, we warp each point to image I^b using the homography: $\mathbf{x}^b_{1..n} = \mathcal{H}(\mathbf{x}^a_{1..n})$. This yields a set of n correspondences between the two images, where we can extract the descriptors from each dense descriptor map: $\mathbf{d}^a_{1..n}$ and $\mathbf{d}^b_{1..n}$. Let us define a generic triplet loss $L_T(I^a, I^b, \text{dist})$ between I^a and I^b, given a descriptor distance $\text{dist}(\mathbf{x}^a, \mathbf{x}^b)$. The triplet loss first enforces a correct correspondence $(\mathbf{x}^a_i, \mathbf{x}^b_i)$ to be close in descriptor space through a positive distance

$$p_i = \text{dist}(\mathbf{x}^a_i, \mathbf{x}^b_i) \ . \tag{1}$$

Additionally, the triplet loss increases the negative distance n_i between $\mathbf{x}^a_i$ and the closest point in I^b which is at least at a distance T from the correct match $\mathbf{x}^b_i$. This distance is computed symmetrically across the two images and the minimum is kept:

$$n_i = \min(\text{dist}(\mathbf{x}^a_i, \mathbf{x}^b_{n_b(i)}), \text{dist}(\mathbf{x}^b_i, \mathbf{x}^a_{n_a(i)})) \ , \tag{2}$$

with $n_b(i) = \arg\min_{j \in [1,n]}(\text{dist}(\mathbf{x}^a_i, \mathbf{x}^b_j))$ s.t. $||\mathbf{x}^a_i - \mathbf{x}^b_j||_2 > T$, and similarly for $n_a(i)$. Given a margin M, the triplet margin loss is then defined as

$$L_T(I^a, I^b, \text{dist}) = \frac{1}{n}\sum_{i=1}^{n} \max(M + (p_i)^2 - (n_i)^2, 0) \ . \tag{3}$$

In our case, the loss L_I for invariant descriptors is an instance of this generic triplet loss between the anchor image I^A and the invariant image I^I, for the L2 descriptor distance:

$$L_I = L_T(I^A, I^I, ||\mathbf{d}^A - \mathbf{d}^I||_2) \ . \tag{4}$$

The loss L_V for variant descriptors is based on the full triplet of images: I^A, I^I and I^V. It enforces variant descriptors to be different between the anchor and the invariant image, while preserving similarity between the anchor and the variant image. Its positive loss is the distance in descriptor space of positive matches between I^A and I^V, and similarly for the negative distance between I^A and I^I:

$$L_V = \frac{1}{n}\sum_{i=1}^{n} \max(fM + ||\mathbf{d}^A_i - \mathbf{d}^V_i||^2_2 - ||\mathbf{d}^A_i - \mathbf{d}^I_i||^2_2, 0) \ , \tag{5}$$

where f is a factor controlling at which point the anchor and the invariant images are different. For rotation changes, $f = \min(1, \frac{\theta_I}{\theta_{max}})$, where θ_I is the absolute angle of rotation between the anchor and the invariant image and θ_{max} is a hyper-parameter representing the threshold beyond which the two images should be considered different. This threshold ensures that only large rotations are penalized by the loss. It is hard to quantify the difference in illumination between two real images, so we set $f = 1$ when the illumination differs between the anchor and invariant image.

When a descriptor d in the set $\mathcal{D}$ of descriptors is supposed to be invariant to all changes (illumination and/or rotation) between I^A and I^I, we use L_I.

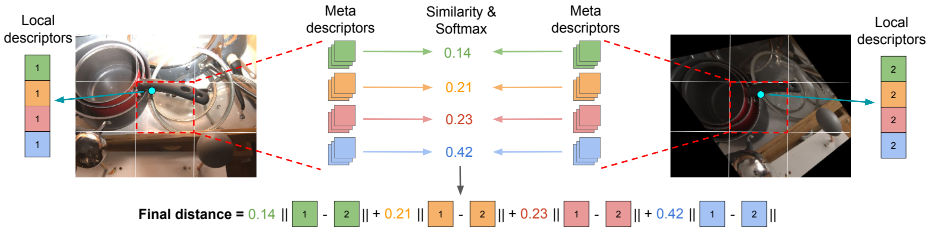

Fig. 3: **The LISRD descriptor distance** between two points is the sum of the four local descriptors distances, weighted by the similarity of the meta descriptors.

