DRFD-Net: Using Dual Receptive Field Descriptors for Multitemporal Optical Remote Sensing Image Registration

Yanan You, *Member, IEEE*, Chao Li, and Wenli Zhou

***Abstract*— Multitemporal optical remote sensing image registration is still a challenging problem for current feature-based image registration algorithms due to the complex nonlinear discrepancies arising from diverse factors, including illumination, weather, and surface condition changes. To address the issue, this article attempts to combine the dual receptive field descriptors (DRFDs) constructed by a novel deep convolutional network. In addition, a novel inner loss function (ILF) that imposes constraints on the intermediate descriptors is adopted in order to consolidate the distinguishability of the descriptors when the overlapping areas of the input image patches are large. Subsequently, the dual feature distance maps (DFDMs) are built on the basis of the DRFDs and combined with features from accelerated segment test (FAST) key points for efficient and accurate correspondence establishment across the source image and the target image. Eventually, an iterative algorithm is proposed to remove the possible outliers. Experiments show that the combination of DRFDs trained with the ILF performs better than current learnable local descriptors, such as L2-Net, HardNet, and SOSNet. The image registration results using our method are more accurate than the methods based on learnable descriptors, such as L2-Net, HardNet, and SOSNet, and handcrafted descriptors, such as scale-invariant feature transform (SIFT), SURF, and ORB.**

***Index Terms*— Dual feature distance maps (DFDMs), dual receptive field description network (DRFD-Net), features from accelerated segment test (FAST), image registration, inner loss function (ILF), scale-invariant feature transform (SIFT).**

I. Introduction

IMAGE registration refers to the process of aligning a set of images with certain spatial relationships, including scaling, rotation, translation, and deformable transformation. The images to be registered might be acquired from different viewpoints, at different times, or by different sensors. Remote sensing image registration is a significant topic in the research of remote sensing data application, which is an indispensable premise for many other tasks, such as change detection [1] and image fusion [2].

The categories of optical remote sensing images are diverse. Image registration based on spectral unmixing is quite significant for hyperspectral optical remote sensing images, in which spectral unmixing is a preprocessing operation to address spectral variability. The newest achievements in the subject of spectral unmixing include the NLTV/NLHTV and NMF in [3], the ALMM in [4], the ones introduced in [5], and so on. However, image registration for the three-channel visible remote sensing images, the main concern in our work, can be achieved by directly extracting image features without considering the mixed pixels.

Up until now, numerous studies have been proposed to solve the issue of image registration, which can be roughly divided into two categories, intensity-based methods and feature-based methods. The intensity-based registration methods attempt to find the optimal transformation that maximizes the similarity between the target image and the source image transformed. The similarity is measured via the intensity information, such as mutual information (MI) [6], cross correlation (CC) [7], phase correlation [8], [9], or sequential similarity detection algorithm (SSDA) [10]. Contrary to the intensity-based methods, the feature-based methods first extract the prominent features, such as points, lines, edges, and contours, and try to match the similar features on the target image and the source image. Generally, most feature-based methods leverage the point features, including scale-invariant feature transform (SIFT) [11], SURF [12], ORB [13], and KAZE [14]. Moreover, in order to make full use of the advantages of both the intensity-based methods and feature-based methods, there are also studies that combine them for image registration, including [15] and [16].

In recent years, with the success of deep learning methods in remote sensing image classification [17], [18], object detection [19], and so on, there have also been studies trying to directly utilize the convolutional neural network (CNN) [20] in transformation parameter regression. Both Miao *et al.* [21] and Park *et al.* [22] adopt CNN to predict the transformation matrix for image registration.

Comparing the methods above, the intensity-based methods are sensitive to intensity changes and computationally complicated, whereas the end-to-end deep learning methods are neither easy to train nor accurate enough. Therefore, feature-based methods are utilized more frequently in remote sensing image registration. Nevertheless, for multitemporal optical remote sensing image registration, due to the fact that the images are obtained asynchronously, there exist remarkable nonlinear discrepancies caused by the illumination, weather, and surface condition changes. As shown in Fig. 1, if conventional handcrafted descriptors, such as SIFT, are directly

Manuscript received April 12, 2021; revised June 19, 2021 and July 16, 2021; accepted August 9, 2021. Date of publication August 23, 2021; date of current version January 31, 2022. This work was supported by Beijing Natural Science Foundation, China, under Grant 4214058. *(Corresponding author: Chao Li.)*

The authors are with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: youyanan@bupt.edu.cn; chaoli1998@163.com; zwl@bupt.edu.cn).

Digital Object Identifier 10.1109/TGRS.2021.3104844

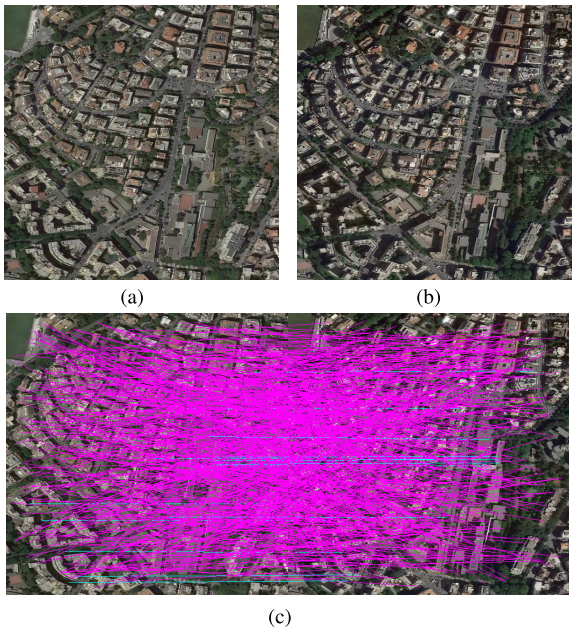

Fig. 1. Key point matching result of SIFT descriptors via nearest neighbor algorithm. The magenta line segments refer to the inliers while the cyan ones refer to the outliers. (a) Source image. (b) Target image. (c) Key point matching using SIFT.

adopted, there can be plenty of outliers with few inliers produced when matching these descriptors. To tackle the barrier, there are two solutions from two perspectives: one is to improve the descriptor construction process and the other is to ameliorate the matching algorithm.

The methods aiming to refine the descriptor matching algorithm attempt to adopt the spatial consistency among the inliers, including LLT [23], LPM [24], GLPM [25], ISIFT [26], and LAF [27]. However, these algorithms still depend on the performance of the descriptors adopted to a large extent, especially when there are fewer key points and inliers in complicated multitemporal optical remote sensing image registration problems. As for the methods intended to improve the descriptor structures, there are studies adopting CNN for descriptor construction instead for the reason that it is difficult for traditional handcrafted descriptors to cope with massive nonlinear changes. These methods include MatchNet [28], PN-Net [29], L2-Net [30], HardNet [31], SOSNet [32], and so on. These deep learning descriptors may perform well on datasets, such as UBC Phototour [33], HPatches [34], and ETH SfM [35]. Nevertheless, when the two key points are close to each other, the intersection of union (IoU) between the two local patches centered on them will be correspondingly large, making it hard for CNN to discriminate. Therefore, it is necessary to improve the current local descriptors based on CNN in order to register the multitemporal optical remote sensing images.

Considering the defect of the current deep convolutional features mentioned above, we try to adopt the intermediate features to enhance the ability to discriminate the local patches with large IoUs in this article. Therefore, we propose the structure of the dual receptive field description network (DRFD-Net) for the construction of the dual receptive field descriptors (DRFDs). During the training process, the inner loss function (ILF) is utilized to impose constraints on the intermediate descriptors, which further promotes distinguishability. In order to reduce the computational redundancy and time cost for building descriptor patch by patch for each key point, dual feature distance maps (DFDMs) are constructed based on the DRFDs and combined with features from accelerated segment test (FAST) [36] key points for efficient and robust correspondence establishment. For the final image registration process, we mainly focus on the transformation of translation, rotation, and scaling, and present an iterative algorithm to remove the possible outliers for accurate image registration results, without considering the complicated misregistration caused by the distortions. The contributions of this article are given in the following.

1) The DRFDs are constructed with the DRFD-Net and combined for better distinguishability.
2) The ILF is introduced to impose constraints on intermediate descriptors, which further improves DRFD-Net's ability to discriminate patches with large IoUs.
3) The DFDMs are built on the basis of DRFDs and then combined with FAST key points for correspondence establishment in a more efficient manner.
4) An iterative algorithm is developed to gradually remove the possible outliers among the correspondences established, which makes the regression of transformation parameters more accurate.

The remainder of this article is organized as follows. Section II concentrates on the related works of descriptor construction methods. Section III introduces the details of our method. Section IV shows the experiments of this article. Eventually, Section V draws a conclusion about this article.

II. Related Works

Local feature description is a basic issue in the field of computer vision, and it has been widespread used in many tasks, such as object detection [37], [38], image retrieval [39], 3-D reconstruction [40], and image registration [41]. For the issue of image registration, local descriptor construction is one of the key procedures, which will directly determine the image registration results. Compared to the local descriptors applied in object detection and image retrieval, there is a stricter requirement in distinguishability for local descriptors applied in image registration in order to ensure the accuracy of the image registration results. Generally, invariance and distinguishability are a contradiction. In other words, when the invariance is strengthened, the distinguishability will be correspondingly weakened. For the multitemporal remote sensing image registration task, we need both better invariance and distinguishability, which is a challenging problem. Therefore, we will concentrate on the related works about descriptor construction in this section.

A. Handcrafted Descriptors

Handcrafted descriptors are usually based on expert knowledge, whose construction procedure consists of two steps. The first step is to extract low-level information in the neighbor

area of the key point, including the intensity and the gradient. Then, the descriptors are built using the pooling or normalizing operations. Generally, the local descriptors need to be scale-invariant and rotation-invariant to be applied in the image registration process.

Typical descriptor construction methods include HOG [42], SIFT, SURF, ORB, KAZE, and so on. HOG descriptors are built on the basis of the gradient histograms, but the algorithm does not rotate the gradient histograms; therefore, HOG descriptors are not rotation-invariant. SIFT first adopts the difference of Gaussian [43] to calculate the position and scale information for each key point and then applies the gradient histogram to calculate its orientation. Finally, SIFT descriptors are built based on the position, scale, and orientation information of the key points. SURF ameliorates SIFT by applying the Hessian matrix's determinant to locate key points and using the Haar wavelet to calculate descriptors, which is more computationally efficient. ORB first uses FAST to detect key points and then uses BRIEF [44] to construct descriptors. Due to the fact that the BRIEF descriptors are not rotation-invariant, ORB takes the direction from the key point itself to the intensity centroid of the image patch centered on it as its orientation. It will then rotate the image patch centered on the key point according to the orientation before the descriptors are calculated, which ensures that the descriptors are rotation-invariant. Different from SIFT and SURF, KAZE applies the additive operator splitting (AOS) for anisotropic diffusion filtering, which is then utilized to build the scale-space pyramids. In addition, KAZE overcomes the defects of Gaussian filtering so that its descriptors are better in rotation-invariance and scale-invariance, but the algorithm is more computationally costly.

However, applying these handcrafted features in complicated multitemporal optical remote sensing image registration will produce massive outliers with few inliers due to the striking nonlinear discrepancies in even correspondent local areas. It is difficult for these algorithms to extract effective consistent information.

B. Learnable Descriptors

1) Overview: In addition to traditional handcrafted algorithms, there are studies utilizing CNN for the construction of local descriptors, such as MatchNet, PN-Net, L2-Net, HardNet, and SOSNet. MatchNet consists of the feature network and the metric network. The feature network is built on the basis of AlexNet [45] and is of a two-tower structure with shared parameters. The metric network uses the fully connected layers that output the similarity between the two input image patches. PN-Net proposes the SoftPN loss function to train the model with two positive image patches and one negative image patch as the input sample. L2-Net adopts the progressive sampling strategy to solve the issue of an extreme unbalanced distribution of positive and negative samples; besides, its loss function is defined with the perspectives of descriptor similarity, descriptor compactness, and intermediate feature maps considered. HardNet takes the same CNN architecture as L2-Net, but it adopts the triplet loss function, which emphasizes the ability of the network to discriminate the positive image patch with the hardest negative image patch. SOSNet develops the second-order similarity (SOS) regularization on the basis of the first-order similarity (FOS) for training, which also adopts the structure of L2-Net.

Nevertheless, the learnable descriptors perform worse when the distribution of key points on the source and target images is denser. With the correctly matched point and incorrectly matched ones getting closer, the convolutional features tend to be ambiguous and the distinguishability declines. We will take HardNet as an example to illustrate the issue in the following.

2) HardNet Loss: The basic idea of HardNet is batch hardest negative sample mining. During the training process, there are two parallel branches that share the same weights. Suppose that the input batch of HardNet is a set of tuples denoted as $X = \{(\boldsymbol{A}_i, \boldsymbol{P}_i)\}, \quad i = 1, 2, \ldots, n$, where $\boldsymbol{A}_i$ and $\boldsymbol{P}_i$ refer to the ith anchor image patch and the ith positive image patch, respectively, which are the input image patches of the two branches. n is the batch size. Here, we take $\boldsymbol{A}_i$ and $\boldsymbol{P}_i$ as two image patches of the same geographic area between which there exist certain transformation of rotation and scaling. There is no explicit negative sample during the HardNet training process. We denote the output feature vectors of the two data streams as a set of tuples $Y = \{(\boldsymbol{a}_i, \boldsymbol{p}_i)\}$ and subsequently define the distance matrix

$$\boldsymbol{D}_{i,j} = \sqrt{2 - \frac{\boldsymbol{a}_i^T \boldsymbol{p}_j}{\left\|\boldsymbol{a}_i^T\right\|_2 \left\|\boldsymbol{p}_j\right\|_2}}, \quad i, j = 1, 2, \ldots, n. \tag{1}$$

With the distance matrix $\boldsymbol{D}_{n\times n}$, we define the distance of the positive correspondence and the batch hardest negative correspondence, respectively

$$\begin{aligned} d_k^{(\mathrm{pos})} &= \boldsymbol{D}_{k,k} \\ d_k^{(\mathrm{neg})} &= \min\left(\min_{i\neq k} \boldsymbol{D}_{i,k}, \min_{j\neq k} \boldsymbol{D}_{k,j}\right) \end{aligned} \tag{2}$$

where $k = 1, 2, \ldots, n$. The loss function of HardNet can be regarded as a mapping of the set Y, and here, we write the loss function as

$$\mathscr{L}_{\mathrm{hn}}(Y) = \frac{1}{n}\sum_{k=1}^{n} \max\left(0, 1 + d_k^{(\mathrm{pos})} - d_k^{(\mathrm{neg})}\right). \tag{3}$$

3) Issue: Denote the target image and the source image as $\boldsymbol{I}^{(\mathrm{tar})}$ and $\boldsymbol{I}^{(\mathrm{src})}$, respectively, and the transformation matrix from the source image to the target image is denoted as $\boldsymbol{T}$; then, we have

$$\boldsymbol{I}^{(\mathrm{tar})}(\boldsymbol{T}\boldsymbol{x}) = g(\boldsymbol{I}^{(\mathrm{src})}(\boldsymbol{x})) \tag{4}$$

where $\boldsymbol{x} = (x, y)^T$ refers to the coordinate of certain pixel on the source image and g represents the intensity mapping function. Denote the key point set on the source image as

$$S^{(\mathrm{src})} = \left\{\boldsymbol{x}_1^{(\mathrm{src})}, \ldots, \boldsymbol{x}_n^{(\mathrm{src})}\right\}. \tag{5}$$

The corresponding key point set on the target image is

$$\begin{aligned} S^{(\mathrm{tar})} &= \left\{\boldsymbol{x}_1^{(\mathrm{tar})}, \ldots, \boldsymbol{x}_n^{(\mathrm{tar})}\right\} \\ &= \left\{\boldsymbol{T}\boldsymbol{x}_1^{(\mathrm{src})}, \ldots, \boldsymbol{T}\boldsymbol{x}_n^{(\mathrm{src})}\right\}. \end{aligned} \tag{6}$$

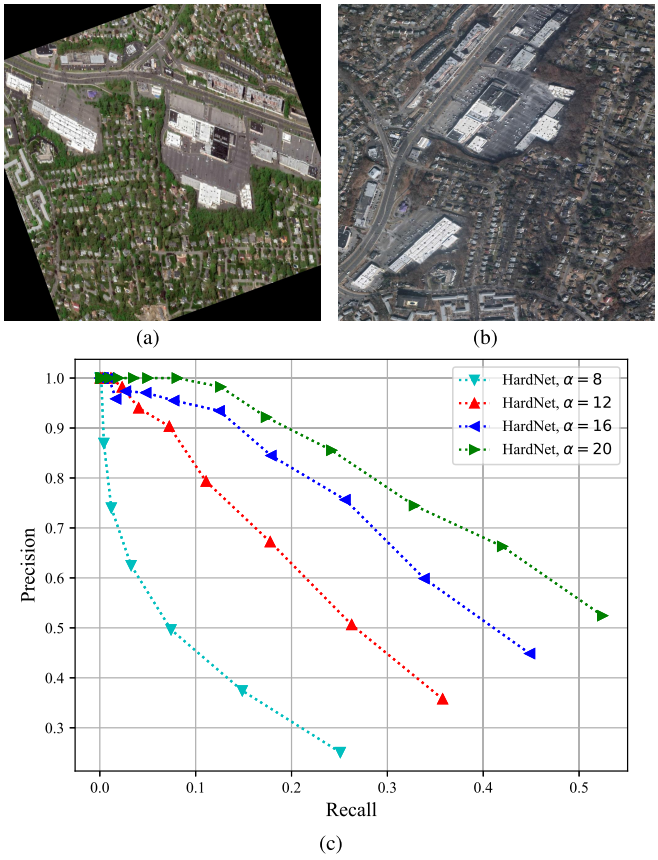

Fig. 2. PR curves for correspondences established through HardNet descriptors using two remote sensing images obtained at different times. α of (7) determines the density of the key point distribution. (a) Source image. (b) Target image. (c) PR curves for correspondence establishment.

