arXiv:2005.05777v1 [cs.CV] 12 May 2020

# HDD-Net: Hybrid Detector Descriptor with Mutual Interactive Learning

Axel Barroso-Laguna[1] Yannick Verdie[2] Benjamin Busam[2,3] Krystian Mikolajczyk[1]

[1]Imperial College London [2]Huawei Noah's Ark Lab [3]Technical University of Munich

{axel.barroso17, k.mikolajczyk}@imperial.ac.uk {yannick.verdie, benjamin.busam}@huawei.com

## Abstract

*Local feature extraction remains an active research area due to the advances in fields such as SLAM, 3D reconstructions, or AR applications. The success in these applications relies on the performance of the feature detector and descriptor. While the detector-descriptor interaction of most methods is based on unifying in single network detections and descriptors, we propose a method that treats both extractions independently and focuses on their interaction in the learning process rather than by parameter sharing. We formulate the classical hard-mining triplet loss as a new detector optimisation term to refine candidate positions based on the descriptor map. We propose a dense descriptor that uses a multi-scale approach and a hybrid combination of hand-crafted and learned features to obtain rotation and scale robustness by design. We evaluate our method extensively on different benchmarks and show improvements over the state of the art in terms of image matching on HPatches and 3D reconstruction quality while keeping on par on camera localisation tasks.*

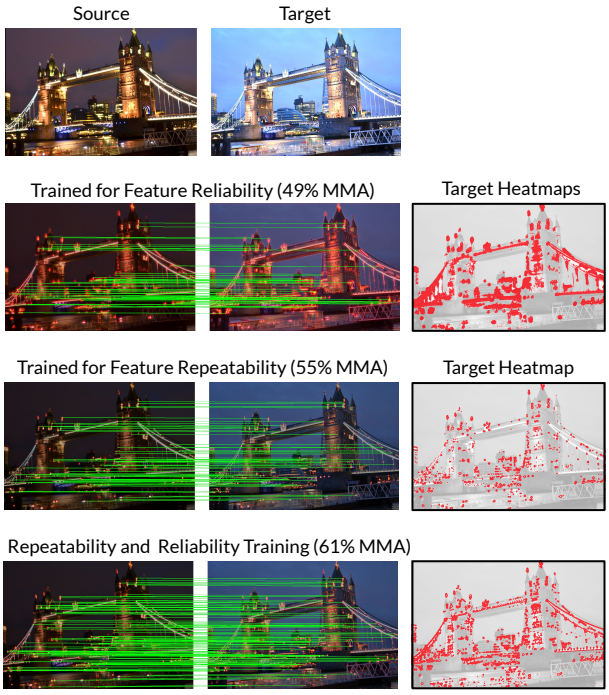


Figure 1: **Effect of Training Strategies on Result.** Correct matches and target detection response maps on *London Bridge* sequence (HPatches) for different training strategies.

## 1. Introduction

At its core, a feature extraction method aims at identifying locations within a scene that are repeatable and distinctive, so that they can be detected with high accuracy under different camera conditions and be matched between different views. The results in vision applications such as image retrieval [1], 3D reconstruction [2], or medical applications [3], among others, have shown the performance advantages of using sparse features over direct methods.

Classical methods [4, 5, 6] independently compute keypoints and descriptors. For instance, SIFT [4] focused on finding blobs on images and extracting gradient histograms as descriptors. Recently proposed descriptors, especially patch-based ones [7, 8, 9, 10], are computed for DoG keypoints, and although they may perform well with other detectors, their test performance is better if the models are trained with patches extracted with the same detector. Most detectors are trained independently of the descriptors and optimise local repeatability of keypoints [11, 12, 13]. The methods that attempt to use the descriptor information to train the detector [14, 15, 16, 17], predicted score maps that either focus on the repeatability or the reliability of a local feature. In our approach, motivated by the limited descriptor influence on the detector, we adapt the descriptor based hard-mining triplet cost function [8] to train the detector model. Thus, keypoint locations are optimised based on the descriptor performance jointly with the detector repeatability. This approach leads to finding both, repeatable

and discriminative features, as shown in figure 1. We extend the models to a multi-scale framework, such that the detector/descriptor networks use different levels of details when making predictions.

Our approach is motivated by the observations that jointly learnt detector-descriptor models [14, 15] lack keypoint localization accuracy, which is critical for SLAM, SfM, or pose estimations. Furthermore, keypoints are typically well localised on simple structures such as edges or corners, while descriptors require more context to be discriminative. We argue that despite the recent trend for end-to-end and joint detector-descriptor methods, separate extractors allow for shallow models that can perform well in terms of accuracy and efficiency.

Besides that, in contrast to patch-based descriptors, dense image descriptors make it more difficult to locally rectify the image regions for invariance. To address this issue, we introduce an approach based on a block of hand-crafted features, and a multi-scale representation within the descriptor architecture, making our network robust to small rotations and scale changes. We term our approach as HDD-Net: Hybrid Detector and Descriptor Network.

In summary, the contributions are: 1) A new detector loss based on the hard-mining triplet cost function. Although the hard-mining triplet is widely used for descriptors it has not been adapted to the training of keypoint detectors. 2) A novel multi-scale sampling scheme to simultaneously train our detector and descriptor. 3) The first dense descriptor architecture that uses a block of hand-crafted features and multi-scale representation to improve the robustness to rotation and scale changes.

