A Multi-feature Embedding Method for Robust Image Matching

Jinhong Yu and Kun Sun(✉)

Hubei Key Laboratory of Intelligent Geo-Information Processing, School of Computer Sciences, China University of Geosciences, Wuhan 430074, China
sunkun@cug.edu.cn

Abstract. In this paper, a feature point matching method that integrates both spatial structure and multiple descriptors is proposed. To be specific, given a set of detected keypoints on both images, multiple feature descriptors are extracted at each keypoint. Then, a subspace that simultaneously encodes both spatial structure and multi-feature similarity is computed. In this subspace, two points from different images will be close if their similarity measured by multiple features are high, and two points from the same image will be close if their distance in the original spatial domain is small. The above task is formulated as a Laplacian Embedding problem, which can be solved by eigen decomposition. Finally, vectors in the subspace are treated as new descriptors of the keypoints, and correspondences are established by searching mutual nearest neighbors. Extensive experiments show remarkable improvement in matching accuracy and downstream tasks such as homography and relative pose estimation by combining both structure information and multiple descriptors.

Keywords: Image matching · Multi-feature fusion · Spatial structure preserving · Subspace embedding

1 Introduction

Establishing sparse feature correspondences between images is a fundamental problem in many computer vision tasks, such as 3D information inferring [1,13,14,24], Structure-from-Motion [22,36,40], robot sensing [25] and image retrieval [3,15]. Given two groups of keypoints, the main steps of image matching are: **i)** computing a high dimensional feature descriptor for each keypoint and **ii)** establishing correspondences between them by for example, finding the nearest neighbor in the feature space. In the above pipeline, feature descriptor is a key factor to improve the final matching result.

In the past two decades, researchers in the community have proposed many excellent handcrafted descriptors [2,5,7,26], as well as modern learned descriptors [10,16,29,32,43,44]. Despite their great success, these methods have their own limitations. Firstly, they still suffer from mismatches in challenging situations such as wide baseline and small scene overlap. Secondly, as observed by

C. Wallraven et al. (Eds.): ACPR 2021, LNCS 13189, pp. 575–589, 2022.
https://doi.org/10.1007/978-3-031-02444-3_43

previous studies [18,19], the performance of different descriptors may vary a lot for the same image. A keypoint might be correctly matched by one descriptor but mismatched by another. This difference implies that using a single descriptor hardly applies to all the scenarios, but different descriptors are complementary and they can cooperate. At last, while feature similarity is given much attention, the useful spatial structure information [20,42,45] is overlooked in these methods. This makes the matching result sensitive to local ambiguity. How to integrate multiple features as well as spatial structure constraint still remains an open problem.

In this paper, an image matching method based on multi-feature embedding is proposed. Different form existing methods that use a single feature, it first extracts multiple feature descriptors at each keypoint. Then, a new representation that encodes both multi-feature similarity and keypoint structure is computed via subspace embedding, which is a widely used methodology [23]. There are two properties of this subspace. On the one hand, if the inter-image similarity between two points measured by multiple descriptors is high, they will be close to each other in the embedded subspace. On the other hand, if two points on the same image are spatially close to each other, the distance between them in the new subspace will also be small. As a result, the structure of each point set is preserved and similar points from different point sets are pulled closer. This task is formulated as a Laplacian Embedding problem, which can be solved via eigen decomposition. Vectors in the computed subspace are treated as the new descriptors for the keypoints. In this way, both multi-feature and spatial structure information are utilized by the proposed method.

To summarize, the proposed method distinguishes itself from existing methods in the following aspects. (1) It generates a novel descriptor for each keypoint by computing a subspace, which is equivalent to the Laplacian Embedding problem. (2) The method is a general framework which fuses multiple off-the-shelf descriptors instead of using only one of them. In this way, the embedded descriptor can adapt to more challenging scenarios. (3) The subspace also preserves the spatial structure of the kepoints, which makes the algorithm robust to local appearance ambiguity.

2 Related Work

2.1 Feature Description Methods

As the most fundamental part of image matching, the performance of feature descriptor is very important. The most famous manual descriptor is SIFT [26], which is obtained by statistical histogram of local image gradient direction of keypoints, and it has been widely used until today. After that, many different kinds of manual feature descriptors have been designed to adapt to different situations, such as faster speed [5,33], smaller memory [7,34], and more robustness [2].

In recent years, feature descriptors based on neural network have developed rapidly, and generally get better matching results than handcrafted descriptors.

Some methods [27,37,43,52] take image patches as input, and can directly calculate the feature vector representations of these patches. HardNet [29] is based on L2-Net [43] network structure, it proposed a triple-network, by introducing a margin and encouraging negative pair feature distance to be greater than the sum of positive pair distance and margin, forcing the network focuses on those negative samples which are most difficult to distinguish. SOSNet [44] achieves better results by using the first-order similarity loss(similar to triplet-loss) and introducing a second-order regularization term between positive matching pairs. Interestingly, one method [53] proposes a *soft margin* relative to *hard margin* in HardNet, it discusses that the traditional *hard margin* is not flexible enough, so this paper proposes a *dynamic soft margin* to overcome this problem.