Otherwise, L_V is used. We define $L_{I/V}(d)$ as the selected loss and the total loss for local descriptors as

$$L_l = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} L_{I/V}(d) \ . \tag{6}$$

3.2 Online selection of the best invariance

Given the local descriptors of the previous section, this section explores how to pick the most relevant invariance when matching images. Since it would be costly to recompute and compare the image variations for every pair of images to be matched, we propose to rely solely on the information contained in the descriptors to perform the selection. A naive approach would be to separately compute the similarity of the different local descriptors and to pick the most similar ones. However, the invariance selection would gain by having more context than the information of a single local descriptor and should be consistent with neighboring descriptors. Therefore, we propose to extract regional descriptors from the local ones and to use them to guide the invariance selection.

The local descriptors are thus gathered in neighboring areas through a NetVLAD layer [1] to get a meta descriptor sharing the same kind of invariance as the subset of local descriptors, but with more context than a single local descriptor. Thus, having similar meta descriptors means sharing the same level of variations. The neighboring areas are created by tiling the image into a $c \times c$ grid and computing a meta descriptor for each tile. Hence, we get four meta descriptors per tile, which are then L2 normalized.

When matching the local descriptors of a tile, the four similarities between the meta descriptors are computed with a scalar product and we can rank the four local descriptors according to these similarities. Instead of making a hard choice by taking only the closest local descriptor, we use a soft assignment. A softmax operation is applied to the four similarities, to get four weights summing to one. These weights are then used to compute the distance between the local descriptors as shown in Figure 3. More precisely, suppose that we want to compute the distance in descriptor space between point $\mathbf{x}^a$ in image I^a and point $\mathbf{x}^b$ in image I^b. Point $\mathbf{x}^a$ is associated with 4 local descriptors $\mathbf{d}^a_{1..4}$ and 4 meta descriptors $\mathbf{m}^a_{1..4}$ corresponding to the region where $\mathbf{x}^a$ lies, and similarly for $\mathbf{x}^b$. Then the final descriptor distance between $\mathbf{x}^a$ and $\mathbf{x}^b$ is

$$\mathrm{dist}(\mathbf{x}^a, \mathbf{x}^b) = \sum_{i=1}^{4} \frac{\exp\left((\mathbf{m}_i^a)^\intercal \cdot \mathbf{m}_i^b\right)}{\sum_{j=1}^{4} \exp\left((\mathbf{m}_j^a)^\intercal \cdot \mathbf{m}_j^b\right)} ||\mathbf{d}_i^a - \mathbf{d}_i^b||_2 \ . \tag{7}$$

Thus, the similarity of the meta descriptors acts as a weighting of the local descriptors distances and can put a stronger emphasis on one specific variance when the corresponding meta descriptors have a high similarity. Matching is then performed with this descriptor distance, and can easily be refined with ratio test [20] or mutual nearest neighbor.

Training loss. The 4 NetVLAD layers are trained with a weak supervision based on another instance of the triplet loss L_T between I^A and I^I with the distance defined above:

$$L_m = L_T(I^A, I^I, \mathrm{dist}) \tag{8}$$

Thanks to this weak supervision, there is no need to explicitly supervise the meta descriptors, which would require knowing the amount of rotation and illumination for every tile in the image. The total loss of the network is finally a combination of the local and meta descriptors, weighted by a factor λ:

$$L = L_l + \lambda L_m \ . \tag{9}$$

3.3 Training details

Datasets. To train descriptors with different levels of variance in terms of rotation and illumination, datasets presenting all possible combinations of changes are needed. Control over the amount of changes is also required in order to know which loss between L_I and L_V should be used for each descriptor. We use in total four datasets to accomplish that. Illumination variations are obtained through the multi illumination dataset in the wild [27] and the style transferred night images of the Aachen day dataset [31]. Both offer pairs of images with fixed viewpoint and different illuminations. Images with fixed illumination come from the MS COCO dataset [18] and the day flow images from the Aachen dataset [31]. For all datasets except the latter, the images are augmented with random homographies containing translation, scaling, rotation and perspective distortion, similarly as in [8]. For the day images of Aachen, the flow is used to create the correspondences and we consider that these images contain only small rotations and no major illumination changes. Overall, there is an equal distribution of images with and without illumination changes, and of rotated and non rotated images.