Therefore, there are n inliers and $n(n-1)$ outliers among all correspondences across $S^{(\mathrm{src})}$ and $S^{(\mathrm{tar})}$. Consider the correspondence establishment as a binary classification issue, and we can apply the precision–recall (PR) curve to evaluate the performance. To illustrate the effects of the key point distribution density, nonmaximum suppression (NMS) is adopted to ensure that

$$\left\|\boldsymbol{x}_i^{(\mathrm{src})} - \boldsymbol{x}_j^{(\mathrm{src})}\right\|_2 \geq \alpha \quad \forall i, j,\ i \neq j. \tag{7}$$

Through changing α, we can adjust the density of the key point distribution on the source image and the target image. Subsequently, we train the HardNet on our dataset using the loss function in (2) and test the performance of the descriptors under the cases of different α's using two optical remote sensing images in Fig. 2(a) and (b). More details about our dataset can be found in Section IV. The PR curves are shown in Fig. 2(c). As substantiated by Fig. 2(c), we find that the HardNet descriptors become less distinguishable with α in (7) being smaller. In other words, when the key point distribution becomes denser, the correctly matched correspondences tend to be vague compared to the incorrectly matched ones.

For the purpose of improving the distinguishability of descriptors without decreasing the invariance obviously, we propose the DRFD-Net that outputs dual features of different receptive fields and combine them, which is the main difference between our method and other learnable feature-based methods. Moreover, the ILF is introduced to force the DRFD-Net to learn to discriminate the input image patches with large IoUs, which further promotes the performance. To establish correspondences across the source and the target image in a more efficient manner instead of constructing descriptors patch by patch, we practically build the DFDMs based on the DRFDs and combine the DFDMs with FAST key points. Eventually, an iterative method is developed to gradually remove the outliers and calculate the transformation parameters. More details about our method will be introduced in Section III.

III. Proposed Method

Conventional handcrafted descriptors are limited in emphasizing the high-level semantic information that is most likely to keep stable in multitemporal remote sensing images, whereas deep convolutional descriptors tend to be ambiguous with the key points getting closer to each other. In order to relieve the issue of the convolutional features, this article combines the DRFDs constructed by the DRFD-Net presented. Moreover, the ILF is introduced to force the DRFD-Net to learn to discriminate the input image patches with large IoUs to further improve the DRFDs. Subsequently, DFDMs are constructed based on DRFDs and then combined with the FAST key points for efficient correspondence establishment. Eventually, an iterative algorithm is proposed to remove the possible outliers for more accurate image registration results.

A. DRFD-Net and ILF

1) DRFD-Net: Contrary to MatchNet, PN-Net, L2-Net, HardNet, SOSNet, and so on, the intermediate features of the DRFD-Net proposed will be utilized to achieve better distinguishability. Therefore, we split the DRFD-Net into two components, which are the small receptive field description network (SRFD-Net) and large receptive field description network (LRFD-Net), as shown in Fig. 3.

For each convolutional layer in DRFD-Net except for the output layer, batch normalization [46] and ReLU are applied. Furthermore, due to the fact that pooling layers are not learnable, we adopt the convolutional layer with a stride of 2 for downsampling operation. Suppose that one input image patch is denoted as $\boldsymbol{X}$, and the small receptive field descriptors (SRFDs) and the large receptive field descriptors (LRFDs) can be written as follows.

1) $\boldsymbol{X}$*:* Input image patch.
2) Θ_s and Θ_l*:* The parameters of SRFD-Net and LRFD-Net.
3) $\mathscr{F}_s(\Theta_s, \boldsymbol{X})$*:* The output tensor of SRFD-Net.
4) $\mathscr{F}_s(\Theta_s, \boldsymbol{X})_{:,i,j}$*:* The feature vector at the ith row and the jth column on $\mathscr{F}_s(\Theta_s, \boldsymbol{X})$.
5) $\mathscr{F}_l(\Theta_l, \mathscr{F}_s(\Theta_s, \boldsymbol{X}))$*:* The output tensor of LRFD-Net.
6) $\mathscr{F}_l(\Theta_l, \mathscr{F}_s(\Theta_s, \boldsymbol{X}))_{:,i,j}$*:* The feature vector at the ith row and the jth column on $\mathscr{F}_l(\Theta_l, \mathscr{F}_s(\Theta_s, \boldsymbol{X}))$.

There are three input streams that share the same weights during the training process. Each input sample contains three image patches whose sizes are 128×128. The three input image patches are denoted as $\boldsymbol{P}_i^{(1)}$, $\boldsymbol{A}_i$, and $\boldsymbol{P}_i^{(2)}$, where i refers

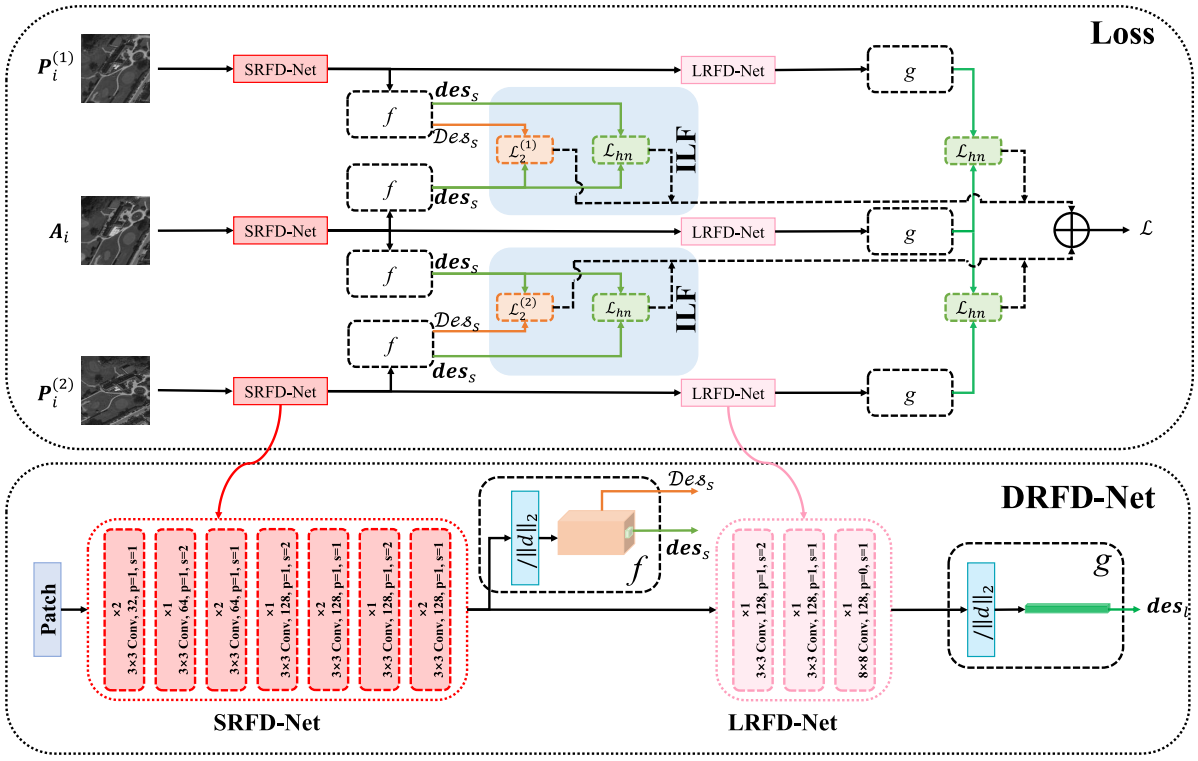

Fig. 3. Architecture and the training process with the ILF of DRFD-Net. Each input sample consists of three image patches of 128 × 128, which are $\boldsymbol{P}_i^{(1)}$, $\boldsymbol{A}_i$, and $\boldsymbol{P}_i^{(2)}$. $\boldsymbol{A}_i$ is the anchor image patch from one scene, while $\boldsymbol{P}_i^{(1)}$ and $\boldsymbol{P}_i^{(2)}$ are two positive image patches from another. There exist random rotation and scaling transformation among the three image patches, which shares the same geographic center. The three streams share the same parameters during the training process. $/\|d\|_2$: the L2 normalization operation. f: to output the SRFD tensor $\mathscr{D}_s$ and the center SRFD vector $\boldsymbol{des}_s$. g: to output the LRFD vector $\boldsymbol{des}_l$. $\mathscr{L}_2^{(1)}$ and $\mathscr{L}_2^{(2)}$: to calculate the two loss terms in (15), respectively. $\mathscr{L}_{\text{hn}}$: to calculate the HardNet loss. $\mathscr{L}$: the total loss function. More details about the DRFD-Net and the ILF can be found in Section III.

to the ith sample. Here, $\boldsymbol{A}_i$ refers to the anchor image patch of the ith sample, and $\boldsymbol{P}_i^{(1)}$ and $\boldsymbol{P}_i^{(2)}$ are two positive image patches of the ith sample. For each input sample, $\boldsymbol{P}_i^{(1)}$ and $\boldsymbol{P}_i^{(2)}$ are originated from the same scene, while $\boldsymbol{A}_i$ is from another that is obtained at a different time. $\boldsymbol{P}_i^{(1)}$, $\boldsymbol{P}_i^{(2)}$, and $\boldsymbol{A}_i$ correspond to the same geographic area with nearly the same geographic center. To ensure rotation-invariance and scale-invariance, there exist random scaling and rotation transformation between each two of the three image patches. More details about the dataset can be found in Section IV.

2) ILF: The inner loss function is proposed to promote the distinguishability of the intermediate descriptors, i.e., SRFDs, which consists of two terms as follows.

The first term is based on the HardNet loss function, as introduced in Section II. Here, we also denote the input batch as $X = \{(\boldsymbol{P}_i^{(1)}, \boldsymbol{A}_i, \boldsymbol{P}_i^{(2)})\}, \quad i = 1, 2, \ldots, n$, where n is the batch size. The output of the SRFD-Net can be denoted as $Y_s = \{(\mathscr{P}_i^{(1)}, \mathscr{A}_i, \mathscr{P}_i^{(2)})\}$, where

$$\begin{aligned} \mathscr{P}_i^{(1)} &= \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(1)}\right) \\ \mathscr{P}_i^{(2)} &= \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(2)}\right) \\ \mathscr{A} &= \mathscr{F}_s(\Theta_s, \boldsymbol{A}_i). \end{aligned} \tag{8}$$

To define the first term that is based on the HardNet loss function, we define another two sets with the similar form of Y in (3) as follows:

$$\begin{aligned} Y_s^{(1)} &= \left\{\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(1)}\right)_{:,9,9}\right)\right\} \\ Y_s^{(2)} &= \left\{\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(2)}\right)_{:,9,9}\right)\right\}. \end{aligned} \tag{9}$$

As depicted previously, $\mathscr{F}_s(\Theta_s, \boldsymbol{X})_{:,i,j}$ refers to the feature vector at the ith row and the jth column on $\mathscr{F}_s(\Theta_s, \boldsymbol{X})$. The input image patch of each sample is of 128 × 128; therefore, the corresponding output tensor of the SRFD-Net is 128 × 16 × 16 according to the constructure in Fig. 3. Thus, $\mathscr{F}_s(\Theta_s, \boldsymbol{P}_i^{(1)})_{:,9,9}$, $\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}$, and $\mathscr{F}_s(\Theta_s, \boldsymbol{P}_i^{(2)})_{:,9,9}$ represent the center vectors on $\mathscr{P}_i^{(1)}$, $\mathscr{A}_i$, and $\mathscr{P}_i^{(2)}$, respectively, which corresponds to the center area of the three input image patches, as shown in Fig. 4. Considering that $\boldsymbol{P}_i^{(1)}$, $\boldsymbol{A}_i$, and $\boldsymbol{P}_i^{(2)}$ share the same geographical center, the first term of the inner loss function can be defined as follows naturally:

$$\mathscr{L}_1 = \mathscr{L}_{\text{hn}}\left(Y_s^{(1)}\right) + \mathscr{L}_{\text{hn}}\left(Y_s^{(2)}\right) \tag{10}$$

where $\mathscr{L}_{\text{hn}}$ is defined in (3).