## 2. Related Work

Classical hand-crafted methods have been extensively studied in [18, 19]. We focus the review of related work on learned methods. For further details we refer to [20, 21, 22, 23].

**Detectors.** Machine learning detectors were introduced with FAST [24], a learned algorithm to speed up the detection of corners in images. Later, TILDE [25] proposed to train multiple piecewise regressors that were robust under photometric changes in images. DNET [26] and TCDET [27] based its learning on a formulation of the covariant constraint, enforcing the architecture to propose the same feature location in corresponding patches. Key.Net [28] expanded the covariant constraint to a multi-scale formulation, and used a hybrid architecture composed of hand-crafted and learned feature blocks.

**Descriptors.** Descriptors have attracted much attention, particularly patch-based methods [29, 7, 8] due to the simplicity of the task and available benchmarks. Recently, SOSNet [9] improved on the state-of-the-art by adding a regularisation term to the triplet loss to include the second-order similarity relationships among descriptors. DOAP [30] reformulated the training of descriptors as a ranking problem, by optimising the mean average precision instead of the distance between patches. GeoDesc [10] integrated geometry constraints to obtain better training data. Following the idea of improving the data, [31] presented a new patch-based dataset containing scenes under different weather and seasonal conditions.

**Joint Detectors and Descriptors.** LIFT [16] was the first CNN based method to integrate detection, orientation estimation, and description. SuperPoint [11] used a single encoder and two decoders to perform dense feature detection and description. It was first pretrained to detect corners on a synthetic dataset, and then improved by applying random homographies to the training images, improving the stability of the ground truth positions under different viewpoints. Similar to LIFT, LF-Net [12] computed position, scale, orientation, and description. LF-Net trained its detector score and scale estimator in full images without external keypoint supervision. RF-Net [13] extended LF-Net by exploiting the information provided by the receptive fields. D2-Net [14] proposed to perform feature detection in the descriptor space, showing that an already pre-trained network could be used for feature extraction even though it was optimized for a different task. R2D2 [15] introduced a dense version of the L2Net [7] to predict descriptors and two keypoint score maps based on their repeatability and reliability. Recently, ASLFeat [17] proposed an accurate detector and invariant descriptor with multi-level connections and deformable convolutional networks [32, 33].

## 3. Method

This section presents the architecture and training of our Hybrid Detector and Descriptor Network (HDD-Net).

### 3.1. HDD-Net Architecture

HDD-Net consists of two independent architectures for inferring the keypoint and descriptor maps, allowing to use different hand-crafted blocks that are designed specifically for each of these two tasks.

**Descriptor.** As our method estimates dense descriptors in the entire image, an affine rectification of independent patches or rotation invariance by construction [34] is not possible. To circumvent this, we design a hand-crafted block which explicitly addresses the robustness to rotation. We incorporate this block into an architecture based on L2-Net [7]. We replace the last convolutional layer by a bilinear upsampling operator to upscale the map to its original image resolution. Moreover, we use a multi-scale

image representation to extract features from different scale levels. Multi-scale L2-Net features are fused into a final descriptor map by a last convolutional layer.

**Rotation Robustness.** Transformation equivariance in CNNs has been extensively discussed in [35, 36, 37, 38]. The two main approaches differ whether the transformations are applied to the input image [39] or to the filters [40]. Rotating the filters is more efficient since they are smaller than the input images, and therefore, have fewer memory requirements. Unlike [40], which applies the rotation to all the layers in their convolutional model, we focus on the input filters only, which further reduces the computational complexity. In contrast, we apply more rotations than [40] to the input filters to provide sufficient robustness. The feature extraction is illustrated in figure 2. At first, we rotate the input filter 16 times and apply a circular mask to avoid artifacts at the filter corners. Consecutively, we extract the feature maps and apply a cyclic max-pooling operator. Max-pooling is applied on the rotation in all three neighbouring feature maps with a channel-wise stride of two. Then, instead of providing a single maximum over the entire rotation space, cyclic pooling returns the maxima in different quadrants. We experimentally found that returning its local maxima provides better results than using only the global one. As the max-pooling operator is driven to positive values, we split the feature maps in three parts [41]: $\mathcal{H}_r(I) = [h(I), (h(I))^+, \text{-}1 \cdot (h(I))^-]$, where the $(\cdot)^+$ and $(\cdot)^-$ operators respectively keep the positive and negative parts of the feature map $h(I)$.

**Scale Robustness.** Gaussian scale-space has been extensively exploited for local feature extraction [5, 42, 16]. In [12, 13, 28], the scale-space representation was used not only to extract multi-scale features but also to learn to combine their information. However, the fusion of multi-scale features is only used during the detection, while, in deep descriptors, it is either implemented via consecutive convolutional layers [11], or as independent multi-scale extraction [15, 14, 17]. In contrast, we extend the Gaussian pyramid to the descriptor extraction and design a network that is able to compute and combine multi-scale information in a single forward pass. The descriptor encoder shares the weights of each multi-scale stream, thus, boosting its ability to extract features robust to scale changes. Figure 3 depicts the multi-scale descriptor.

**Detector.** We adopt the architecture of Key.Net [28], which combines specific hand-crafted filters for feature detection and a multi-scale shallow network. It has recently been shown to achieve the state of the art results in repeatability.