Implementation details. We describe here the details of our architecture. The backbone network, inspired by the VGG16 [37], is composed of successive 3×3 convolutional layers with channel size 64-64-64-64-128-128-256-256. Each conv layer is followed by ReLU activation and batch normalization. Every two layers, a 2×2 average pooling with stride 2 is applied to reduce the spatial resolution by 2. For an image of size $H \times W \times 3$, the output feature map will have a size of $H/8 \times W/8 \times 256$. The local descriptor heads are all composed of the following operations: 3×3 conv of channel size 256 - ReLU - Batch Norm - 1×1 conv of channel size 128. The final dimension of each local descriptor is thus $H/8 \times W/8 \times 128$, and each concatenated descriptor is 512-dimensional. The semi-dense descriptors can then be bilinearly interpolated to the locations of any keypoint. Note that in order to achieve a better robustness to scale changes, one can also detect the keypoints and describe them at multiple image resolutions and aggregate the results in the original image resolution, similarly as in [9] and [31]. The NetVLAD layers consists in 8 clusters of 128-dimensional descriptors, hence a meta descriptor size of 1024. We used $c \times c = 3 \times 3$ tiles per image.

The network is trained on RGB images resized to 240×320 with the following hyper-parameters: distance threshold $T = 8$, $\theta_{max} = \frac{\pi}{4}$, margin $M = 1$, loss factor $\lambda = 1$. It comprises roughly 3.7M parameters, which are optimized with the Adam solver [16] (learning rate = 0.001 and $\beta = (0.9, 0.999)$). In practice, the local descriptors are pre-trained first and then fine-tuned by an end-to-end training with the meta descriptors. At test time, a single forward pass on a GeForce RTX 2080 Ti with 480×640 images takes 6ms on average.

4 Experimental results

We present here experiments validating the relevance of our method. Section 4.2 highlights the importance of learning different invariances, validates the proposed approach with an ablation study, and shows that LISRD can be extended to other descriptors such as SIFT and Upright SIFT. LISRD is then compared to the state of the art on a benchmark homography dataset (Section 4.3), on a challenging dataset with diverse conditions where the presence or lack of invariance is essential (Section 4.4) and on a visual localization task in the real world (Section 4.5).

4.1 Metrics

Since we want to compare the performance of the descriptors only, all the following metrics are computed on SIFT keypoints if not stated otherwise. The metrics are computed on pairs of images resized to 480×640 and related by a known homography. Resizing is performed by upscaling/downscaling the images to have each edge greater or equal respectively to 480 and 640, and a central crop is applied to get the target resolution. We keep a maximum of 1000 points among the keypoints shared between the two views and matches are obtained after mutual nearest neighbor filtering.

Homography estimation. We follow the procedure of [8] to compute a homography estimation score. Given a pair of images, RANSAC is used to fit a homography between the clouds of matched keypoints. The score is obtained by warping the four corners of the first image $\hat{c}_{1\ldots4}$ with the predicted homography and comparing their distance to the same points $c_{1\ldots4}$ warped by the ground truth homography. The homography is considered as correct when the average distance is below a threshold ϵ, which is set to 3 pixels in all experiments: $\text{HEstimation} = \frac{1}{4}\sum_{i=1}^{4} ||\hat{c}_i - c_i||_2 \leq \epsilon$.

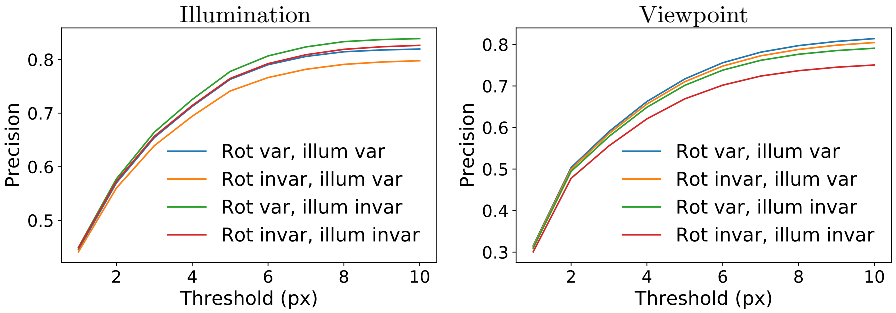

Fig. 4: **Precision on HPatches of the 4 local descriptors.** Variant ones are better when invariance is not needed (e.g. rotation for the illumination dataset).