With $\mathscr{L}_1$, the intermediate descriptors can be discriminative in a degree; however, it is essential for DRFD-Net to learn to discriminate the local patches with large overlapping area directly. As mentioned before, $\mathscr{F}_s(\Theta_s, \boldsymbol{P}_i^{(1)})_{:,9,9}$, $\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}$, and $\mathscr{F}_s(\Theta_s, \boldsymbol{P}_i^{(2)})_{:,9,9}$ are corresponding to the center areas of $\boldsymbol{P}_i^{(1)}$, $\boldsymbol{A}_i$ and $\boldsymbol{P}_i^{(2)}$, respectively. Then, we can

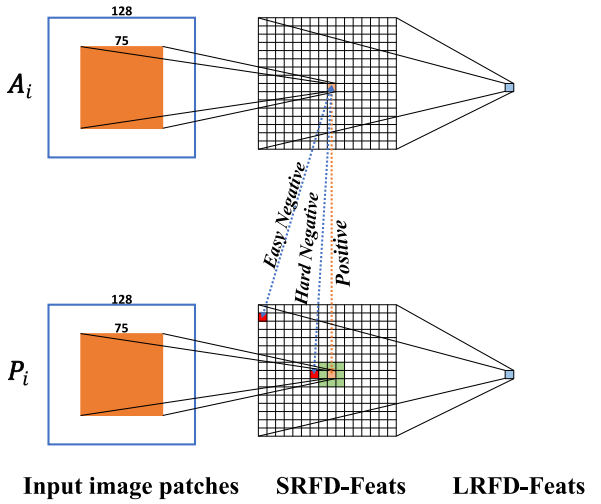

Fig. 4. DFs when calculating $\mathscr{L}_2^{(j)}$ in (15). For the output features of SRFD-Net and LRFD-Net, each grid refers to a feature of 128-d. The orange dashed double arrow refers to the correctly matched features, while the blue ones refer to the incorrectly matched features. The grids colored green refer to DFs, which do not contribute to the calculation of $\mathscr{L}_2^{(j)}$ in (15).

define

$$\begin{aligned}\Delta d_i^{(1)} &= \min_{\substack{p\neq 9\\ q\neq 9}} d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(1)}\right)_{:,p,q}\right) \\ &\quad - d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(1)}\right)_{:,9,9}\right) \\ \Delta d_i^{(2)} &= \min_{\substack{p\neq 9\\ q\neq 9}} d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(2)}\right)_{:,p,q}\right) \\ &\quad - d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(2)}\right)_{:,9,9}\right) \end{aligned} \tag{11}$$

where $d(\boldsymbol{u}, \boldsymbol{v})$ is defined as follows:

$$d(\boldsymbol{u}, \boldsymbol{v}) = \left\| \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|_2} - \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2} \right\|_2. \tag{12}$$

Nevertheless, it is nearly impossible for the geometric center of $\boldsymbol{P}_i^{(1)}$, $\boldsymbol{A}_i^{(1)}$, and $\boldsymbol{P}_i^{(2)}$ to coincide totally. Therefore, for the purpose of robustness, sometimes, we will ignore the features that are too close to the center feature, which are the dead features (DFs) represented by the grids colored green in Fig. 4. Then, (11) is rewritten as follows:

$$\begin{aligned}\Delta d_i^{(1)} &= \min_{p,q\in A} d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(1)}\right)_{:,p,q}\right) \\ &\quad - d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(1)}\right)_{:,9,9}\right) \\ \Delta d_i^{(2)} &= \min_{p,q\in A} d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(2)}\right)_{:,p,q}\right) \\ &\quad - d\left(\mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)_{:,9,9}, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(2)}\right)_{:,9,9}\right) \end{aligned} \tag{13}$$

where A is defined as

$$A = \{n \in \mathbb{N} \mid 1 \leq n < 9-\omega \ \vee \ 9+\omega < n \leq 16\}. \tag{14}$$

In (14), ω is an integer that defines the features that do not contribute to the ILF loss function, i.e., DFs during the process of calculating ILF. The second term in ILF is defined as

$$\begin{aligned}\mathscr{L}_2^{(j)} &= \frac{1}{n}\sum_{i=1}^{n} \max\left(0, 1 - \Delta d_i^{(j)}\right) \\ \mathscr{L}_2 &= \mathscr{L}_2^{(1)} + \mathscr{L}_2^{(2)}. \end{aligned} \tag{15}$$

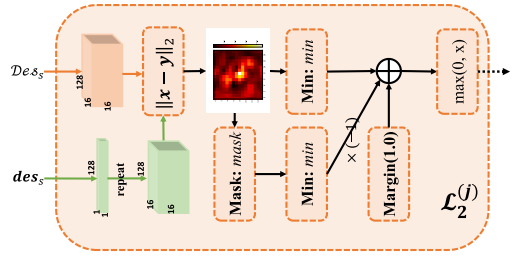

Fig. 5. Process of calculating $\mathscr{L}_2^{(j)}$ in (15), which shows the detailed structure of $\mathscr{L}_2^{(j)}$ components in Fig. 3. Mask: *mask* is used to prevent the DFs from being considered, which are the features colored green in Fig. 4. Min: *min* is to calculate the minimal value of the input matrix.

The main process of calculating the second term of ILF $\mathscr{L}_2^{(j)}$ is shown in Fig. 5. The ILF is a combination of the two terms, which is

$$\mathscr{L}_{\text{ilf}} = \mathscr{L}_1 + \mathscr{L}_2. \tag{16}$$

Apart from $\mathscr{L}_{\text{ilf}}$, we also impose constraints on the output features of LRFD-Net, which is also based on HardNet loss. Define two sets with the output features of the LRFD-Net as

$$\begin{aligned} Y_l^{(1)} &= \left\{\left(\mathscr{F}_l(\Theta_l, \mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)), \mathscr{F}_l\left(\Theta_l, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(1)}\right)\right)\right)\right\} \\ Y_l^{(2)} &= \left\{\left(\mathscr{F}_l(\Theta_l, \mathscr{F}_s(\Theta_s, \boldsymbol{A}_i)), \mathscr{F}_l\left(\Theta_l, \mathscr{F}_s\left(\Theta_s, \boldsymbol{P}_i^{(2)}\right)\right)\right)\right\}. \end{aligned} \tag{17}$$

Here, the input image patch is of 128×128, and then, the output feature of the LRFD-Net is of $128 \times 1 \times 1$; we take it as a 128-d vector in this article. The entire loss function is

$$\mathscr{L} = \mathscr{L}_{\text{ilf}} + \mathscr{L}_{\text{hn}}\left(Y_l^{(1)}\right) + \mathscr{L}_{\text{hn}}\left(Y_l^{(2)}\right) \tag{18}$$

where $\mathscr{L}_{\text{hn}}$ is defined in (3).

B. Correspondence Establishment

Feeding the DRFD-Net with an image patch whose size is 128×128, we can obtain the dual descriptors of different receptive fields. The next step is to use the DRFDs to establish correspondences across the source image and the target image. However, if, for each key point detected, an image patch is cropped and fed into the network for descriptor construction, the time cost will be high. To address the issue, DFDMs are constructed and combined with FAST key points to improve the efficiency of this step.

1) DFDMs: DFDMs consist of the small receptive field feature distance map (SRF-FDM) and the large receptive field feature distance map (LRF-FDM). Select a key point on the source image $\boldsymbol{I}^{(\text{src})}$ whose location is $(x^{(\text{src})}, y^{(\text{src})})$, and the distance from the key point to the image boundaries should be no less than 64. Then, feeding the DRFD-Net with $\boldsymbol{I}^{(\text{src})}$ and $\boldsymbol{I}^{(\text{tar})}$, we have

$$\begin{aligned} \mathscr{U}_s^{(\text{src})} &= \mathscr{F}_s\left(\Theta_s, \boldsymbol{I}^{(\text{src})}\right) \\ \mathscr{U}_l^{(\text{src})} &= \mathscr{F}_l\left(\Theta_l, \mathscr{F}_s\left(\Theta_s, \boldsymbol{I}^{(\text{src})}\right)\right) \\ \mathscr{V}_s^{(\text{tar})} &= \mathscr{F}_s\left(\Theta_s, \boldsymbol{I}^{(\text{tar})}\right) \\ \mathscr{V}_l^{(\text{tar})} &= \mathscr{F}_l\left(\Theta_l, \mathscr{F}_s\left(\Theta_s, \boldsymbol{I}^{(\text{tar})}\right)\right). \end{aligned} \tag{19}$$

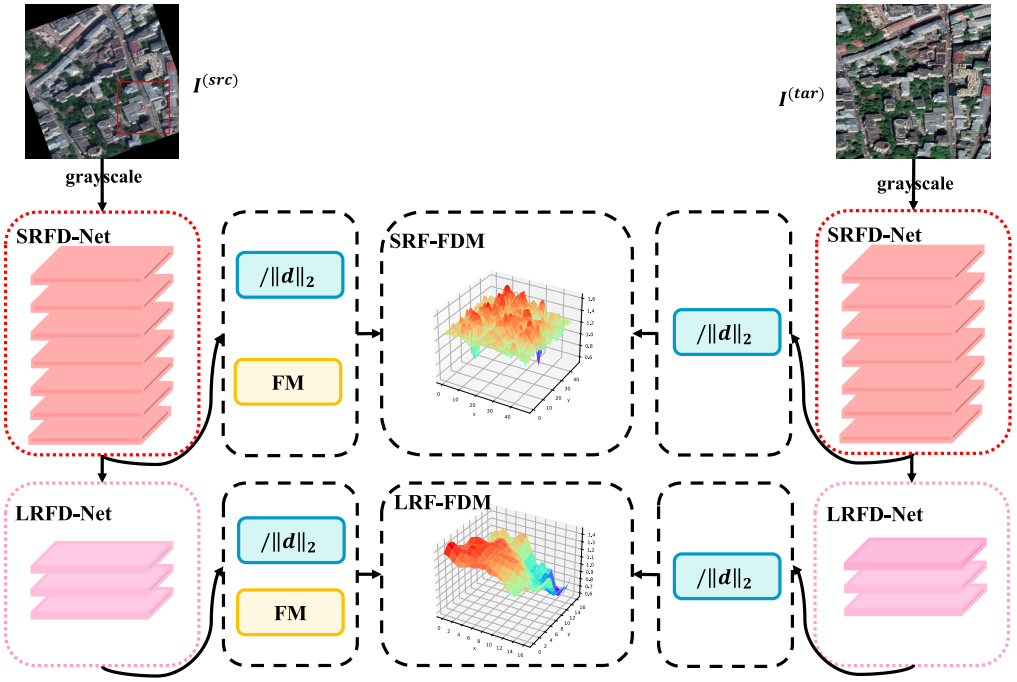

Fig. 6. Main process of calculating the SRF-FDM $\boldsymbol{FDM}^{(s)}$ and the LRF-FDM $\boldsymbol{FDM}^{(l)}$ in (23). $/\|d\|_2$: L2 normalization operation. FM: feature mapping, which refers to the process of mapping feature in (20) and (21).

Then, the corresponding SRFD and LRFD for the key point on the $\mathscr{U}_s^{(\mathrm{src})}$ and $\mathscr{U}_l^{(\mathrm{src})}$ are

$$\begin{aligned} \boldsymbol{U}_s^{(\mathrm{src})} &= \left(\mathscr{U}_s^{(\mathrm{src})}\right)_{:,y_s,x_s} \\ \boldsymbol{U}_l^{(\mathrm{src})} &= \left(\mathscr{U}_l^{(\mathrm{src})}\right)_{:,y_l,x_l} \end{aligned} \tag{20}$$

where we have

$$\begin{aligned} x_s &= \lfloor (x^{(\mathrm{src})} + 11)/8 \rfloor \\ y_s &= \lfloor (y^{(\mathrm{src})} + 11)/8 \rfloor \\ x_l &= \lfloor (x^{(\mathrm{src})} - 33)/16 \rfloor \\ y_l &= \lfloor (y^{(\mathrm{src})} - 33)/16 \rfloor. \end{aligned} \tag{21}$$

Subsequently, each element of the SRF-FDM and the LRF-FDM is defined as follows, respectively:

$$\begin{aligned} d_{i,j}^{(s)} &= \left\| \frac{\boldsymbol{U}_s^{(\mathrm{src})}}{\left\|\boldsymbol{U}_s^{(\mathrm{src})}\right\|_2} - \frac{\left(\mathscr{V}_s^{(\mathrm{tar})}\right)_{:,i,j}}{\left\|\left(\mathscr{V}_s^{(\mathrm{tar})}\right)_{:,i,j}\right\|_2} \right\|_2 \\ d_{p,q}^{(l)} &= \left\| \frac{\boldsymbol{U}_l^{(\mathrm{src})}}{\left\|\boldsymbol{U}_l^{(\mathrm{src})}\right\|_2} - \frac{\left(\mathscr{V}_l^{(\mathrm{tar})}\right)_{:,p,q}}{\left\|\left(\mathscr{V}_l^{(\mathrm{tar})}\right)_{:,p,q}\right\|_2} \right\|_2 \end{aligned} \tag{22}$$

where $(\mathscr{V}_s^{(\mathrm{tar})})_{:,i,j}$ refers to the feature on the ith row and the jth column of $\mathscr{V}_s^{(\mathrm{tar})}$. The SRF-FDM and the LRF-FDM can be written as follows accordingly:

$$\begin{aligned} \boldsymbol{FDM}^{(s)} &= \left(d_{i,j}^{(s)}\right)_{h^{(s)} \times w^{(s)}} \\ \boldsymbol{FDM}^{(l)} &= \left(d_{p,q}^{(l)}\right)_{h^{(l)} \times w^{(l)}} \end{aligned} \tag{23}$$

where $h^{(s)} \times w^{(s)}$ and $h^{(l)} \times w^{(l)}$ refer to the number of features in $\mathscr{V}_s^{(\mathrm{tar})}$ and $\mathscr{V}_l^{(\mathrm{tar})}$, respectively. The process of calculating SRF-FDM and LRF-FDM is shown in Fig. 6. From the SRF-FDM and the LRF-FDM shown in Fig. 6, we find that the global lowest point of SRF-FDM is sharper, which gives a more accurate location, while LRF-FDM has less disturbing local lowest points. Based on this point, we take advantage of both the SRF-FDM and LRF-FDM and combine the DFDMs with FAST key points for correspondence establishment.

2) Correspondence Establishment: The correspondence establishment algorithm is shown in Algorithm 1. Key points in the source image are first detected using the FAST algorithm. In order to decrease the computational complexity, NMS is adopted to prevent the distribution of key points detected from being too dense. In addition, we ensure that the distance from each key point to the image boundaries should be no less than 64. Denote the key points that satisfy the conditions above as

$$S^{(\mathrm{src})} = \left\{ \left(x_1^{(\mathrm{src})}, y_1^{(\mathrm{src})}\right), \ldots, \left(x_n^{(\mathrm{src})}, y_n^{(\mathrm{src})}\right) \right\} \tag{24}$$

where n is the volumn of $S^{(\mathrm{src})}$. For each key point $(x_k^{(\mathrm{src})}, y_k^{(\mathrm{src})})$ in the key point set $S^{(\mathrm{src})}$, (20)–(23) are used to calculate $\boldsymbol{FDM}_k^{(s)}$ and $\boldsymbol{FDM}_k^{(l)}$. After that, we then define

$$\begin{aligned} a_k^{(s)}, b_k^{(s)} &= \arg\min_{i,j} \left(\boldsymbol{FDM}_k^{(s)}\right)_{i,j} \\ a_k^{(l)}, b_k^{(l)} &= \arg\min_{p,q} \left(\boldsymbol{FDM}_k^{(l)}\right)_{p,q} \\ u_k^{(s)}, v_k^{(s)} &= \arg\min_{\substack{i \in G^{(s)} \\ j \in H^{(s)}}} \left(\boldsymbol{FDM}_k^{(s)}\right)_{i,j} \\ u_k^{(l)}, v_k^{(l)} &= \arg\min_{\substack{p \in G^{(l)} \\ q \in H^{(l)}}} \left(\boldsymbol{FDM}_k^{(l)}\right)_{p,q} \end{aligned} \tag{25}$$

Algorithm 1 Correspondence Establishment Using DFDMs and FAST Key Points

Input: $\boldsymbol{I}^{(src)}$: source image; $\boldsymbol{I}^{(tar)}$: target image; ϵ_0: threshold; ϵ_1: threshold;
Output: CP: interest point correspondences;
Initialize $CP = \phi$, $S^{(src)} = NMS(FAST(\boldsymbol{I}^{(src)}))$;
Initialize $S^{(tar)} = FAST(\boldsymbol{I}^{(tar)})$ and $S^{(tar)}_{r,c}$;
for $\boldsymbol{p}_k^{(src)} \in S^{(src)}$ **do**
  if $distToBoundary(\boldsymbol{p}_k^{(src)}, \boldsymbol{I}^{(src)}) < 64$ **then**
    Remove $\boldsymbol{p}_k^{(src)}$ from $S^{(src)}$;
  end if
end for
Do calculations in Eq. (19)
for $\boldsymbol{p}_k^{(src)} \in S^{(src)}$ **do**
  Calculate $\boldsymbol{FDM}_k^{(s)}$ and $\boldsymbol{FDM}_k^{(l)}$ via Eq. (20)-(23);
  Do calculations in Eq. (25)-(26);
  if Eq. (29) is satisfied **then**
    if $C = (S^{(tar)})_{a_k^{(s)}, b_k^{(s)}} \neq \phi$ **then**
      Find the $\boldsymbol{p}_k^{(tar)} \in C$ with the highest response;
      $cp = (\boldsymbol{p}_k^{(src)}, \boldsymbol{p}_k^{(tar)})$;
      Add cp to CP;
    end if
  else if Eq. (31) is satisfied **then**
    if Eq. (32) is satisfied **then**
      $C = (S^{(tar)})_{a_k^{(s)}, b_k^{(s)}}$;
      Find the $\boldsymbol{p}_k^{(tar)} \in C$ with the highest response;
      $cp = (\boldsymbol{p}_k^{(src)}, \boldsymbol{p}_k^{(tar)})$;
      Add cp to CP;
    else if Eq. (33) is satisfied **then**
      $C = (S^{(tar)})_{u_k^{(s)}, v_k^{(s)}}$;
      Find the $\boldsymbol{p}_k^{(tar)} \in C$ with the highest response;
      $cp = (\boldsymbol{p}_k^{(src)}, \boldsymbol{p}_k^{(tar)})$;
      Add cp to CP;
    end if
  end if
end for
Output CP.