Another kind of end-to-end methods use image as input to obtain more reliable matching results by calculating dense features. Aiming at a large number of multi-views geometry problems in computer vision, SuperPoint [10] proposes a self supervised training framework of keypoint detection and description, which outputs highly abstract features of the input image. Subsequent end-to-end methods will also compute dense feature representation. D2-Net [11] proposes a "detect-and-describe" method, which uses a single CNN for joint feature detection and description, so an image can only get a 3D tensor. The goal of R2D2 [32] is to learn repeatable and reliable keypoints and powerful descriptors, and its outputs are dense descriptors, reliability map and repeatability map.

The existing deep learning methods all need ground truth correspondences to train, and the acquisition of correspondences is costly in some cases. Therefore, CAPS [47] proposes a method that directly uses the relative camera pose between image pairs as the supervision, thus greatly reducing the training costs. However, dense features tend to occupy more memory and computation is time-consuming.

2.2 Feature Matching Methods

Some researchers try to improve the results of image matching from another perspective. The most basic feature matching relationship is usually obtained by finding the mutual nearest neighbor features in feature space. SIFT [26] proposes ratio test based on mutual nearest neighbor searching and greatly improves the matching accuracy. Some methods [12,28,38,39,41,42] use Gaussian mixture model for image matching, where each keypoint in the first image is treated as a Gaussian component, and the probability of each keypoint in the second image being assigned to each Gaussian component is modeled. Other methods [48] to treat the matching problem as a classification problem, in this case, the keypoints in one image can be regarded as cluster centers, while the keypoints in another image are the keypoints to be assigned. Some multi-image matching methods [17,56] can promote the matching accuracy of image pairs to some extent by establishing the cycle-consistency constraint between multiple images.

The feature matching correspondence can also be restored from the feature similarity matrix, which is very common in graph matching [50,55] and multi-graph matching [8,31,46,49]. A spectral method [21] proposes to find the correspondences from the feature similarity matrix, this spectral method is also

used in many subsequent graph matching methods. Besides feature similarity, some methods [20,35,42,45] also considers the spatial structure of keypoints in the same image, and the better matching results are obtained by combining feature and spacial information, but this approach only takes into account a single feature. Recently, a novel method, SuperGlue [35], uses neural network to find correspondences, which fully considers the relationship of cross-image keypoints and self-image keypoints, this is also reflected in this paper.

The above image matching methods can not solve the inherent problem of features, that is, a good correspondence basically depends on a good feature descriptor. As we can not guarantee that a certain feature can be widely used in all scenes, from another perspective, the method of fusing multiple different existing features in this paper is a good choice.

2.3 Feature Fusion Methods

There are also some matching methods from the perspective of multiple features fusion. Hu *et al.* proposed in [19] that the best feature can be selected for each keypoint in the homography space for matching, but each keypoint essentially uses a single descriptor information. Yu *et al.* proposed a multi-feature fusion matching method [51], but their fusion features are geometric, gray, color and texture features. LISRD [30] proposes a method to separate invariants from local descriptors. In its framework, it includes the structure of learning multiple local descriptors, which makes people think it is a multi-feature fusion method. In fact, LISRD does not fuse features.

The goal of this paper is to design a multi-feature fusion method, in which each feature has its own contribution. And for different keypoints, different features have different contributions. In this way, different features complement effectively, and image matching accuracy can also be improved.

3 The Proposed Method

Given two images I_1 and I_2, we detect two groups of keypoints $X_1 \in R^{m\times 2}$ and $Y_2 \in R^{n\times 2}$ on each image. For each keypoint, K kinds of descriptors are extracted, which are denoted as $P_1^k \in R^{m\times d_k}$ and $Q_2^k \in R^{n\times d_k}$. $k = 1, .., K$ is the k-th feature and d_k is the dimension of it.

Different from existing methods which use a single descriptor, we want to fuse multiple features and impose structural constraint at the same time. To this end, we compute a new representation $E_1 = \left\{e_1^1, e_2^1, ..., e_m^1\right\}^T \in R^{m\times c}$ and $E_2 = \left\{e_1^2, e_2^2, ..., e_n^2\right\}^T \in R^{n\times c}$ of the original keypoints by projecting all these keypoints information into a subspace. The superscript 1 or 2 indicates the first or the second image, and c is the dimension of the subspace feature. E_1 and E_2 can be computed by minimizing the following objective function [45]:

$$\min \sum_{l=1,2} \sum_{i,j} \left\|e_i^l - e_j^l\right\|^2 S_{l,ij} + \sum_{i,j} \left\|e_i^1 - e_j^2\right\|^2 U_{ij}. \tag{1}$$

The first term in Eq. (1) encodes intra-image spatial information, where $S_{l,ij}$ represents the spatial similarity between keypoints i and j in image l. $S_{1,ij}$ and $S_{2,ij}$ can be computed by the following kernel function $K_s(\cdot,\cdot)$:

$$S_{1,ij} = K_s(x_i, x_j) = e^{-\frac{(x_i - x_j)^2}{2\sigma^2}},\ x_i, x_j \in X_1, \tag{2a}$$

$$S_{2,ij} = K_s(y_i, y_j) = e^{-\frac{(y_i - y_j)^2}{2\sigma^2}},\ y_i, y_j \in Y_2. \tag{2b}$$

According to Eq. (2), if two points on the same image are spatially close to each other, the corresponding similarity in $S_{1,ij}$ would be large. To minimize Eq. (1), their distance in the subspace should be small.