Precision. Precision (also known as mean matching accuracy) is the percentage of correct matches over all the predicted matches [9,31]. We use by default a threshold of 3 pixels to consider a match to be correct.

Recall. Recall is the ratio of correctly predicted matches over the total number of ground truth matches, where a ground truth correspondence is the *closest* point within an error threshold of 3 pixels. A predicted match with the *second closest* point but still within the correct threshold is considered as incorrect.

4.2 Method Validation

Impact of the different invariances. One can check the validity of our approach by comparing the 4 local descriptors. We use the HPatches dataset [4], which is standard in descriptor evaluation. It is composed of 116 sequences of 5 pairs of images, with either viewpoint changes (given by a known homography) or illumination changes with fixed viewpoint. Figure 4 shows the comparison between the 4 descriptors in terms of precision. On viewpoint changes, the illumination variant descriptors are superior as the lighting is fixed in these images and they are thus more discriminative. Since HPatches contains few rotations, there is no significant difference in terms of rotation invariance and being rotation variant brings a small advantage on average. The precision on illumination changes shows that the best performing descriptors are the illumination invariant ones and that being rotation variant helps since the viewpoint is fixed. Thus there is no descriptor outperforming the others in all cases, and our hypothesis that variant descriptors are more discriminative than invariant ones is validated.

Ablation study. To confirm the benefit of our online selection of invariance and choice of parameters, we compare LISRD on homography estimation on the HPatches dataset with other selection methods of the local descriptors as well as with variants of our approach (Table 1). *Best of the 4* computes the metrics for the 4 local descriptors separately and picks the best score. *Greedy* computes the pairwise distances of all points for each local descriptor and greedily chooses the local descriptor with smallest distance for each pair of points. *Hard assignment* selects the local descriptor that maximizes the meta descriptor similarity, instead

Table 1: **Ablation study on the HPatches dataset.**

	HEstimation
Best of the 4	0.778
Greedy	0.774
Hard assignment	0.762
No tiling	0.752
5 × 5 tiles	0.773
Single desc	0.766
LISRD (ours)	**0.784**

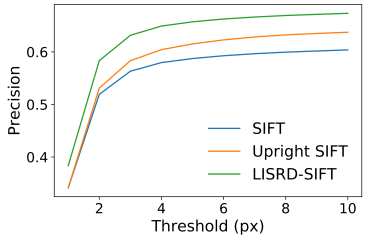

Fig. 5: **Variants of SIFT vs. our method fusing them (LISRD-SIFT).** Precision is computed on HPatches viewpoint.

of choosing a soft assignment as in our proposed method. *No tiling* and *5 × 5 tiles* are variants of our method with no tiling or with *5 × 5* tiles for the meta descriptors. Finally, *Single desc* is a descriptor trained with exactly the same architecture as ours, but with the 4 local descriptors concatenated and trained with invariance in both illumination and rotation.

On the full HPatches dataset, *Best of the 4* corresponds to the descriptor invariant to both illumination and rotation, as both changes are present. However, our selection method can still leverage the other descriptors: for example an illumination variant descriptor for the viewpoint part. The disparity between LISRD and *Greedy* and *Hard assignment* highlights the added value of the meta descriptors, and shows that a soft assignment can better leverage the 4 descriptors at the same time. Finally, the comparison with *Single desc* confirms our hypothesis that disentangling the types of invariance is beneficial compared to learning a single invariant descriptor with the same number of weights.

Generalization to other descriptors. LISRD can be easily generalized to other kinds of descriptors, and not only to our proposed learned local descriptors. We demonstrate this by applying our approach to the duo of local descriptors SIFT and Upright SIFT – SIFT without rotation invariance, as presented in Figure 1. Instead of having four local descriptors, there are only two of them, one invariant to rotation and one variant, and similarly for the meta descriptors. Our method is evaluated against SIFT and Upright SIFT on the viewpoint part of HPatches. This dataset contains indeed sequences with no rotation, where Upright SIFT performs better, and other sequences with strong rotations, where SIFT takes over. Figure 5 shows that our method can effectively leverage both SIFT and Upright SIFT and outperforms the two.

