ISPRS Journal of Photogrammetry and Remote Sensing 181 (2021) 205–217

Contents lists available at ScienceDirect

ISPRS Journal of Photogrammetry and Remote Sensing

journal homepage: www.elsevier.com/locate/isprsjprs

# A deep learning semantic template matching framework for remote sensing image registration

Liangzhi Li [a,*], Ling Han [b,c], Mingtao Ding [a], Hongye Cao [a], Huijuan Hu [a]

[a] College of Geological Engineering and Geomatics, Chang'an University, Xi'an 710064, China
[b] School of Land Engineering, Chang'an University, Xi'an 710064, China
[c] Shaanxi Key Laboratory of Land Consolidation, Xi'an 710064, China

ARTICLE INFO

*Keywords:*
Registration
Deep learning
Semantic template
Semantic distribution probability
Remote sensing image
CNN

ABSTRACT

We propose a deep learning framework by the probability of the predicting semantic spatial position distribution for remote sensing image registration. Traditional matching methods optimize similarity metrics with pixel-by-pixel searching, which is time consuming and sensitive to radiometric differences. Driven by learning-based methods, we take the reference and template images as inputs and map them to the semantic distribution position of the corresponding reference image. We apply the affine invariant to obtain a correspondence between the overlap of the barycenter position of the semantic template and the center pixel point, which transforms the semantic boundary alignment into a point-to-point matching problem. Additionally, two loss functions are proposed, one for optimizing the subpixel matching position and the other for determining the semantic spatial probability distribution of the matching template. In this work, we explore the influence of the template radius size, the filling form of training labels, and the weighted combination of loss function on the matching accuracy. Our experiments show that the trained model is robust to template deformation while still operating orders of magnitude faster. Furthermore, our proposed method implements high matching accuracy in four large scene images with radiometric differences. This ensures the improved speed of remote sensing image analysis and pipeline processing while facilitating novel directions in learning-based registration. Our code is freely available at https://github.com/liliangzhi110/semantictemplatematching.

## 1. Introduction

Different satellite sensors, such as optical and Synthetic Aperture Radar (SAR), can provide multiple remote sensing images. These multimodal images have highly complementary information, and therefore, there is a need to integrate these images to form a comprehensive representation of the observed scenes. Image registration is a basic task in remote sensing image processing. Multimodal remote sensing images are used for fusion, and strict matching must be implemented before spatial analysis and processing (Jiang et al., 2020). Although the automatic registration method of remote sensing images has endured considerable progress over the past few decades, matching between multimodal images is still difficult due to the influence of nonlinear radiation differences (Ryu et al., 2017).

At present, remote sensing images are registered using orbit parameters and positioning models (Wang et al., 2017), which eliminate almost all global geometric deformations, such as obvious rotation and scale differences. Since optical images usually contain significant geolocation errors, we cannot rely on this geocoding method to provide an accurate correspondence. Therefore, we need to perform an image matching process that is constrained by the large geometric and radiometric differences between multimodal remote sensing images. Optical and SAR images of the same scene have different intensities and texture features, making matching between the two images still difficult (Xiang et al., 2018). To this end, this paper proposes a semantic template matching method that is robust to the nonlinear radiation differences between multimodal images.

Remote sensing image registration methods can be roughly classified into two categories: feature-based methods and area-based methods. Feature-based methods first extract the image features, including point features (Moravec operator, Harris operator), line features (edge and contour), and surface features, and then use the similarity metric between the features for matching (Yang et al., 2017). The representative feature-based method is the scale-invariant feature transform (SIFT)

* Corresponding author.
*E-mail address:* liliangzhi@chd.edu.cn (L. Li).

https://doi.org/10.1016/j.isprsjprs.2021.09.012
Received 3 January 2021; Received in revised form 17 June 2021; Accepted 10 September 2021
Available online 24 September 2021

(Lowe, 2004). SIFT is invariant to translation, rotation, and scale change and has been widely used in image registration. However, the feature points extracted by the SIFT designed for normal images may not be retained on the remote sensing image (Wang et al., 2018), owing to the complex imaging mechanism of multimodal remote sensing images. Researchers have proposed new local invariant features based on SIFT, such as PSO-SIFT (Ma et al., 2016) and Affine-SIFT (Cai et al., 2013). However, the abovementioned feature-based matching method is greatly affected by the gray differences between images, and thus, it fails to reflect the common features between multimodal images (Xu et al., 2016). Furthermore, those methods lead to severe misregistration due to the insufficiency of extraction from multimodal images at corresponding positions.

Area-based methods use image similarity metrics to search matching information from the entire image or intercepted images. Compared with feature-based methods, area-based methods have the following advantages: (1) area-based methods avoid the step of feature detection and directly search for overlapping patches with the greatest feature similarity (Ye et al., 2017); (2) area-based methods allow the searching of the initial geographic position within remote sensing images over a small area for geo-corrected images with a position offset of only a few pixels. Common similarity metrics include the sum of squared differences (SSD) (Hisham et al., 2015), the normalized cross-correlation (NCC) (Kern and Pattichis, 2007; Suri and Reinartz, 2009), and the mutual information (MI) (Maes et al., 1997). SSD is the simplest similarity metric that works by calculating the intensity difference between two images. Although SSD has high computational efficiency, it is sensitive to large radiometric differences. NCC is widely used in remote sensing image matching due to its invariance on linear radiation differences. However, NCC is vulnerable to nonlinear radiometric differences in images. For this reason, neither of the above two similarity metrics can effectively handle remote sensing images with nonlinear radiation differences, lighting variations, and noise sources. MI is widely used for multimodal remote sensing image registration because of its greater robustness to complex radiation variations (Chen et al., 2018b). However, MI is sensitive to the size of the matching template window and has a high computational cost, which limits its application in remote sensing image registration.

In recent years, deep learning methods have achieved great success in many aspects (LeCun et al., 2015). The major reason is that deep learning uses multiple levels of nonlinear operations to abstract features from a large number of sample data, which is a completely data-driven solution. Additionally, its mechanism can optimize the entire network end-to-end through information feedback. Currently, many deep learning methods have been proposed, especially for semantic segmentation. These methods have been successfully applied in land cover classification using remote sensing images (Yuan et al., 2018). Inspired by the deep learning semantic segmentation model structure, we propose a semantic spatial distribution framework for remote sensing image registration. By taking patches in sensed and reference images as inputs, the framework is able to learn the semantic spatial position probability relationship of sensed images in the reference image. Different from the conventional method based on feature extraction and feature matching, our proposed method directly obtains the corresponding pixel matching position.

However, our experiments show that the boundaries predicted by the semantic spatial distribution framework are blurred, and thus, it is not able to determine exactly where the template image is located. For instance, Fig. 1 illustrates this problem, where Fig. 1(a), (b) are the input data, Fig. 1(c) is the training label and Fig. 1(d) represents the predicted semantic spatial position distribution probability. Fig. 1(d) fails to determine which probability value belongs to which pixel due to the presence of a rough boundary and an irregular heatmap shape.

To this end, we introduce two loss functions based on the template barycenter affine invariant to optimize the network in terms of both point matching and semantic spatial location probability. Additionally,

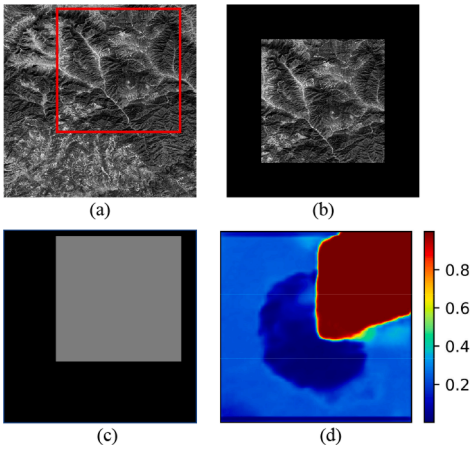


**Fig. 1.** (a and b) Show that the reference image and matching template patch cropped from a sensed image as input to the network. (c) and (d) are the training labels as ground truth matching values and the semantic template matching results obtained by using the deep learning semantic segmentation model in the experiment, respectively.

deep learning methods can approach complex functions and optimize a large number of abstract parameters. To train our correspondence network, we require a large dataset and template patches with known coordinate transformation relations. For such large-scale remote sensing data, manual production is time consuming and laborious, introducing uncertain factors. In addition, using traditional methods to obtain matching samples may introduce bias. Thus, we warp one of the two pixel-level aligned images using a randomly generated affine transformation matrix, where the warped image and the other image are used as matching image pairs, and the geometric transformation matrix is used as the ground truth for matching. The detailed process is described in Section 3.6.

The main contributions of this work are developed from three aspects:

(1) This paper uses a deep neural network to map the distribution probability of the template image pixel, achieving higher registration accuracy.
(2) We propose a loss function based on the constant barycenter position of matching templates after affine transformation, transforming the boundary alignment into a point-to-point matching problem.
(3) We explore the effects of the filling form of the training labels, the size of template images, and the setting of loss function weights on the matching accuracy.

The paper is organized as follows. Section 2 introduces the related work of traditional and deep learning image registration. Section 3 describes our correspondence network for remote sensing image matching. Section 4 shows extensive experiments and analyses of our proposed method. We conclude in Section 5.

## 2. Related work

In this section, we review image registration methods for the aforementioned two categories and deep learning registration methods: (1) area-based methods; (2) feature-based methods; and (3) supervised and unsupervised deep learning image registration methods.