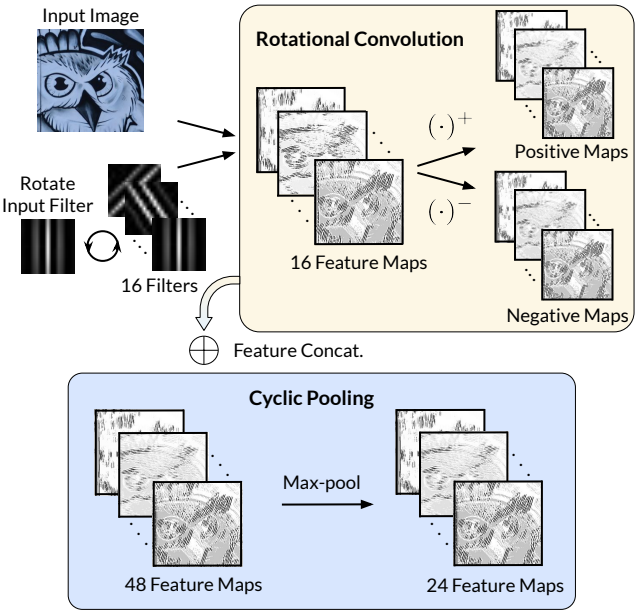


Figure 2: **Hand-crafted Block.** Rotation robustness is given by rotating an input filter and sampling from its rotation space. $(\cdot)^+$ and $(\cdot)^-$ operators split positive and negative parts before the cyclic max-pooling is applied to all features.

### 3.2. Descriptor-Detector Training

Detector learning has focused on localising features that are repeatable in a sequence of images [11, 12, 13, 25, 21, 28], with a few works that determine whether these features are adequate for the matching stage [43, 15, 16]. Since a good feature should be repeatable as well as discriminative [18], we formulate the descriptor triplet loss function as a new detector learning term to refine the feature candidates towards more discriminative positions. Unlike AffNet [43], which estimates the affine shape of the features, we refine only their locations, as these are the main parameters that are often used for the end tasks such as SfM, SLAM or AR. R2D2 [15] inferred two independent response maps, seeking for discriminativeness of the features and their repeatability. Our approach combines both objectives into a single detection map. LIFT [16] training was based on finding the locations with closest descriptors, in contrast, we propose a function based on a triplet loss with a hard-negative mining strategy.

**Detector Learning with Triplet Loss.** Hard-negative triplet learning maximises the Euclidean distance between a positive pair and their closest negative sample. In the original work [8], the optimisation happens in the descriptor part, however, we propose to freeze the descriptor such that the sampling locations proposed by the detector are updated to

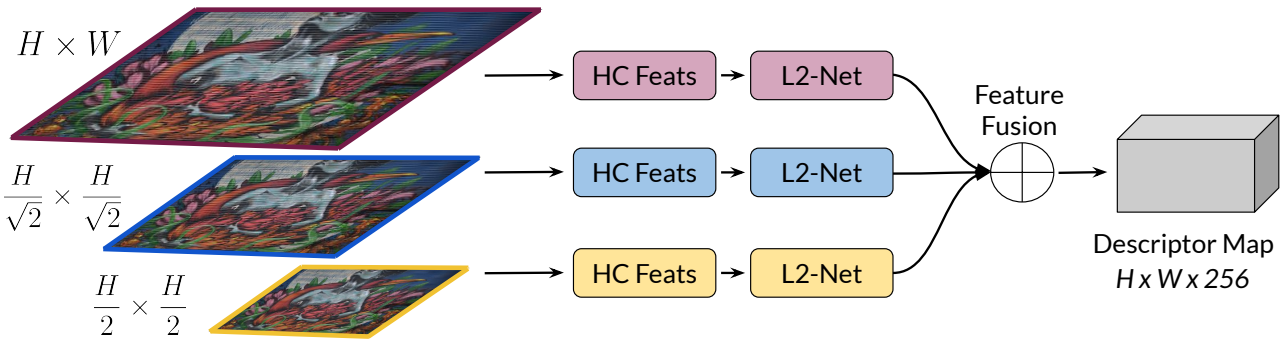


Figure 3: **Multi-Scale Hybrid Descriptor**. A Gaussian pyramid is fed into the block of hand-crafted features that serve as the input to L2-Net. Multi-scale L2-Net features are upsampled and combined through a final convolution.

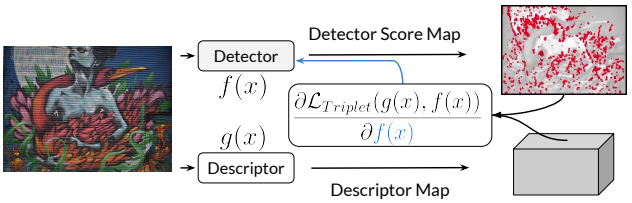


Figure 4: **Triplet loss function** optimises detections based on their descriptor map, refining the feature candidates towards more discriminative positions.

minimise the loss term as shown in figure 4.