4.3 Descriptor evaluation on HPatches

This section compares the performance of LISRD against state-of-the-art local descriptors on the benchmark dataset HPatches. Since our approach requires global context from full images, we cannot run it on the patch level dataset. We use the full sequences of images instead, similarly as in [8,9,31]. We consider

Table 2: **Comparison to the state of the art on HPatches.** Homography estimation, precision and recall are computed for error thresholds of 3 pixels. The best score is in bold and the second best one is underlined.

		Root SIFT	HardNet	SOSNet	SP	D2-Net	R2D2	GIFT	Ours
HP Illum	HEstimation	0.898	0.884	0.919	0.877	0.818	0.916	**0.923**	0.884
	Precision	0.554	0.574	0.591	0.629	0.650	**0.666**	0.573	0.665
	Recall	0.431	0.483	0.519	0.565	0.564	0.580	0.521	**0.655**
HP View	HEstimation	0.644	0.688	**0.742**	0.651	0.553	0.627	0.715	0.688
	Precision	0.515	0.582	0.598	0.595	0.564	0.550	0.552	**0.626**
	Recall	0.350	0.422	0.448	0.446	0.382	0.371	0.429	**0.495**

the following baselines: Root SIFT with the default Kornia[2] implementation; HardNet [24] (trained on the PS-dataset [26]), SOSNet [39] (trained on the Liberty dataset of UBC Phototour [6]), SuperPoint (SP) [8], D2-Net [9], R2D2 [31] and GIFT [19] with the authors implementation. Since we want to evaluate the descriptors only, SIFT keypoints are detected in the images and for each method, we extract the local descriptors at these locations. For Root SIFT, HardNet and SOSNet, we sample 32×32 patches at each SIFT keypoint and rotate them according to the SIFT orientation. As we want to evaluate the impact of rotation and illumination invariance only, we use single scale implementations for all methods[3]. Our method could however be made scale invariant using similar multi-scale approaches as in [9,31].

The results are summarized in Table 2. Overall, LISRD ranks among the two best methods in precision and recall. The possibility to leverage rotation variant descriptors on the fixed pairs of the illumination part and to alternatively select the right level of lighting invariance given the amount of illumination changes probably explains our superior performance on the illumination part. Note the comparison with SuperPoint, whose architecture and training procedure are very similar to LISRD, and where our method displays better results in all metrics, thus showing the gain of our approach. The weaker results in homography estimation can be explained by a limitation of our method. Since our meta descriptors have a very coarse spatial resolution (3×3 grid), if one of them fails to pick the right invariance, this will impact all the matches of its region. Thus, the correct matches predicted by LISRD can in that case become very concentrated in a specific part of the image, which makes the homography estimation with RANSAC less accurate. This issue could be avoided with a finer tiling of the meta descriptors, but at the price of a reduced global context.

4.4 Evaluation in challenging and cross-modal situations

The HPatches dataset offers a fair benchmark, but is limited to only few rotations and medium illumination changes. Our approach is designed to be used in a

[2] `https://kornia.github.io/`

[3] In the case of GIFT, which is both rotation and scale invariant, we sample images with scale 1 to make it rotation invariant only.

Table 3: **Evaluation on a use case where invariance selection matters.** Homography estimation, precision and recall are computed with SuperPoint keypoints on a dataset with day-night changes and various levels of rotation. Selecting the relevant variant or invariant descriptors boosts the precision and recall of our method compared to the previous state-of-the-art methods.