### 2.1. Area-based image registration

Area-based methods are the earliest approaches used for matching the remote sensing image, processing the intensity value of the image

(Feng et al., 2019), and searching for the best matching similarity in the window template. These methods can be roughly divided into three categories: correlation methods, Fourier methods, and MI methods (Karthick and Maniraj, 2019). Correlation methods implement the registration by calculating the similarity of two images or template windows, in which NCC is an early representative in area-based methods. However, NCC is not robust for multimodal image registration in areas lacking texture (Li et al., 2018b). Another idea is to find the best parameters in the frequency domains. For instance, Reddy and Chatterji (1996) proposed a fast Fourier transform-based method to find the best matching in the frequency domain. This method has advantages in efficiency and robustness to noise in the frequency domain. Building on these approaches, De Castro and Morandi (1987) proposed some extended phase correlation (EPC) methods to solve the rotation and scaling factors. Due to the radiometric difference between the images, the correlation between the phases will gradually decrease. In Tong et al. (2015), a new EPC matching method was proposed in the frequency domain, using singular value segmentation and uniform random sampling consistency, thereby achieving affine transformation. However, it does not focus on multimodal images with obvious radiometric differences. Taking a different approach, Ye et al. (2017, 2016) proposed automatic registration of remote sensing images (ARRSI) and orientation phase consistency histograms (HOPC) of remote sensing images. These methods use the NCC-based phase consistency similarity measure in the template window. ARRSI uses patches based on phase coherence moments as descriptors, which are invariant to the matching process of intensity mapping (Wong and Clausi, 2007). HOPC captures the internal structural feature descriptors of the images, allowing the NCC framework to be used to measure two images with nonlinear radiometric differences for matching. However, the prerequisite for using HOPC is to eliminate rotation and scale differences, and its success is limited to imagery that obeys specific geometric and radiometric constraints (Ye and Shen, 2016).

### 2.2. Feature-based image registration

Different from area-based methods, feature-based methods mainly extract salient features instead of all gray information, making them robust to geometric deformation. Many works have proposed a variety of remote sensing image registration methods based on SIFT (Paul and Pati, 2016; Tareen and Saleem, 2018; Sedaghat and Mohammadi, 2018). However, with the development of deep learning, many studies have begun to use deep neural networks to extract abstract features for description, which compensates for the inability to guarantee the rotation and scale invariance of feature points in remote sensing image matching due to radiometric differences.

This section focuses on feature-based methods for remote sensing image registration related to deep learning. Ye et al. (2018) proposed a combination function of convolutional neural network (CNN) features and SIFT features, and integrated the combined features into the registration algorithm, achieving better registration performance. Ma et al. (2016) introduced a two-part registration method from ranging coarse to fine, as well as the strategy of combining depth and local features, which increased the correct correspondence. Yang et al. (2018) introduced a multitemporal remote sensing image registration method based on CNN features, using CNNs to generate robust multiscale feature descriptors. Wang et al. (2018) developed a deep neural network that directly learned the mapping between input pairs and matching labels for remote sensing image matching with an end-to-end network optimization strategy. Han et al. (2015) proposed a unified feature learning and matching method (MatchNet), which used the fewest descriptors to obtain better results. MatchNet was composed of a deep CNN layer for extracting features and a fully connected layer network for metric feature description, which improved the matching accuracy and reduced the storage requirement for descriptors.

The abovementioned feature description method based on deep neural networks for extracting points and image blocks is optimized iteratively with traditional methods used for matching objective functions. Although we have seen substantial progress in matching deep abstract features, these methods rely on the selection of good feature points to extract matchable candidate searches and template patches. Given the large differences between multimodal images, under normal circumstances, no prominent features are seen in either domain. Additionally, these methods also require manual intervention, resulting in the inability to take advantage of deep neural network structures for end-to-end matching optimization.

### 2.3. Supervised and unsupervised deep learning image registration

We briefly divide the existing deep learning-based image registration methods into two categories: supervised and unsupervised deep learning methods (Haskins et al., 2020). The supervised deep learning method uses a deep neural network to regress the geometric transformation parameters from the reference and sensed images (Haskins et al., 2020). Miao et al. (2016a,b) transformed the registration problem to a regression problem using supervised methods to learn parameters (affine transformation matrix or matching point displacement) for image registration. DeTone et al. (2016) proposed a VGG-style projection transformation parameter regression model for the evaluation of homography between natural images. DeTone et al. (2018) proposed a deep learning network for feature point detection and feature description (SuperPoint), which obtained a richer feature point set compared with traditional methods. However, since the position of the interest point is uncertain, it must be obtained by simulation, making the interest point learned by SuperPoint inapplicable to real scenarios. In particular, it is difficult for a supervised learning method based on a regression network to optimize rotation and scale terms of different dimensions synchronously, which limits the accuracy of supervised learning registration.

Unsupervised deep learning matching methods use a spatial transformation network (Jaderberg et al., 2015) to warp the sensed image to the coordinates of the reference image, which is trained without any human annotations. These methods optimize the geometric constraints between two images by the similarity loss function. Balakrishnan et al. (2019) proposed an unsupervised image registration framework based on a CNN, which formulated the mapping between registration pairs and deformation fields and distorted the sensed image by a spatial transformation network, in which the target loss function was the gray similarity between the distorted image and the reference image. Balakrishnan et al. (2018) used a CNN to quickly model the variability registration algorithm, which achieved a faster matching operation without supervised information training. Dalca et al. (2018) proposed a probabilistic model and derived an unsupervised learning inference algorithm based on the latest development of CNNs. de Vos et al. (2019) designed flexible ConvNets for affine image registration and deformable image registration to stack multiple ConvNets into a larger architecture, achieving image registration from coarse to fine.

The dual-supervised learning matching method not only guides the regression parameters but also optimizes the similarity loss between the sensed and reference images. Fan et al. (2019) designed a fully convolutional network to guide the training in two ways simultaneously, which reduced the reliance on ground truth through a hierarchical loss and multisource strategy. Although unsupervised and supervised learning methods apply different strategies for accurate registration at different levels, these methods are limited by the input window size of the network, rendering the optimization of matching images limited by the input window. As unsupervised and supervised learning methods rely on a small region of support, they cannot deal with large remote sensing images with geometric deformation. Theoretically, increasing the size of the network window can address large-scale remote sensing image matching; however, additional complexity and network training difficulty are introduced, which limits the application of the network in

registration.

## 3. The proposed method

Fig. 2 presents an overview of our method. Let $r$ be the reference image for remote sensing image matching and $s$ be the template image cropped from the sensed image. The template patch is placed in the middle of $s$, and other positions are filled with 0.0 values. For simplicity, we assume that $r$ and $s$ contain single-channel grayscale image data. We also assume that the template patch is not strictly aligned with a patch in $r$, and the relationship between them is an affine transformation, which is a situation encountered in practice.

We use a deep neural network to model the function $P = N_\theta(r, s)$, where $\theta$ is the network parameter. Semantic corresponding positions (spatial distribution probability) between $r$ and $s$ are stored in $P$, which are predicted values. As shown in Fig. 2, $r$ and $s$ are input from different channels and then superimposed on the same channel after multilayer network feature extraction. The trained parameter $\theta$ is used to calculate $P$. We take the semantic position corresponding to the template patch as the truth value ($T$) and calculate the loss with $P$ ($L_1$). $P$ fails to determine the specific matching coordinate due to its fuzzy boundary and irregular shape. Thus, we calculate the barycenter position $(x, y)$ of $P$ as the matching position(the affine invariance of the barycenter) and optimize the loss ($L_2$) with the barycenter point $(x, y)$ of $T$.

Next, we introduce the deep neural network architecture, loss function, selection of the semantic matching template, sample data generation, training and matching method in detail.

### 3.1. Deep learning network architecture

In this section, we describe our CNN architecture used in the experiment, but it should be noted that various semantic deep learning networks may work well. The parameterization of $N_\theta(r, s)$ is based on a semantic segmentation network, which is similar to DeepLabv3 plus (Chen et al., 2018a). Reference and template images are input into their respective channels. As shown in Fig. 3, the network inputs $r$ and $s$ into different channels, followed by multiple convolutions. Subsequently, the generated feature maps are connected to multiple channels as a single input to the encoding network. For the encoding network, we use ResNet-50 (Boroumand et al., 2018) that has been trained and adjusted to output, where Res_1 and Res_2 generate feature maps as $R_1$ and $R_2$, respectively. In the decoding stage, we input the feature maps $R_1$ and $R_2$ into two different branching structures, which propagate them to the layers generating the registration. The branch I architecture consists of atrous spatial pyramid pooling (ASPP) (Chen et al., 2017a), $1 \times 1$ convolution, and upsampling layers, where each convolutional layer is a sequence of convolutions, activated by rectified linear units (ReLU) and batch normalization (BN). The branch II architecture fuses the feature maps generated by branch I, followed by an upsampling layer implemented recovering the original image dimensions. Next, we introduce the adjusted ResNet-50 and ASPP modules.

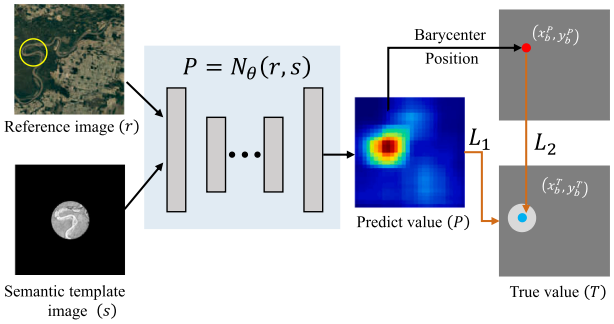


**Fig. 2.** The overall structure of our proposed method. We learn parameters $\theta$ for a function $P = N_\theta(r, s)$, where $r$ and $s$ represent the input reference image and template image cropped from the sensed image, respectively. $P$ represents the semantic spatial position distribution probability predicted by the network. $T$ is the truth value, that is, the training label. $(x_b^P, y_b^P)$ and $(x_b^T, y_b^T)$ are the barycenter positions of $P$ and $T$, respectively. $L_1$ and $L_2$ represent two loss functions: pixel-level barycenter position loss and semantic spatial probability distribution loss of the semantic matching template.

### 3.2. ResNet-50

The residual network has skip connections in each block, which is easier to train and optimize, especially for very deep neural networks. Since the residual network has achieved outstanding performance in feature extraction, we use ResNet-50 as the main structure of our network. The dual-channel architecture replaces the ResNet-50 single-channel architecture intending to apply this mapping reference and template image features in a channel-separated strategy while improving the feature extraction ability. Fig. 3 shows the structure of the adjusted network. In the original residual block, instead of using 64 as the number of final generated feature maps, we use 128, as in Res_1. Additionally, experimental results show that changing the number of feature maps in Res_2 to 256 can further improve the network matching accuracy.

