Applied Intelligence (2024) 54:5406–5416
https://doi.org/10.1007/s10489-024-05418-w

AFSRNet: learning local descriptors with adaptive multi-scale feature fusion and symmetric regularization

Dong Li[1] · Haowen Liang[1] · Kin-Man Lam[2]

Accepted: 23 March 2024 / Published online: 20 April 2024

Abstract

Multi-scale feature fusion has been widely used in handcrafted descriptors, but has not been fully explored in deep learning-based descriptor extraction. Simple concatenation of descriptors of different scales has not been successful in significantly improving performance for computer vision tasks. In this paper, we propose a novel convolutional neural network, based on center-surround adaptive multi-scale feature fusion. Our approach enables the network to focus on different center-surround scales, resulting in improved performance. We also introduce a novel regularization technique that uses second-order similarity to constrain the learning of local descriptors, based on the symmetric property of the similarity matrix. The proposed method outperforms single-scale or simple-concatenation descriptors on two datasets and achieves state-of-the-art results on the Brown dataset. Furthermore, our method demonstrates excellent generalization ability on the HPatches dataset. Our code is released on GitHub: https://github.com/Leung-GD/AFSRNet/tree/main.

Keywords Local descriptor · Multi-scale feature fusion · Symmetric regularization

1 Introduction

In recent years, there has been a growing interest in 3D construction [1, 2], image matching [3, 4], and image registration [5]. Efficient descriptor learning has emerged as a crucial research area within these computer vision applications. The dominant approach to generate local feature descriptors is to encode image patches into representative vectors.

In previous studies, the focus was on handcrafted descriptors, which require sophisticated engineering knowledge and mathematical derivation processes [6]. Traditional image-matching pipelines rely on handcrafted descriptors, which have been successful in various applications. However, learning-based descriptors have demonstrated better scalability, robustness, and discriminability, achieving higher matching performance compared to handcrafted descriptors [7]. In this case, a single-scale network cannot realize the specific size of specific learning and cannot integrate more rich information.

Previous studies primarily focused on handcrafted descriptors, which require sophisticated engineering knowledge and mathematical derivation processes [6]. Traditional image matching pipelines, based on handcrafted descriptors like SIFT [8] and SURF [9], have been successful in various applications. However, these crafted descriptors still face challenges, such as the omission of important image details. Recently, learning-based descriptors [7, 10, 11] have demonstrated better scalability, robustness, and discriminability, achieving higher matching performance compared to handcrafted descriptors. These studies show that deep learning can greatly improve the efficiency of descriptors. In this case, a single-scale network is typically used to learn local features. However, a single-scale network cannot realize specific-scale learning, thereby failing to integrate richer information. Center-surround multi-scale feature extraction is performed by generating sub-patches of different sizes cropped from the central regions of input patches. Features are then extracted from the respective sub-patches to form multi-scale features. The adoption of deep learning-based descriptor extraction, as seen in CS L2Net [10] and Patch-NetVLAD [12], is more effective in introducing richer information into the learned

✉ Haowen Liang
lianghaowen163@163.com

Dong Li
dong.li@gdut.edu.cn

Kin-Man Lam
kin.man.lam@polyu.edu.hk

[1] Guangdong University Of Technology, Guangzhou 510006, China

[2] The Hong Kong Polytechnic University, HongKong 999077, China

features. However, it is important to highlight that simple concatenation and fixed-weight fusion, commonly used in this approach, cannot be optimized during backpropagation.

The descriptor network based on the triplet loss function proves effective in enhancing descriptor performance under challenging conditions. Numerous studies have focused on refining this loss function, incorporating regularization for modification [13–17]. This function is commonly employed to train descriptor networks, emphasizing that the distance between descriptors of negative samples should exceed that between descriptors of positive samples. Despite its effectiveness, prior works often overlook the characteristics of the descriptor distance measure. The relative distance between matching pairs can also be constrained in the regularization term. Additionally, while previous works commonly simplify calculations using a similarity matrix, they neglect that the symmetry of the similarity matrix can be used for the network's training constraint.

To tackle the previously mentioned challenges, we propose a novel approach called AFSRNet, a center-surround multi-scale convolutional neural network designed for extracting local descriptors. AFSRNet employs three branches as its input, each dedicated to a distinct center-surround scale for processing input patches. These branches generate features of the same size using dilated convolution. In addition to the network architecture, we address descriptor learning challenges by introducing a unique regularization term based on symmetry of the descriptor similarity matrix. Unlike previous works, our approach considers not only the distance between descriptors of negative and positive samples, but also the relative distance between matching pairs in the regularization term. Moreover, we perform the regularization by enhancing symmetry of the similarity matrix to further decrease the calculation of descriptor learning. In summary, the main contributions of our work include the following:

1) We introduce a novel neural network based on center-surround Adaptive Multi-Scale Feature Fusion (AMSF) for learning local descriptors, which can be optimized during backpropagation.
2) We propose a new regularization term called Symmetric Regularization (SR), which constrains the similarity matrix of the descriptors to improve the robustness of the learned descriptors.
3) By combining SR with triplet loss, our descriptor-learning network can be trained to achieve state-of-the-art results for local descriptor learning.

2 Related work

The research of designing local descriptors has gradually moved from handcrafted ones to learning-based ones. Since the purpose of this paper is descriptor learning, below we give a brief review of descriptor learning methods in this paper, ranging from traditional methods to the recently proposed learning-based methods and various applications of multi-scale feature fusion.

2.1 Descriptors learning network

Handcrafted descriptors for local patches have primarily focused on mathematical derivations, such as gradient filters and intensity comparisons. SIFT [8] is widely considered as the most commonly used real-valued descriptor, which computes smoothed histograms using the gradient field of the image patch. SURF[9] utilizes a box filter to extract image gradient information and employs a multi-scale filter in the scale space, replacing the downsampling operation in SIFT to improve computational efficiency. This modification effectively enables many practical applications to be realized. While SURF has demonstrated the significance of multi-scale features in traditional descriptor learning, they are rarely incorporated into deep learning-based methods.