where $G^{(s)}$, $H^{(s)}$, $G^{(l)}$, and $H^{(l)}$ are defined as

$$\begin{aligned} G^{(s)} &= \left\{ n \in \mathbb{N} \mid n < a_k^{(s)} - \omega_1 \ \vee \ n > a_k^{(s)} + \omega_1 \right\} \\ H^{(s)} &= \left\{ n \in \mathbb{N} \mid n < b_k^{(s)} - \omega_1 \ \vee \ n > b_k^{(s)} + \omega_1 \right\} \\ G^{(l)} &= \left\{ n \in \mathbb{N} \mid n < a_k^{(l)} - \omega_2 \ \vee \ n > a_k^{(l)} + \omega_2 \right\} \\ H^{(l)} &= \left\{ n \in \mathbb{N} \mid n < b_k^{(l)} - \omega_2 \ \vee \ n > b_k^{(l)} + \omega_2 \right\}. \end{aligned} \tag{26}$$

In this article, we fix $\omega_1 = \omega$ and $\omega_2 = 2$, where ω is the hyperparameter defined in (14) during the training process. According to the lowest point locations on the SRF-FDM and LRF-FDM, we can predict correspondent locations for key points detected on the source image. The feature at $(a_k^{(s)}, b_k^{(s)})$ on $\boldsymbol{FDM}_k^{(s)}$ corresponds to the point at $(8b_k^{(s)} - 7, 8a_k^{(s)} - 7)$, while the feature at $(a_k^{(l)}, b_k^{l})$ on $\boldsymbol{FDM}_k^{(l)}$ corresponds to $(41 + 16b_k^{(l)}, 41 + 16a_k^{(l)})$. However, due to the downsampling operation, there are inevitable location errors if we directly use the locations of SRF-FDM and LRF-FDM. Therefore, we further detect the FAST key points on the target image, which are denoted as

$$S^{(\text{tar})} = \left\{ \left(x_1^{(\text{tar})}, y_1^{(\text{tar})}, R_1\right), \ldots, \left(x_m^{(\text{tar})}, y_m^{(\text{tar})}, R_m\right) \right\} \tag{27}$$

where m is the volumn of $S^{(\text{tar})}$ and R_k refers to the response of the kth key point. Then, sprinkle the FAST points on the SRF-FDM. In other words, for each $(x_k^{(\text{tar})}, y_k^{(\text{tar})}, R_k)$ in $S^{(\text{tar})}$, it falls on the r_kth row and c_kth column grid of the SRF-FDM, where

$$\begin{aligned} c_k &= \left\lfloor \left(x_k^{(\text{tar})} + 11\right)/8 \right\rfloor \\ r_k &= \left\lfloor \left(y_k^{(\text{tar})} + 11\right)/8 \right\rfloor. \end{aligned} \tag{28}$$

Define the key points on the rth row and cth column grid as a subset of $S^{(\text{tar})}$, which is denoted as $(S^{(\text{tar})})_{r,c}$. If

$$\left(\boldsymbol{FDM}_k^{(s)}\right)_{a_k^{(s)}, b_k^{(s)}} - \left(\boldsymbol{FDM}_k^{(s)}\right)_{u_k^{(s)}, v_k^{(s)}} \geq \epsilon_0 \tag{29}$$

$$\left(S^{(\text{tar})}\right)_{a_k^{(s)}, b_k^{(s)}} \neq \phi \tag{30}$$

we directly assume that the key point with the highest response in $(S^{(\text{tar})})_{a_k^{(s)}, b_k^{(s)}}$ is correspondent to $(x_k^{(\text{src})}, y_k^{(\text{src})})$. Otherwise, if

$$\left(\boldsymbol{FDM}_k^{(l)}\right)_{a_k^{(l)}, b_k^{(l)}} - \left(\boldsymbol{FDM}_k^{(l)}\right)_{u_k^{(l)}, v_k^{(l)}} \geq \epsilon_1 \tag{31}$$

then we will see whether $(8b_k^{(s)} - 7, 8a_k^{(s)} - 7)$ or $(8v_k^{(s)} - 7, 8u_k^{(s)} - 7)$ is in the neighborhood of $(41 + 16b_k^{(l)}, 41 + 16a_k^{(l)})$. In other words, if

$$\begin{aligned} \left|8b_k^{(s)} - 16b_k^{(l)} - 48\right| &\leq 32 \\ \left|8a_k^{(s)} - 16a_k^{(l)} - 48\right| &\leq 32 \\ (S^{(\text{tar})})\_a_k^{(s)}, b_k^{(s)} &\neq \phi \end{aligned} \tag{32}$$

we still take the key point with the highest response in $(S^{(\text{tar})})_{a_k^{(s)}, b_k^{(s)}}$ as the correspondent point. Else if,

$$\begin{aligned} \left|8v_k^{(s)} - 16b_k^{(l)} - 48\right| &\leq 32 \\ \left|8u_k^{(s)} - 16a_k^{(l)} - 48\right| &\leq 32 \\ (S^{(\text{tar})})_{u_k^{(s)}, v_k^{(s)}} &\neq \phi \end{aligned} \tag{33}$$

we will take the key point with the highest response in $(S^{(\text{tar})})_{u_k^{(s)}, v_k^{(s)}}$ as the correspondent point.

For the two thresholds ϵ_0 and ϵ_1 in (29) and (31), in fact, the values should depend on the similarity extent of the source and target images, and the number of key points detected. In this article, we fix $\epsilon_0 = \epsilon_1$, both within the range of $[0, 0.15]$.

C. Iterative Image Registration

After the correspondences are established, the next step is to solve the transformation parameters based on these correspondences and complete the registration process. In light of the possible outliers, we develop an iterative algorithm for image registration to make the regression of transformation parameters more accurate.

1) Optimal Transformation: We assume that there only exist translation, rotation, and scaling transformation between the source image and the target image. Suppose that the key point

correspondences are denoted as $CP = \{(\boldsymbol{p}_1, \boldsymbol{q}_1), \ldots, (\boldsymbol{p}_z, \boldsymbol{q}_z)\}$, where z refers to number of correspondences. Denote the transformation parameters as s, $\boldsymbol{R}$, and $\boldsymbol{t}$ for scaling, rotation, and translation. Then, the cost function is defined as follows:

$$C(s, \boldsymbol{R}, \boldsymbol{t}) = \sum_{i=1}^{z} \|s\boldsymbol{R}\boldsymbol{p}_i + \boldsymbol{t} - \boldsymbol{q}_i\|_2^2. \tag{34}$$

The problem of solving s, $\boldsymbol{R}$, and $\boldsymbol{t}$ can be written as

$$\begin{aligned} s, \boldsymbol{R}, \boldsymbol{t} = \arg\min_{s,\boldsymbol{R},\boldsymbol{t}} & C(s, \boldsymbol{R}, \boldsymbol{t}) \\ \text{s.t.} \quad & \boldsymbol{R}_{1,1} = \boldsymbol{R}_{2,2} \\ & \boldsymbol{R}_{1,2} = -\boldsymbol{R}_{2,1} \\ & \boldsymbol{R}_{1,1}^2 + \boldsymbol{R}_{2,1}^2 = 1. \end{aligned} \tag{35}$$

Then, define the centroids of CP

$$\begin{aligned} \boldsymbol{\mu}_p &= \frac{1}{z}\sum_{i=1}^{z} \boldsymbol{p}_i \\ \boldsymbol{\mu}_q &= \frac{1}{z}\sum_{i=1}^{z} \boldsymbol{q}_i. \end{aligned} \tag{36}$$

Let $\boldsymbol{p}_i' = \boldsymbol{p}_i - \boldsymbol{\mu}_p$, and $\boldsymbol{q}_i' = \boldsymbol{q}_i - \boldsymbol{\mu}_q$, and (34) can be rewritten as

$$C(s, \boldsymbol{R}, \boldsymbol{t}) = \sum_{i=1}^{z} \left(\left\|\boldsymbol{q}_i' - s\boldsymbol{R}\boldsymbol{p}_i'\right\|_2^2 + \left\|\boldsymbol{\mu}_q - s\boldsymbol{R}\boldsymbol{\mu}_p - \boldsymbol{t}\right\|_2^2\right). \tag{37}$$

Then, we can infer

$$\begin{aligned} (s\boldsymbol{R}) &= \arg\min_{s\boldsymbol{R}} \sum_{i=1}^{z} \|\boldsymbol{q}_i' - s\boldsymbol{R}\boldsymbol{p}_i'\|_2^2 \\ \boldsymbol{t} &= \boldsymbol{\mu}_q - (s\boldsymbol{R})\boldsymbol{\mu}_p. \end{aligned} \tag{38}$$

Let

$$\begin{aligned} C_1 &= \sum_{i=1}^{z} \left\|\boldsymbol{q}_i' - s\boldsymbol{R}\boldsymbol{p}_i'\right\|_2^2 \\ &= \sum_{i=1}^{z} \left(\boldsymbol{q}_i'^T\boldsymbol{q}_i' + s^2\boldsymbol{p}_i'^T\boldsymbol{p}_i' - 2s\boldsymbol{q}_i'^T\boldsymbol{R}\boldsymbol{p}_i'\right). \end{aligned} \tag{39}$$

To minimize C_1, let $(\partial C_1/\partial s) = 0$, and we have

$$\begin{aligned} s^2 &= \frac{\sum_{i=1}^{z} \boldsymbol{q}_i'^T s\boldsymbol{R}\boldsymbol{p}_i'}{\sum_{i=1}^{z} \boldsymbol{p}_i'^T\boldsymbol{p}_i'} \approx \frac{\sum_{i=1}^{z} \boldsymbol{q}_i'^T\boldsymbol{q}_i'}{\sum_{i=1}^{z} \boldsymbol{p}_i'^T\boldsymbol{p}_i'} \\ s &\approx \sqrt{\frac{\sum_{i=1}^{z} \boldsymbol{q}_i'^T\boldsymbol{q}_i'}{\sum_{i=1}^{z} \boldsymbol{p}_i'^T\boldsymbol{p}_i'}}. \end{aligned} \tag{40}$$

Then, we calculate the optimal rotation and translation

$$\boldsymbol{R} = \arg\min_{\boldsymbol{R}} \sum_{i=1}^{z} \left(-2\boldsymbol{q}_i'^T\boldsymbol{R}s\boldsymbol{p}_i'\right). \tag{41}$$

Let $\boldsymbol{W} = s\sum_{i=1}^{z} \boldsymbol{q}_i'\boldsymbol{p}_i'^T$, and factorize $\boldsymbol{W}$ via SVD; we have

$$\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T. \tag{42}$$

Then, the optimal rotation and translation can be calculated as follows:

$$\boldsymbol{R} = \boldsymbol{U}\boldsymbol{V}^T, \ \boldsymbol{t} = \boldsymbol{\mu}_q - s\boldsymbol{R}\boldsymbol{\mu}_p. \tag{43}$$

2) Iterative Image Registration: Based on the correspondences built by Algorithm 1, we propose an iterative method to gradually remove the possible outliers and optimize the transformation parameters for more accurate registration. For the reason that we do not need DRFD-Net to build descriptors in this process, the iterative method is practically computationally efficient. Denote the correspondences obtained by Algorithm 1 as

$$\mathrm{CP}_0 = \left\{\left(\left(\boldsymbol{p}_1^{(\mathrm{src})}\right)_0, \boldsymbol{p}_1^{(\mathrm{tar})}\right), \ldots, \left(\left(\boldsymbol{p}_{z_0}^{(\mathrm{src})}\right)_0, \boldsymbol{p}_{z_0}^{(\mathrm{tar})}\right)\right\} \tag{44}$$

where $(\boldsymbol{p}_k^{(\mathrm{src})})_0$ is the kth key point in the source image, z_0 is the volume of CP_0, and $\boldsymbol{p}_k^{(\mathrm{tar})}$ refers to the key point in the target image, which is corresponding to $(\boldsymbol{p}_k^{(\mathrm{src})})_0$. First, s_0, $\boldsymbol{R}_0$, and $\boldsymbol{t}_0$ are calculated using the method through (34)–(43). Then,

$$\begin{aligned} \left(\boldsymbol{p}_k^{(\mathrm{src})}\right)_1 &= s_0\boldsymbol{R}_0\left(\boldsymbol{p}_k^{(\mathrm{src})}\right)_0 + \boldsymbol{t}_0 \\ \delta_k^{(0\to1)} &= \left\|\boldsymbol{p}_k^{(\mathrm{tar})} - \left(\boldsymbol{p}_k^{(\mathrm{src})}\right)_1\right\|_2 \\ m^{(0\to1)} &= \frac{1}{z_0}\sum_{k=1}^{z_0} \delta_k^{(0\to1)} \\ \sigma^{(0\to1)} &= \frac{1}{z_0}\sum_{k=1}^{z_0} \left(\delta_k^{(0\to1)} - m^{(0\to1)}\right)^2. \end{aligned} \tag{45}$$

If $\delta_k^{(0\to1)} \le m^{(0\to1)} + \alpha_0\sigma^{(0\to1)}$, then $((\boldsymbol{p}_k^{(\mathrm{src})})_1, \boldsymbol{p}_k^{(\mathrm{tar})})$ is preserved. After the first iteration, we have

$$\mathrm{CP}_1 = \left\{\left(\left(\boldsymbol{p}_1^{(\mathrm{src})}\right)_1, \boldsymbol{p}_1^{(\mathrm{tar})}\right), \ldots, \left(\left(\boldsymbol{p}_{z_1}^{(\mathrm{src})}\right)_1, \boldsymbol{p}_{z_1}^{(\mathrm{tar})}\right)\right\}. \tag{46}$$

If $z_0 = z_1$, then $\alpha_1 = (1-\eta)\alpha_0$. Repeat the process above until the number of iterations is reached. Besides, if the number of correspondences left is not enough, the iterative algorithm will be ceased either. The least number of correspondences reserved is set to be 40 in this article. Eventually, we can calculate the homographic matrix $\boldsymbol{H}$

$$\begin{aligned} \boldsymbol{H}_{\mathrm{it}} &= \begin{Bmatrix} s_{\mathrm{it}}\boldsymbol{R}_{\mathrm{it}} & \boldsymbol{t}_{\mathrm{it}} \\ \boldsymbol{0} & 1 \end{Bmatrix} \\ \boldsymbol{H} &= \prod_{\mathrm{it}=N}^{0} \boldsymbol{H}_{\mathrm{it}} \end{aligned} \tag{47}$$

where it refers to the itth iteration, and N marks the iteration when the process is ceased. $\boldsymbol{H}$ is the solved transformation matrix from $\boldsymbol{I}^{(\mathrm{src})}$ to $\boldsymbol{I}^{(\mathrm{tar})}$. The iterative registration method is shown in Algorithm 2.

IV. Experiments

A. Dataset and Training

1) Dataset: Currently, there are many different datasets proposed for local descriptor construction, such as UBC Phototour, HPatches, and ETH SfM. Nevertheless, these datasets are originated from natural images instead of remote sensing images, which are improper for the construction of local descriptors for remote sensing images.