### 3.3. Atrous spatial pyramid pooling

ASPP is a model proposed in DeepLab v3 (Chen et al., 2017b) to expand the receptive field, which is a stacking combination of dilated convolutions with different rates. Fig. 4 depicts the difference between dilated convolution operations and original convolution operations, with a discontinuous, where Fig. 4(b) is the dilated convolution with rate $= 2$. Alternatively, the original convolution is the dilated convolution with rate $= 1$, as shown in Fig. 4(a). Successive dilated convolutions using the same rate will result in including discontinuous pixels in the operation. Although the receptive field is enlarged, the continuity of information is also lost. Therefore, ASPP is composed of dilated convolutions with different rates, i.e., rate $= 1, 6, 12, 18$, and max-pool. The details are shown in Fig. 5.

### 3.4. Loss function

Fig. 6 shows the affine invariant property of the geometric template on the barycenter position, which we use to transform the semantic spatial position alignment problem into a point-to-point matching. Building on this, we propose two loss functions: one is used to determine the barycenter position loss $L_b$ of the matching template; the other is used to predict probability distribution loss $L_m$ for overall semantic spatial positions, assisting in determining the approximate semantic position. This combined optimization is performed using two different but not independent loss functions, each of which produces a loss value indicating the likelihood and matching accuracy of a template being matchable.

**The semantic position weighted average loss:** $L_b$ is composed of two parts, penalizing the loss of coordinates $x$ and $y$. The network architecture with $L_m$ cannot determine the true template location due to the fuzzy matching boundary of the network prediction, as shown in Fig. 1(d). Therefore, we calculate the semantic position weighted average (the barycenter position of prediction results) as the matching point of the template. The detailed formula is as follows:

$$M_{ij} = \sum_I \sum_J V(i, j) \tag{1}$$

where $V(i, j)$ is the semantic spatial position probability value of point $(i, j)$ and $M_{ij}$ indicates the sum of the semantic spatial position probability value of all points $(i, j)$.

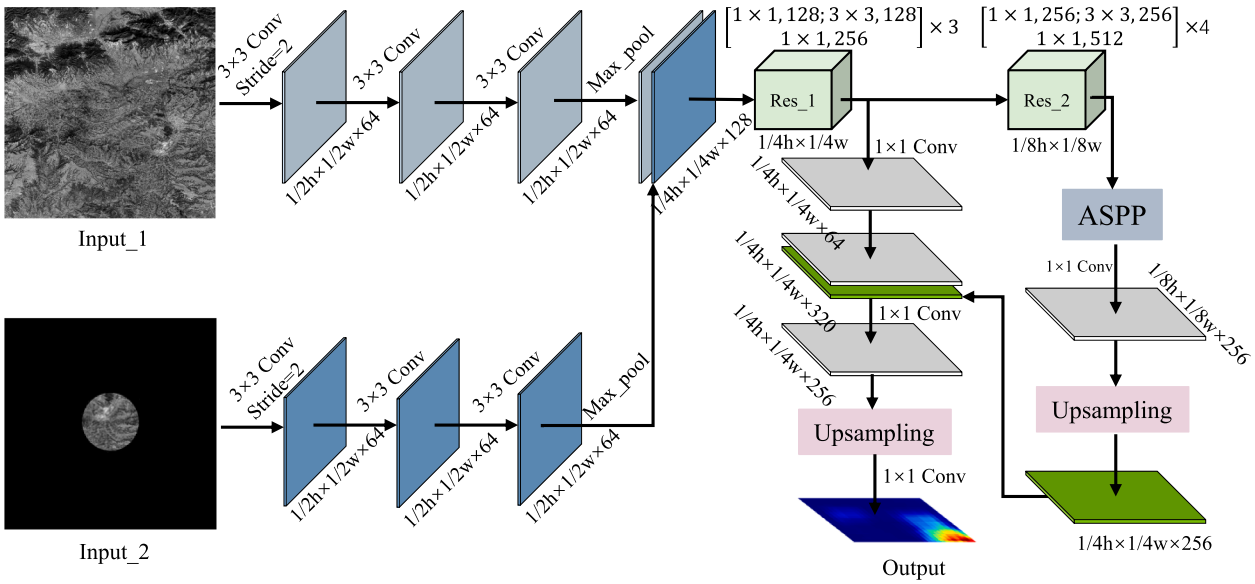


**Fig. 3.** The overall structure of the network. The reference and sensed images are input from two channels, where ASPP is the abbreviation of atrous spatial pyramid pooling, and Res_1, Res_2 are the residual blocks of ResNet-50.

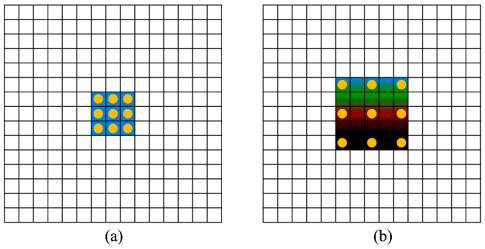


**Fig. 4.** Illustration of the receptive field of 3 × 3 dilated convolutional kernels. (a) The dilated convolution with rate = 1. (b) The dilated convolution with rate = 2.

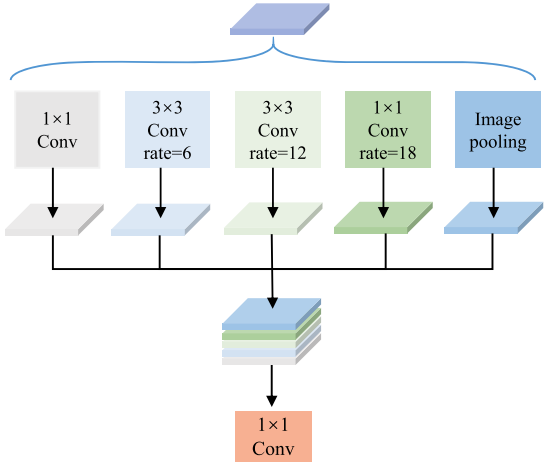


**Fig. 5.** ASPP structure, including multiple parallel dilated convolution layers with different rates, i.e., rate = 1, 6, 12, 18, and max-pool.

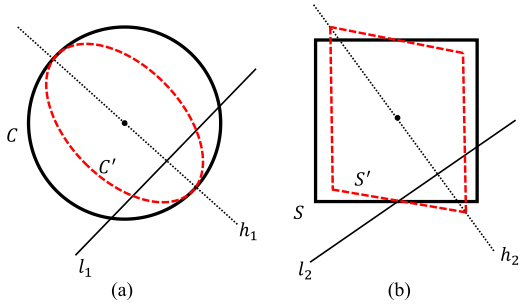


**Fig. 6.** Affine invariance of the barycenter. (a and b) Are the circular matching template and the square matching template, where the template $C$ is affine transformed to obtain $C'$, $l$ is the stretching direction and $h$ is the vertical line, while their barycenter and center pixels are still coincident. We use this invariance to compute the weighted average of the predicted semantic spatial distribution probabilities (barycenter position) indicating the matching position of the template center pixel. Similarly, the square matching template (b) has the same property.

$$M_i = \sum_I \sum_J i \cdot V(i,j) \tag{2}$$

$$M_j = \sum_I \sum_J j \cdot V(i,j) \tag{3}$$

where $M_i$ and $M_j$ represent the weights of all output positions in $i$ and $j$ coordinates, respectively. Then, the weighted averages of predicted results in $i,j$ coordinates are calculated using $M_{ij}, M_i$ and $M_j$, to obtain barycenter position coordinates. Thus, the formula can be expressed as:

$$x_b = \frac{M_i}{M_{ij}}, y_b = \frac{M_j}{M_{ij}} \tag{4}$$

The $L_b$ loss function is calculated as follows:

$$L_b = (x_b - x_{true})^2 + (y_b - y_{true})^2 \tag{5}$$

where $x_{true}, y_{true}$ denote the true coordinate position of the template barycenter in the reference image.

**The semantic spatial probability distribution loss:** $L_m$ is used to determine the probability distribution of templates with the aim of using

this distribution to obtain the likelihood of a template position. $L_m$ is composed of the cross entropy loss function and the mean square error. The overall $L_m$ is written as:

$$loss_1 = -\sum_{i=1}^{n} \widehat{y}_i \log y_i + (1-\widehat{y}_i)\log(1-\widehat{y}_i) \tag{6}$$

$$loss_2 = \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2 \tag{7}$$

$$L_m = loss_1 + loss_2 \tag{8}$$

where $y_i$ and $\widehat{y}_i$ represent true values and predicted values respectively. The loss is 0.0 only when $y_i$ and $\widehat{y}_i$ are equal.

In the training process, we apply different loss weights to $L_b, L_m$, as they operate on template positions independently. Furthermore, in training the network for registration in this way, we set a combination of weights for $L_b, L_m$, including 0.0, 1.0, 5.0, and 10.0. The selection of an optimal weight combination is discussed in detail in Section 4.2.

### 3.5. Selection of semantic matching template

To obtain highly superior matching results, it is imperative to assign high positional weights to the barycenter in the matching template. This is achieved by populating the training templates with different values to form template matching labels with different levels of positional weights. The operation is responsible for filling the template from the barycenter to the edge position according to an equal-decreasing paradigm, where the maximum value is filled with 1.0 and the minimum value is 0.0. The position weighted distribution can be generated by any probability distribution function; however, in this paper, we study the correspondence of the semantic template barycenter whose position must be assigned a higher weight, which represents the matching accuracy of the whole template.