Given a pair of corresponding images, we create a grid on each image with a fixed window size of $s \times s$. From each window, we extract a soft-descriptor and its positive and negative samples as illustrated in figure 5. To compute the soft-descriptor, we aggregate all the descriptors within the window based on the detection score map, so that the final soft-descriptor and the scores within a window are entangled. Note that if Non-Maximum Suppression (NMS) was used to select the maximum coordinates and its descriptor, we would only be able to back-propagate through the selected pixels and not the entire map. Consider a window $w_i$ of size $s \times s$ with the score value $r$ at each coordinate $[u, v]$ within the window. A softmax provides:

$$p_i(u,v) = \frac{e^{r_i(u,v)}}{\sum_{j,k}^{s} e^{r_i(j,k)}}. \quad (1)$$

Window $w_i$ has the associated score map $R$, and descriptor vector $D$, at each coordinate $[u, v]$ within the window. We compute the soft-score, $\bar{r}_i$, and soft-descriptor, $\bar{d}_i$, as:

$$\bar{r}_i = \sum_{u,v}^{s} R(u,v) \odot p_i(u,v) \quad \text{and} \quad \bar{d}_i = \sum_{u,v}^{s} D(u,v) \odot p_i(u,v). \quad (2)$$

We use L2 normalisation for the soft-descriptor by projecting it onto the unit hypersphere. Similar to [44, 13], we sample the hardest negative candidate from a non-neighbouring area. This geometric constraint is illustrated in figure 5. We can define our detector triplet loss with soft-descriptors in window $w_i$ as:

$$\mathcal{L}_i(w_i) = \mathcal{L}_i(\delta_+, \delta_-, \mu, \bar{r}_i) = \bar{r}_i \; max(0, \mu + \delta_+ - \delta_-), \quad (3)$$

where $\mu$ is a margin parameter, and $\delta_+$ and $\delta_-$ are the Euclidean distances between positive and negative soft-descriptors pairs. We weight the contribution of each window by its soft-score to control the participation of meaningless windows *e.g.*, flat areas. The final loss is defined as the aggregation of losses on all windows of size $s \times s$:

$$\mathcal{L}_{Trip}(s) = \sum_i \mathcal{L}_i(\delta_+, \delta_-, \mu, \bar{r}_i). \quad (4)$$

**Multi-Scale Context Aggregation.** We extend equation 4 to a multi-scale approach to learn features that are discriminative across a range of scales. Multi-scale learning was used in keypoint detection [28, 12, 13], however, we extend these works by using the multi-scale sampling strategy on the descriptor part. Thus, we sample local soft-descriptors with varying window sizes, $s$, as shown in figure 5, and combine their losses with control parameters $\lambda_s$ in a final term:

$$\mathcal{L}_{MS-Trip} = \sum_s \lambda_s \mathcal{L}_{Trip}(s), \quad (5)$$

**Repeatable & Discriminative.** The detector triplet loss optimises the model to find locations that can potentially be matched. As stated in [18], discriminativeness is not sufficient to train a suitable detector. Therefore, we combine our discriminative loss and the repeatability term M-SIP proposed in [28] with control parameter $\beta$ to balance their contributions:

$$\mathcal{L}_{R\,\&D} = \mathcal{L}_{M-SIP} + \beta \mathcal{L}_{MS-Trip}, \quad (6)$$

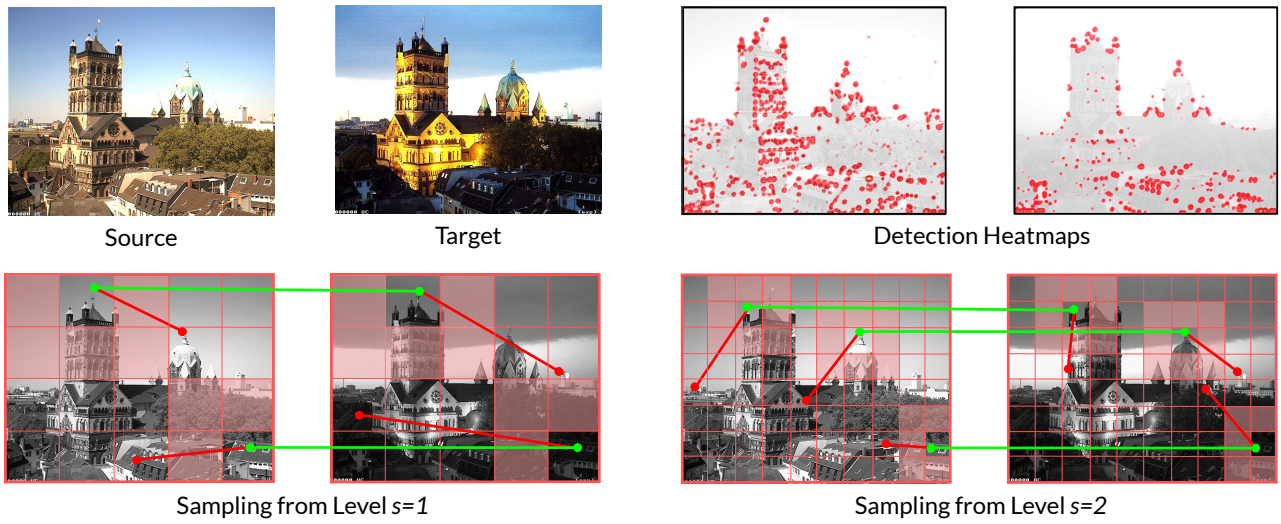


Figure 5: **Triplet Formation Pipeline.** Soft-descriptors are extracted from each window together with their respective positives (green lines) and the hardest negatives (red lines). The negatives are extracted only from non-neighbouring areas (denoted as non-red areas).