In order to train the DRFD-Net, we construct a new dataset using 13 groups of optical remote sensing images obtained from Google Earth. Each group contains two remote sensing images that are produced at different times and cover the same

Algorithm 2 Iterative Image Registration (IIR)

Input: $\boldsymbol{I}^{(src)}$: source image; CP_0: correspondeces obtained via Algorithm 1;
Output: $\boldsymbol{I}^{(pred)}$: the registered image;
Initialize $\alpha_0 = 3$, $\eta = 0.05$, $It = 50$, $it = 0$;
Initialize s_0, $\boldsymbol{R}_0$, $\boldsymbol{t}_0$ using Eq. (34)-(43);
while $it < It$ **do**
  Initialize $CP_{it+1} = \phi$;
  for each $k \leq z_{it}$ **do**
    $(\boldsymbol{p}_k^{(src)})_{it+1} = s_{it}\boldsymbol{R}_{it}(\boldsymbol{p}_k^{(src)})_{it} + \boldsymbol{t}_{it}$;
    $\delta_k^{(it\to it+1)} = \|\boldsymbol{p}_k^{(tar)} - (\boldsymbol{p}_k^{(src)})_{it+1}\|_2$;
  end for
  $m^{(it\to it+1)} = \frac{1}{z_{it}}\sum_{k=1}^{z_{it}} \delta_k^{(it\to it+1)}$;
  $\sigma^{(it\to it+1)} = \frac{1}{z_{it}}\sum_{k=1}^{z_{it}} (\delta_k^{(it\to it+1)} - m^{(it\to it+1)})^2$;
  for each $k \leq z_{it}$ **do**
    if $\delta_k^{(it\to it+1)} \leq m^{(it\to it+1)} + \alpha_{it}\sigma^{(it\to it+1)}$ **then**
      Add $((\boldsymbol{p}_k^{(src)})_{it+1}, \boldsymbol{p}_k^{(tar)})$ to CP_{it+1};
    end if
  end for
  if $z_{it+1} < 40$ **then**
    break;
  end if
  Calculate s_{it+1}, $\boldsymbol{R}_{it+1}$, $\boldsymbol{t}_{it+1}$ with CP_{it+1} via Eq. (34)-(43);
  if $z_{it} = z_{it+1}$ **then**
    $\alpha_{it+1} = (1-\eta)\alpha_{it}$;
  end if
  $it++$;
end while
$N = it$;
$\boldsymbol{H} = \prod_{it=N}^{0} \boldsymbol{H}_{it}$;
Build the registered image $\boldsymbol{I}^{(pred)}$ using $\boldsymbol{H}$;

geographical areas. The original remote sensing images are from urban areas around the world, whose spatial resolutions are around 1 m. The width and the height of the original images range from 6000 to 10 000 pixels. For the dataset constructed, each sample of the dataset contains one anchor image patch and two positive image patches. The method of how to extract a sample from the original data is introduced in the following.

Denote the two registered images in the kth group as $\boldsymbol{I}_k^{(1)}$ and $\boldsymbol{I}_k^{(2)}$, respectively, and we adopt FAST algorithm to detect the key points on $\boldsymbol{I}_k^{(1)}$. To make sure that the key points detected are significant, we set the response threshold of the FAST algorithm to be 32, which means that the key points whose responses are under 32 will be abandoned. To reduce the data redundancy, NMS is used to ensure that the horizontal and vertical distances between any two key points selected are no less than 64. For each FAST key point selected, we crop two patches in both $\boldsymbol{I}_k^{(1)}$ and $\boldsymbol{I}_k^{(2)}$ with the sizes of 256×256 centered at the key point, which are denoted as $\boldsymbol{BP}_i^{(1)}$ and $\boldsymbol{BP}_i^{(2)}$. Then, a smaller image patch whose size is 128×128 is cropped at the center of $\boldsymbol{BP}_i^{(1)}$, which is the anchor image patch denoted as $\boldsymbol{A}_i$. After that, $\boldsymbol{BP}_i^{(2)}$ is transformed with random scaling and rotation, where the rotation angle obeys the uniform distribution of $[-\pi, \pi]$ and the scaling factor obeys the uniform distribution of $[0.8, 1.25]$. After the transformation, crop a patch at the center of the transformed $\boldsymbol{BP}_i^{(2)}$ with the sizes of 128×128, which is one of the positive image patches denoted as $\boldsymbol{P}_i^{(1)}$. Perform the same operation again, and we can obtain the other positive image patch $\boldsymbol{P}_i^{(2)}$. $\boldsymbol{P}_i^{(1)}$ and $\boldsymbol{P}_i^{(2)}$ are originated from one scene, while $\boldsymbol{A}_i$ is from the other. The three image patches $\boldsymbol{A}_i$, $\boldsymbol{P}_i^{(1)}$, and $\boldsymbol{P}_i^{(2)}$ make up a sample of the dataset. Fig. 7 shows three examples of the dataset.

2) Training: We apply the Nvidia Tesla V100 GPU for the training of the DRFD-Net. Throughout the training process, the stochastic gradient descent (SGD) algorithm [47] is used for parameter optimization. In this article, we set the learning rate lr = 0.01 and momentum mmt = 0.5. To inhibit overfitting, dropout [48] layer and weight decay are applied during the process of training.

B. Evaluation of Correspondence Establishment

1) Testing Data: To measure the performance of the DRFDs, we select four groups of remote sensing images for testing. Each group contains two images covering the same geographical areas, which are acquired at different times, as shown in Fig. 8. The detailed information about the four groups of remote sensing images is shown in Table I. The transformation parameters are from the target images to the source images.

2) Evaluation Methods: We will introduce two methods for evaluation. One is the classic PR curve, and the other is the ρ–γ curve. In this article, the PR curves are applied to evaluate the performance of descriptors when the correspondence set is offered across the source image and the target image. However, in real image registration cases, we do not know the exact transformation between the source image and the target image; therefore, we also apply the ρ–γ curves to evaluate the location performance in this article. The details about the two evaluation methods are presented as follows.

a) Precision–recall curve: Suppose that there are two ordered key point sets from the source image and the target image, respectively

$$\begin{aligned} S^{(\text{src})} &= \left\{\boldsymbol{p}_1^{(\text{src})}, \ldots, \boldsymbol{p}_e^{(\text{src})}\right\} \\ S^{(\text{tar})} &= \left\{\boldsymbol{p}_1^{(\text{tar})}, \ldots, \boldsymbol{p}_e^{(\text{tar})}\right\} \end{aligned} \tag{48}$$

where $(\boldsymbol{p}_k^{(\text{src})}, \boldsymbol{p}_k^{(\text{tar})})$, $k = 1, \ldots, e$ is a correspondence correctly matched. Therefore, the number of the outliers is $e(e-1)$. Taking it as a binary classification, we can apply PR index to evaluate the performance. Suppose that the judgment function is $\hat{I}$, and if $\hat{I}(\boldsymbol{p}_k^{(\text{src})}, \boldsymbol{p}_k^{(\text{tar})}) = 1$, the correspondence is then regarded to be correctly matched. Therefore, we can define the precision and recall as follows:

$$\begin{aligned} p &= \frac{\sum_{i,j} \hat{I}\left(\boldsymbol{p}_i^{(\text{src})}, \boldsymbol{p}_i^{(\text{tar})}\right) I(i=j)}{\sum_{i,j} \hat{I}(\boldsymbol{p}_i^{(\text{src})}, \boldsymbol{p}_i^{(\text{tar})})} \\ r &= \frac{\sum_{i,j} \hat{I}\left(\boldsymbol{p}_i^{(\text{src})}, \boldsymbol{p}_i^{(\text{tar})}\right) I(i=j)}{e} \end{aligned} \tag{49}$$

where I is the indication function.

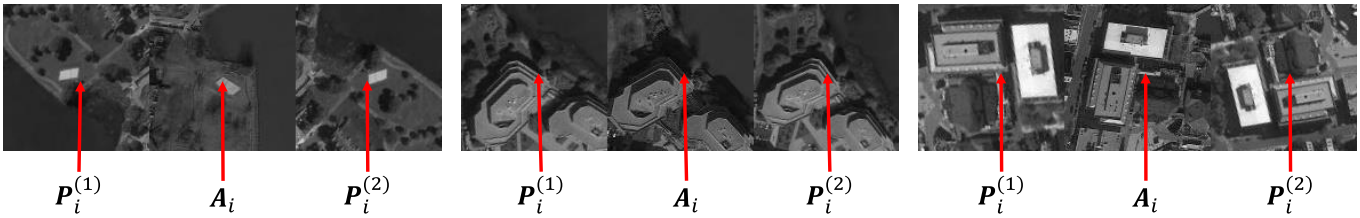

Fig. 7. Three examples in the dataset. A_i represents the anchor image patch, while $P_i^{(1)}$ and $P_i^{(2)}$ refer to the two positive image patches. It can be seen that A_i is originated from a different scene compared to $P_i^{(1)}$ and $P_i^{(2)}$.

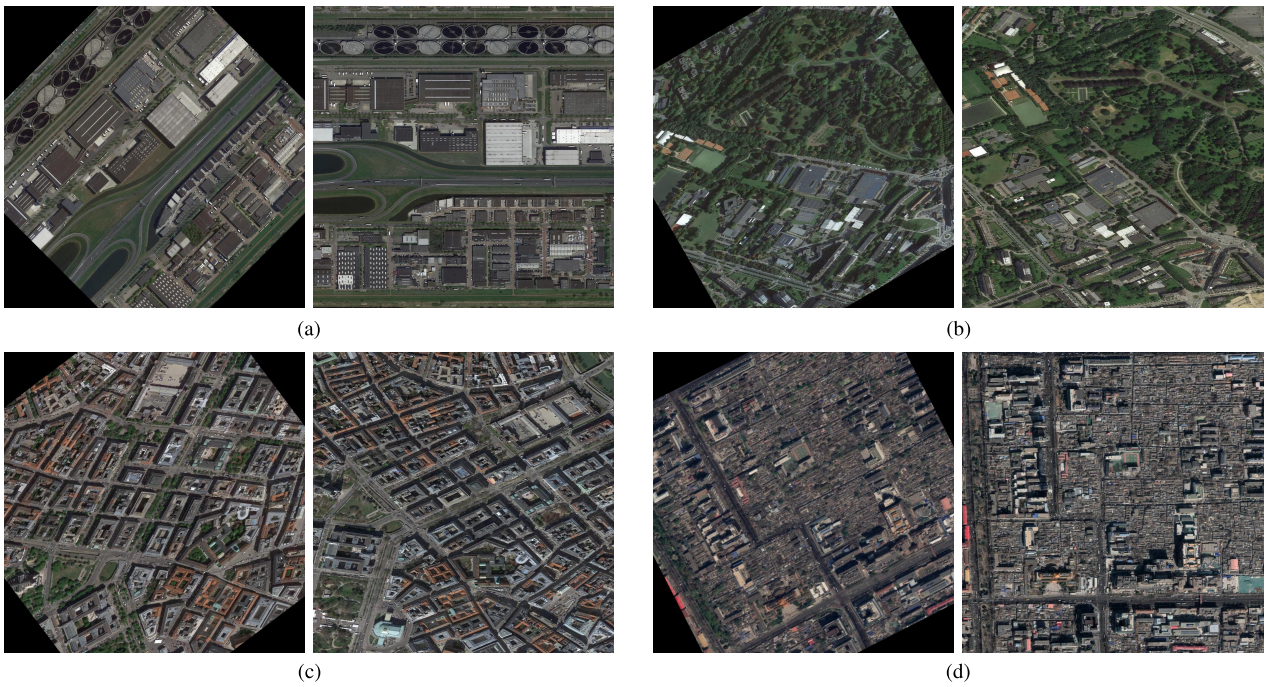

Fig. 8. Testing data for evaluation of the descriptors. The images of column 1 and column 3 are the source images, and the images of column 2 and column 4 are the target images. (a) Group A. (b) Group B. (c) Group C. (d) Group D.

TABLE I
INFORMATION OF TESTING DATA IN FIG. 8

Group	Area	Time (Target)	Time (Source)	Resolution Level[1]	Scaling	Rotation	Tranformation center
A	Amsterdam	2018-4-18	2019-4-20	17	0.97	$\pi/4$	$(511.5, 511.5)$
B	Brussels	2013-7-7	2015-10-1	17	1.0	$\pi/6$	$(511.5, 511.5)$
C	Vienna	2017-3-31	2020-4-25	17	1.05	$37\pi/180$	$(511.5, 511.5)$
D	Beijing	2019-2-21	2020-4-6	17	0.9	$3\pi/20$	$(511.5, 511.5)$

[1] The resolution at Level 17 in Google Earth is approximately $1m$.

b) ρ–γ curve: Denote the true transformation matrix from $\boldsymbol{I}^{(\mathrm{src})}$ to $\boldsymbol{I}^{(\mathrm{tar})}$ as

$$\boldsymbol{T} = \begin{pmatrix} s\cos\phi & -s\sin\phi & t_x \\ s\sin\phi & s\cos\phi & t_y \end{pmatrix} \tag{50}$$

and the key point set in the source image is written as

$$P^{(\mathrm{src})} = \left\{ \boldsymbol{p}_1^{(\mathrm{src})}, \ldots, \boldsymbol{p}_e^{(\mathrm{src})} \right\}. \tag{51}$$

The correspondences obtained are denoted as

$$\mathrm{CP} = \left\{ \left(\boldsymbol{p}_{g_1}^{(\mathrm{src})}, \boldsymbol{p}_{g_1}^{(\mathrm{tar})} \right), \ldots, \left(\boldsymbol{p}_{g_m}^{(\mathrm{src})}, \boldsymbol{p}_{g_m}^{(\mathrm{tar})} \right) \right\} \tag{52}$$

where $1 \le g_k \le e, \quad k = 1, \ldots, m$. Then, we define

$$\begin{aligned} \rho &= \frac{\sum_{j=1}^{m} I\left(\left\| \boldsymbol{T}\boldsymbol{p}_{g_j}^{(\mathrm{src})} - \boldsymbol{p}_{g_j}^{(\mathrm{tar})} \right\|_2 < \tau \right)}{m} \\ \gamma &= \frac{\sum_{j=1}^{m} I\left(\left\| \boldsymbol{T}\boldsymbol{p}_{g_j}^{(\mathrm{src})} - \boldsymbol{p}_{g_j}^{(\mathrm{tar})} \right\|_2 < \tau \right)}{e} \end{aligned} \tag{53}$$

where τ is the tolerance of location error. When ρ and γ are both larger, the performance is better.