The second term in Eq. (1) encodes inter-image feature information, in which U_{ij} is the feature similarity defined by multiple descriptors between x_i and y_j. U_{ij} can be computed from the following equation:

$$U_{ij} = \frac{1}{K}\sum_{k=1}^{K} U_{ij}^k, \tag{3}$$

where

$$U_{ij}^k = K_u\left(p_i^k, q_j^k\right) = e^{-\frac{\left(p_i^k - q_j^k\right)^2}{2\beta^2}},\ p_i^k \in P_1^k \textbf{ and } q_j^k \in Q_2^k \tag{4}$$

is a kernel function representing the feature similarity between x_i and y_j with the k-th descriptor. As we can see from Eq. (3) and Eq. (4), the feature information in Eq. (1) is jointly defined by multiple descriptors. If two points from different images are similar to each other, the correponding similarity in U_{ij} would be large. To minimize Eq. (1), their distance in the subspace should be small as well. As a result, the subspace defined by Eq. (1) has the following properties: similar points from different images measured by multiple descriptors are pulled closer and the relative structure of points from the same image is preserved.

The feature information and spatial information can be expressed in a compact matrix form, which is shown in Eq. (5).

$$A = \begin{bmatrix} S_1 & U \\ U^T & S_2 \end{bmatrix}. \tag{5}$$

Here A is a 2×2 block matrix. Its diagonal blocks $S_1 \in R^{m\times m}$ and $S_2 \in R^{n\times n}$ are the spatial information matrices computed from Eq. (2). Its off-diagonal block $U \in R^{m\times n}$ is the feature information matrix computed from Eq. (3). Denoting $E = \left[E_1^T, E_2^T\right]$ and applying some simple derivation, Eq. (1) can be rewritten in the following form:

$$\min tr(E^T A E), \tag{6}$$

which can be seen as the Laplacian Embedding problem [6]. The optimal embedding features E in Eq. (6) can be obtained by solving the following problem,

$$\min_{E^T D E = I} tr(E^T L E), \tag{7}$$

where $L = D - A$ is the Laplacian matrix of A, and D is a diagonal matrix whose non-zero elements are computed from $D_{ii} = \sum_j A_{ij}$. Equation (7) is a generalized eigenvector problem, whose solution is the eigenvectors corresponding to the c smallest non-zero eigenvalues.

After computing E from Eq. (7), we have a new c-dimensional representation for each keypoint in X_1 and Y_2. This new descriptor not only fuses multi-feature information, but also encodes spatial structure constraint. We then match the keypoints by searching for mutual nearest neighbors in the subspace.

4 Experiments

4.1 Evaluation Metrics

The experiments are performed on a machine equipped with Xeon E5-2620 2.1GHz, 64GB RAM and one GTX 1080Ti. Following SuperPoint [10], D2-Net [11], UCN [9] and CAPS [47], the proposed method is evaluated in terms of Mean Matching Accuracy (MMA) and several downstream tasks such as homography estimation accuracy and relative pose estimation accuracy.

Mean Matching Accuracy (MMA). For a certain keypoint, if the distance between its estimated matching position and the ground truth matching position is smaller than a threshold, this match would be deemed as correct. The Mean Matching Accuracy (MMA) is the ratio of correct correspondences in the whole dataset. Higher MMA is preferable.

Homography Estimation Accuracy: Homography is a 3×3 matrix which plays an important role in a variety of areas such as panorama generation and planar surface detection. It can be estimated from correspondences between two views. To be specific, we use the OpenCV function to estimate the homography matrix and compare it with the ground truth. Following SuperPoint [10], the *four-corner accuracy* is used to check whether the estimated homography is correct. That is, the four corners of an image are warped by the estimated homography and the ground truth homography, respectively. If the average distance error between them is less than a threshold ε, then the estimated homography is admitted to be correct.

Relative Pose Estimation Accuracy: Another application of image feature point matching is 3D reconstruction, which requires to estimate the relative pose between two cameras. The pose parameters, *i.e.* the rotation matrix $R \in R^{3\times3}$ and the translation vector $t \in R^{3\times1}$ can also be computed from correspondences. For rotation, we compute the angle error between the estimation and the ground truth. As for translation, we simply compute the directional error with the ground truth because its magnitude is determined up to an unknown scale factor. The estimation is deemed as correct if the error is below a threshold.