We design two forms of labels: (1) each pixel position of the template is assigned a weight of 1.0, which means that the training label is filled with 1.0 on the corresponding position of the reference image and the rest are filled with 0.0 (called zero-one label); (2) the result of a discrete Gaussian distribution function is used as the label value of the semantic spatial distribution (called the Gaussian kernel label). The figure of the Gaussian kernel function is a bell shape with a positive-terrestrial distribution, meaning that the closer the label value to the barycenter is, the greater the probability, and the farther away the label value further from the centroid is, the smaller the probability. The detailed equation of the Gaussian distribution function is as follows:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{9}$$

where $x$ is the input, $\mu$ is the mean, and $\sigma$ is the standard deviation.

In our experiments, we make circular and square matching templates using the two filling forms mentioned above and set different values for Gaussian distribution functions about $\sigma$ and $\mu$. The experimental results show that the training loss of the circular template is slightly lower than that of the square template, while the loss of the discrete Gaussian kernel labels with different $\sigma$ and $\mu$ values does not differ significantly; in contrast, the difference is larger for zero-one labels. Thus, we focus on comparing the matching accuracy of circular zero-one labels and Gaussian kernel labels with $\sigma$=1.0 and $\mu$=0.0 (i.e., discrete filling values of 0.2, 0.4, 0.6, 0.8, 1.0) in different remote sensing image registrations. Three template labels with different filling forms are shown in Fig. 7. Finally, in addition to evaluation on filling forms, we also perform experiments on matching templates with different radii, where the radii of matching templates are set to 12, 18, 25, 36, 124, and 180 (the unit is a pixel).

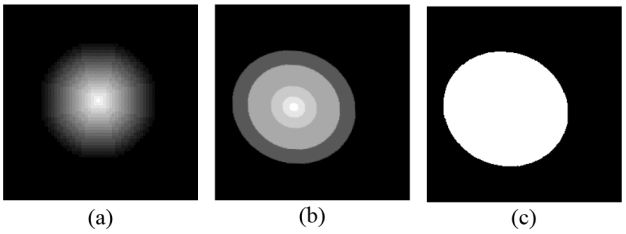


**Fig. 7.** Filling form for circular training labels. Gaussian kernel labels are depicted in (a) and (b), where $\sigma, \mu$ parameters are both 0.5, 1.0, while (b) has different sampling intervals, i.e., the templates are filled with $0.2, 0.4, 0.6, 0.8, 1.0$. A zero-one label is shown in (c), which is only filled with 0.0 and 1.0.

### 3.6. Generating sample data

We develop a stochastic generation method for obtaining training data with the goal of applying these generated samples to adequately train the network model. Fig. 8 presents an overview of our method. Let $I_{optical}$ and $I_{SAR}$ be two remote sensing images aligned in pixels. The first stage of the sample generation method aims at randomly selecting a matching template on two pixel-aligned images, such as the location of the circles in $I_{optical}$ and $I_{SAR}$. While simulating the displacement of three points in $I_{optical}$ according to a certain range and warping the circle by the correspondence generated by three points to perform the affine transformation and obtain $I'_{optical}$, we wish to generate a random transformation matrix with a specific relationship. In other words, we deform the template as much as possible to train a matching network model with geometric invariance. Finally, we crop $I'_{optical}$ to obtain a patch $T_1$ as the reference image for training. Similarly, the circular patch in $I_{SAR}$ is cut and filled to 0.0 with the same size as $T_1$ to obtain $T_2$. $T_1$ and $T_2$ are combined as training sample pairs, where the corresponding positions $L_{true}$ after disturbance are used as matching labels.

### 3.7. Training and matching

Our correspondence framework argues for using global optimization with the function parameter $\theta$ of $N(r,s)$ to obtain the semantic spatial position distribution probability of template pixels while mapping it to template barycenter coordinates. To make the model robust, sample images acquired at different times with the same scene are used to train the network. For the sake of simplicity, input data are both single-band grayscale images. Since each template identifies a match point position at the template barycenter, after sliding the cropped reference and sensed images, we expect to use these matching image pairs to build the

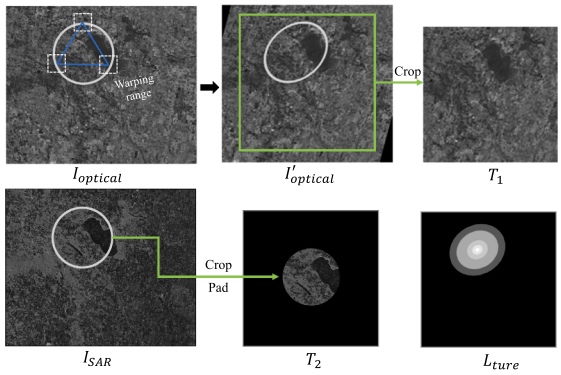


**Fig. 8.** Example of training data generation process. $I_{optical}$ and $I_{SAR}$ represent optical and SAR images, respectively, which are strictly aligned at the pixel level. $I'_{optical}$ represents the $I_{optical}$ after affine transformation, where its transformation relation is recorded by $L_{true}$ and used as the training label. $T_1$ and $T_2$ cropped from $I'_{optical}$ and $I_{SAR}$, respectively represent training images.

set of matching points between images.

To address the obtained matching point sets, the random sampling consensus (RANSAC) (Li et al., 2018a) algorithm is used to globally constrain the invalid matching points, which eliminates incorrect corresponding points from a set of matching points through random sampling and voting schemes. Based on the obtained matching points, they are refined by the least square algorithm to calculate the transformation matrix.

## 4. Experiment

Remote sensing images with different sensors and scenes are analyzed quantitatively and qualitatively to evaluate the matching performance of our proposed pipeline. First, our training data and evaluation indicators are described in detail in Section 4.1. Then, the influence of the training label and loss function weight on the matching accuracy is analyzed in Section 4.2. Section 4.3 introduces the optimal selection of the template radius to improve matching accuracy. Sections 4.4 and 4.5 describe the performance comparison with the latest technologies, such as NCC, MI, SIFT, SUFT (Bay et al., 2008), Affine-SIFT, SAR-SIFT (Dellinger et al., 2014) and the radiation-variation insensitive feature transform (RIFT) (Li et al., 2019).

### 4.1. The training data

We require a large number of remote sensing images to train the correspondence network. We rely on the abovementioned sample generation method to produce 4 kinds of remote sensing datasets, including Google Earth images, GF-2 images, Landsat-8 images, and optical-SAR images. Each dataset contains different types of scenes, including urban, industrial, rural and suburban scenes, which have a range of feature variations. For the Google Earth image datasets, we select different scenes to be cropped at the same position, with a size of 512 × 512. For the GF-2 and Landsat-8 image datasets, we apply the same strategy to obtain them. SAR images are based on the enhanced ellipsoid correction TerraSAR-X data product, while optical images are obtained from Google Earth. In addition, SAR and optical images are precisely coregistered using hundreds of manually selected points to align each pair of images. In the training stage, we shuffle each data set, ensuring each dataset has a ratio of approximately 0.7/0.2/0.1 among training, validation, and test data, respectively.

In the experiment, we use the number of matching points ($N$) and the root mean square error (RMSE) index to evaluate the image registration performance. The detailed calculation formula of RMSE is as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i^p - x_i^t)^2 + (y_i^p - y_i^t)^2} \tag{10}$$

where $(x^p, x^p)$ is the template barycenter position of prediction, $(x^t, x^t)$ is the true matching position of a template, and $n$ represents the number of matching point pairs eliminated by RANSAC.

### 4.2. Selection of the training label and weighted combination of loss functions

We conduct experiments on the above four datasets to evaluate the impact of the training label form described in Section 3, as well as the weighted combination of loss functions on the matching accuracy. We crop 64 matching patch pairs on the four data sets and calculate the average value of RMSE to compare the matching accuracy of the zero-one label and the Gaussian kernel label with different weighted combination $(L_b, L_m)$ loss functions. The settings of the weight parameters and experimental results are detailed in Table 1. It shows that when $L_b$ and $L_m$ weights are set to 10.0 and 1.0, respectively, the matching error of the zero-one label reaches the minimum RMSE on the four data sets. However, the matching error of the Gaussian kernel label is minimized with the weighted combination of 1.0 and 10.0 for $L_b$ and $L_m$, respectively. Particularly, the accuracy is low when using only $L_b$ or $L_m$ loss functions, regardless of whether the training labels are set to the zero-one or Gaussian kernel form, which is insufficient to extract subpixel-level feature fitting. In addition, the matching accuracy of optical and SAR images using zero-one labels is higher than that using Gaussian kernel labels, with an average RMSE of 2.216. In contrast, Google Earth image, GF-2 image, and Landsat-8 image data sets in the case of the Gaussian kernel label achieve a high matching accuracy, reaching the sub-pixel level. Therefore, in the experiment, for Google Earth image, GF-2 image, and Landsat-8 image datasets, we apply Gaussian kernel labels to train the network, while zero-one labels are used for the registration of optical and SAR images.

We use the corresponding networks of zero-one and Gaussian kernel labels to predict the position of templates and generate correspondence heatmaps, as shown in Fig. 9. With a weighted combination ($1.0L_b$+$10.0L_m$) of loss functions, in Fig. 9(c), the zero-one label network produced an irregular circle, the size of which almost coincides with the label, for which the matching accuracy depends on the coincidence of the network's predicted boundaries. As depicted in Fig. 9(f), with the weighted combination ($10.0L_b$+$1.0L_m$) loss functions, the model trained with Gaussian kernel labels generates a square with smooth edges, which is surrounded by a circle similar to a template to reconcile the barycenter of feature maps, achieving subpixel matching accuracy. In this case, the network prediction results do not generate the same shape as the given Gaussian kernel template, which may be due to the further abstraction of the probability distribution of the position similarity at the semantic level by the deep neural network.