3) Ablation Studies: In this article, we propose the idea to combine SRFDs and LRFDs to promote the distinguishability for local descriptors. In addition, we develop the ILF to train the network to further improve the performance of

TABLE II
CORRESPONDING AP VALUES OF PR CURVES IN FIG. 9

	Group A		Group B		Group C		Group D	
	$\alpha = 8$	$\alpha = 16$	$\alpha = 8$	$\alpha = 16$	$\alpha = 8$	$\alpha = 16$	$\alpha = 8$	$\alpha = 16$
SRFD-128	0.1006	0.2189	0.0761	0.1714	0.2438	0.4612	0.0276	0.0926
LRFD-128	0.3767	0.6390	0.1832	0.4144	0.5196	0.8260	0.0471	0.2186
CatDRFD-256	0.4168	0.6626	0.2482	0.4678	0.6450	0.8530	0.1228	0.3434
LRFD-256	0.4066	0.6781	0.2141	0.4509	0.5666	0.8535	0.0435	0.2166
SRFD-128 + ILF	0.5039	0.6148	0.2413	0.3568	0.6483	0.7659	0.1186	0.2226
LRFD-128 + ILF	0.5008	0.7419	0.2285	0.4395	0.6678	0.8938	0.1040	0.3623
CatDRFD-256 + ILF	**0.6570**	**0.7819**	**0.3638**	**0.5277**	**0.8172**	**0.9140**	**0.2459**	**0.4674**

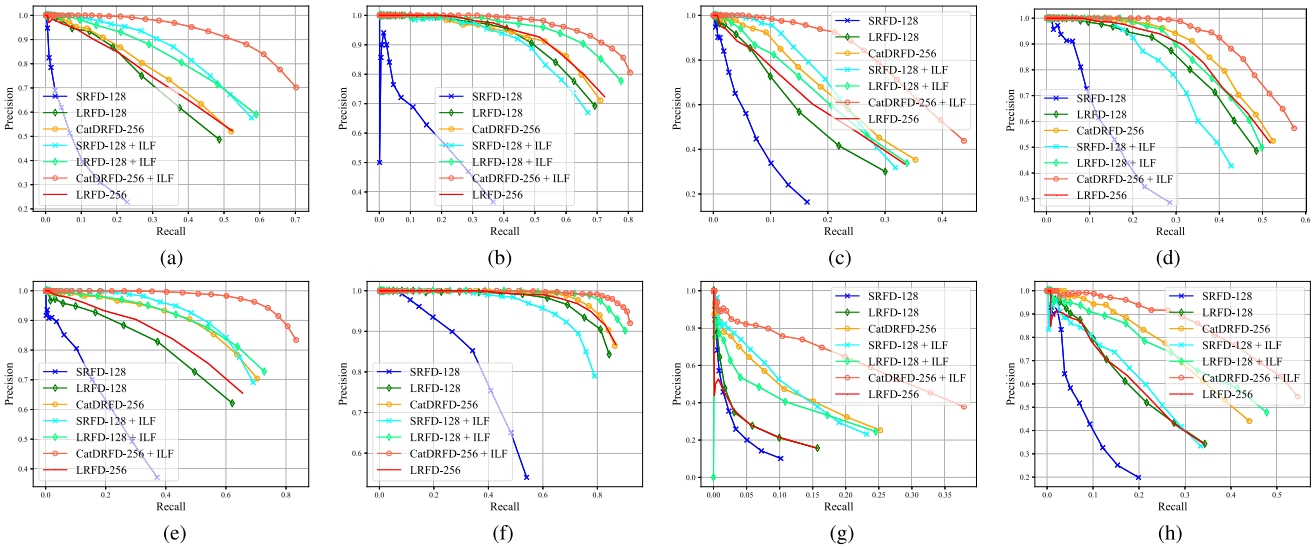

Fig. 9. PR curves for ablation studies. SRFD-128: 128-d SRFD. LRFD-128: 128-d LRFD. CatDRFD-256: 256-d catenated descriptor of DRFDs. LRFD-256: the 256-d LRFD with the channels of convolutional layers in LRFD-Net being 256. "+ILF" means that the inner loss function ILF is introduced during the training process of the DRFD-Net; otherwise, the loss function is $\mathscr{L}_{\text{hn}}(Y_l^{(1)}) + \mathscr{L}_{\text{hn}}(Y_l^{(2)})$ without $\mathscr{L}_{\text{ilf}}$ in (10). For descriptors trained with ILF, ω in (14) is fixed to be 1. (a) A($\alpha = 8$). (b) A($\alpha = 16$). (c) B($\alpha = 8$). (d) B($\alpha = 16$). (e) C($\alpha = 8$). (f) C($\alpha = 16$). (g) D($\alpha = 8$). (h) D($\alpha = 16$).

the descriptors. We would like to investigate how combining the DRFDs and introducing the ILF will influence the performance of the local descriptors. To this end, we give two ordered point sets on the source image and the target image as (48), treating the matching process as a binary classification problem. Then, we use the PR curves for evaluation, which are shown in Fig. 9, and the corresponding average precision (AP) values are shown in Table II. We test the SRFDs, LRFDs, and the catenation of SRFDs and LRFDs under the cases when ILF are used and not used, respectively. Here, we fix $\omega = 1$ in (14) when testing the descriptors with ILF. From Table II and Fig. 9, we can see that the catenated DRFDs (CatDRFDs) outperform SRFDs and LRFDs, no matter whether ILF is introduced during the training process or not. For better illustration, we subsequently modify the channels of convolutional layers in the LRFD-Net to 256 and train again. We find that the 256-d LRFDs perform worse than the 256-d CatDRFDs, which further proves that SRFDs do help in improving the performance of the convolutional descriptors. In addition, comparing the case when ILF is not adopted with the case when ILF is used, we find that the ILF not only improves the distinguishability of the SRFDs but also the LRFDs and the CatDRFDs. To conclude, the combination of the DRFDs and the novel ILF both promotes the performance of the learnable local descriptors.

However, to construct descriptors for key points, patch by patch will introduce enormous computational redundancy, which abates efficiency. To solve this issue, we would not directly catenate the DRFDs for correspondence establishment but construct the DFDMs that consist of the SRF-FDM and the LRF-FDM and combine the DFDMs with FAST key points instead. Therefore, we will also try to explore the feature location performance of the SRF-FDM, LRF-FDM, and the combination of DFDM (C-DFDM) under the cases when ILF is adopted or not. Rather than offering two corresponding key point sets across the source image and the target image, here, we only give the key point set of the source image and use SRF-FDM, LRF-FDM, and C-DFDM for location in the target image, respectively. The ρ–γ curves are used for evaluation, which is shown in Fig. 10. When testing SRF-FDM, LRF-FDM, and C-DFDM with ILF, we set $\omega = 1$ in (14). Moreover, for C-DFDM, there are two thresholds ϵ_0 and ϵ_1 in (29) and (31), and we set $\epsilon_0 = \epsilon_1$.

Based on Fig. 10, we have the following findings.

1) The combination of DFDMs (C-FDM) performs better than single SRF-FDM and single LRF-FDM in a key

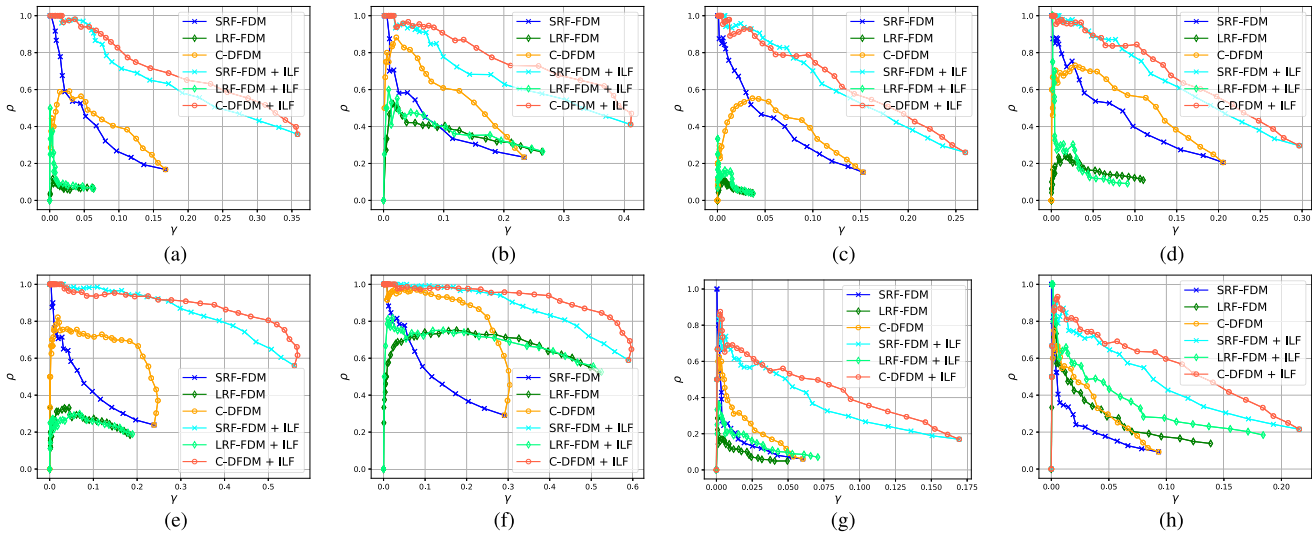

Fig. 10. ρ–γ curves for ablation studies. C-DFDM: the combination of DFDMs. "+ILF" means that the inner loss function ILF is introduced during the training process of the DRFD-Net; otherwise, the loss function is $\mathscr{L}_{\text{hn}}(Y_l^{(1)}) + \mathscr{L}_{\text{hn}}(Y_l^{(2)})$ without $\mathscr{L}_{\text{ilf}}$ in (10). For descriptors trained with ILF, ω in (14) is fixed to be 1. When testing C-DFDM, we set $\epsilon_0 = \epsilon_1$ in (29) and (31). It can be observed that there might exist a period when ρ and γ both increase for Group C, and that is because DRF-FDM can correct the errors of single SRF-FDM in a degree. (a) A($\alpha = 8$). (b) A($\alpha = 16$). (c) B($\alpha = 8$). (d) B($\alpha = 16$). (e) C($\alpha = 8$). (f) C($\alpha = 16$). (g) D($\alpha = 8$). (h) D($\alpha = 16$).

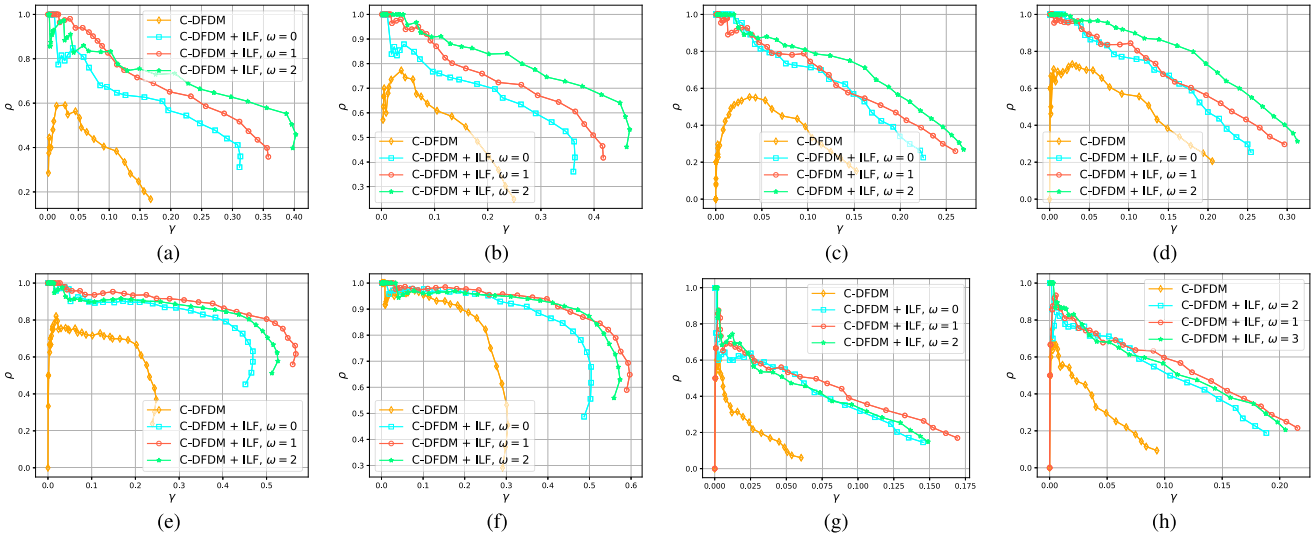

Fig. 11. ρ–γ curves to investigate the influence of hyperparameter ω in (14). "+ILF" means that the ILF is introduced during the training process of the DRFD-Net; otherwise, the loss function is $\mathscr{L}_{\text{hn}}(Y_l^{(1)}) + \mathscr{L}_{\text{hn}}(Y_l^{(2)})$ without $\mathscr{L}_{\text{ilf}}$ in (10). For C-DFDM, there are two thresholds ϵ_0 and ϵ_1 in (29) and (31), and we set $\epsilon_0 = \epsilon_1$. Cases when $\omega = 0$, $\omega = 1$, and $\omega = 2$ are tested. (a) A($\alpha = 8$). (b) A($\alpha = 16$). (c) B($\alpha = 8$). (d) B($\alpha = 16$). (e) C($\alpha = 8$). (f) C($\alpha = 16$). (g) D($\alpha = 8$). (h) D($\alpha = 16$).

point location, which is consistent with the previous experimental results of CatDRFDs. In fact, on the one hand, for the reason that DRFDs have larger receptive fields, DRF-FDM can correct the errors of a single SRF-FDM to a degree. On the other hand, due to more details kept in SRFDs and a higher sampling rate, SRF-FDM can be more accurate than LRF-FDM. Integrating both advantages, C-DFDM has a better performance.

2) For key point location results using C-DFDM, there might exist a period when the ρ and γ both increase, as shown in Fig. 10(e) and (f). This is because, with ϵ_0 increasing, the correct correspondences abandoned by SRF-FDM are picked up again by LRF-FDM.
3) With the ILF being added, the key point location results based on SRF-FDM and C-DFDM are improved remarkably. Also, the key point location results using LRF-FDM are improved in most cases with ILF. This also substantiates that the ILF proposed in this article can improve the performance of the DRFDs.
4) In most cases, LRF-FDM performs not so well as the location results of SRF-FDM when $\alpha = 8$. This is because the downsampling rate of the LRF-FDM is

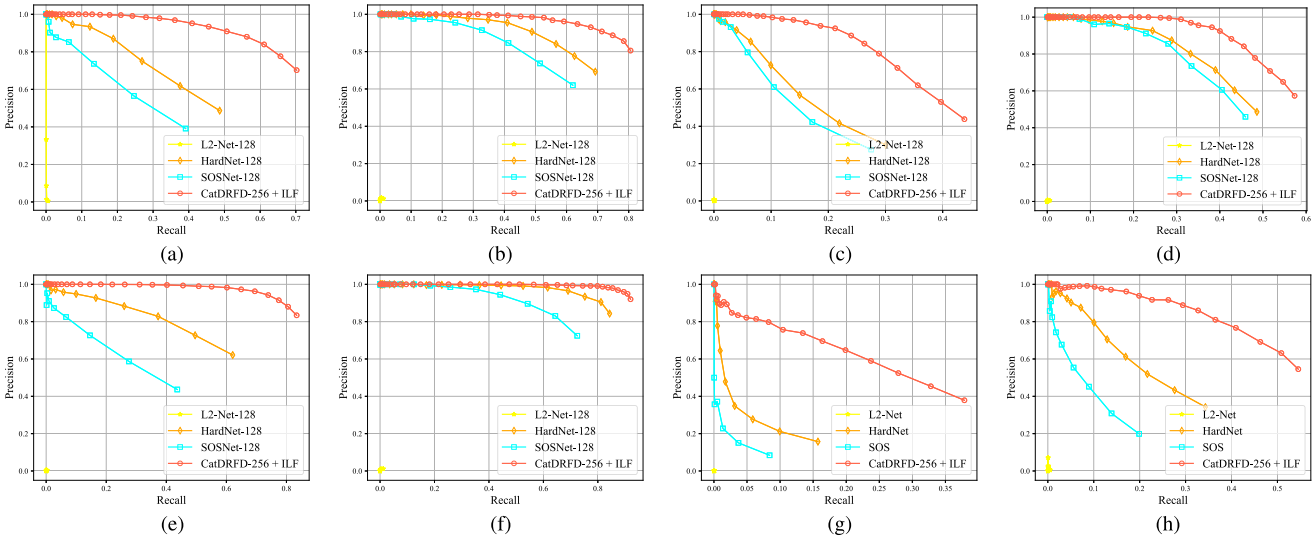

Fig. 12. PR curves for comparison with other learnable descriptors. L2-Net-128: 128-d L2-Net descriptor. HardNet-128: 128-d HardNet descriptor. SOSNet-128: 128-d SOSNet descriptor. For CatDRFD-256 trained with ILF, ω in (14) is fixed to be 1. (a) A($\alpha = 8$). (b) A($\alpha = 16$). (c) B($\alpha = 8$). (d) B($\alpha = 16$). (e) C($\alpha = 8$). (f) C($\alpha = 16$). (g) D($\alpha = 8$). (h) D($\alpha = 16$).