4.2 Datasets

Similar to CAPS [47], the experiments are carried out on two datasets: HPatches [4] and COLMAP [54].

Table 1. The MMA on the HPatches dataset. The pixel threshold is from 1 to 10. Best results are in bold.

Method	1	2	3	4	5	6	7	8	9	10
2-Hand	.177	.353	.410	.441	.464	.484	.501	.517	.531	.542
2-Depth	**.212**	.413	.478	.511	.535	.554	.569	.581	.592	.600
4-Descs	.197	.388	.452	.486	.511	.535	.554	.573	.589	.602
4-Depth	**.212**	**.416**	**.483**	**.518**	**.541**	**.561**	**.576**	**.588**	**.598**	**.607**
F-Only	.167	.329	.384	.411	.428	.440	.449	.456	.460	.465

Table 2. Average homography estimation accuracy on HPatches under different thresholds ε. Best results are in bold.

Method	$\varepsilon = 1$	$\varepsilon = 3$	$\varepsilon = 5$
2-Hand	0.303	0.497	0.595
2-Depth	0.322	0.541	0.654
4-Descs	0.311	0.534	0.663
4-Depth	**0.325**	**0.560**	**0.690**
F-Only	0.324	0.525	0.642

HPatches is used to evaluate MMA and homography estimation accuracy. It consists of 116 scenes, among which 57 scenes are for illumination change and the other 59 scenes are for viewpoint change. Each scene contains 6 images and 5 pairs by matching the first image to the others, leading to a total of 580 image pairs. For every image pair, a homography is provided as the ground truth. SuperPoint [10] is applied to detect at most 1000 keypoints on each image except for the *i_dc* scene, because SuperPoint is not able to handle its resolution.

COLMAP is used for the evaluation of relative pose estimation accuracy. It contains four scenes: *gerrard*, *graham*, *person* and *south*, with 100, 560, 330 and 128 images respectively. These images, which are captured by different users and collected from the Internet, present great challenges such as viewpoint changes, scaling and occlusion. The camera parameters estimated in a standard SfM pipeline are provided as ground truth. Similar to [47], we divide all the image pairs in this dataset into three groups according to the viewing angle difference: *easy* $[0, 15°]$, *moderate* $[15°, 30°]$ and *hard* $[30°, 60°]$. In each group, we randomly select 200 image pairs, resulting a total of 600 image pairs for testing. SuperPoint [10] is also applied to detect at most 1000 keypoints on each image.

The proposed method is compared with several state-of-the-art descriptors including SIFT [26], RootSIFT [2], HardNet [29], SOSNet [44], SoftMargin [53] and SuperPoint [10]. The first two are famous handcrafted descriptors while the last three are outstanding deep learned descriptors. Our method is also compared with the OS [45] matching algorithm, which is closely related to our method, but it considers only a single descriptor. To evaluate the performance of each descriptor itself, we do not apply ratio test and all the matches are established by simply finding mutual nearest neighbors.

Table 3. Average relative pose (*rotation*/*translation*) estimation accuracy on the COLMAP dataset. The angle threshold is strictly set to 5°. Best results are in bold.

Method	*Easy*	*Moderate*	*Hard*
2-Hand	0.550/0.455	0.270/0.170	0.085/0.050
2-Depth	**0.695**/0.600	0.410/0.325	0.225/**0.155**
4-Descs	0.605/0.520	0.390/0.245	0.195/0.105
4-Depth	0.690/**0.610**	**0.445/0.335**	**0.245**/0.135
F-Only	0.530/0.445	0.360/0.235	0.160/0.120

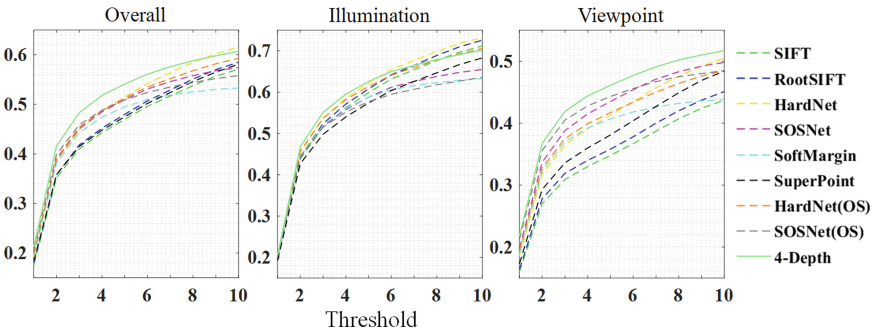

Fig. 1. The mean matching accuracy (MMA) for different thresholds on HPatches. From left to right are: results on the whole dataset, the illumination subset and the viewpoint subset.

4.3 Ablation Studies

Existing descriptors are either handcrafted or deep learned. Here we test 4 different combinations of them and analyze the results. 2-Hand uses two handcrafted descriptors SIFT and RootSIFT. 2-Depth uses two of the outstanding deep descriptors, HardNet and SOSNet. 4-Descs uses a mixture of both handcrafted and deep learned descriptors. Two of them are from 2-Hand and the others from 2-Depth. 4-Depth uses four deep learned descriptors, including HardNet, SOSNet, SoftMargin and SuperPoint.