**Table 1**
Matching error with the combination of $\alpha L_b + \beta L_m$ loss function. Z and P are abbreviations for the zero-one and Gaussian kernel label, respectively. $T_1, T_2, T_3, T_4$ represent Google Earth image, GF-2, Landsat-8, and optical and SAR datasets, respectively.

| | Loss function weight combinations | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $\alpha$= | 0.0 | 1.0 | 1.0 | 1.0 | 1.0 | 5.0 | 5.0 | 10.0 | 10.0 |
| | $\beta$= | 1.0 | 1.0 | 0.0 | 5.0 | 10.0 | 1.0 | 10.0 | 1.0 | 5.0 |
| Z | $T_1$ | 2.61 | 2.26 | 2.61 | 2.35 | 2.32 | 2.21 | 2.14 | **2.07** | 2.09 |
| | $T_2$ | 2.8 | 2.29 | 2.6 | 2.47 | 2.33 | 2.27 | 2.13 | **2.05** | 2.06 |
| | $T_3$ | 2.69 | 2.26 | 2.69 | 2.50 | 2.31 | 2.33 | 2.10 | 2.08 | **2.04** |
| | $T_4$ | 2.99 | 2.33 | 2.67 | 2.43 | 2.32 | 2.26 | 2.14 | **2.01** | 2.14 |
| P | $T_1$ | 1.52 | 1.06 | 1.48 | 1.27 | 1.13 | 1.02 | 0.91 | **0.85** | 0.81 |
| | $T_2$ | 1.47 | 1.12 | 1.49 | 1.18 | 1.12 | 1.04 | 0.93 | **0.86** | 0.91 |
| | $T_3$ | 1.64 | 1.17 | 1.48 | 1.10 | 1.14 | 1.02 | 0.93 | **0.82** | 0.83 |
| | $T_4$ | 2.47 | 2.12 | 2.4 | **1.28** | 1.33 | 1.64 | 1.61 | 1.45 | 1.58 |

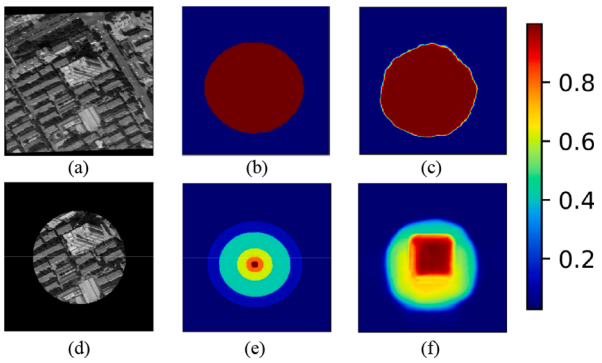


**Fig. 9.** Examples of training label filling forms and corresponding prediction results. (a and d) Represent the reference image and semantic matching template, respectively. (b and e) Are zero-one and Gaussian kernel labels, respectively. (c) Shows the generation result of the zero-one label under the optimal weight combination ($1L_b+10L_m$). (f) Depicts the semantic prediction results of the Gaussian kernel label under the $10L_b+1L_m$ combination.

### 4.3. Selection of template radius

Choosing a template with an appropriate radius size is the basis of semantic information matching. Theoretically, the template radius takes values in the range [1, network window size], where the minimum value of the template radius is 1 and the maximum value is 1/2 of the input image size. While a template with only one pixel can be processed in our matching model, it does not contain contextual information about the pixel, which will result in mismatching. Thus, to make the template contain spatial contextual information, we choose the appropriate radius size for templates that can be used for network evaluation. However, template patches with a large size may result in the identification of no matching position on the reference image. Therefore, under the condition of the input window size limitation, we must obtain the maximum registration accuracy on the smallest possible matching template, leaving enough freedom for subsequent template position selection.

We select reference images with sizes of 128 × 128, 256 × 256, 320 × 320, and 400 × 400 to evaluate the effect of the corresponding template radius on the matching accuracy, with the goal of optimizing the selection of the appropriate template radius. To ensure a fair comparison, we test the above four data sets. The size of the reference images is 128 × 128, while the corresponding template radii are set to 15, 20, 25, 30, 35, 40, 45, and 50. The selection of other reference image sizes and their radii are shown in Fig. 10. The hyperparameters used for training, including the batch size, epoch, and learning rate, are consistent across the four test data sets, except for the reference and template image radii.

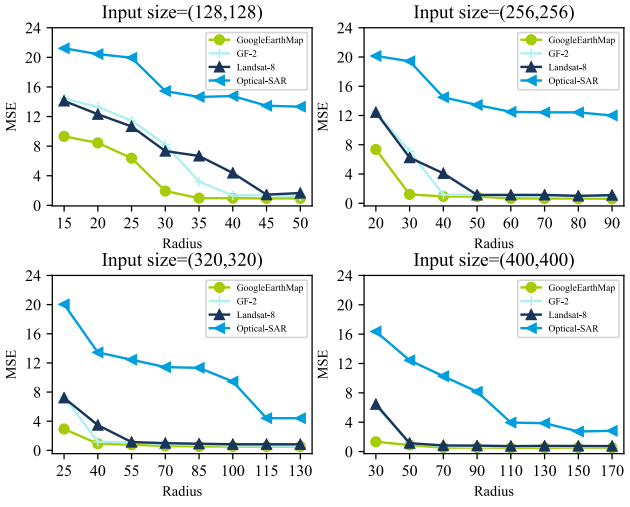


**Fig. 10.** Effect of network input window and template size on accuracy. The four sub-figures respectively show the matching RMSE of window sizes 128 × 128, 256 × 256, 320 × 320 and 400 × 400 with different template radii.

In the evaluation stage, we apply RMSE to assess the image matching accuracy at different template radii. As shown in Fig. 10, the same trend of variation is obtained on the four datasets, i.e., the matching error gradually decreases as the template radius increases, and the decreased amplitude becomes small when the template radius increases to a certain range. For example, the Google Earth image, GF-2 image, and Landsat-8 image datasets all start to have a smooth decrease in the matching error with a radius size of 45 when the reference image size is set to 128 × 128, and matching minima begin to develop. However, the RMSE of the optical and SAR image matching remains at approximately 16.0 compared with the other three sets of data, which fails to meet the requirements of accurate image matching. The matching accuracy of the optical and SAR images gradually improves with increasing the reference image size and template radius. When the reference image size is set to 400 × 400 and the matching template radius is set between 110 and 170, the optical and SAR images achieve a higher matching accuracy. To unify the model and its accuracy comparison, we set the size of the reference image and template radius to 400 × 400 and 128 × 128, respectively.

### 4.4. Semantic template matching

Since the matching accuracy of the proposed network plays a crucial role throughout the whole matching process, it is imperative that we apply existing methods to evaluate their performance. We use two methods for comparison: NCC and MI matching methods. Moreover, both of these methods use a global searching strategy to match the best alignment position, in which the results with the minimum RMSE are used for comparisons. For a fair comparison, we randomly cut 16 pairs of matching templates on each test data set for experiments. In the evaluation stage, we use the proposed template matching method and two comparison methods to obtain matching positions, while RMSE and average time consumed ($AT$) are used to evaluate the matching accuracy and speed performance, respectively. Table 2 shows the matching RMSE and $AT$ results of 64 image pairs on four test data sets. The results demonstrate that the matching accuracy of the proposed matching method is significantly improved compared with NCC and MI. The average RMSEs of our proposed method on the four test datasets are 0.63, 0.65, 0.52, and 1.40, which are significantly lower than those of NCC (13.55, 13.94, 14.24, 14.14) and MI (8.12, 8.50, 7.76, 7.64). The NCC and MI matching methods have a large RMSE on the test data, which may be caused by template deformation. Furthermore, our correspondence network directly generates the spatial matching probability distribution of template positions without using pixel-by-pixel search, as in the NCC and MI methods, which saves considerable time. As shown in Table 2, the NCC and MI methods take approximately 320 s to match a template on average, while our method takes only 0.95 s (TensorFlow with RTX2070), which greatly improves the matching efficiency.

Table 2 shows that the matching errors of $i_{13}$ in $T_1$, $i_6$ in $T_2$, $i_8$ in $T_3$ and $i_1$ in $T_4$ are significantly larger than the average values in the test dataset. We further study the quality of the heatmap generated by the model prediction results through the process of data comparison. Fig. 11 shows examples of generated matching pair heatmaps with large deviations from the average RMSE in the matching results on the Google Earth image, GF-2 image, Landsat-8 image and optical-SAR image datasets. The template in Fig. 11(a) only contains the vegetation category, with a matching RMSE of 1.38, which may be because the reference and sensed images have different shadow directions of the ground features, causing the network to give more weight to the homogeneous vegetation zone in the lower left, thereby generating an irregular heatmap. The radius of the template in Fig. 11(b) almost coincides with the

**Table 2**
Matching accuracy results. We show the matching results on four test data sets through NCC, MI and the proposed method, where PRO represents the abbreviation of our proposal and *AT* is an abbreviation for average time consumed.