HardNet [11] employs a simple but effective strategy known as hard negative mining, which highlights the significance of proper sampling. This sampling strategy aims to select the most indistinguishable image descriptors to train the network, resulting in improved robustness in patch matching. Consequently, many models, including our own, have adopted this sampling strategy since its introduction. Furthermore, Liang et al. [18] proposed a multi-level aggregation technique that facilitates descriptor learning across the entire network. Each level of their network extracts a feature vector after feature fusion, and the final descriptor concatenates these outputs. CRBNet [19] is built upon L2Net and starts by using strided convolutional layers for downsampling. They bring in a residual learning framework for enhancement. The architecture has four stages with shortcut connections, and they strategically put strided convolutional layers in the last two stages to effectively keep the input patch information. Zhang et al. DarkFeat [20] addresses the descriptor challenges in low-light visual perception at night. The local feature descriptor is crucial for such applications, face performance degradation in extreme low-light scenarios due to low signal-to-noise ratio in images. To overcome this, the paper proposes a deep learning model designed for end-to-end detection and description of local features directly from RAW format images captured in extreme low-light conditions.

The majority of the aforementioned research on descriptor matching primarily focuses on single-scale approaches. In contrast, we propose a center-surround multi-scale fusion network to enhance matching accuracy and enable the network to focus on different parts of the image patch. In the following section, we provide a detailed description of our network architecture and its components.

2.2 Multi-scale feature fusion

Multiscale feature fusion has greatly contributed to target detection and semantic segmentation tasks by combining semantic and spatial information from different feature scales to improve overall performance. Multiscale input methods include building multiscale pyramids, using multiscale intermediate feature maps, and parallel inputting of multiscale image information.

In serial multi-scale architectures, FPN [21] integrates multi-scale features with a modified top-down path using center-cropped patches, creating a pyramid-like structure. EFPN [22] enhances object detection with a Pyramid Network featuring an extra high-res level for small object detection. MSPFN [23] achieves rain streak removal with recurrent calculation and a multi-scale pyramid. MFANet [24] improves detection accuracy using channel attention. In parallel multi-scale networks, SPP [25] extracts features with multiple pooling layers. ASPP [26] captures multi-scale information with parallel atrous convolutions. Trident-net [27] adjusts the receptive field with parallel multi-branch architectures, enhancing feature extraction efficiency.

Building upon insights gained from prior methods, CS L2Net [10] introduces a novel architectural paradigm in the realm of feature extraction. Departing from conventional approaches, CS L2Net embraces a concatenation tower structure that leverages the strength of two distinct L2Net models, each characterized by parallel towers. PatchNetVLAD [12] deviates from conventional concatenation by employing fixed weights for the fusion of descriptors across center-surround scales. The process involves cutting patches, generating descriptors, and applying predetermined weights for fusion.

However, an important observation is that existing multiscale fusion methods lack learnability or relies on a direct connection. This limitation restricts the optimization of the fusion process during backpropagation. To overcome this challenge, our proposed approach introduces an adaptive center-surround multi-scale fusion method. This innovative strategy addresses the non-learnable nature of existing techniques, allowing dynamic optimization of descriptor fusion across various center-surround scales through adaptive weight adjustment during backpropagation.

To effectively utilize multi-scale features, we also design a parallel multi-branch and center-surround network. Each branch focuses on studying features at a particular scale, and an efficient fusion technique is employed instead of simple concatenation. This approach allows us to fully exploit the benefits of multi-scale information. We propose a feature fusion approach that can make adaptive learning and achieve better performance.

2.3 Descriptor distance constraint

Modifying the loss function and regularization terms has proven to be an effective approach for constraining the distance of descriptors. Triplet loss [28] enable the learning of more suitable network weights by comparing Euclidean distances between samples. While Euclidean distances provide an absolute measurement, relative distance measurement is more appropriate for vector embedding learning. To incorporate relative measurement into descriptor learning, RALNet [13] utilizes the angle distance between feature vectors instead of the L2 distance to measure their similarity. In our approach, the regularization of the loss function is also designed based on angular distance. In addition to modifying how descriptor distances are measured, researchers have explored modifications to other aspects of the loss function. For instance, CDF [17] enhances the triplet loss by utilizing a dynamic margin based on cumulative distribution instead of a fixed margin. HSD [15] projects the entire network onto hyperspherical space by altering the normalization, demonstrating that hyperspherical learning is more suitable for descriptors. RDLNet [16] focuses on learning hard samples and compact descriptors through triplet networks, directing the network's attention to challenging examples and promoting the generation of concise descriptors.

The constraint of descriptor distances solely based on matched and unmatched pairs may not be sufficiently robust. In addition to first-order optimization techniques, SOSNet [14] demonstrates that second-order constraints further enhance the quality of descriptors. However, SOSNet overlooks the fact that the properties distance between matching pairs can also be constrained within the regularization term. In our approach, we introduce this constraint into the second-order constraint and implement it using the similarity matrix. This regularization term incorporates the consideration of matching pair properties, improving the overall performance and robustness of the descriptors.

3 Method

3.1 Network architecture

Figure 1 shows the main architecture of AFSRNet, which mainly contains three parts: multi-scale input, feature extraction, and feature fusion. Each part is described in detail in the following.

Multi-scale input Previous work has demonstrated that focusing on pixels near the center of a patch and paying more