		Root SIFT	HardNet	SOSNet	SP	D2-Net	R2D2	GIFT	Ours
Day ref	HEstimation	0.121	**0.199**	0.178	0.146	0.094	0.170	0.187	0.198
	Precision	0.188	0.232	0.228	0.195	0.195	0.175	0.152	**0.291**
	Recall	0.112	0.194	0.203	0.178	0.117	0.162	0.133	**0.317**
Night ref	HEstimation	0.141	**0.262**	0.211	0.182	0.145	0.196	0.241	**0.262**
	Precision	0.238	0.366	0.297	0.264	0.259	0.237	0.236	**0.371**
	Recall	0.164	0.323	0.269	0.255	0.182	0.216	0.209	**0.384**

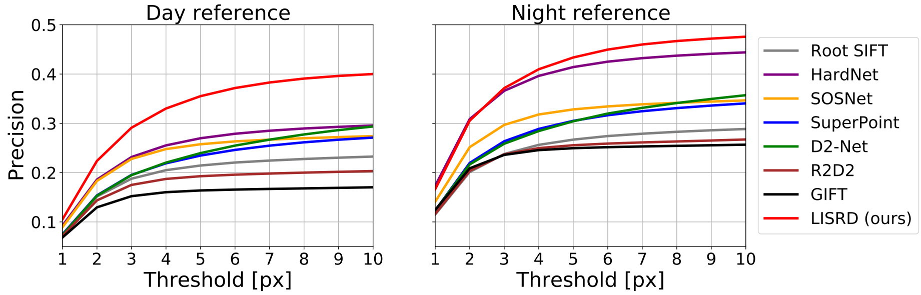

Fig. 6: **Precision curves on the DNIM dataset [44] augmented with rotations.** LISRD leverages its variant and more discriminative descriptors whenever possible and is thus more accurate than the state-of-the-art descriptors for all pixel error thresholds.

variety of scenarios and with changing conditions, so that all our local descriptors can be leveraged. In order to evaluate our method on such a versatile task, we designed a new benchmark dataset, based on the day-night image matching (DNIM) dataset [44]. This dataset is composed of sequences of images of a fixed camera taking pictures at regular time intervals and across day and night, with a total of 1722 images. For each sequence, the image with timestamp closest to noon is taken as day reference and the image closest to midnight as night reference. We create two benchmarks, where the images of each sequence are paired with either the day reference or the night one. We then synthetically warp the pairs with the same homography sampling scheme as in [8] with an equal distribution of homographies with and without rotations. We plan to release the homographies used in this benchmark to let other researchers compare with their own methods. Examples of images and matches for this dataset can be found in the supplementary material.

Table 3 and Figure 6 show the evaluation with the state-of-the-art descriptors, using SuperPoint keypoints. LISRD can adapt its invariance to illumination and rotations to alternatively select the most relevant descriptor and it outperforms the other methods by a large margin both in terms of precision and recall.

Table 4: **Visual localization performance on the Aachen Day-Night dataset [34].** We report the percentage of correctly localized queries for various distance and orientation error thresholds for SIFT, SuperPoint and D2-Net multi-scale (MS). Our method shows a good generalization when evaluated on different keypoints (KP) and can improve the original descriptor performance.

Error threshold	SIFT KP		SuperPoint KP		D2-Net KP	
	Up-Root SIFT	Ours	SuperPoint	Ours	D2-Net (MS)	Ours (MS)
$0.5m, 2^\circ$	54.1	**72.4**	73.5	**78.6**	67.3	**73.5**
$1m, 5^\circ$	66.3	**82.7**	79.6	**86.7**	87.8	**88.8**
$5m, 10^\circ$	75.5	**94.9**	88.8	**98.0**	**100.0**	99.0

4.5 Application to localization in challenging conditions

A typical application of image matching including adverse conditions such as strong illumination changes and wide baselines is the visual localization task. We evaluate our method on the local feature challenge of CVPR 2019 based on the Aachen Day-Night dataset [34]. The goal is to localize 98 night time query images as accurately as possible, given 20 day images per query with known camera pose. As the keypoint quality is essential in this task, we compare our method with other descriptors for various types of keypoints: SIFT, SuperPoint and D2-Net multi-scale (MS). The numbers for the baseline methods are taken from the benchmark on the official website[4]. The results in Table 4 show that our method is not limited to SIFT keypoints and can effectively improve the performance of local descriptors in challenging conditions. Note in particular the improvement over SuperPoint, which shares a similar architecture as ours.

5 Conclusion

We presented a novel approach to learn local feature descriptors able to adapt to multiple variations in images, while remaining discriminative. We unified the learning of several local descriptors with multiple levels of invariance and of meta descriptors leveraging regional context to guide the local descriptors matching.