The results on MMA, homography estimation accuracy and relative pose estimation accuracy are shown in Table 1, Table 2 and Table 3, respectively. As we can see from the data, when using the same number of descriptors (for example 2-Hand and 2-Depth), deep learned descriptors outperforms traditional handcrafted ones. It also reveals that using more descriptors will improve the results (see 2-Depth and 4-Depth). However, we also find that 4-Descs is lower than 4-Depth and 2-Depth. This indicates that not all the descriptors will contribute to the results. Some descriptors that are not good enough might even make the results worse. Based on the above observations, we recommend to use 4-Depth in the following experiments.

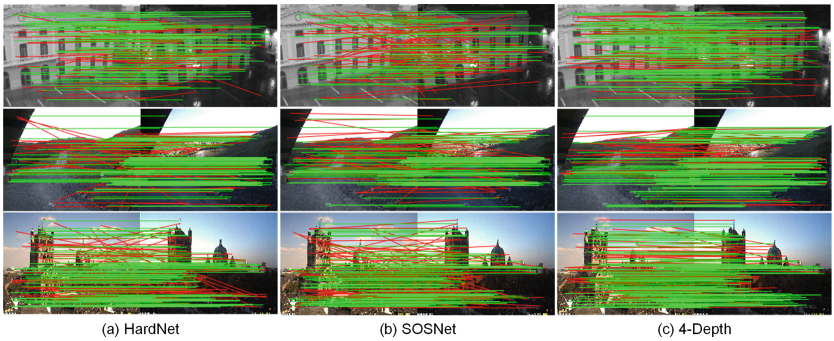

Fig. 2. Visualization results of the correspondences on three typical image pairs. *Green* and *red* lines indicate correct and incorrect matches, respectively. (Color figure online)

We also test the role of spatial structure information. According to [45], we replace the diagonal blocks of A in Eq. (5) with identity matrices for the method 4-Depth. In this case, spatial structure information is removed and only feature information is considered. The results, denoted as F-Only, is also shown in Table 1, Table 2 and Table 3. As we can see, F-Only is significantly lower than 4-Depth, showing that integrating spatial structure information is beneficial.

4.4 Mean Matching Accuracy Evaluation

Figure 1 shows the result of MMA under different thresholds (from 1 to 10). We plot the statistics on the whole dataset (Overall), as well as two subsets (Illumination and Viewpoint). HardNet (OS) and SOSNet (OS) represent the matching results of [45] when using HardNet and SOSNet, respectively.

The proposed method achieves the best performance on the whole dataset and the viewpoint subset. It also returns the best results on the illumination subset when the threshold is less than 6. HardNet and SOSNet are the top two compared methods. The two handcrafted descriptors, SIFT and RootSIFT, fall behind other learned descriptors on the viewpoint subset but receive good results on the illumination subset. Figure 2 gives some visualization results of the correspondences on some example image pairs. It shows that our method returns more correct and fewer incorrect matches.

4.5 Results on Downstream Tasks

Table 4 shows the average homography estimation accuracy on HPatches for different methods. Three thresholds are used. The proposed 4-Depth method achieves the best result for $\varepsilon = 3$ and $\varepsilon = 5$, and ranks second for $\varepsilon = 1$. HardNet(OS) and SOSNet(OS) outperform HardNet and SOSNet, respectively by involving spatial constraint. There is a remarkable improvement between our method and [45], showing that using multiple descriptors is beneficial.

Table 4. Average homography estimation accuracy on HPatches under different thresholds ε. A.M.P is the number of Average Matching Points and F.L.R is the Forecast Loss Rate. The best and second best results are in bold and blue.

Method	$\varepsilon = 1$	$\varepsilon = 3$	$\varepsilon = 5$	F.L.R	A.M.P
SIFT [26]	0.296	0.499	0.588	0	475.2
RootSIFT [2]	0.296	0.489	0.584	0	467.9
HardNet [29]	0.315	0.513	0.638	0	534.3
SOSNet [44]	0.322	0.529	0.650	0	525.7
SoftMargin [53]	0.308	0.508	0.637	0	526.7
SuperPoint [10]	0.290	0.470	0.595	0	504.5
HardNet(OS) [45]	0.322	0.523	0.626	0	609.5
SOSNet(OS) [45]	**0.329**	0.525	0.671	0	614.5
4-Depth	0.325	**0.560**	**0.690**	0	625.1
Un.	0.315	0.490	0.635	0	858.6
Vo.	0.325	0.567	0.664	0.016	287.7
In.	0.283	0.464	0.565	0.049	173.4

Table 5 shows the average relative pose estimation accuracy on the COLMAP dataset for different methods. The angle thresholds error is strictly set to 5°. For all these methods, the score drops from *easy* to *hard*. Our 4-Depth method achieves the best results except for translation on the *hard* subset. HardNet (OS) and SOSNet (OS) defeat HardNet and SOSNet, and rank the top two among the remaining compared methods.