**Entangled Detector-Descriptor Learning.** We frame our joint optimisation strategy as follows. The detector is optimised by equation 6, meanwhile, the descriptor learning is based on the hard-mining triplet loss [8]. For the descriptor learning, we use the same sampling approach as in figure 5, however, instead of sampling soft-descriptors, we sample a point-wise descriptor per window. The location to sample the descriptor is provided by an NMS on the detector score map. Hence, our detector refines its candidate positions using the descriptor space, while, the descriptor learning is conditioned by the detector score map sampling. The interaction between parts tightly couples the two tasks and allows for mutual refinement. We alternate the detector and descriptor optimisation steps during training until a mutual convergence is reached. Although it is possible to formulate our optimisation as a single objective minimisation problem, in practice the alternation helped the optimiser converge to a satisfactory minimum.

## 4. Implementation Details

This section introduces relevant implementation details, such as dataset generation and HDD-Net training methodology.

**Training Dataset.** We synthetically create pairs of images by applying random homography transformations to ImageNet images [45]. The random homography parameters are: rotation $[-30°, 30°]$, scale $[0.5, 2.0]$ and skew $[-0.6, 0.6]$. For tackling illumination changes, we use the AMOS dataset [31], which contains sequences of images taken from the same position at different times of the year. We further filter the AMOS dataset and keep only images that are taken during summer between sunrise and midnight time. We generate a total of $12,000$ and $4,000$ images for training and validation, respectively.

**HDD-Net Training and Testing.** Although the detector triplet loss function is applied to the full image, we only use the top $K$ detections for training the descriptor. We select $K = 20$ with a batch size of 8. Thus, in every training batch, there is a total of 160 triplets for training the descriptor. On the detector site, we use $s = [8, 16, 24, 32]$, $\lambda_s = [64, 16, 4, 1]$, and set $\beta = 0.4$. The hyper-parameter search was done on the validation set. We fix HDD-Net descriptor size to 256 dimensions. During test time, we apply a $15 \times 15$ NMS to select candidate locations on the detector score map. Networks and dataset generation were implemented in TensorFlow 1.15 and will be released. Training concludes within 48 hours on a single GTX 1080Ti.

## 5. Experimental Evaluation

This section presents the evaluation results of our method in several application scenarios. The comparison focuses against joint detector and descriptor state of the art approaches.

### 5.1. Architecture Design

**Dataset.** We use the Heinly dataset [46] to validate our architecture design choices. It is a small SfM and homography dataset, we focus on its homography set and

| L2Net-Backbone | $1^{st}$ Order | $2^{nd}$ Order | Gabor Filter | Fully Learnt | $(\cdot)^+$ & $(\cdot)^-$ | Multi-Scale | Heinly MMA (%) |
|---|---|---|---|---|---|---|---|
| ✓ | - | - | - | ✓ | - | - | 41.8 |
| ✓ | - | - | - | - | - | - | 42.0 |
| ✓ | ✓ | - | - | - | - | - | 42.5 |
| ✓ | - | ✓ | - | - | - | - | 43.1 |
| ✓ | - | - | ✓ | - | - | - | 43.3 |
| ✓ | - | - | - | - | - | ✓ | 43.4 |
| ✓ | - | - | ✓ | - | ✓ | - | 43.6 |
| ✓ | - | - | ✓ | - | - | ✓ | 44.1 |
| ✓ | - | - | ✓ | - | ✓ | ✓ | **44.5** |

Table 1: **Ablation Study.** MMA (%) on Heinly dataset [46] for different descriptor designs. Best results are obtained with Gabor filters in the hand-crafted block, $(\cdot)^+$ and $(\cdot)^-$ operators, and multi-scale feature fusion.

use only the sequences that are not part of HPatches [20]. We compute the Mean Matching Accuracy (MMA) [47] as the ratio of correctly matched features within a threshold of 5 pixels and the total number of detected features.

**Ablation Study.** We evaluate a set of hand-crafted filters for extracting features that are robust to rotation. Specifically, $1^{st}$ and $2^{nd}$ order derivatives as well as Gabor filters. In addition, we further test a fully learnt approach without the hand-crafted filters. We also report results showing the impact of splitting the hand-crafted positive and negative features. Finally, our multi-scale approach is also tested against a single-pass architecture without multi-scale feature fusion.

**Results** in table 1 show that Gabor filters obtain better results than $1^{st}$ or $2^{nd}$ order derivatives. They are especially effective for rotation since they are designed to detect patterns under specific orientations. Besides, results without constraining the rotational block to any specific filter are slightly lower than the baseline. The fully learnt model could be improved by adding more filters, but if we restrict the design to a single filter, hand-crafted filter with $(\cdot)^+$ and $(\cdot)^-$ operators give the best performance. Lastly, a notable boost over the baseline comes from our proposed multi-scale pyramid and feature fusion within the architecture.

## 5.2. Image Matching

**Dataset.** We use the HPatches dataset [20] with 116 sequences, including viewpoint and illumination changes. We compute results for sequences with image resolution smaller than 1200 × 1600 following the approach in [14]. To demonstrate the impact of the detector and to make a fair comparison between different methods, we extend the detector evaluation protocol proposed in [21] to the matching metrics by computing the MMA score for the top 100, 500, and 1,000 keypoints.