While restricted to illumination and rotation invariance, our framework can be generalized to more variations, at the cost of an exponentially growing number of descriptors however. A future direction of work would be to reduce the amount of redundancy between each descriptor by enforcing a stronger disentanglement separating each factors of variation. Since our approach is able to enforce different levels of invariance, one can also add another head to our network to predict invariant keypoints, while keeping discriminative descriptors, thus solving the current issue in joint learning of invariant detectors and descriptors.

Overall, this work is a first step towards disentangled descriptors. Separating the types of invariances paves the way to a full disentanglement of the factors of variations of images and could lead to flexible and interpretable local descriptors.

Acknowledgments. This work has been supported by an ETH Zurich Postdoctoral Fellowship and Innosuisse funding (Grant No. 34475.1 IP-ICT).

[4] `https://www.visuallocalization.net/`

References

1. Arandjelović, R., Gronat, P., Torii, A., Pajdla, T., Sivic, J.: NetVLAD: CNN architecture for weakly supervised place recognition. In: Computer Vision and Pattern Recognition (CVPR) (2016)
2. Balntas, V., Tang, L., Mikolajczyk, K.: Bold - binary online learned descriptor for efficient image matching. In: Computer Vision and Pattern Recognition (CVPR) (2015)
3. Balntas, V., Johns, E., Tang, L., Mikolajczyk, K.: Pn-net: Conjoined triple deep network for learning local image descriptors. In: Computer Vision and Pattern Recognition (CVPR) (2016)
4. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: Computer Vision and Pattern Recognition (CVPR) (2017)
5. Balntas, V., Riba, E., Ponsa, D., Mikolajczyk, K.: Learning local feature descriptors with triplets and shallow convolutional neural networks. In: British Machine Vision Conference (BMVC) (2016)
6. Brown, M., Hua, G., Winder, S.: Discriminative learning of local image descriptors. Trans. Pattern Analysis and Machine Intelligence (PAMI) (2010)
7. Cohen, T., Welling, M.: Group equivariant convolutional networks. In: International Conference on Machine Learning (ICML) (2016)
8. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: Computer Vision and Pattern Recognition Workshops (CVPRW) (2018)
9. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-Net: A Trainable CNN for Joint Detection and Description of Local Features. In: Computer Vision and Pattern Recognition (CVPR) (2019)
10. Ebel, P., Mishchuk, A., Yi, K.M., Fua, P., Trulls, E.: Beyond cartesian representations for local descriptors. In: International Conference on Computer Vision (ICCV) (2019)
11. Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A.: Matchnet: Unifying feature and metric learning for patch-based matching. In: Computer Vision and Pattern Recognition (CVPR) (2015)
12. Harris, C., Stephens, M.: A combined corner and edge detector. In: In Proc. of Fourth Alvey Vision Conference (1988)
13. He, K., Lu, Y., Sclaroff, S.: Local descriptors optimized for average precision. In: Computer Vision and Pattern Recognition (CVPR) (2018)
14. Heinly, J., Schönberger, J.L., Dunn, E., Frahm, J.M.: Reconstructing the World* in Six Days *(As Captured by the Yahoo 100 Million Image Dataset). In: Computer Vision and Pattern Recognition (CVPR) (2015)
15. Kaliroff, D., Gilboa, G.: Self-supervised unconstrained illumination invariant representation. In: arXiv (2019)
16. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR) (2014)
17. Li, Z., Snavely, N.: Megadepth: Learning single-view depth prediction from internet photos. In: Computer Vision and Pattern Recognition (CVPR) (2018)
18. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft COCO: Common objects in context. In: European Conference on Computer Vision (ECCV) (2014)