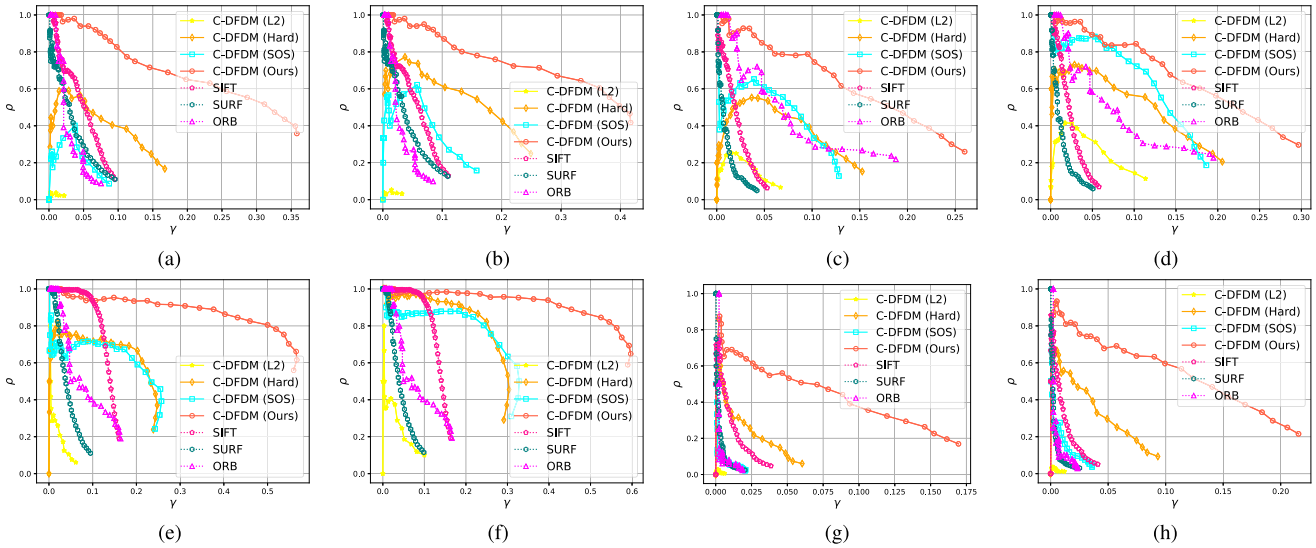

Fig. 13. ρ–γ curves for comparison of other descriptors. C-DFDM (L2): using the C-DFDM constructed via DRFD-Net trained with L2-Net loss. C-DFDM (Hard): using the C-DFDM constructed via DRFD-Net trained with HardNet loss. C-DFDM (SOS): using the C-DFDM constructed via DRFD-Net trained with SOSNet loss. C-DFDM (Ours): using the C-DFDM constructed via DRFD-Net trained with the proposed loss function, including ILF. When testing the deep learning methods, we still keep $\epsilon_0 = \epsilon_1$ in (29) and (31). (a) A($\alpha = 8$). (b) A($\alpha = 16$). (c) B($\alpha = 8$). (d) B($\alpha = 16$). (e) C($\alpha = 8$). (f) C($\alpha = 16$). (g) D($\alpha = 8$). (h) D($\alpha = 16$).

twice as the SRF-FDM and the ILF does not constrain the LRFDs directly.

In conclusion, we can infer that the CatDRFDs perform better than single SRFDs or LRFDs. The C-DFDM also has better location performance than single SRF-FDM and LRF-FDM. Besides, the ILF proposed in this article can also improve the performance of the DRFDs remarkably.

4) Hyperparameter ω: To select a proper hyperparameter ω in (14), we attempt to investigate the influence of the key point location for the C-DFDM. The ρ–γ curves are still applied for evaluation, where $\omega = 0$, $\omega = 1$, and $\omega = 2$ are tested. For ϵ_0 and ϵ_1 in (29) and (31), we still keep $\epsilon_0 = \epsilon_1$. The results are shown in Fig. 11.

Based on the experiment, we can see that the C-DFDM with $\omega = 2$ performs best in Group A and Group B, and the one with $\omega = 1$ performs best in Group C and Group D. Intuitively, when ω is smaller, the constraints of ILF are more strict, and the distinguishability of the descriptors should be better. However, experiments show that C-DFDM with $\omega = 0$ performs worse than that with $\omega = 1$ and $\omega = 2$ in most cases, which proves that a too small ω can be too much of a good thing. We believe that it is because, when ω is too small, the DRFDs will be more sensitive, and the DRFD-Net is more likely to be overfitted.

5) Comparison With Other Descriptors: For the reason that L2-Net, HardNet, and SOSNet take the same structure of

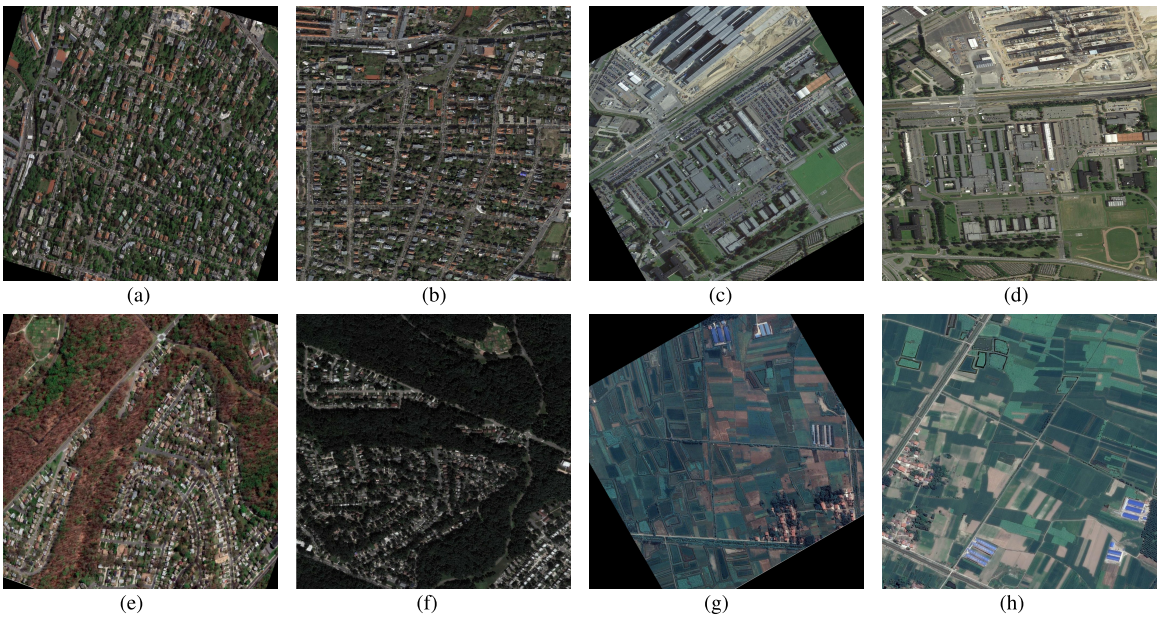

Fig. 14. Testing data for evaluation of image registration. (a) Source (I). (b) Target (I). (c) Source (II). (d) Target (II). (e) Source (III). (f) Target (III). (g) Source (IV). (h) Target (IV).

TABLE III
CORRESPONDING AP VALUES OF PR CURVES IN FIG. 12

	Group A		Group B		Group C		Group D	
	$\alpha = 8$	$\alpha = 16$	$\alpha = 8$	$\alpha = 16$	$\alpha = 8$	$\alpha = 16$	$\alpha = 8$	$\alpha = 16$
L2-Net-128	<0.01	<0.01	<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
HardNet-128	0.3767	0.6390	0.1832	0.4144	0.5196	0.8260	0.0471	0.2186
SOSNet-256	0.2546	0.5425	0.1565	0.3856	0.2869	0.6771	0.0144	0.0900
CatDRFD-256 + ILF	**0.6570**	**0.7819**	**0.3638**	**0.5277**	**0.8172**	**0.9140**	**0.2459**	**0.4674**

L2-Net and are not compatible with our dataset, we only take their loss functions and train with the DRFD-Net architecture for comparison.

First, we apply PR curves to test the distinguishability of the CatDRFDs trained with ILF and the L2-Net, HardNet, and SOSNet descriptors. The results are shown in Fig. 12 and Table III, from which we can see that CatDRFD trained with ILF outperforms other descriptors remarkably.

Practically, we also compare the key point location performance of C-DFDM trained with ILF, L2-Net loss, HardNet loss, and SOSNet loss, respectively, via ρ–γ curves. In addition, the key point location performance of traditional descriptors, such as SIFT, SURF, and SURF, is also included. The comparison results are shown in Fig. 13, based on which we have the following observations.

1) C-DFDM trained with the ILF performs better than that trained with L2-Net loss, HardNet loss, and SOSNet loss.
2) Within the certain scope of accuracy requirements, the location results based on C-DFDM are better than conventional descriptors, such as SIFT, SURF, and ORB, to a large extent, which illustrates that it is more likely for deep convolutional descriptors to extract the high-level semantic information that is more possible to keep stable in multitemporal remote sensing images.

C. Evaluation of Registration

1) RMSE: The root mean square error (RMSE) is adopted in this article to measure the accuracy of image registration. Pick up n key points in the source image, which are denoted as

$$S^{(\mathrm{src})} = \left\{ \boldsymbol{p}_1^{(\mathrm{src})}, \ldots, \boldsymbol{p}_n^{(\mathrm{src})} \right\}. \tag{54}$$

Denote the predicted transformation matrix and the true transformation matrix from the source image to the target image as $\hat{\boldsymbol{T}}$ and $\boldsymbol{T}$, respectively. Then, the RMSE can be defined as

$$\mathrm{err} = \frac{1}{n} \sum_{i=1}^{n} \left\| \boldsymbol{T}\, \boldsymbol{p}_i^{(\mathrm{src})} - \hat{\boldsymbol{T}}\, \boldsymbol{p}_i^{(\mathrm{src})} \right\|_2. \tag{55}$$

When RMSE is smaller, the image registration result is more accurate.

2) Registration Results: To test the accuracy of the registration method that we propose, we select another four groups of multitemporal optical remote sensing images. Among the testing groups, Group (I) and Group (II) are of easy mode with more similar local features, and the other two groups are of difficult mode with fewer similar local features. The testing images are all of 1024×1024, as shown in Fig. 14. The image registration results are shown in Fig. 15, where the form of

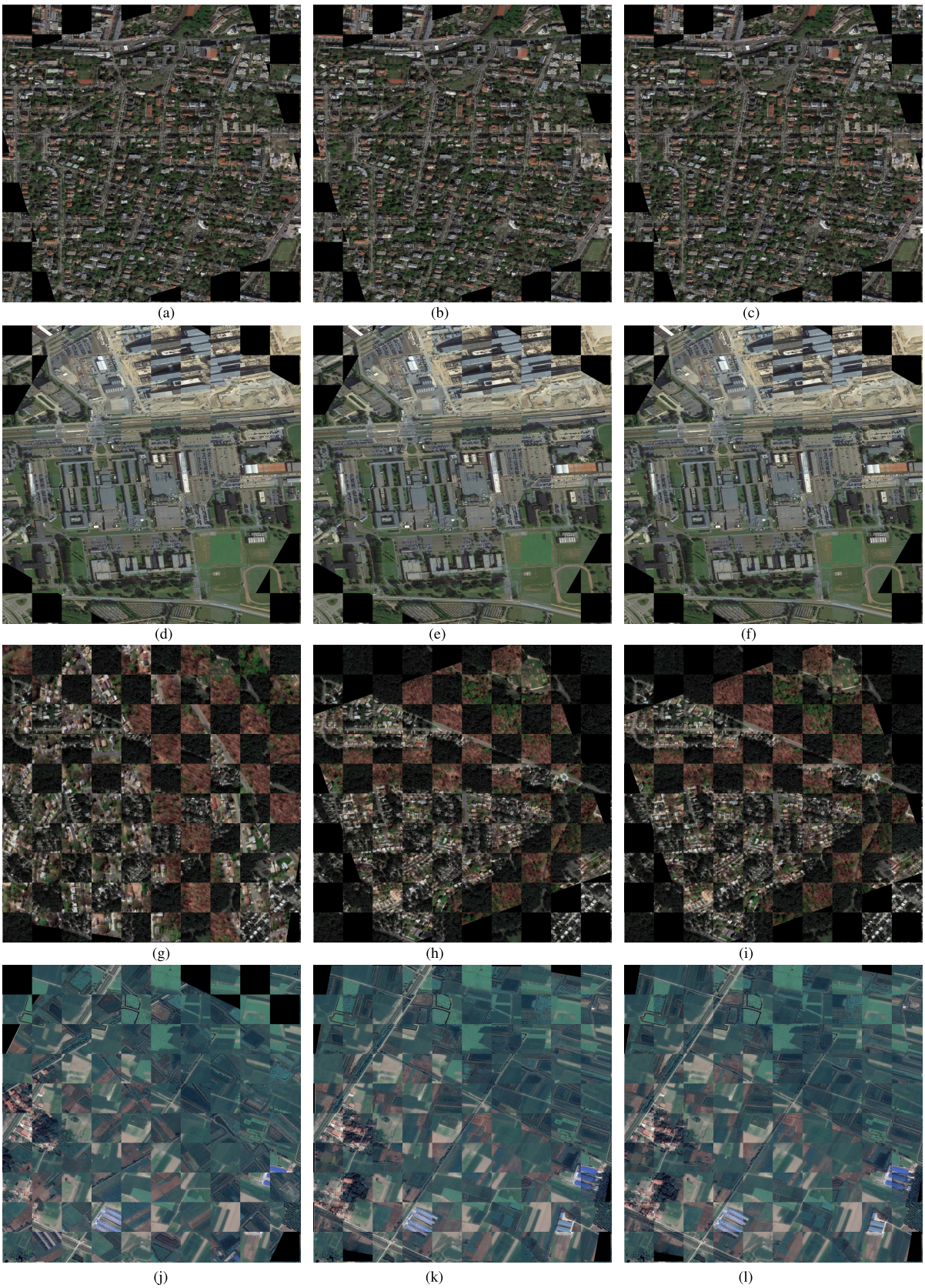

Fig. 15. Image registration results of Group I–IV using SIFT, C-DFDM based on HardNet, and C-DFDM based on the proposed ILF. It can be seen that the method using C-DFDM based on the proposed ILF has the best registration results. (a) SIFT + IIR (I). (b) C-DFDM (Hard) + IIR (I). (c) C-DFDM (Ours) + IIR (I). (d) SIFT + IIR (II). (e) C-DFDM (Hard) + IIR (II). (f) C-DFDM (Ours) + IIR (II). (g) SIFT + IIR (III). (h) C-DFDM (Hard) + IIR (III). (i) C-DFDM (Ours) + IIR (III). (j) SIFT + IIR (IV). (k) C-DFDM (Hard) + IIR (IV). (l) C-DFDM (Ours) + IIR (IV).