To test some other simple feature fusing strategies, we use intersection, union and voting of four deep features in Table 4 and Table 5. They are denoted as **In.**, **Un.** and **Vo.**, respectively. For **Vo.**, a correspondence is required to be found by at least three out of four descriptors. As we can see, **Un.** contains too many false matches so its results are generally not as good as ours. **Vo.** shows much higher score in Table 5, but it's worth noting that the increase of accuracy is at the cost of sacrificing many correct matches. To prove this, we give the number of *Average Matching Points*(A.M.P) and the *Forecast Loss Rate* (F.L.R) in both tables. It shows that **Vo.** sacrifices nearly 50% and 82% matches in Table 4 and Table 5, while the statistics for **In.** is 71% and 93%. Losing too many correspondences may lead to failure when estimating the geometry models due to insufficient data. The Forecast Loss Rate of **Vo.** and **In.** can range from 1% up to 33%. As a result, although **Vo.** and **In.** can achieve higher accuracy in easy situations, they are infeasible in harder situations due to high failure rate.

4.6 Parameters and Efficiency

In our method, the dimension c of the subspace is an important parameter. To investigate its influence, an experiment is carried out on the **v_grace** scene of HPatches, in which c increase from 5 to 400 with a step size of 5. The average

Table 5. Average relative pose (*rotation*/*translation*) estimation accuracy. The angle error threshold is strictly set to 5°. A.M.P is the number of Average Matching Points. F.e, F.m and F.h are the Forecast Loss Rate for each subset. The best and second best results are in bold and blue.

Method	*Easy*	*Moderate*	*Hard*	*F.e*	*F.m*	*F.h*	A.M.P
SIFT [26]	.540/.395	.250/.135	.105/.050	0	0	0	349.4
RootSIFT [2]	.555/.410	.260/.155	.105/.050	0	0	0	338.3
HardNet [29]	.580/.500	.340/.215	.150/.115	0	0	0	445.1
SOSNet [44]	.565/.456	.350/.215	.160/.080	0	0	0	435.9
SoftMargin [53]	.580/.450	.350/.240	.150/.090	0	0	0	451.4
SuperPoint [10]	.565/.445	.245/.150	.125/.065	0	0	0	384.2
HardNet(OS) [45]	.615/.515	.385/.265	.190/**.140**	0	0	0	536.3
SOSNet(OS) [45]	.625/.535	.345/.255	.170/.130	0	0	0	538.6
4-Depth	**.690/.610**	**.445/.335**	**.245**/.135	0	0	0	534.2
Un.	.110/.020	.075/.000	.040/.010	0	0	0	950.2
Vo.	.750/.650	.520/.425	.285/.220	.015	.082	.097	87.9
In.	.700/.590	.330/.260	.185/.115	.100	.333	.335	36.2

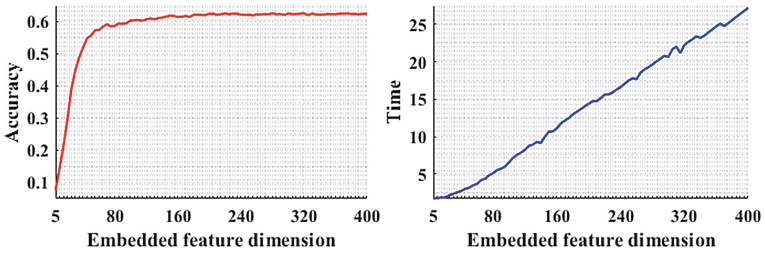

Fig. 3. The average matching accuracy and running time for different embedded feature dimension c. As a trade-off, we set the embedding feature dimension to $c = 55$ in all the experiments.

matching accuracy and running time are shown in Fig. 3. The results show that the matching accuracy of our method will increase when the embedded dimension becomes higher, but it will cost more time as well. In particular, the running time keeps growing but the average matching accuracy remains stable when the feature dimension c exceeds 60. As a trade-off, we set the embedding feature dimension to $c = 55$ in all the experiments.

5 Conclusions

This paper proposes a novel image matching method based on multi-feature fusion and subspace embedding. The basic idea is to compute a subspace, in which intra-image structures of the keypoints are preserved and inter-image

multi-feature similarities are encoded. This goal is achieved by solving a Laplacian Embedding problem. The proposed method is tested on a variety of scenes. Both the mean matching accuracy and performance on downstream tasks such as homography estimation and relative pose estimation are evaluated. Results show that the proposed method achieves the best performance when combining four deep descriptors: HardNet, SOSNet, SoftMargin and SuperPoint.

Acknowledgment. This work is supported by National Natural Science Foundation of China (62176242, 61802356), also in part by NSFC (41925007, 62076228) and Open Research Project of The Hubei Key Laboratory of Intelligent Geo-Information Processing (KLIGIP-2019B03).