19. Liu, Y., Shen, Z., Lin, Z., Peng, S., Bao, H., Zhou, X.: Gift: Learning transformation-invariant dense visual descriptors via group cnns. In: Advances in Neural Information Processing Systems (NeurIPS) (2019)
20. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV) **60** (2004)
21. 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: Computer Vision and Pattern Recognition (CVPR) (2019)
22. Luo, Z., Shen, T., Zhou, L., Zhu, S., Zhang, R., Yao, Y., Fang, T., Quan, L.: Geodesc: Learning local descriptors by integrating geometry constraints. In: European Conference on Computer Vision (ECCV) (2018)
23. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.V.: A comparison of affine region detectors. International Journal of Computer Vision (IJCV) (2005)
24. Mishchuk, A., Mishkin, D., Radenović, F., Matas, J.: Working hard to know your neighbor's margins:local descriptor learning loss. In: Advances in Neural Information Processing Systems (NIPS) (2017)
25. Mishkin, D., Radenovic, F., Matas, J.: Repeatability is not enough: Learning affine regions via discriminability. In: European Conference on Computer Vision (ECCV) (2018)
26. Mitra, R., Doiphode, N., Gautam, U., Narayan, S., Ahmed, S., Chandran, S., Jain, A.: A Large Dataset for Improving Patch Matching. arXiv (2018)
27. Murmann, L., Gharbi, M., Aittala, M., Durand, F.: A multi-illumination dataset of indoor object appearance. In: International Conference on Computer Vision (ICCV) (2019)
28. Noh, H., Araujo, A., Sim, J., Weyand, T., Han, B.: Large-scale image retrieval with attentive deep local features. In: International Conference on Computer Vision (ICCV) (2017)
29. Ono, Y., Trulls, E., Fua, P., Yi, K.M.: Lf-net: Learning local features from images. In: Advances in Neural Information Processing Systems (NIPS) (2018)
30. Revaud, J., Weinzaepfel, P., Harchaoui, Z., Schmid, C.: EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow. In: Computer Vision and Pattern Recognition (CVPR) (2015)
31. Revaud, J., Weinzaepfel, P., de Souza, C.R., Humenberger, M.: R2D2: repeatable and reliable detector and descriptor. In: Advances in Neural Information Processing Systems (NeurIPS) (2019)
32. Sarlin, P.E., Cadena, C., Siegwart, R., Dymczyk, M.: From coarse to fine: Robust hierarchical localization at large scale. In: Computer Vision and Pattern Recognition (CVPR) (2019)
33. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: Computer Vision and Pattern Recognition (CVPR) (2020)
34. Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., Safari, D., Okutomi, M., Pollefeys, M., Sivic, J., Kahl, F., Pajdla, T.: Benchmarking 6dof outdoor visual localization in changing conditions. In: Computer Vision and Pattern Recognition (CVPR) (2018)
35. Sattler, T., Torii, A., Sivic, J., Pollefeys, M., Taira, H., Okutomi, M., Pajdla, T.: Are large-scale 3d models really necessary for accurate visual localization? In: Computer Vision and Pattern Recognition (CVPR) (2017)
36. Schönberger, J.L., Pollefeys, M., Geiger, A., Sattler, T.: Semantic visual localization. In: Computer Vision and Pattern Recognition (CVPR) (2017)

37. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR) (2014)
38. Tian, Y., Fan, B., Wu, F.: L2-net: Deep learning of discriminative patch descriptor in euclidean space. In: Computer Vision and Pattern Recognition (CVPR) (2017)
39. Tian, Y., Yu, X., Fan, B., Wu, F., Heijnen, H., Balntas, V.: Sosnet: Second order similarity regularization for local descriptor learning. In: Computer Vision and Pattern Recognition (CVPR) (2019)
40. Tola, E., Lepetit, V., Fua, P.: Daisy: An efficient dense descriptor applied to wide baseline stereo. Trans. Pattern Analysis and Machine Intelligence (PAMI) **32** (2010)
41. Wu, C., Li, X., Frahm, J., Pollefeys, M.: 3d model matching with viewpoint-invariant patches (vip). In: Computer Vision and Pattern Recognition (CVPR) (2008)
42. Yang, T.Y., Nguyen, D.K., Heijnen, H., Balntas, V.: Ur2kid: Unifying retrieval, keypoint detection, and keypoint description without local correspondence supervision. In: arXiv (2020)
43. Yi, K., Trulls, E., Lepetit, V., Fua, P.: Lift: Learned invariant feature transform. In: European Conference on Computer Vision (ECCV) (2016)
44. Zhou, H., Sattler, T., Jacobs, D.W.: Evaluating local features for day-night matching. In: European Conference on Computer Vision Workshops (ECCVW) (2016)