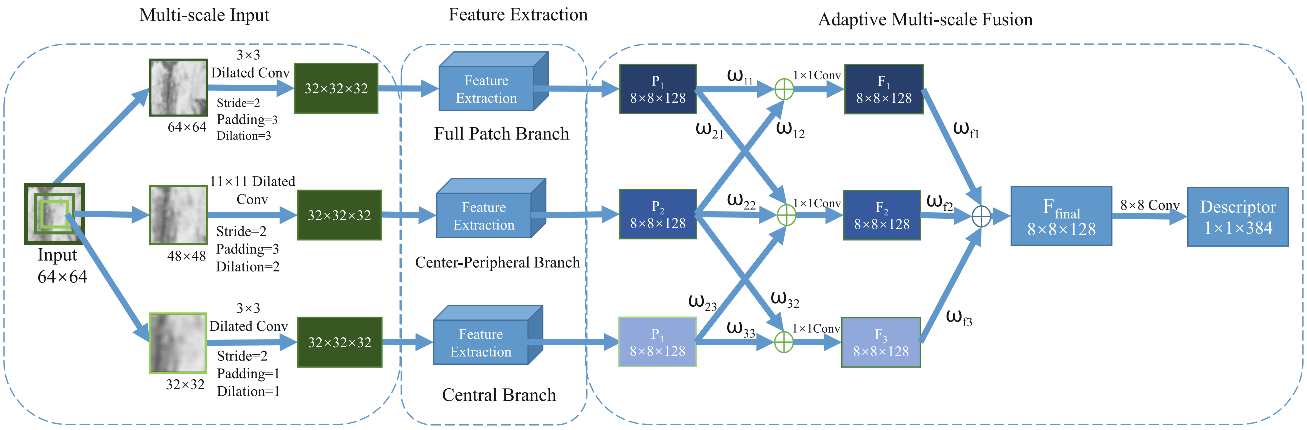


Fig. 1 The main architecture of AFSRNet consists of three parts: multi-scale input, feature extraction, and adaptive multi-scale fusion. The architecture includes three branches, named Full Patch Branch, Center-Peripheral Branch, and Central Branch from top to bottom

attention to the central region of input patches can enhance the accuracy of descriptor matching. To this end, we propose a three-stream structure that contains the central region at all three different scales. As shown in Fig. 1, the three multi-scale branches are referred to as the full-patch branch, center-peripheral branch, and central branch, with sizes of 64×64, 48×48, and 32×32, respectively. The latter two parts are obtained by cutting in a center-surround way. The one closest to the center, and also the smallest, is called the central part, corresponding to the center branch. The second smallest part is the center-periphery section between the full patch and the center, containing the center part, and it corresponds to the center-periphery branch. The three inputs are processed using dilated convolution with different strides in the three branches to generate features of dimension 32×32×32. Such a central-surround multi-scale patch information input is critical for enhancing the performance of descriptor matching.

Feature Extraction After adjusting the size of the feature maps from the cropped patches, we feed them into feature extraction module. The architecture of the feature extraction module of AFSRNet is shown in Fig. 2. Instead of pooling layers, the spatial size is reduced using stridden convolutions since pooling layers tend to negatively impact the performance of the descriptor [11]. The output of this module is a feature map of size 8×8×128.

Feature Fusion In our proposed method, we utilize adjacent sub-networks in parallel for feature fusion. This allows the descriptor to focus on feature information from different center-surround receptive fields. To achieve adaptive feature map fusion, we implement a fusion method based on normalized weights, as follows:

$$F_1 = Conv(\frac{\omega_{11} \cdot P_1 + \omega_{12} \cdot P_2}{\omega_{11} + \omega_{12} + \varepsilon}), \tag{1}$$

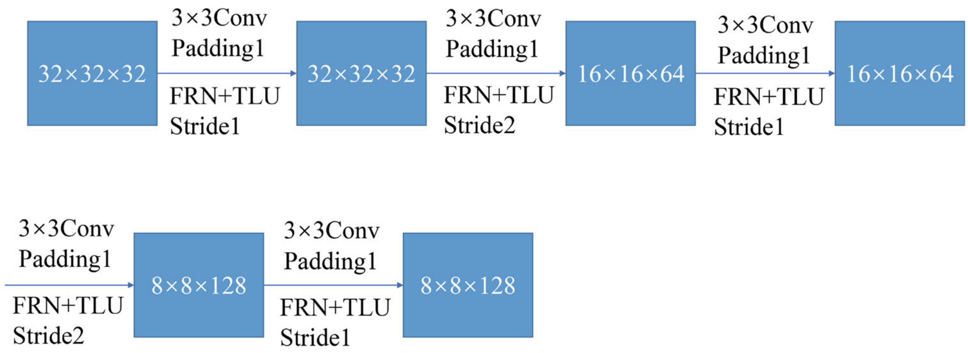
$$F_2 = Conv(\frac{\omega_{21} \cdot P_1 + \omega_{22} \cdot P_2 + \omega_{23} \cdot P_3}{\omega_{21} + \omega_{22} + \omega_{23} + \varepsilon}), \tag{2}$$

$$F_3 = Conv(\frac{\omega_{32} \cdot P_2 + \omega_{33} \cdot P_3}{\omega_{32} + \omega_{33} + \varepsilon}), \tag{3}$$

$$F_{final} = \frac{\omega_{f_1} \cdot F_1 + \omega_{f_2} \cdot F_2 + \omega_{f_3} \cdot F_3}{\omega_{f_1} + \omega_{f_2} + \omega_{f_3} + \varepsilon}. \tag{4}$$

The weight ω_{ij} of the i^{th} output and the j^{th} input is learnable. To ensure that $\omega_{ij} \geq 0$, we apply a ReLu after each ω_{ij}.

Fig. 2 The feature extraction architecture is adapted from HyNet [29]. Each convolutional layer is followed by Filtere Response Normalization (FRN) and Thresholded Linear Unit(TLU)



Besides, $\varepsilon = 0.0001$ is a small value to avoid numerical instability.

3.2 Loss function and regularization

Triplet loss Triplet loss is a commonly used loss function for training local descriptors. It enforces smaller distances between positive matches and larger distances between negative matches. The common expression for triplet loss is as follows:

$$L_{triplet} = \frac{1}{N}\sum_{i}^{N} max(s(a_i, p_i) - s(a_i, n_i) + m, 0), \tag{5}$$

where a, p and n represent an anchor, a positive and a negative of the triplet tuple, m represents the margin and function $s(x, y)$ represents the similarity score between the two features x and y, and x and y are L2 normalized. Sampling is crucial to achieve both performance gain and computational efficiency, thus, we adopt the same sampling strategy as HardNet [11]. RALNet [13] has demonstrated that cosine similarity is superior to Euclidean distance in comparing descriptor distances. Therefore, we define the similarity function $s(x, y)$ as follows, hereafter denoted as s_{xy} for simplicity, as follows:

$$s(x, y) = 1 - x \cdot y^T = 1 - ||x|| \, ||y|| \cos\theta_{xy}, \tag{6}$$

where θ_{xy} represents the angle between x and y.