**Effect of Triplet Learning on Detector.** Table 2 shows HDD-Net results when training its detections to be repeatable or/and discriminative. The performance of $\mathcal{L}_{MS-Trip}$ only is lower than $\mathcal{L}_{M-SIP}$, which is in line with [15]. Repeatable features are crucial for matching images, however, best results are obtained when combining repeatable and discriminative loss terms for the detector learning. The results show that the combination of both principles into a single score map detection is effective.

**Comparison to SOTA.** Figure 6 compares our HDD-Net to different algorithms. HDD-Net outperforms all the other methods for viewpoint and illumination sequences on every threshold, excelling especially in the viewpoint change, that includes the scale and rotation transformations for which HDD-Net was designed. SuperPoint [11] performance is lower when using only top 100 keypoints, and even though no method was trained with such constraint, the other models keep their performance very close to their 500 or 1,000 results. When constraining the number of keypoints, D2Net-SS [14] results are higher than for its multi-scale version D2Net-MS, D2Net-MS was reported in [14] to achieve higher performance when using an unlimited number of features.

| | HPatches (MMA) | |
|---|---|---|
| | View | Illum |
| $\mathcal{L}_{MS-Trip}$ | 26.4 | 34.9 |
| $\mathcal{L}_{M-SIP}$ | 38.3 | 35.5 |
| $\mathcal{L}_{M-SIP}$ & $\mathcal{L}_{MS-Trip}$ | **38.9** | **41.5** |

Table 2: MMA (%) results for different detector optimisation objectives on HPatches [20] dataset.

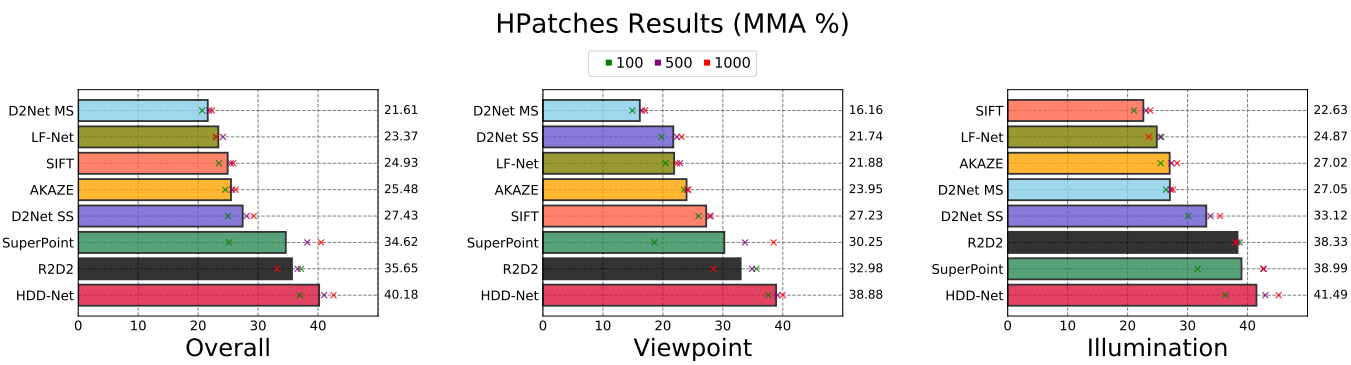


Figure 6: Mean Matching Accuracy (MMA) on HPatches dataset [20] for top 100, 500 and 1,000 extracted points. Methods are sorted on descending order by their score on each of the splits.

### 5.3. 3D Reconstruction

**Dataset**. We use the ETH SfM benchmark [48] for the 3D reconstruction task. We select three sequences; *Madrid Metropolis*, *Gendarmenmarkt*, and *Tower of London*. We report results in terms of registered images, sparse points, track length, and reprojection error. The top 2,048 points are used as in [23], which still provides a fair comparison between methods at a much lower cost. The sparse and dense reconstructions are performed using COLMAP [2] software. In addition, we used one-third of the images in each dataset to reduce the computational time.

**Results**. Table 3 presents the results for the 3D reconstructions experiment. HDD-Net and SuperPoint obtain the best results overall. While HDD-Net recovers more sparse points and registers more images in *Madrid Metropolis* and *Tower of London*, SuperPoint does it for *Geendarmenmarkt*. Their accuracy leads to more dense reconstructions than D2-Net or R2D2 networks. D2-Net features did not allow to reconstruct any model on *Madrid Metropolis* within the evaluation protocol *i.e.*, small regime on the number of extracted keypoints. Due to challenging examples with moving objects within the images and sometimes the object of interest being in distant views, recovering a 3D model from a subset of keypoints makes the reconstruction task even harder. Even though, limiting the total number of extracted points for each method also gives an indicator of the precision and relevance of such keypoints. In terms of a track length, that is the number of images in which at least one feature was successfully tracked, R2D2 and HDD-Net outperform all the other methods. LF-Net reports a smaller reprojection error followed by SIFT and HDD-Net. Although the reprojection error is small in LF-Net, their number of sparse points and registered images are below other competitors.