References

1. Albarelli, A., Rodolà, E., Torsello, A.: Imposing semi-local geometric constraints for accurate correspondences selection in structure from motion: a game-theoretic perspective. Int. J. Comput. Vis. **97**(1), 36–53 (2012)
2. Arandjelovic, R., Zisserman, A.: Three things everyone should know to improve object retrieval. In: CVPR, pp. 2911–2918. IEEE (2012)
3. Babenko, A., Slesarev, A., Chigorin, A., Lempitsky, V.: Neural codes for image retrieval. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 584–599. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_38
4. Balntas, V., Lenc, K., Vedaldi, A., Mikolajczyk, K.: HPatches: a benchmark and evaluation of handcrafted and learned local descriptors. In: CVPR, pp. 3852–3861. IEEE (2017)
5. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_32
6. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput. **15**(6), 1373–1396 (2003)
7. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: BRIEF: binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_56
8. Chen, Z., Xie, Z., Yan, J., Zheng, Y., Yang, X.: Layered neighborhood expansion for incremental multiple graph matching. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12355, pp. 251–267. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58607-2_15
9. Choy, C.B., Gwak, J., Savarese, S., Chandraker, M.K.: Universal correspondence network. In: NIPS, pp. 2406–2414. Curran Associates, Inc. (2016)
10. DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperPoint: self-supervised interest point detection and description. In: CVPR, pp. 224–236. IEEE (2018)
11. Dusmanu, M., et al.: D2-Net: a trainable CNN for joint description and detection of local features. In: CVPR, pp. 8092–8101. IEEE (2019)
12. Fang, L., Sun, Z., Lam, K.: An effective membership probability representation for point set registration. IEEE Access **8**, 9347–9357 (2020)

13. Forster, C., Pizzoli, M., Scaramuzza, D.: Appearance-based active, monocular, dense reconstruction for micro aerial vehicles. In: Robotics: Science and Systems X (2014)
14. Gao, X., Luo, J., Li, K., Xie, Z.: Hierarchical RANSAC-based rotation averaging. IEEE Signal Process. Lett. **27**, 1874–1878 (2020)
15. Gordo, A., Almazán, J., Revaud, J., Larlus, D.: Deep image retrieval: learning global representations for image search. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 241–257. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_15
16. Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A.C.: MatchNet: unifying feature and metric learning for patch-based matching. In: CVPR, pp. 3279–3286. IEEE (2015)
17. Havlena, M., Schindler, K.: VocMatch: efficient multiview correspondence for structure from motion. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8691, pp. 46–60. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10578-9_4
18. Hu, Y., Lin, Y.: Progressive feature matching with alternate descriptor selection and correspondence enrichment. In: CVPR, pp. 346–354. IEEE (2016)
19. Hu, Y., Lin, Y., Chen, H., Hsu, K., Chen, B.: Matching images with multiple descriptors: an unsupervised approach for locally adaptive descriptor selection. IEEE Trans. Image Process. **24**(12), 5995–6010 (2015)
20. Jiang, X., Ma, J., Jiang, J., Guo, X.: Robust feature matching using spatial clustering with heavy outliers. IEEE Trans. Image Process. **29**, 736–746 (2020)
21. Leordeanu, M., Hebert, M.: A spectral technique for correspondence problems using pairwise constraints. In: International Conference on Computer Vision, pp. 1482–1489. IEEE (2005)
22. Li, Z., Snavely, N.: MegaDepth: learning single-view depth prediction from internet photos. In: CVPR, pp. 2041–2050. IEEE (2018)
23. Li, Z., Liu, H., Zhang, Z., Liu, T., Xiong, N.N.: Learning knowledge graph embedding with heterogeneous relation attention networks. IEEE Trans. Neural Netw. Learn. Syst., 1–13 (2021). https://doi.org/10.1109/TNNLS.2021.3055147
24. Liu, H., Fang, S., Zhang, Z., Li, D., Lin, K., Wang, J.: MFDNet: collaborative poses perception and matrix fisher distribution for head pose estimation. IEEE Trans. Multimedia (2021). https://doi.org/10.1109/TMM.2021.3081873
25. Liu, T., Liu, H., Li, Y., Chen, Z., Zhang, Z., Liu, S.: Flexible FTIR spectral imaging enhancement for industrial robot infrared vision sensing. IEEE Trans. Industr. Inform. **16**(1), 544–554 (2020)
26. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004). https://doi.org/10.1023/B:VISI.0000029664.99615.94
27. Luo, Z., et al.: GeoDesc: learning local descriptors by integrating geometry constraints. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11213, pp. 170–185. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01240-3_11
28. Ma, J., Jiang, X., Jiang, J., Gao, Y.: Feature-guided gaussian mixture model for image matching. Pattern Recognit. **92**, 231–245 (2019)
29. Mishchuk, A., Mishkin, D., Radenovic, F., Matas, J.: Working hard to know your neighbor's margins: local descriptor learning loss. In: NIPS, pp. 4826–4837. Curran Associates, Inc. (2017)