Regularization During training, a training batch consists of N pairs of matched patches. The size of the descriptor matrices A and P are $N \times 384$ while the similarity matrix D is $N \times N$. The similarity matrix D is shown as follows:

$$D = (1 - A \cdot P^T) = \begin{bmatrix} s_{a_1 p_1} & s_{a_1 p_2} & \cdot & s_{a_1 p_N} \\ s_{a_2 p_1} & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ s_{a_N p_1} & \cdot & \cdot & s_{a_N p_N} \end{bmatrix}. \tag{7}$$

The foundational principle of second-order similarity (SOS) posits that vertices with similar neighbors are likely to exhibit similarity. SOSNet [14] employs second-order similarity regularization (SOSR) by minimizing the Euclidean distance between a_i,a_j and p_i,p_j to enhance descriptor similarity. However, the efficacy of using Euclidean distance as a second-order similarity measure may be limited [13]. Additionally, SOSNet overlooks the potential improvement that could result from integrating first-order similarity properties into second-order similarity constraints. While constructing similarity matrices for training is common in descriptor methodologies, these matrices are rarely utilized for regularization terms within constraints.

In response to these limitations, we introduce a novel regularization term, symmetric regularization (SR). The motivation behind SR lies in addressing the inadequacies of relying solely on Euclidean distance and exploring the untapped potential of incorporating first-order and second-order similarity properties. To achieve this, SR strategically leverages distance matrices, which comprehensively capture the pairwise relationships between descriptors.

Distance matrices provide a detailed representation of the similarity landscape by encapsulating the distances between all descriptor pairs. This nuanced understanding of pairwise relationships enables a more refined and context-aware descriptor learning process. Moreover, SR places emphasis on the symmetry of similarity relationships.

By incorporating distance matrices and symmetry considerations into the regularization process, SR aims to refine descriptor training. The regularization term encourages bidirectional consistency in similarity relationships, enhancing the overall robustness and accuracy of the descriptor model. This strategic incorporation of distance matrices and symmetry considerations represents a departure from conventional approaches, ensuring a more detailed exploration of the underlying structures within descriptor spaces. The introduction of SR contributes to a more nuanced understanding of the intrinsic relationships between descriptors, ultimately leading to improved descriptor learning outcomes.

Proposition 1 *Let a_i, p_i are pairs of matched descriptors in a batch and they are second order similar, that is*

$$First\ Order\ Similitry\,(FOS): \; s_{a_i p_i} = s_{a_j p_j} = 0, \tag{8}$$

$$Second\ Order\ Similitry\,(SOS): \; s_{a_i a_j} = s_{p_i p_j}, \tag{9}$$

then the similarity matrix D is symmetric.

Proof Based on (8), we can conclude that

$$\begin{cases} a_i \cdot {p_i}^T = ||a_i|| \, ||p_i|| \cos\theta_{a_i p_i} = 1, \\ a_j \cdot {p_j}^T = ||a_j|| \, ||p_j|| \cos\theta_{a_j p_j} = 1 \end{cases} \tag{10}$$

The descriptors are all L2 normalized, so the norm of them equal to 1, and thus can be conclude from (6) and (10) that

$$\theta_{a_i p_i} = \theta_{a_j p_j} = 0. \tag{11}$$

Based on (9) and (6), we can conclude that

$$\theta_{a_i a_j} = \theta_{p_i p_j}, \tag{12}$$

and we have that

$$\begin{cases} \theta_{a_i p_j} = \theta_{a_i a_j} \pm \theta_{a_j p_j}, \\ \theta_{a_j p_i} = \theta_{p_i p_j} \pm \theta_{a_i p_i}. \end{cases} \tag{13}$$

Then, according to (6), (10), (11) and (12), we derive that

$$\begin{cases} \theta_{a_i p_j} = \theta_{a_j p_i}, \\ s_{a_i p_j} = s_{a_j p_i} \end{cases} \tag{14}$$

Then referring to (7), $s_{a_i p_j}$ and $s_{a_j p_i}$ are the symmetrical elements of the similarity matrix D. Therefore, it can conclude that if $s_{a_i p_j} = s_{a_j p_i}$, D=D^T, so the similarity matrix D is a symmetric matrix.

If D is a symmetric matrix and a_i, p_i are fisrt order matched, we have (11) and (14), then we conclude that:

$$\begin{cases} \theta_{a_i a_j} = \theta_{a_i p_j} \pm \theta_{a_j p_j} = \theta_{a_i p_j}, \\ \theta_{p_i p_j} = \theta_{a_j p_i} \pm \theta_{a_j p_j} = \theta_{a_j p_i}. \end{cases} \tag{15}$$

Then referencing to (6), we conclude that $s_{a_i a_j} = s_{p_i p_j}$. So if D is a symmetric matrix and a_i, p_i are fisrt order matched, we can conclude that a_i, p_i are second order similar.

Finally, we have that if and only if a_i, p_i are pairs of matched descriptors and they are second order similar, the similarity matrix of them is symmetric. □

This suggests that enhancing the symmetry of the descriptor similarity matrix can lead to second-order similarity among the descriptors within the same batch. We formulate the symmetric regularization term as follows:

$$SR = \frac{\|\mathbf{D} - \mathbf{D}^\top\|_\mathbf{F}}{\|\mathbf{D}\|_\mathbf{F}}, \tag{16}$$

where $\|\mathbf{D}\|_\mathbf{F}$ denotes the Frobenius norm of D. Essentially, (7) quantifies the proportion of asymmetry in the similarity matrix D, with a decreasing value indicating reduced asymmetry in the matrix.

In contrast to SOSNet, which employs a KNN algorithm and constructs three matrices for computation with an algorithmic complexity of $O(N)$, our proposed symmetric regularizatio only requires the construction of a single matrix and has an algorithmic complexity of $O(1)$. The total loss function is expressed as:

$$L = \frac{1}{N}\sum_{i}^{N} max(s(a_i, p_i) - s(a_i, n_i) + m, 0) + \lambda SR. \tag{17}$$

4 Experiments

In this section, we compare our proposed descriptor learning method with several methods on two benchmark datasets including Brown dataset [30] and HPatches dataset [31].