| | Method | $i_1$ | $i_2$ | $i_3$ | $i_4$ | $i_5$ | $i_6$ | $i_7$ | $i_8$ | $i_9$ | $i_{10}$ | $i_{11}$ | $i_{12}$ | $i_{13}$ | $i_{14}$ | $i_{15}$ | $i_{16}$ | *AT* | ARMSE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $T_1$ | NCC | 13.2 | 11.98 | 9.51 | 15.74 | 17.93 | 9.29 | 11.91 | 9.62 | 11.90 | 17.31 | 19.44 | 13.62 | 11.58 | 17.17 | 14.45 | 12.22 | 316 | 13.55 |
| | MI | 6.11 | 8.34 | 9.60 | 8.53 | 7.37 | 7.80 | 9.82 | 8.48 | 7.24 | 8.22 | 6.91 | 7.25 | 9.29 | 6.73 | 9.53 | 8.69 | 323 | 8.12 |
| | PRO | 0.76 | 0.30 | 0.44 | 0.76 | 1.01 | 0.71 | 1.12 | 0.72 | 0.37 | 0.26 | 0.42 | 0.53 | 1.38 | 0.20 | 0.31 | 0.86 | **0.98** | **0.63** |
| $T_2$ | NCC | 11.31 | 10.18 | 15.92 | 18.25 | 18.6 | 10.71 | 11.52 | 13.83 | 11.12 | 9.75 | 16.45 | 14.53 | 19.44 | 19.41 | 9.26 | 12.84 | 319 | 13.94 |
| | MI | 6.24 | 9.26 | 9.37 | 9.20 | 9.49 | 8.37 | 9.14 | 7.83 | 9.73 | 6.25 | 7.82 | 8.10 | 9.91 | 7.49 | 9.20 | 8.57 | 322 | 8.50 |
| | PRO | 1.11 | 0.97 | 0.87 | 0.34 | 0.23 | 1.23 | 1.05 | 0.45 | 0.55 | 0.64 | 0.95 | 0.31 | 0.67 | 0.16 | 0.59 | 0.26 | **0.93** | **0.65** |
| $T_3$ | NCC | 15.74 | 12.29 | 13.6 | 17.77 | 11.21 | 10.47 | 11.47 | 11.68 | 16.86 | 12.83 | 18.76 | 11.64 | 9.79 | 19.28 | 17.97 | 16.98 | 318 | 14.27 |
| | MI | 7.37 | 6.5 | 6.17 | 7.20 | 6.48 | 9.38 | 7.82 | 6.13 | 8.34 | 7.6 | 7.60 | 9.28 | 8.8 | 9.57 | 9.87 | 6.18 | 326 | 7.76 |
| | PRO | 0.43 | 0.33 | 0.53 | 0.65 | 0.53 | 0.37 | 0.39 | 1.17 | 0.47 | 0.77 | 0.25 | 0.28 | 0.29 | 0.63 | 0.68 | 0.47 | **0.96** | **0.52** |
| $T_4$ | NCC | 19.45 | 10.88 | 18.17 | 16.20 | 19.98 | 19.71 | 15.74 | 17.92 | 19.70 | 12.89 | 15.12 | 14.38 | 16.71 | 18.65 | 13.25 | 9.52 | 320 | 14.14 |
| | MI | 6.12 | 6.76 | 8.38 | 9.41 | 6.20 | 7.27 | 7.13 | 8.51 | 8.50 | 8.18 | 8.77 | 6.22 | 7.77 | 6.35 | 8.96 | 7.85 | 314 | 7.64 |
| | PRO | 2.73 | 1.18 | 1.07 | 1.05 | 1.65 | 0.63 | 2.22 | 0.89 | 1.56 | 2.52 | 1.31 | 0.95 | 2.17 | 0.72 | 0.87 | 0.91 | **0.95** | **1.40** |

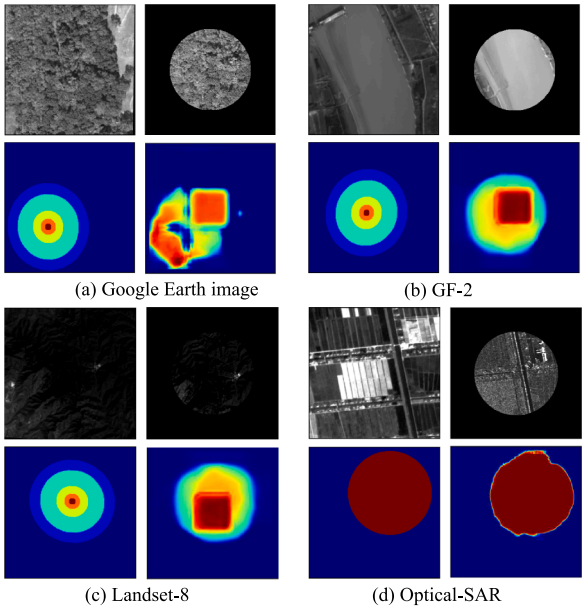


**Fig. 11.** Example of a matching pair with a large deviation from the average value of the RMSE in the test data matching result. (a), (b), (c) and (d) Represent the Google Earth image, GF-2, Landsat-8, and optical-SAR image respectively, where each of them is represented as the reference image, the semantic matching template, the training label, and the feature map predicted by the network.

width of the river, with only a few recognizable shore features on the left side. The matching RMSE reaches 1.23. Therefore, when the image has a higher resolution, a fixed-size matching template is likely to contain the features with the same texture, making the matching unstable. The matching RMSE in Fig. 11(c) is 1.17, which is similar to the case in Fig. 11(b). Specifically, there are fewer distinguishable features in the template and most of them are homogeneous objects without location identification. For the matching of optical-SAR images, the network requires generating a semantic spatial position probability distribution aligned with the template boundaries. However, the boundary of a template generated in Fig. 11(d) is rough and irregular because the nonlinear radiation between the optical and SAR images has a relatively large difference. The objects with larger contrast in the optical image are all affected by the same radiation instead of SAR images.

### 4.5. Large-scale scene matching

While we have evaluated the label filling form, template radius, optimal selection for loss function weight parameters, and experiments on four test data sets, these are limited to patch-based tests. Therefore, to comprehensively evaluate the applicability of our proposed method, we apply it to determine the correspondence relationship in large-scale remote sensing images. In the experiment, we choose the same sensor image as the training datasets. Fig. 12(a) and (b) depict images of the Nashville city scene in the United States obtained from Google Earth, acquired in December 2019 and June 2016, respectively, with a size of 1735 × 1500. Fig. 13(a) and (b) show the high-resolution remote sensing data taken by GF-2 over Chuzhou City, China, with sizes of 1600 × 1620. The shooting time was July 2016 and November 2017. Fig. 14 (a) and (b) are rural land cover scenes in Guizhou Province, China. They were taken by Landsat-8 in June 2015 and July 2016, respectively, with sizes of 1274 × 1287. Fig. 15(a) was obtained from Google Earth, and was taken in December 2013. Fig. 15(b) was taken by TerraSAR-X in December 2017 with the size of 2035 × 1943. $I_1$,$I_2$,$I_3$, and $I_4$ represent the large-scale scene of Google Earth image, GF-2 images, Landsat-8 images, and optical-SAR images, respectively.

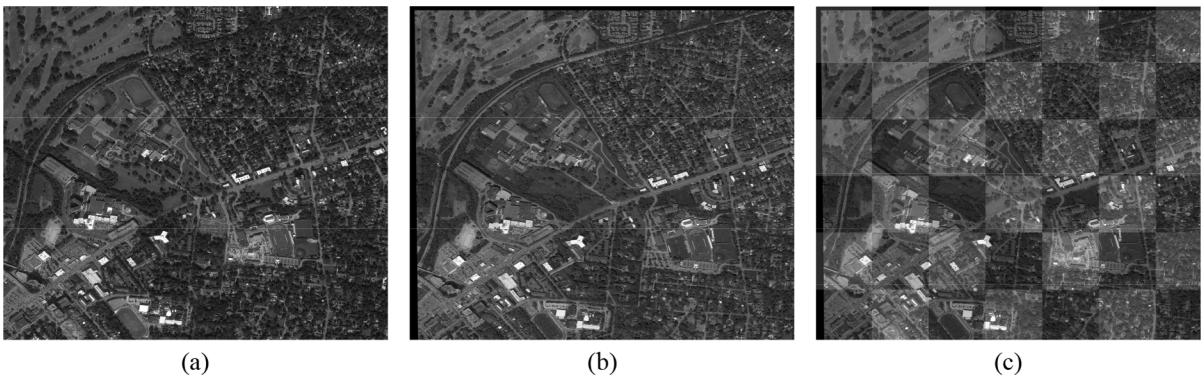


**Fig. 12.** The images acquired from Google Earth image over an urban area in Nashville, United States, with a size of 1735 × 1500. (a) Is the sensed image,taken in December 2019. (b) Is the reference image, taken in December 2019. (c) Is our result on the checkerboard mosaicked image.

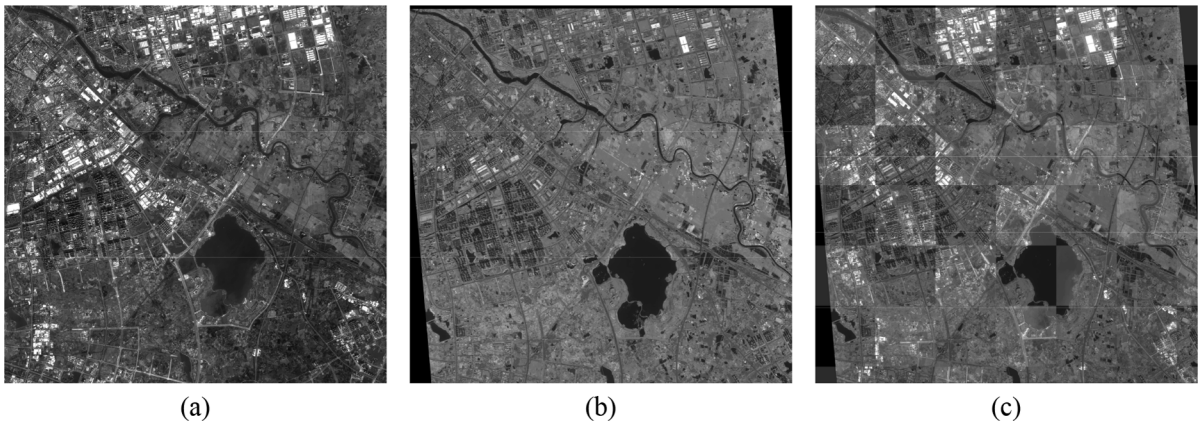


**Fig. 13.** The images acquired by GF-2 over the suburb area in Chuzhou, China, and the size is of 1600 × 1620. (a) Is the sensed image,taken in July 2016. (b) Is the reference image, taken in November 2017. (c) Is our result on the checkerboard mosaicked image.