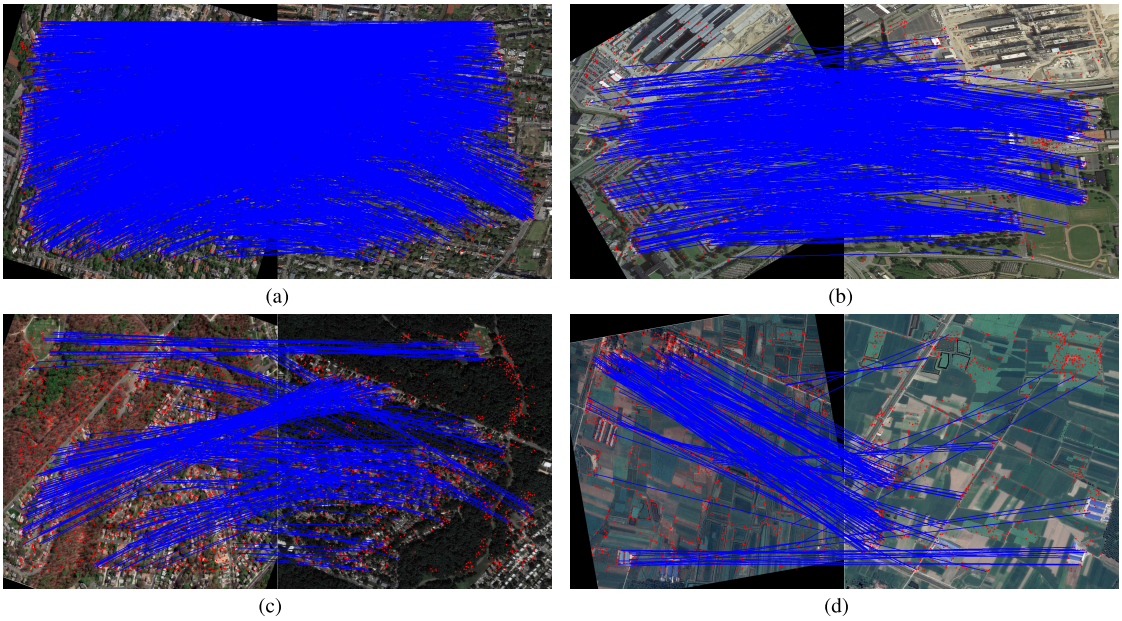

Fig. 16. Key points selected in the source image and the target image and the correctly matched correspondences when using C-DFDM trained with ILF to establish correspondences for Group I–IV. (a) C-DFDM (Ours) (I). (b) C-DFDM (Ours) (II). (c) C-DFDM (Ours) (III). (d) C-DFDM (Ours) (IV).

TABLE IV
RMSE AND RUNNING TIME (RT) FOR METHODS TESTED

	Group I		Group II		Group III		Group IV	
	RMSE	RT	RMSE	RT	RMSE	RT	RMSE	RT
SIFT + RANSAC[1]	0.8523	24.9557(s)	1.2801	9.9345(s)	N/A	N/A	N/A	N/A
SIFT + IIR[2]	0.5680	24.9108(s)	0.5307	10.5123(s)	N/A	N/A	N/A	N/A
SURF + RANSAC[1]	0.6344	9.6979(s)	1.4634	8.6061(s)	N/A	N/A	N/A	N/A
SURF + IIR[2]	0.3579	9.9295(s)	1.5817	7.4095(s)	N/A	N/A	N/A	N/A
ORB + RANSAC[1]	0.8194	0.2119(s)	1.0805	0.2775(s)	N/A	N/A	N/A	N/A
ORB + IIR[2]	1.3377	0.2186(s)	1.2331	0.2384(s)	N/A	N/A	N/A	N/A
C-DFDM (Hard)[3]+ RANSAC[1]	5.1325	12.9098	2.0263	11.1928(s)	N/A	N/A	N/A	N/A
C-DFDM (Hard)[3]+ IIR[2]	1.1865	13.7850(s)	0.5917	11.5323(s)	3.7761	12.7489(s)	5.0684	11.2306(s)
C-DFDM (SOS)[4]+ RANSAC[1]	2.9210	12.5786(s)	3.1667	13.6513(s)	N/A	N/A	N/A	N/A
C-DFDM (SOS)[4]+ IIR[2]	0.3027	13.8750(s)	0.5126	9.9452(s)	5.5744	13.5092(s)	4.8112	12.6970(s)
C-DFDM (Ours)[5]+ RANSAC[1]	0.5815	13.7359(s)	1.2336	9.1062(s)	5.2945	12.4906(s)	7.9825	10.6263(s)
C-DFDM (Ours)[5]+ IIR[2]	**0.2905**	12.7348(s)	**0.1934**	9.3414(s)	**1.1279**	13.1029(s)	**0.6913**	10.9764(s)

[1] Using the RANSAC algorithm for image registration.
[2] Using the iterative algorithm proposed for image registration.
[3] The model is trained using the HardNet loss.
[4] The model is trained using the SOSNet loss.
[5] The model is trained using the loss function proposed, including ILF.
[6] N/A means the RMSE is more than 16.

checkerboard mosaic images is adopted for intuitive visualization. In order to evaluate the image registration accuracy of our method, we compare the RMSE values and running time (RT) of our registration methods with the methods based on SIFT, SURF, ORB, and the deep learning methods, including HardNet and SOSNet, as shown in Table IV. For the reason that L2-Net performs obviously worse than HardNet and SOSNet, as shown in Figs. 12 and 13, we do not include it here. In addition, we also show the initial correspondence establishment results in Table V and Fig. 16.

From Table IV and Fig. 15, we have the following observations.

1) The proposed iterative algorithm outperforms the traditional RANSAC algorithm in removing the possible outliers for more accurate registration results in most cases.
2) The registration method based on C-DFDM trained with ILF is more accurate than that trained with HardNet loss and SOSNet loss, which demonstrates that DRFDs have better performance than HardNet descriptors and SOSNet descriptors.
3) The registration method based on C-DFDM trained with ILF performs better than methods based on handcrafted features, including SIFT, SURF, and ORB, which

TABLE V
NUMBER OF THE INITIAL CORRESPONDENCES AND THE NUMBER OF THE CORRECTLY MATCHED ONES

	Group I		Group II		Group III		Group IV	
	N_s	N_c	N_s	N_c	N_s	N_c	N_s	N_c
SIFT	386	349	273	216	21	1	48	5
SURF	191	124	257	149	46	1	87	2
ORB	211	45	227	44	194	3	223	13
C-DFDM (Hard)	5023	326	3871	725	756	22	598	21
C-DFDM (SOS)	3646	389	4932	555	2419	29	1223	31
C-DFDM (Ours)	3229	2315	1834	772	3304	262	2016	164

[1] N_s refers to the number of correspondences established.
[2] N_c refers to the number of the correctly matched correspondences.

substantiates that DRFDs have better performance than SIFT, SURF, and ORB.

4) Apart from the ORB algorithm, the time cost of the proposed method based on C-DFDM and FAST key points detected is about equal to other handcrafted methods, including SIFT and SURF.

Combining Tables IV and V, we find that the proposed method does not always has the best correctly matching rate (CMR), i.e., N_c/N_s, but has the best registration results, especially for Group III and IV that are of hard mode. Since the iterative algorithm can filter out the outliers and reserve the inliers to a degree, the crucial factor becomes whether the number of the correctly matched correspondences is enough when CMR is more than a certain value. According to Table V, we can see that the proposed method produces much more inliers than other methods and, therefore, is more accurate in image registration results.

In conclusion, the proposed method performs the best in image registration results among the methods tested. This shows that our method outperforms the methods based on learnable descriptors, such as L2-Net, HardNet, and SOSNet, as well as handcrafted descriptors, including SIFT, SURF, and ORB. Besides, the time cost using our method is about equal to methods based on SIFT and SURF, which demonstrates that our method is proper for practical application.

V. CONCLUSION

In this article, we propose the DRFD-Net to construct DRFDs and combine them for better performance. In order to further enhance the distinguishability for descriptors, we also develop the novel ILF to impose constraints on the intermediate features. Experiment shows that the catenation of DRFDs trained with ILF outperforms L2-Net, HardNet, and SOSNet descriptors remarkably. Considering the computational redundancy of patch-by-patch descriptor construction, we propose to method to construct DFDMs and combine them with FAST key points for correspondence establishment. Experiment shows that C-DFDM trained with the ILF performs better than that trained with L2-Net loss, HardNet loss, and SOSNet loss, as well as handcrafted descriptors, including SIFT, SURF, and ORB. Eventually, in order to remove possible outliers for better registration accuracy, we propose the iterative algorithm and demonstrate that it is better than classical RANSAC.

In conclusion, compared to conventional handcrafted features, such as SIFT, SURF, and ORB, our method is more likely to extract the high-level semantic information that is more possible to keep stable in multitemporal remote sensing images. While compared to other learnable methods, our method also promotes distinguishability and invariance, which relieves the defect of current learnable features. Besides, combining the C-DFDM with FAST key points, the time cost of our method is about equal to methods based on SIFT and SURF. However, our method still needs FAST key point detection to cope with the inevitable error caused by the downsampling operation (e.g., pooling) of the DRFD-Net. In further work, we will try to balance the accurate key point location and rational convolutional feature extraction.

REFERENCES

[1] Y. You, J. Cao, and W. Zhou, "A survey of change detection methods based on remote sensing images for multi-source and multi-objective scenarios," *Remote Sens.*, vol. 12, no. 15, p. 2460, Jul. 2020.
[2] W. Dong, Y. Yang, J. Qu, W. Xie, and Y. Li, "Fusion of hyperspectral and panchromatic images using generative adversarial network and image segmentation," *IEEE Trans. Geosci. Remote Sens.*, early access, May 21, 2021, doi: 10.1109/TGRS.2021.3078711.
[3] J. Yao, D. Meng, Q. Zhao, W. Cao, and Z. Xu, "Nonconvex-sparsity and nonlocal-smoothness-based blind hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2991–3006, Jun. 2019.
[4] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1923–1938, Apr. 2019.
[5] D. Hong *et al.*, "Interpretable hyperspectral artificial intelligence: When nonconvex modeling meets hyperspectral remote sensing," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 2, pp. 52–87, Jun. 2021.
[6] J. P. Kern and M. S. Pattichis, "Robust multispectral image registration using mutual-information models," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 5, pp. 1494–1505, May 2007.
[7] L. Pallotta, G. Giunta, and C. Clemente, "Subpixel SAR image registration through parabolic interpolation of the 2-D cross correlation," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4132–4144, Jun. 2020.
[8] Y. Dong, T. Long, W. Jiao, G. He, and Z. Zhang, "A novel image registration method based on phase correlation using low-rank matrix factorization with mixture of Gaussian," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 446–460, Jan. 2018.
[9] Y. Xiang, R. Tao, L. Wan, F. Wang, and H. You, "OS-PC: Combining feature representation and 3-D phase correlation for subpixel optical and SAR image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 9, pp. 6451–6466, Mar. 2020.
[10] D. I. Barnea and H. F. Silverman, "A class of algorithms for fast digital image registration," *IEEE Trans. Comput.*, vol. C-21, no. 2, pp. 179–186, Feb. 1972.
[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
[12] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.
[13] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

[14] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "KAZE features," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 214–227.
[15] R. Feng, Q. Du, X. Li, and H. Shen, "Robust registration for remote sensing images by combining and localizing feature- and area-based methods," *ISPRS J. Photogramm. Remote Sens.*, vol. 151, pp. 15–26, May 2019.
[16] R. Feng, Q. Du, H. Shen, and X. Li, "Region-by-region registration combining feature-based and optical flow methods for remote sensing images," *Remote Sens.*, vol. 13, no. 8, p. 1475, Apr. 2021.
[17] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2021.
[18] D. Hong *et al.*, "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, May 2021.
[19] Y. You, B. Ran, G. Meng, Z. Li, F. Liu, and Z. Li, "OPD-Net: Prow detection based on feature enhancement and improved regression model in optical remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 6121–6137, Jul. 2021.
[20] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[21] S. Miao, Z. J. Wang, Y. Zheng, and R. Liao, "Real-time 2D/3D registration via CNN regression," in *Proc. Int. Symp. Biomed. Imag.*, 2016, pp. 1430–1434.
[22] J.-H. Park, W.-J. Nam, and S.-W. Lee, "A two-stream symmetric network with bidirectional ensemble for aerial image matching," *Remote Sens.*, vol. 12, no. 3, p. 465, Feb. 2020.
[23] J. Ma, H. Zhou, J. Zhao, Y. Gao, J. Jiang, and J. Tian, "Robust feature matching for remote sensing image registration via locally linear transforming," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6469–6481, Dec. 2015.
[24] J. Ma, J. Zhao, J. Jiang, H. Zhou, and X. Guo, "Locality preserving matching," *Int. J. Comput. Vis.*, vol. 127, no. 5, pp. 512–531, 2019.
[25] J. Ma, J. Jiang, H. Zhou, J. Zhao, and X. Guo, "Guided locality preserving feature matching for remote sensing image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4435–4447, Aug. 2018.
[26] S. Chen, S. Zhong, B. Xue, X. Li, L. Zhao, and C.-I. Chang, "Iterative scale-invariant feature transform for remote sensing image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3244–3265, Apr. 2021.
[27] X. Jiang *et al.*, "Robust feature matching for remote sensing image registration via linear adaptive filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1577–1591, Feb. 2021.
[28] X. Han, T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg, "MatchNet: Unifying feature and metric learning for patch-based matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3279–3286.
[29] V. Balntas, E. Johns, L. Tang, and K. Mikolajczyk, "PN-Net: Conjoined triple deep network for learning local image descriptors," 2016, *arXiv:1601.05030*. [Online]. Available: http://arxiv.org/abs/1601.05030
[30] Y. Tian, B. Fan, and F. Wu, "L2-Net: Deep learning of discriminative patch descriptor in Euclidean space," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 661–669.
[31] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4829–4840.
[32] Y. Tian, X. Yu, B. Fan, F. Wu, H. Heijnen, and V. Balntas, "SOS-Net: Second order similarity regularization for local descriptor learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 11016–11025.
[33] M. Brown, G. Hua, and S. Winder, "Discriminative learning of local image descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 43–57, Jan. 2011.
[34] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk, "HPatches: A benchmark and evaluation of handcrafted and learned local descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5173–5182.
[35] J. L. Schonberger, H. Hardmeier, T. Sattler, and M. Pollefeys, "Comparative evaluation of hand-crafted and learned local features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1482–1491.
[36] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 430–443.
[37] X. Wu, D. Hong, J. Chanussot, Y. Xu, R. Tao, and Y. Wang, "Fourier-based rotation-invariant feature boosting: An efficient framework for geospatial object detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 302–306, Feb. 2020.
[38] X. Wu, D. Hong, J. Tian, J. Chanussot, W. Li, and R. Tao, "ORSIm detector: A novel object detection framework in optical remote sensing imagery using spatial-frequency channel features," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5146–5158, Jul. 2019.
[39] M. Sotoodeh, M. R. Moosavi, and R. Boostani, "A novel adaptive LBP-based descriptor for color image retrieval," *Expert Syst. Appl.*, vol. 127, pp. 342–352, Aug. 2019.
[40] Z. Kuang, J. Yu, S. Zhu, Z. Li, and J. Fan, "Effective 3-D shape retrieval by integrating traditional descriptors and pointwise convolution," *IEEE Trans. Multimedia*, vol. 21, no. 12, pp. 3164–3177, Dec. 2019.
[41] A. Sedaghat and H. Ebadi, "Remote sensing image matching based on adaptive binning SIFT descriptor," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 10, pp. 5283–5293, Oct. 2015.
[42] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2005, pp. 886–893.
[43] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Roy. Soc. London B, Biol. Sci.*, vol. 207, no. 1167, pp. 187–217, 1980.
[44] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 778–792.
[45] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.
[46] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.
[47] L. Bottou, "Stochastic gradient descent tricks," in *Neural Networks: Tricks of the Trade*. Berlin, Germany: Springer, 2012, pp. 421–436.
[48] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

Yanan You (Member, IEEE) received the Ph.D. degree from the School of Electronic and Information Engineering, Beihang University, Beijing, China, in 2015.

He held a post-doctoral position at Beihang University from 2015 to 2017. Since September 2017, he has been a Lecturer with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing. His research interests are in remote sensing image processing, synthetic aperture radar (SAR) interferometry processing, deep learning, big data technology, and so on.

Chao Li received the B.E. degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2020, where he is currently pursuing the M.S. degree with the School of Artificial Intelligence.

His research interests are in remote sensing image processing, especially in image registration and local descriptor construction.

Wenli Zhou received the Ph.D. degree in engineering in signal and information processing from Beijing University of Posts and Telecommunications, Beijing, China, in 2006.

She is currently an Associate Professor with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. Her research interests include network traffic monitoring, user behavior analysis, telecommunications and Internet big data processing, and other research.