4.1 Implementation details

To prevent overfitting, a dropout rate of 0.2 is applied. The PyTorch library is utilized to train our local descriptor network. SGD is chosen as the optimizer with an initial learning rate of 0.1, momentum of 0.9, and weight decay of 0.0001. The value of λ in the loss function is set to 0.15.

4.2 Experimental results and analysis

4.2.1 Brown phototour

For network training, we use the Brown Dataset [30], which is composed of local patches extracted from different scenes. Brown Dataset consists of three subsets: Yosemite, Notredame, and Liberty. Usually, we take one of the subsets as training set while the other two are used for testing. Each patch in the dataset has a unique 3D point indexes and patches with identical 3D point index are matching ones. What is more, for each 3D point, there are at least 2 matching patches. There are approximately 500K (1.5M) and 3D points (patches) in the Brown dataset. The original size of each patch is 64×64. In addition, we extract 1000K triplets of patches from the training set with a batch size of 384 for training. We follow the standard evaluation protocol of it by using the 100K pairs provided by the authors and report the false positive rate at 95% recall.

The impact of effective data-driven neural networks on improving results compared to traditional methods is unquestionable. The deep learning based approach is a major leap forward, especially the modifications to the network structure and regularization terms that improve the descriptor learning performance. As shown in Table 1, experimental results on the Brown dataset highlight the superiority of CNN-based approaches compared to SIFT, such as L2Net [10], HardNet [11], RAL-Net [13] and SOSNet [14] .

Thanks to our adaptive fusion and simplified computational but efficient SR regularization, our approach outperforms previous methods. Notably, our method employs novel three-branch center-surround multiscale learning, again outperforming similar methods such as CS L2Net [10].

Based on the experimental results, we present the following observations. It is evident that deep learning-based methods can significantly outperform traditional methods by using data-driven and efficient neural networks. As shown in Table 1, our approach outperforms other descriptor-learning methods on the Brown dataset. Our method outperforms

Table 1 Patch-verification performance on the UBC phototour dataset. False positive rates at 95% recall are reported

Train	Notredam	Yosemite	Liberty	Yosemite	Liberty	Notredam	Mean
Test	Liberty		Notredam		Yosemite		
SIFT [8]	29.84		22.53		27.29		26.55
TBLD [32]	20.4	21.95	14.47	16.53	36.88	35.09	18.25
BLCD [33]	10.07	11.90	4.90	5.26	9.02	10.03	8.53
HybridDesc [34]	2.70	3.63	0.81	1.00	3.17	2.67	2.33
L2Net [10]	2.36	4.70	0.72	1.29	2.57	1.71	2.23
CS L2Net [10]	1.71	3.87	0.56	1.09	2.03	1.30	1.76
HardNet [11]	1.49	2.51	0.53	0.79	1.96	1.84	1.51
RALNet [13]	1.30	2.39	0.37	0.67	1.52	1.31	1.26
SOSNet [14]	1.08	2.12	0.35	0.67	1.03	0.95	1.03
CDF [17]	1.21	2.01	0.39	0.68	1.51	1.29	1.18
CDF-STC [35]	1.16	1.86	0.36	0.61	1.18	1.09	1.04
HSD+ [15]	1.19	1.91	0.37	0.64	1.38	1.14	1.11
MR3A [36]	1.47	2.09	0.50	0.77	1.69	1.75	1.38
MFD-Net [37]	1.21	2.10	0.40	0.74	1.85	1.77	1.35
HyNet [29]	0.89	**1.37**	0.34	0.61	**0.88**	0.96	0.84
AFSRNet	**0.84**	1.40	**0.28**	**0.45**	0.94	**0.67**	**0.76**

unsupervised approaches such as TBLD [32], BLCD [33], and HybridDesc [34]. Notably, it consistently surpasses these methods and notably excels when compared to HybridDesc, acknowledged as the leading unsupervised feature descriptor learning approach.

4.2.2 HPatches

HPatches [31], a local descriptor evaluation benchmark, provides a huge dataset and evaluation criteria for modern descriptors. HPatches dataset consists of over 1.5 million patches extracted from 116 viewpoint and illumination changing scenes and different from the Brown dataset, it contains more diversity and noisy changes. According to the different levels of geometric noise, the extracted patches can be divided into three groups: easy, hard, and tough. There are three evaluation tasks of HPatches: patch verification, image matching, and patch retrieval. In the evaluation on the HPatches dataset, all learning methods shown in Fig. 3 are trained on *Liberty*, a subset of the Brown Dataset.

It is essential to note that all the methods presented in Fig. 3 were trained using the Liberty subset of the Brown dataset. This specific training dataset provides a consistent benchmark for assessing the performance of various methods. Our proposed approach stands out with substantial enhancements across multiple evaluation metrics, encompassing patch verification, image matching, and patch retrieval. The detailed analysis depicted in Fig. 3 illustrates the comparative performance of different methods in these key aspects. Notably, our method consistently outperforms existing approaches, underscoring its effectiveness in handling various challenges posed by patch-based tasks. This comprehensive evaluation on HPatches reaffirms the robustness and superiority of our approach in comparison to the other methods, thereby contributing to its credibility and relevance in real-world applications.

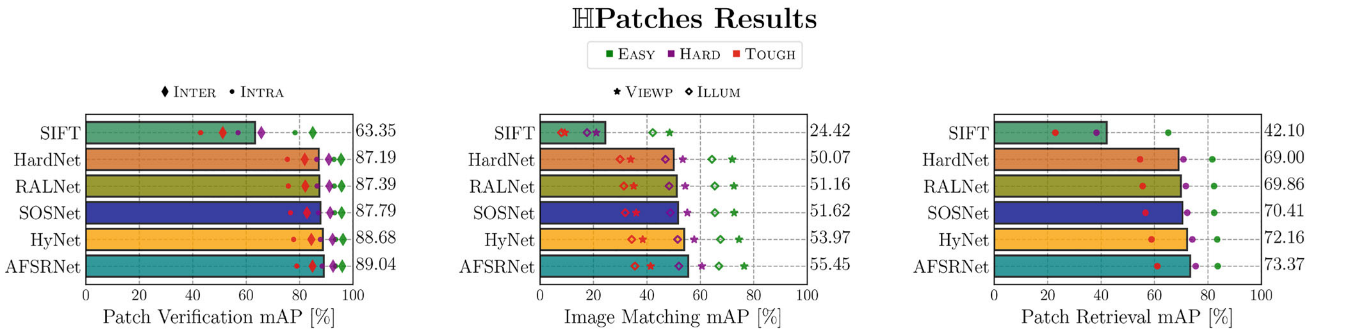


Fig. 3 Results of the patch-verification, patch-matching, and patch-retrieval tasks on HPatches [31]. Our proposed method outperforms all other methods in all three tasks

4.3 Ablation study

To validate the efficacy of our proposed method, we performed ablation experiments on various components, including the AMSF module, the SR, the number of branches, and the value of λ. These experiments were conducted on the Brown dataset.