| | Reg. Images | Sparse Points | Track Length | Reproj. Err. |
|---|---|---|---|---|
| | **Madrid Metropolis** (448 Images) | | | |
| SIFT [4] | 27 | 1140 | 4.34 | 0.69 |
| LF-Net [12] | 19 | 467 | 4.22 | **0.62** |
| SuperPoint [11] | 39 | 1258 | 5.08 | 0.96 |
| D2Net-SS [14] | – | – | – | – |
| D2Net-MS [14] | – | – | – | – |
| R2D2 [15] | 22 | 984 | 4.85 | 0.88 |
| HDD-Net | **43** | **1374** | **5.25** | 0.80 |
| | **Gendarmenmarkt** (488 Images) | | | |
| SIFT [4] | 132 | 5332 | 3.68 | **0.86** |
| LF-Net [12] | 99 | 3460 | 4.65 | 0.90 |
| SuperPoint [11] | **156** | **6470** | 5.93 | 1.21 |
| D2-Net SS [14] | 17 | 610 | 3.31 | 1.04 |
| D2-Net MS [14] | 14 | 460 | 3.02 | 0.99 |
| R2D2 [15] | 115 | 3834 | **7.12** | 1.05 |
| HDD-Net | 154 | 6174 | 6.30 | 0.98 |
| | **Tower of London** (526 Images) | | | |
| SIFT [4] | 75 | 4621 | 3.21 | 0.71 |
| LF-Net [12] | 76 | 3847 | 4.63 | **0.56** |
| SuperPoint [11] | 111 | 5760 | 5.41 | 0.75 |
| D2-Net SS [14] | 10 | 360 | 2.93 | 0.94 |
| D2-Net MS [14] | 10 | 64 | 5.95 | 0.93 |
| R2D2 [15] | 81 | 3756 | **6.02** | 1.03 |
| HDD-Net | **116** | **6039** | 5.45 | 0.80 |

Table 3: 3D Reconstruction results on ETH 3D benchmark [48]. Best results are in bold. Dash symbol (–) means that COLMAP could not reconstruct any model.

| | Aachen Day-Night | | |
|---|---|---|---|
| | Correct Localised Queries (%) | | |
| Localisation Thres. | 0.5m, 2° | 1m, 5° | 5m, 10° |
| SIFT [4] | 33.7 | 52.0 | 65.3 |
| SuperPoint [11] | 42.9 | 61.2 | 85.7 |
| D2-Net SS [14] | 44.9 | 65.3 | **88.8** |
| D2-Net MS [14] | 41.8 | **68.4** | **88.8** |
| R2D2 [15] | **45.9** | 66.3 | **88.8** |
| HDD-Net | 43.9 | 62.2 | 82.7 |

Table 4: Aachen Day-Night [49] results on camera localisation.

### 5.4. Camera Localisation

**Dataset.** The Aachen Day-Night [49] contains more than 5,000 images, with separate queries for day and night[1]. Due to the challenging data, and to avoid convergence issues, we increase the number of keypoints to 8,000. Despite that, LF-Net features did not allow to converge and are not included in table 4.

**Results.** The best results for the most permissive error threshold are reported by D2-Net networks and R2D2. Note that D2-Net and R2D2 are trained on MegaDepth [50], and Aachen datasets, respectively, which contains real 3D scenes under similar geometric conditions. In contrast, SuperPoint and HDD-Net use synthetic training data, and while they perform better on image matching or 3D reconstruction, their performance is lower on localisation. As a remark, results are much closer in the most restrictive error, showing that HDD-Net and SuperPoint are on par with their competitors for more accurate camera localisation.

## 6. Conclusion

In this paper, we have introduced a new detector-descriptor method based on a hand-crafted block and multi-scale image representation within the descriptor. Moreover, we have reformulated the triplet loss function to not only learn the descriptor part but also to refine the proposed keypoint locations from the detector. We validate our contributions in the image matching task, where HDD-Net outperforms the baseline with a wide margin. Furthermore, we show through extensive experiments across different tasks that our approach outperforms or performs as well as the top joint detector-descriptor algorithms in terms of matching accuracy, number of registered images and reconstructed 3D points, despite using only synthetic and much fewer data samples for training.

[1]We use the benchmark from the CVPR 2019 workshop on Long-term Visual Localization.

## References


[1] Marvin Teichmann, Andre Araujo, Menglong Zhu, and Jack Sim. Detect-to-retrieve: Efficient regional aggregation for image search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5109–5118, 2019. 1

[2] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016. 1, 7

[3] Benjamin Busam, Patrick Ruhkamp, Salvatore Virga, Beatrice Lentes, Julia Rackerseder, Nassir Navab, and Christoph Hennersperger. Markerless inside-out tracking for 3d ultrasound compounding. In *Simulation, Image Processing, and Ultrasound Systems for Assisted Diagnosis and Navigation*, pages 56–64. Springer, 2018. 1

[4] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 1, 7, 8

[5] Pablo Fernández Alcantarilla, Jesús Nuevo, and Adrien Bartoli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. *BMVC*, 2013. 1, 3

[6] Stefan Leutenegger, Margarita Chli, and Roland Siegwart. Brisk: Binary robust invariant scalable keypoints. In *2011 IEEE international conference on computer vision (ICCV)*, pages 2548–2555. Ieee, 2011. 1

[7] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 661–669, 2017. 1, 2

[8] Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor's margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, pages 4826–4837, 2017. 1, 2, 3, 5

[9] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11016–11025, 2019. 1, 2

[10] Zixin Luo, Tianwei Shen, Lei Zhou, Siyu Zhu, Runze Zhang, Yao Yao, Tian Fang, and Long Quan. Geodesc: Learning local descriptors by integrating geometry constraints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 168–183, 2018. 1, 2

[11] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 224–236, 2018. 1, 2, 3, 6, 7, 8

[12] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: learning local features from images. In *Advances in Neural Information Processing Systems*, pages 6234–6244, 2018. 1, 2, 3, 4, 7