30. Pautrat, R., Larsson, V., Oswald, M.R., Pollefeys, M.: Online invariance selection for local feature descriptors. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12347, pp. 707–724. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58536-5_42
31. Phillips, S., Daniilidis, K.: All graphs lead to Rome: learning geometric and cycle-consistent representations with graph convolutional networks. CoRR arXiv:1901.02078 (2019)
32. Revaud, J., de Souza, C.R., Humenberger, M., Weinzaepfel, P.: R2D2: reliable and repeatable detector and descriptor. In: NIPS, pp. 12405–12415. Curran Associates, Inc. (2019)
33. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3951, pp. 430–443. Springer, Heidelberg (2006). https://doi.org/10.1007/11744023_34
34. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: ORB: an efficient alternative to SIFT or SURF. In: ICCV, pp. 2564–2571. IEEE (2011)
35. Sarlin, P., DeTone, D., Malisiewicz, T., Rabinovich, A.: SuperGlue: learning feature matching with graph neural networks. In: CVPR, pp. 4937–4946. IEEE (2020)
36. Schönberger, J.L., Frahm, J.: Structure-from-motion revisited. In: CVPR, pp. 4104–4113. IEEE (2016)
37. Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P., Moreno-Noguer, F.: Discriminative learning of deep convolutional feature point descriptors. In: International Conference on Computer Vision, pp. 118–126. IEEE (2015)
38. Sun, J., Sun, Z., Lam, K., Zeng, Z.: A robust point set registration approach with multiple effective constraints. IEEE Trans. Ind. Electron. **67**(12), 10931–10941 (2020)
39. Sun, K., Tao, W., Qian, Y.: Guide to match: multi-layer feature matching with a hybrid Gaussian mixture model. IEEE Trans. Multim. **22**(9), 2246–2261 (2020)
40. Taira, H., et al.: InLoc: indoor visual localization with dense matching and view synthesis. In: CVPR, pp. 7199–7209. IEEE (2018)
41. Tao, W., Sun, K.: Asymmetrical gauss mixture models for point sets matching. In: CVPR, pp. 1598–1605. IEEE (2014)
42. Tao, W., Sun, K.: Robust point sets matching by fusing feature and spatial information using nonuniform gaussian mixture models. IEEE Trans. Image Process. **24**(11), 3754–3767 (2015)
43. Tian, Y., Fan, B., Wu, F.: L2-Net: deep learning of discriminative patch descriptor in Euclidean space. In: CVPR, pp. 6128–6136. IEEE (2017)
44. Tian, Y., Yu, X., Fan, B., Wu, F., Heijnen, H., Balntas, V.: SOSNet: second order similarity regularization for local descriptor learning. In: CVPR, pp. 11016–11025. IEEE (2019)
45. Torki, M., Elgammal, A.M.: One-shot multi-set non-rigid feature-spatial matching. In: CVPR, pp. 3058–3065. IEEE (2010)
46. Wang, Q., Zhou, X., Daniilidis, K.: Multi-image semantic matching by mining consistent features. In: CVPR, pp. 685–694. IEEE (2018)
47. Wang, Q., Zhou, X., Hariharan, B., Snavely, N.: Learning feature descriptors using camera pose supervision. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12346, pp. 757–774. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58452-8_44
48. Wang, Y., Mei, X., Ma, Y., Huang, J., Fan, F., Ma, J.: Learning to find reliable correspondences with local neighborhood consensus. Neurocomputing **406**, 150–158 (2020)

49. Yu, T., Yan, J., Liu, W., Li, B.: Incremental multi-graph matching via diversity and randomness based graph clustering. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11217, pp. 142–158. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01261-8_9
50. Yu, T., Yan, J., Wang, Y., Liu, W., Li, B.: Generalizing graph matching beyond quadratic assignment model. In: NIPS, pp. 861–871. Curran Associates, Inc. (2018)
51. Yu, X., Guo, Y., Li, J., Cai, F.: An image patch matching method based on multi-feature fusion. In: 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, 2017, pp. 1–6. IEEE (2017)
52. Zagoruyko, S., Komodakis, N.: Learning to compare image patches via convolutional neural networks. In: CVPR, pp. 4353–4361. IEEE (2015)
53. Zhang, L., Rusinkiewicz, S.: Learning local descriptors with a CDF-based dynamic soft margin. In: International Conference on Computer Vision, pp. 2969–2978. IEEE (2019)
54. Zhao, C., Cao, Z., Li, C., Li, X., Yang, J.: NM-Net: mining reliable neighbors for robust feature correspondences. In: CVPR, pp. 215–224. IEEE (2019)
55. Zhou, F., la Torre, F.D.: Factorized graph matching. IEEE Trans. Pattern Anal. Mach. Intell. **38**(9), 1774–1789 (2016)
56. Zhou, X., Zhu, M., Daniilidis, K.: Multi-image matching via fast alternating minimization. In: International Conference on Computer Vision, pp. 4032–4040. IEEE (2015)