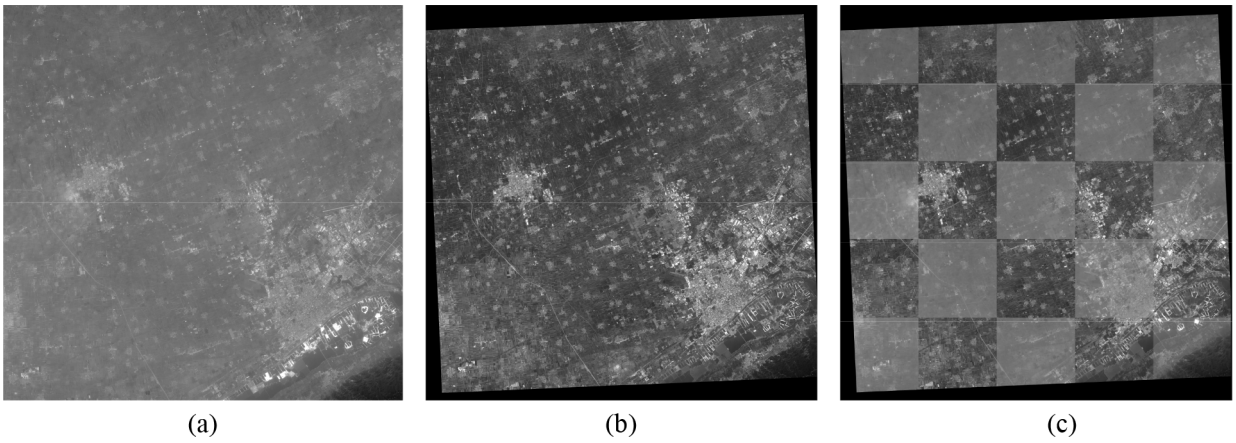


**Fig. 14.** The images acquired by Landsat-8 over the rural land cover scenes in Guizhou Province, China, with a size of 1274 × 1287. (a) Is the sensed image,taken in June 2015. (b) Is the reference image, taken in July 2016. (c) Is our result on the checkerboard mosaicked image.

In the experiment, our proposed method is compared with the remaining 5 latest technologies: SIFT, SURF, Affine-SIFT, SAR-SIFT, and RIFT. We use the number of the matching points ($N$) and RMSE to evaluate the accuracy of matching points. Theoretically, our proposed method can crop a large number of matching templates from the sensed images based on a certain image overlap. Although increasing the number of matching templates can improve the accuracy of registration, the computation time will increase accordingly. Thus, to unify the comparison, we crop the same number of matching pairs on the four large scene images for registration. As noted in Section 4.4, a fixed-size matching template radius can lead to templates containing indistinguishable homogeneous objects, resulting in increased errors. Therefore, we use RANSAC to eliminate these erroneous matching points based on the fixed number of matching image pairs.

For SIFT, SURF, SAR-SIFT, they are matched with the Euclidean distance ratio between the nearest and second nearest neighbors of the corresponding feature, where the ratio is set to 0.6, 0.7, 0.8, 0.9, and RANSAC is set to 5.0. Then the minimum RMSE is selected for com-

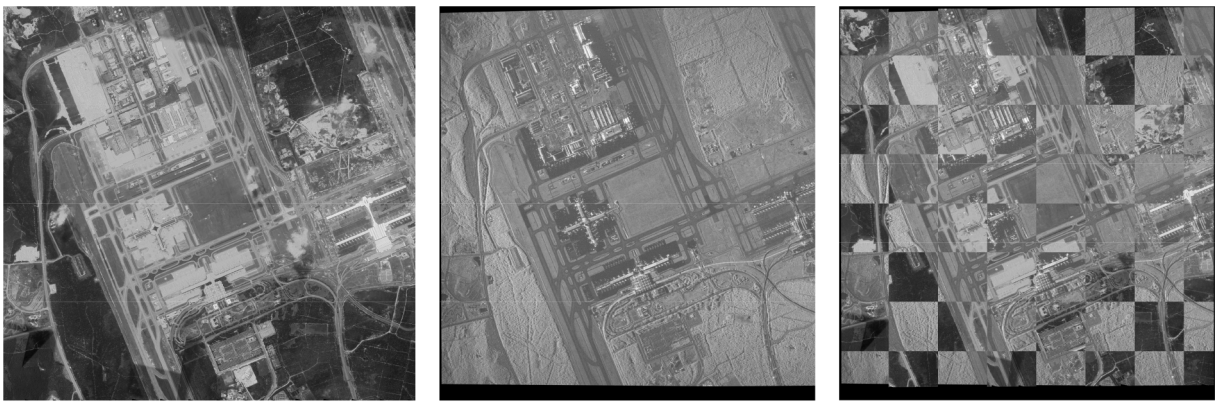


**Fig. 15.** The images over the airport scene Kuala Lumpur, Malaysia. (a) Is the sensed image acquired from the Google Earth image, taken in December 2015. (b) Is the reference image acquired by TerraSAR-X in July 2020. (c) Is our result on the checkerboard mosaicked image.

**Table 3**
Results for the comparison and our proposal methods. We show the number of matching point and matching error for SIFT, SURF, Affine-SIFT, SAR-SIFR, RIFT and our proposed method.

| method | $I_1$ | | $I_2$ | | $I_3$ | | $I_4$ | |
|---|---|---|---|---|---|---|---|---|
| | *N* | RMSE | *N* | RMSE | *N* | RMSE | *N* | RMSE |
| SIFT | 152 | 1.05 | 114 | 1.21 | 350 | 1.61 | 11 | 21.67 |
| SURF | 128 | 1.11 | 31 | 1.29 | 162 | 1.72 | 75 | 25.35 |
| Affine-SIFT | 4213 | 1.13 | 1130 | 1.30 | 2328 | 2.57 | 327 | 62.02 |
| SAR-SIFT | 215 | 1.16 | 124 | 1.35 | 164 | 1.76 | 31 | 31.26 |
| RIFT | 176 | **1.03** | 121 | **1.16** | 215 | 1.65 | 124 | 3.41 |
| Proposal | 149 | 1.12 | 165 | 1.20 | 123 | **1.27** | 96 | **2.15** |

parison. For Affine-SIFT, we use the results of the online website algorithm. For RIFT, we set the parameters and algorithms given in the paper (Li et al., 2018b). Table 3 lists the matching results of large-scale scene images. It shows that all methods perform well on $I_1$ and $I_2$ (urban scenes) with higher resolution and rich texture. Especially, for $I_3$ of the rural land cover scene and $I_4$ with nonlinear radiometric differences, our method achieves high matching accuracy.

The results show that our proposed template matching method can learn more features from complex and nonlinear radiation images, which has advantages over traditional manual feature-based methods.

On $I_1$ and $I_2$, SIFT and RIFT achieve higher matching accuracy (RMSE 0.975, 0.769). The RMSE of our proposed method is 0.986 and 0.768 on $I_1$ and $I_2$, respectively, which also achieve sub-pixel matching accuracy. For $I_3$ and $I_4$, our proposed method achieves the highest matching accuracy (RMSE 1.126, 1.567), which is 45 % higher than the best RIFT in the comparison method. In addition, although Affine-SIFT produces a larger number of matching points on $I_1, I_2, I_3$, and $I_4$, it does not obtain higher matching accuracy. For our proposed method, we set a fixed number of matching templates in four large scene images, which means that about 169 templates can be cropped from each image without pixel overlap. After RANSAC eliminates the invalid matching points, the *N* on $I_1, I_2, I_3$ and $I_4$ are 149, 165, 123, and 96, which indicates that our method achieves a small RMSE with a certain number of matching points while obtaining a higher matching accuracy.

The qualitative evaluation results of our proposed method are illustrated in the chessboard mosaic image. Fig. 12(c), Fig. 13(c), and Fig. 14 (c) show that the edges of ground objects are continuous, and the overall area overlaps very well in our results. For the optical and SAR images of $I_4$, a small number of regions appear discontinuous under the influence of nonlinear radiometric differences and regional feature variations. However, our proposed method can still perform registration well overall, as shown in Fig. 15(c). The above quantitative and qualitative evaluations demonstrate the effectiveness of our template matching method in rural land scenes, especially for optical and SAR image matching with nonlinear radiometric differences.

The experimental results show that our proposed method can accurately determine the spatial position between large scene remote sensing images. However, the matching accuracy of the proposed framework is slightly lower than that of SIFT, SURF and RIFT on Google Earth images and lower than that of RIFT in GF-2. In this case, our proposed framework requires fine-tuning of diverse datasets and matching templates of high spatial resolution remote sensing image pairs compared with the method based on feature point matching. In addition, our proposed method maps the position probability distribution of one image in the other image, which requires manual selection of matching templates in the case of large deformation of the two images or small overlapping areas. Although our method can calculate whether the images match each other and the coordinate positions by training positive and negative matching samples, it increases the matching computation time as the number of image pairs increases. In SAR and optical image matching, the matching accuracy of the proposed framework is generally higher than that of SIFT, SURF, Affine-SIFT, SAR-SIFR, and RIFT. However, the RMSE of SAR and optical images is 2.15, which is lower than the matching accuracy between optical images. Therefore, the network framework requires increasing the size of the input window to further improve the matching accuracy of SAR and optical images.

## 5. Conclusion

In this paper, we propose a deep learning semantic template matching framework for remote sensing image registration. In contrast to conventional image registration, the framework does not obtain the transform parameters by feature extraction and matching optimization but directly by mapping the spatial position probabilities of templates on the reference image. Two loss functions are introduced in the matching template framework, which transforms the boundary alignment task

into a point-to-point matching problem while avoiding mismatching due to fuzzy semantic boundaries. Additionally, we develop a filling form using a discrete Gaussian kernel for training labels, which improves matching accuracy.

In the experiment, we demonstrate the following:

(1) Our proposed method achieves a high matching accuracy while spending less time using a deep neural network to map the spatial position probability distribution of the template.
(2) The image matching accuracy in this study is sensitive to the size of reference images and the radius of templates, especially for optical and SAR image matching, which requires a larger input size and template radius;
(3) The form of training labels affects the matching accuracy. For instance, training labels populated with Gaussian kernels can significantly improve the matching accuracy between identical sensor images. In contrast, matching optical and SAR images requires zero-one labels.