[13] Xuelun Shen, Cheng Wang, Xin Li, Zenglei Yu, Jonathan Li, Chenglu Wen, Ming Cheng, and Zijian He. Rf-net: An end-to-end image matching network based on receptive field. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8132–8140, 2019. 1, 2, 3, 4

[14] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint detection and description of local features. *arXiv preprint arXiv:1905.03561*, 2019. 1, 2, 3, 6, 7, 8

[15] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yohann Cabon, and Martin Humenberger. R2d2: Repeatable and reliable detector and descriptor. *arXiv preprint arXiv:1906.06195*, 2019. 1, 2, 3, 6, 7, 8

[16] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *European Conference on Computer Vision*, pages 467–483. Springer, 2016. 1, 2, 3

[17] Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, , and Long Quan. Aslfeat: Learning local features of accurate shape and localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1, 2, 3

[18] Tinne Tuytelaars and Krystian Mikolajczyk. Local invariant feature detectors: a survey. *Foundations and Trends in Computer Graphics and Vision*, 2008. 2, 3, 4

[19] Gabriela Csurka, Christopher R Dance, and Martin Humenberger. From handcrafted to deep local features. *arXiv preprint arXiv:1807.10254*, 2018. 2

[20] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5173–5182, 2017. 2, 6, 7

[21] Karel Lenc and Andrea Vedaldi. Large scale evaluation of local image feature detectors on homography datasets. *BMVC*, 2018. 2, 3, 6

[22] David Bojanić, Kristijan Bartol, Tomislav Pribanić, Tomislav Petković, Yago Diez Donoso, and Joaquim Salvi Mas. On the comparison of classic and deep keypoint detector and descriptor methods. In *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pages 64–69. IEEE, 2019. 2

[23] Jin Yuhe, Dmytro Mishkin, Anastasiia Mishchuk, Jiri Matas, Pascal Fua, Kwang Moo Yi, and Eduard Trulls. Image matching across wide baselines: From paper to practice. In *arXiv preprint arXiv:2003.01587*, 2020. 2, 7

[24] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European conference on computer vision*, pages 430–443. Springer, 2006. 2

[25] Yannick Verdie, Kwang Yi, Pascal Fua, and Vincent Lepetit. Tilde: a temporally invariant learned detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5279–5288, 2015. 2, 3

[26] Karel Lenc and Andrea Vedaldi. Learning covariant feature detectors. In *European Conference on Computer Vision*, pages 100–117. Springer, 2016. 2

[27] Xu Zhang, Felix X Yu, Svebor Karaman, and Shih-Fu Chang. Learning discriminative and transformation covariant local feature detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6818–6826, 2017. 2

[28] Axel Barroso-Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Key.net: Keypoint detection by handcrafted and learned cnn filters. *International Conference on Computer Vision*, 2019. 2, 3, 4

[29] Vassileios Balntas, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks. In *BMVC*, volume 1, page 3, 2016. 2

[30] Kun He, Yan Lu, and Stan Sclaroff. Local descriptors optimized for average precision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 596–605, 2018. 2

[31] Milan Pultar, Dmytro Mishkin, and Jiří Matas. Leveraging outdoor webcams for local descriptor learning. *arXiv preprint arXiv:1901.09780*, 2019. 2, 5

[32] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 2

[33] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019. 2

[34] Patrick Ebel, Anastasiia Mishchuk, Kwang Moo Yi, Pascal Fua, and Eduard Trulls. Beyond cartesian representations for local descriptors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 253–262, 2019. 2

[35] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999, 2016. 3

[36] Patrick Follmann and Tobias Bottger. A rotationally-invariant convolution module by feature map back-rotation. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 784–792. IEEE, 2018. 3

[37] Daniel E Worrall and Max Welling. Deep scale-spaces: Equivariance over scale. *arXiv preprint arXiv:1905.11697*, 2019. 3

[38] Sander Dieleman, Kyle W Willett, and Joni Dambre. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly notices of the royal astronomical society*, 450(2):1441–1459, 2015. 3

[39] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015. 3

[40] Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu. Exploiting cyclic symmetry in convolutional neural networks. *arXiv preprint arXiv:1602.02660*, 2016. 3

[41] Alberto Crivellaro and Vincent Lepetit. Robust 3d tracking with descriptor fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3414–3421, 2014. 3

[42] Krystian Mikolajczyk and Cordelia Schmid. Indexing based on scale invariant interest points. *ICCV*, 2001. 3

[43] Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Repeatability is not enough: Learning affine regions via discriminability. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 284–300, 2018. 3

[44] Dmytro Mishkin, Jiri Matas, and Michal Perdoch. Mods: Fast and robust method for two-view matching. *Computer Vision and Image Understanding*, 141:81–93, 2015. 4

[45] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 5

[46] Jared Heinly, Enrique Dunn, and Jan-Michael Frahm. Comparative evaluation of binary features. In *European Conference on Computer Vision*, pages 759–773. Springer, 2012. 5, 6

[47] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. 2005. 6

[48] Johannes L Schonberger, Hans Hardmeier, Torsten Sattler, and Marc Pollefeys. Comparative evaluation of hand-crafted and learned local features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1482–1491, 2017. 7

[49] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, et al. Benchmarking 6dof outdoor visual localization in changing conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8601–8610, 2018. 8

[50] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 8