1)*Impact of AMSF*: We evaluate the effect of AMSF on descriptor learning by comparing AMSF with single-scale learning methods, such as L2Net [10], and simple concatenation methods, such as CS L2Net [10]on the Brown dataset. Our AFSRNet comprises three sub-networks, so the dimension of the output produced by simply concatenating the descriptors from the three sub-networks is 3×128, i.e., 384. Therefore, for a fair comparison, all compared methods are trained such that the dimension of their descriptors is also 384. Table 2 shows that our proposed adaptive multi-scale fusion module is superior to concatenation.

In addition to the previously mentioned experimental results, we further conducted a detailed analysis by examining the Area Under the Curve (AUC) throughout the entire training process. The AUC curve, illustrated in Fig. 4-(a), serves as a dynamic metric to evaluate the performance evolution over different epochs. Remarkably, our model consistently exhibits superior performance when compared to the other two models, surpassing them notably around the tenth epoch. As shown in the Fig. 4-(a) , Concatenation also surpasses Single scale in about the 15th cycle, because one more scale information can make the feature descriptor have better matching performance, while AMSF surpasses Concatenation in about the 10th cycle, which shows that not only scale information is important, but also how to fuse them into one descriptor is also important, and our AMSF solves the problem of how to fuse them in a better way based on the utilization of multiscale information, and it can be seen that this superiority is embodied in the whole process of training.

This in-depth investigation not only provides a more comprehensive understanding of the superiority of the AMSF module but also emphasizes that this superiority is sustained throughout the entire training process. The AUC curve showcases the continuous and progressive excellence of our model, going beyond occasional instances of favorable experimental results. This extended analysis not only bolsters the evidence supporting the effectiveness of the AMSF module but also enhances the overall persuasiveness and credibility of our work.

2)*Impact of SR*: Moreover, we conducted experiments to evaluate the effect of symmetric regularization(SR) on the performance of our network. Our results show that networks trained with symmetric regularization outperform networks trained without symmetric regularization, as evidenced by the decrease in FPR95. Specifically, Table 3 shows that, for the same descriptor distance, using symmetric regularization results in a 30.1% reduction in FPR95.

3)*Impact of the number of branches*: To understand how the number of branches affects the final matching results, we conducted experiments comparing two-branch and four-branch models with our proposed three-branch model. The input patch size for a two-branch network is 64×64 and 32×32, while for a four-branch network, it is 64×64, 56×56, 48×48, and 32×32. The network configuration is the same, utilizing the AMSF module and SR. As shown in Table 4, our three-branch model performs better. Despite the computational intensity of the four-branch network, it doesn't outperform our approach in matching effectiveness. This highlights that a network's efficacy isn't solely determined by the number of branches or scales. Regarding the two-branch network, althoug it's performance is worse than four- and three-branch ones, it outperforms other two-branch networks like CS L2Net [10], showcasing the effectiveness of our adaptive fusion module.

4) *Impact of* λ: We also explored the effect of the hyperparameter λ on the performance of our method by varying its value from 0.05 to 0.4. The training was conducted on the *Liberty* subset, and the evaluation was performed on the remaining two subsets. As shown in Fig. 4-(b), we observed that the best performance of the learned descriptors was achieved when λ was set to 0.15. Therefore, we selected λ=0.15 as the weight value to appropriately balance the first-order and second-order loss functions in our approach.

4.4 Visualization result

In order to thoroughly assess the efficacy of our proposed method, we present a comprehensive performance validation based on visual results obtained from the HPatches dataset [31]. The initial phase of our experimentation involved the utilization of the SIFT algorithm [8] for both keypoint

Table 2 Comparison of AMSF, concatenation, and single-scale descriptors. AMSF stands for adaptive multi-scale fusion

Train Test	Notredam Liberty	Yosemite	Liberty Notredam	Yosemite	Liberty Yosemite	Notredam	Mean
Single Scale	1.19	2.03	0.31	0.67	1.07	0.93	1.04
Concatenation	1.28	1.58	0.34	0.50	1.00	0.69	0.88
AMSF	**0.84**	**1.40**	**0.28**	**0.45**	**0.94**	**0.67**	**0.76**