The matching accuracy of our method does not differ much from the best matching accuracy in the registration of high-resolution remote sensing images of large scenes compared with the latest existing matching methods. Furthermore, our framework still has high stability in rural land cover scenes, particularly in optical and SAR image registration with nonlinear radiometric differences. The proposed deep learning semantic template matching method provides a new problem-solving idea for remote sensing image registration.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

This work is supported by the Ministry of Education Joint Foundation (No. 6141A02022376), the Fund Project of Shaanxi Key Laboratory of Land Consolidation, the Fundamental Research Funds for the Central Universities (No. 300102350401)

## References

Balakrishnan, G., Zhao, A., Sabuncu, M.R., Guttag, J., Dalca, A.V., 2018. An unsupervised learning model for deformable medical image registration. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 9252–9260.
Balakrishnan, G., Zhao, A., Sabuncu, M.R., Guttag, J., Dalca, A.V., 2019. Voxelmorph: a learning framework for deformable medical image registration. IEEE Trans. Med. Imag. 38, 1788–1800.
Bay, H., Ess, A., Tuytelaars, T., Van Gool, L., 2008. Speeded-up robust features (surf). Comput. Vis. Image Understand. 110, 346–359.
Boroumand, M., Chen, M., Fridrich, J., 2018. Deep residual network for steganalysis of digital images. IEEE Trans. Inf. Forensics Secur. 14, 1181–1193.
Cai, G.R., Jodoin, P.M., Li, S.Z., Wu, Y.D., Su, S.Z., Huang, Z.K., 2013. Perspective-sift: An efficient tool for low-altitude remote sensing image registration. Signal Process. 93, 3088–3110.
Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2017a. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans. Pattern Anal. Mach. Intell. 40, 834–848.
Chen, L.C., Papandreou, G., Schroff, F., Adam, H., 2017b. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587.
Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018a. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV), pp. 801–818.
Chen, S., Li, X., Zhao, L., Yang, H., 2018b. Medium-low resolution multisource remote sensing image registration based on sift and robust regional mutual information. Int. J. Remote Sens. 39, 3215–3242.
Dalca, A.V., Balakrishnan, G., Guttag, J., Sabuncu, M.R., 2018. Unsupervised learning for fast probabilistic diffeomorphic registration. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 729–738.
De Castro, E., Morandi, C., 1987. Registration of translated and rotated images using finite fourier transforms. IEEE Trans. Pattern Anal. Mach. Intell. PAMI-9 700–703. https://doi.org/10.1109/TPAMI.1987.4767966.
Dellinger, F., Delon, J., Gousseau, Y., Michel, J., Tupin, F., 2014. Sar-sift: a sift-like algorithm for sar images. IEEE Trans. Geosci. Remote Sens. 53, 453–466.
DeTone, D., Malisiewicz, T., Rabinovich, A., 2016. Deep image homography estimation. arXiv preprint arXiv: 1606.03798.
DeTone, D., Malisiewicz, T., Rabinovich, A., 2018. Superpoint: Self-supervised interest point detection and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 224–236.
Fan, J., Cao, X., Yap, P.T., Shen, D., 2019. Birnet: Brain image registration using dual-supervised fully convolutional networks. Med. Image Anal. 54, 193–206.
Feng, R., Du, Q., Li, X., Shen, H., 2019. Robust registration for remote sensing images by combining and localizing feature-and area-based methods. ISPRS J. Photogramm. Remote Sens. 151, 15–26.
Han, X., Leung, T., Jia, Y., Sukthankar, R., Berg, A.C., 2015. Matchnet: Unifying feature and metric learning for patch-based matching. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3279–3286.
Haskins, G., Kruger, U., Yan, P., 2020. Deep learning in medical image registration: a survey. Mach. Vis. Appl. 31, 8.
Hisham, M., Yaakob, S.N., Raof, R.A., Nazren, A.A., Embedded, N.W., 2015. Template matching using sum of squared difference and normalized cross correlation. In: 2015 IEEE Student Conference on Research and Development (SCOReD). IEEE. pp. 100–104.
Jaderberg, M., Simonyan, K., Zisserman, A., et al., 2015. Spatial transformer networks. In: Advances in neural information processing systems, pp. 2017–2025.
Jiang, X., Ma, J., Fan, A., Xu, H., Lin, G., Lu, T., Tian, X., 2020. Robust feature matching for remote sensing image registration via linear adaptive filtering. IEEE Trans. Geosci. Remote Sens. 1–15 https://doi.org/10.1109/TGRS.2020.3001089.
Karthick, S., Maniraj, S., 2019. Different medical image registration techniques: A comparative analysis. Curr. Med. Imag. 15, 911–921.
Kern, J.P., Pattichis, M.S., 2007. Robust multispectral image registration using mutual-information models. IEEE Trans. Geosci. Remote Sens. 45, 1494–1505.
LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444.
Li, H., Qin, J., Xiang, X., Pan, L., Ma, W., Xiong, N.N., 2018a. An efficient image matching algorithm based on adaptive threshold and ransac. IEEE Access 6, 66963–66971.
Li, J., Hu, Q., Ai, M., 2018b. Rift: Multi-modal image matching based on radiation-invariant feature transform. arXiv preprint arXiv:1804.09493.
Li, J., Hu, Q., Ai, M., 2019. Rift: Multi-modal image matching based on radiation-variation insensitive feature transform. IEEE Trans. Image Process. 29, 3296–3310.
Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision 60, 91–110.
Ma, W., Wen, Z., Wu, Y., Jiao, L., Gong, M., Zheng, Y., Liu, L., 2016. Remote sensing image registration with modified sift and enhanced feature matching. IEEE Geosci. Remote Sens. Lett. 14, 3–7.
Maes, F., Collignon, A., Vandermeulen, D., Marchal, G., Suetens, P., 1997. Multimodality image registration by maximization of mutual information. IEEE Trans. Med. Imag. 16, 187–198.
Miao, S., Wang, Z.J., Liao, R., 2016a. A cnn regression approach for real-time 2d/3d registration. IEEE Trans. Med. Imag. 35, 1352–1363.
Miao, S., Wang, Z.J., Zheng, Y., Liao, R., 2016b. Real-time 2d/3d registration via cnn regression. In: 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI). IEEE, pp. 1430–1434.
Paul, S., Pati, U.C., 2016. Remote sensing optical image registration using modified uniform robust sift. IEEE Geosci. Remote Sens. Lett. 13, 1300–1304.
Reddy, B.S., Chatterji, B.N., 1996. An fft-based technique for translation, rotation, and scale-invariant image registration. IEEE Trans. Image Process. 5, 1266–1271. https://doi.org/10.1109/83.506761.
Ryu, S., Kim, S., Sohn, K., 2017. Lat: Local area transform for cross modal correspondence matching. Pattern Recogn. 63, 218–228.
Sedaghat, A., Mohammadi, N., 2018. Uniform competency-based local feature extraction for remote sensing images. ISPRS J. Photogramm. Remote Sens. 135, 142–157.
Suri, S., Reinartz, P., 2009. Mutual-information-based registration of terrasar-x and ikonos imagery in urban areas. IEEE Trans. Geosci. Remote Sens. 48, 939–949.
Tareen, S.A.K., Saleem, Z., 2018. A comparative analysis of sift, surf, kaze, akaze, orb, and brisk. In: 2018 International conference on computing, mathematics and engineering technologies (iCoMET). IEEE, pp. 1–10.
Tong, X., Ye, Z., Xu, Y., Liu, S., Li, L., Xie, H., Li, T., 2015. A novel subpixel phase correlation method using singular value decomposition and unified random sample consensus. IEEE Trans. Geosci. Remote Sens. 53, 4143–4156.
de Vos, B.D., Berendsen, F.F., Viergever, M.A., Sokooti, H., Staring, M., Išgum, I., 2019. A deep learning framework for unsupervised affine and deformable image registration. Med. Image Anal. 52, 128–143.
Wang, M., Fan, C., Pan, J., Jin, S., Chang, X., 2017. Image jitter detection and compensation using a high-frequency angular displacement method for yaogan-26 remote sensing satellite. ISPRS J. Photogramm. Remote Sens. 130, 32–43.
Wang, S., Quan, D., Liang, X., Ning, M., Guo, Y., Jiao, L., 2018. A deep learning framework for remote sensing image registration. ISPRS J. Photogramm. Remote Sens. 145, 148–164.
Wong, A., Clausi, D.A., 2007. Arrsi: Automatic registration of remote-sensing images. IEEE Trans. Geosci. Remote Sens. 45, 1483–1493.
Xiang, Y., Wang, F., You, H., 2018. Os-sift: A robust sift-like algorithm for high-resolution optical-to-sar image registration in suburban areas. IEEE Trans. Geosci. Remote Sens. 56, 3078–3090.
Xu, X., Li, X., Liu, X., Shen, H., Shi, Q., 2016. Multimodal registration of remotely sensed images based on jeffrey's divergence. ISPRS J. Photogramm. Remote Sens. 122, 97–115.

Yang, K., Pan, A., Yang, Y., Zhang, S., Ong, S.H., Tang, H., 2017. Remote sensing image registration using multiple image features. Remote Sens. 9, 581.
Yang, Z., Dan, T., Yang, Y., 2018. Multi-temporal remote sensing image registration using deep convolutional features. IEEE Access 6, 38544–38555.
Ye, F., Su, Y., Xiao, H., Zhao, X., Min, W., 2018. Remote sensing image registration using convolutional neural network features. IEEE Geosci. Remote Sens. Lett. 15, 232–236.
Ye, Y., Shan, J., Bruzzone, L., Shen, L., 2017. Robust registration of multimodal remote sensing images based on structural similarity. IEEE Trans. Geosci. Remote Sens. 55, 2941–2958.
Ye, Y., Shen, L., 2016. Hopc: A novel similarity metric based on geometric structural properties for multi-modal remote sensing image matching. ISPRS Ann. Photogramm. Remote Sens. Spatial Inform. Sci. 3, 9.
Yuan, Y., Fang, J., Lu, X., Feng, Y., 2018. Remote sensing image scene classification using rearranged local features. IEEE Trans. Geosci. Remote Sens. 57, 1779–1792.