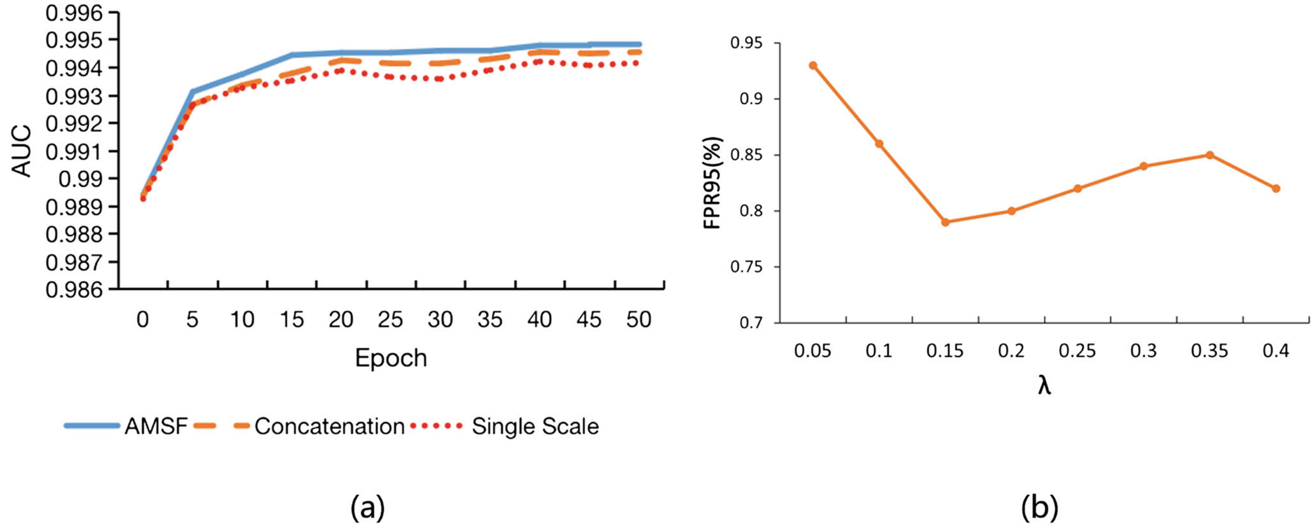


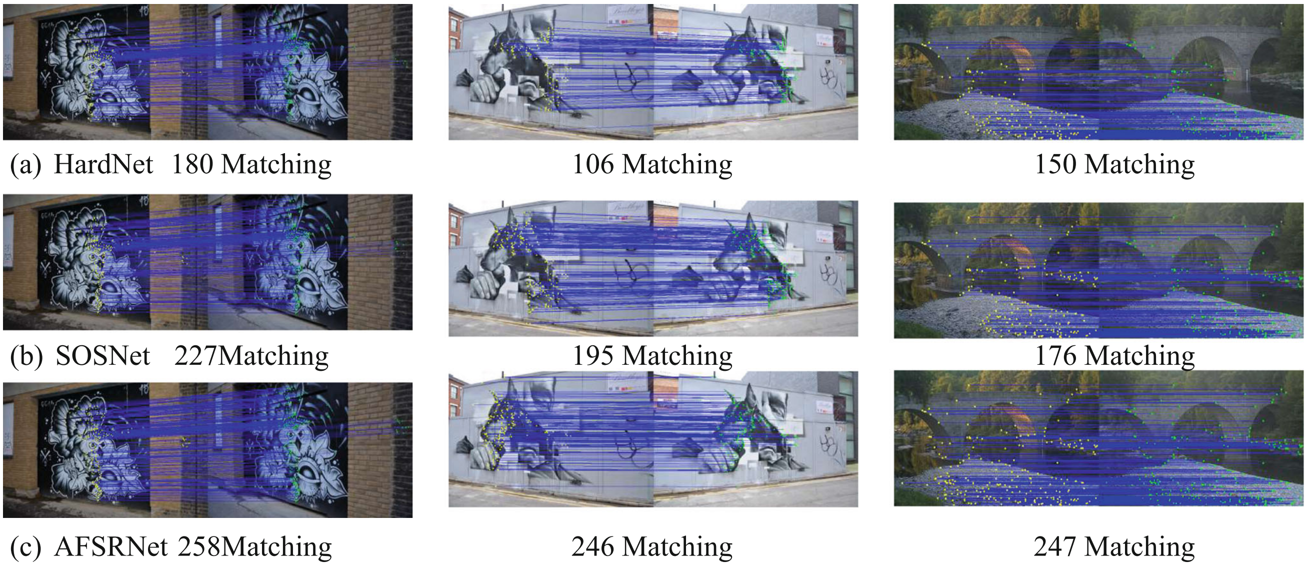
Fig. 4 Model analysis. AFSRNet is trained and tested on tested on Brown dataset.(a):Effect of AMSF. Area under the ROC curve (AUC) of different training epoch (train on *Liberty* and test on *Notredam*) is served as an indicator. (b):Effect of the λ. The curve shows the relation between FPR95 and λ

Table 3 Effect of SR, where "w/" and "w/o" mean with and without, respectively

Train Test	Notredam Liberty	Yosemite	Liberty Notredam	Yosemite	Liberty Yosemite	Notredam	Mean
w/o SR	1.05	1.92	0.38	0.59	1.40	0.75	1.02
w/ SOSR	0.89	1.63	0.29	0.51	**0.88**	0.73	0.81
w/ SR	**0.84**	**1.38**	**0.28**	**0.45**	0.94	**0.67**	**0.76**

Table 4 Comparison of different numbers of branches

Train Test	Notredam Liberty	Yosemite	Liberty Notredam	Yosemite	Liberty Yosemite	Notredam	Mean
Two Branches	1.21	1.51	0.41	0.46	1.03	0.86	0.91
Four Branches	0.86	1.42	0.29	**0.42**	0.94	0.74	0.78
Three Breanches	**0.84**	**1.40**	**0.28**	0.45	**0.94**	**0.67**	**0.76**



(a) HardNet 180 Matching 106 Matching 150 Matching

(b) SOSNet 227Matching 195 Matching 176 Matching

(c) AFSRNet 258Matching 246 Matching 247 Matching

Fig. 5 Visualization matching results on HPatches [31]. The correct matching pairs are indicated by blue lines

detection and descriptor extraction. Three distinct methods, namely HardNet [11], SOSNet [14], and our novel AFSRNet, were employed for the extraction of descriptors.

Following the descriptor extraction process, we applied a nearest neighbor distance ratio matching strategy with a specified threshold of 0.75, as illustrated in Fig. 5. This matching strategy, depicted in the figure, serves to establish correspondences between descriptors and plays a crucial role in evaluating the performance of our method.

As shown in Fig. 5, it serves as a valuable reference, visually highlighting the superior performance of our proposed method in terms of descriptor pair matching accuracy. Our results demonstrate that our AFSRNet consistently outperforms both HardNet and SOSNet, producing more accurately matched descriptor pairs. This superiority is particularly evident in the visual results obtained from . The visualization evaluation underscores the robustness of our approach, showcasing its ability to generate more precise and reliable descriptor matches when compared to the other two established methods.

5 Conclusion

In this paper, we introduce a novel feature extraction model based on adaptive multi-scale feature fusion (AMSF) and a new regularization term, called symmetric regularization (SR). Our proposed method achieves state-of-the-art performance on the Brown dataset, and also exhibits strong generalization ability on the HPatches dataset. Furthermore, we conducted a comprehensive ablation study to reveal the contribution of each proposed component to the final performance. Our results show that our proposed AMSF module and SR play critical roles in enhancing the performance of descriptor learning.

Acknowledgements This work was supported by the Guangdong Basic and Applied Basic Research Foundation (Grant No. 2021A1515011867).

Data Availability The datasets generated during and analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Conflicts of interest The authors declare that they have no conflicts of interest.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Xue J, Hou X, Zeng Y (2021) Review of image-based 3d reconstruction of building for automated construction progress monitoring. Appl Sci 11(17)
2. Ganesan K, Ganapathi II, Javed S et al (2023) Multimodal hybrid features in 3d ear recognition. Appl Intell 53(10):11,618-11,635
3. Cai Y, Li L, Wang D et al (2023) Htmatch: An efficient hybrid transformer based graph neural network for local feature matching. Signal Process 204(108):859
4. Di Y, Liao Y, Zhou H et al (2023) Femip: detector-free feature matching for multimodal images with policy gradient. Appl Intell 53(20):24068–24088
5. Zhu F, Zhu X, Huang Z et al (2021) Deep learning based data-adaptive descriptor for non-rigid multi-modal medical image registration. Signal Process 183(108):023
6. Ma J, Jiang X, Fan A et al (2021) Image matching from handcrafted to deep features: A survey. Int J Comput Vis 129(1):23–79
7. Jin Y, Mishkin D, Mishchuk A et al (2021) Image matching across wide baselines: From paper to practice. Int J Comput Vis 129(2):517–547
8. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. International journal of computer vision 60:91–110
9. Bay H, Ess A, Tuytelaars T et al (2008) Speeded-up robust features (surf). Comput. Vis. Image Underst 110(3):346–359
10. Tian Y, Fan B, 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 (CVPR)
11. Mishchuk A, Mishkin D, Radenovic F et al (2017) Working hard to know your neighbor's margins: Local descriptor learning loss. In: Guyon I, Luxburg UV, Bengio S et al (eds) Advances in Neural Information Processing Systems, vol 30. Curran Associates Inc
12. Hausler S, Garg S, Xu M, et al (2021) Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), pp 14,141–14,152
13. Xu Y, Gong M, Liu T et al (2019) Robust angular local descriptor learning. In: Jawahar C, Li H, Mori G et al (eds) Computer Vision - ACCV 2018. Springer International Publishing, Cham, pp 420–435
14. Tian Y, Yu X, Fan B, et al (2019) Sosnet: Second order similarity regularization for local descriptor learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)
15. Wang S, Guo X, Tie Y, et al (2021) Local feature descriptors with deep hypersphere learning. In: 2021 IEEE international conference on image processing (ICIP), pp 1524–1528
16. Zhang J, Jiao L, Ma W et al (2023) Rdlnet: A regularized descriptor learning network. IEEE Trans Neural Netw Learn Syst 34(9):5669–5681
17. Zhang L, Rusinkiewicz S (2019) Learning local descriptors with a cdf-based dynamic soft margin. In: Proceedings of the IEEE/CVF international conference on computer vision (ICCV)
18. Liang P, Ji H, Cheng E et al (2021) Learning local descriptors with multi-level feature aggregation and spatial context pyramid. Neurocomputing 461:99–108
19. Zhang P, Zhang C, Liu B et al (2022) Leveraging local and global descriptors in parallel to search correspondences for visual localization. Pattern Recognit 122(108):344
20. He Y, Hu Y, Zhao W, et al (2023) Darkfeat: noise-robust feature detector and descriptor for extremely low-light raw images. In: Proceedings of the AAAI conference on artificial intelligence, pp 826–834
21. Lin TY, Dollár P, Girshick R, et al (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125
22. Deng C, Wang M, Liu L et al (2022) Extended feature pyramid network for small object detection. IEEE Trans Multimed 24:1968–1979
23. Jiang K, Wang Z, Yi P, et al (2020) Multi-scale progressive fusion network for single image deraining. In: Proceedings of the

IEEE/CVF conference on computer vision and pattern recognition, pp 8346–8355

24. Wang G, Gan X, Cao Q et al (2023) Mfanet: multi-scale feature fusion network with attention mechanism. Vis Comput 39(7):2969–2980
25. He K, Zhang X, Ren S et al (2015) Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell 37(9):1904–1916
26. Chen LC, Papandreou G, Kokkinos I et al (2017) Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans Pattern Anal Mach Intell 40(4):834–848
27. Li Y, Chen Y, Wang N, et al (2019) Scale-aware trident networks for object detection. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 6054–6063
28. Balntas V, Riba E, Ponsa D, et al (2016) Learning local feature descriptors with triplets and shallow convolutional neural networks. In: BMVC, p 3
29. Tian Y, Barroso Laguna A, Ng T, et al (2020) Hynet: Learning local descriptor with hybrid similarity measure and triplet loss. In: Larochelle H, Ranzato M, Hadsell R, et al (eds) Advances in neural information processing systems, vol 33. Curran Associates, Inc., pp 7401–7412
30. Brown M, Hua G, Winder S (2011) Discriminative learning of local image descriptors. IEEE Trans Pattern Anal Mach Intell 33(1):43–57
31. Balntas V, Lenc K, Vedaldi A, et al (2017) Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)
32. Miao Y, Lin Z, Ma X et al (2021) Learning transformation-invariant local descriptors with low-coupling binary codes. IEEE Trans Image Process 30:7554–7566
33. Fan B, Liu H, Zeng H et al (2021) Deep unsupervised binary descriptor learning through locality consistency and self distinctiveness. IEEE Trans Multimed 23:2770–2781
34. Wang W, Zhang L, Huang H (2023) Revisiting unsupervised local descriptor learning. In: Proceedings of the AAAI conference on artificial intelligence, pp 2680–2688
35. Yin J, Liu Q, Meng F et al (2022) Stcdesc: Learning deep local descriptor using similar triangle constraint. Knowl Based Syst 248(108):799
36. Quan D, Wang S, Li Y et al (2021) Multi-relation attention network for image patch matching. IEEE Trans Image Process 30:7127–7142
37. Yu C, Liu Y, Li C et al (2022) Multibranch feature difference learning network for cross-spectral image patch matching. IEEE Trans Geosci Remote Sensing 60:1–15

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.