remote sensing

MDPI

Article

Matching of Remote Sensing Images with Complex Background Variations via Siamese Convolutional Neural Network

Haiqing He [1,*], Min Chen [2], Ting Chen [3] and Dajun Li [1]

[1] School of Geomatics, East China University of Technology, Nanchang 330013, China; djli@ecit.cn
[2] Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu 611756, China; minchen@home.swjtu.edu.cn
[3] School of Water Resources & Environmental Engineering, East China University of Technology, Nanchang 330013, China; ct_201607@ecit.cn
* Correspondence: hhq201360010@ecit.cn

Received: 22 January 2018; Accepted: 22 February 2018; Published: 24 February 2018

Abstract: Feature-based matching methods have been widely used in remote sensing image matching given their capability to achieve excellent performance despite image geometric and radiometric distortions. However, most of the feature-based methods are unreliable for complex background variations, because the gradient or other image grayscale information used to construct the feature descriptor is sensitive to image background variations. Recently, deep learning-based methods have been proven suitable for high-level feature representation and comparison in image matching. Inspired by the progresses made in deep learning, a new technical framework for remote sensing image matching based on the Siamese convolutional neural network is presented in this paper. First, a Siamese-type network architecture is designed to simultaneously learn the features and the corresponding similarity metric from labeled training examples of matching and non-matching true-color patch pairs. In the proposed network, two streams of convolutional and pooling layers sharing identical weights are arranged without the manually designed features. The number of convolutional layers is determined based on the factors that affect image matching. The sigmoid function is employed to compute the matching and non-matching probabilities in the output layer. Second, a gridding sub-pixel Harris algorithm is used to obtain the accurate localization of candidate matches. Third, a Gaussian pyramid coupling quadtree is adopted to gradually narrow down the searching space of the candidate matches, and multiscale patches are compared synchronously. Subsequently, a similarity measure based on the output of the sigmoid is adopted to find the initial matches. Finally, the random sample consensus algorithm and the whole-to-local quadratic polynomial constraints are used to remove false matches. In the experiments, different types of satellite datasets, such as ZY3, GF1, IKONOS, and Google Earth images, with complex background variations are used to evaluate the performance of the proposed method. The experimental results demonstrate that the proposed method, which can significantly improve the matching performance of multi-temporal remote sensing images with complex background variations, is better than the state-of-the-art matching methods. In our experiments, the proposed method obtained a large number of evenly distributed matches (at least 10 times more than other methods) and achieved a high accuracy (less than 1 pixel in terms of root mean square error).

Keywords: image matching; Siamese convolutional neural network; background variation; sub-pixel Harris algorithm; Gaussian pyramid coupling quadtree

1. Introduction

Image matching refers to a fundamental task of establishing correspondences between two or more images of the same scene taken at different times, from different sensors, or from different viewpoints. It is widely used in various applications of computer vision and remote sensing, such as image registration and fusion, change detection, and environment monitoring. Automatic image matching technology has been widely studied in the fields of computer vision and remote sensing in the past decades [1–4]. Unlike those in computer vision applications, the images in remote sensing (multi-temporal and multi-source images) are usually affected by complex background variations, such as noise caused by cloud and haze weather conditions and land cover changes caused by human construction activities and disasters (earthquakes and floods) [5]. These variations make image matching difficult.

Existing matching methods are mainly divided into area-based and feature-based methods [3]. The second group is more popular than the first one due to the robustness and reliability of those methods against image geometric distortion and radiometric difference [6,7]. Feature-based methods generally consist of three steps: feature detection, description, and matching. Scale-invariant feature transform (SIFT) is one of the popular feature-based matching methods [8]. Considering the success of the SIFT algorithm, many improved versions have been proposed to enhance the performance of feature detection, description, and matching. The improved algorithms for feature detection include speeded-up robust features (SURF) [9] and complex SIFT (CSIFT) [10], among others. SURF can accelerate feature detection using FAST-Hessian and Haar wavelets, and CSIFT can detect features of complex-valued images. Many improved descriptors, such as principal component analysis–SIFT [11], gradient location and orientation histogram [12], and Affine–SIFT [13], have been investigated to make the SIFT features distinctive in image deformation. Feature descriptors are combined with several similarity metrics or constraints, such as scale-orientation joint restriction criteria [14], weight-based topological map-matching algorithm [15], normalized cross correlation and least square matching [16], perspective scale invariant feature [17], l_q-estimator [18], and L_2-minimizing estimation [6], to match remote sensing images. Despite significant improvements to the feature-based matching method, the manually designed methods (e.g., SIFT) cannot fully obtain the invariant descriptors with the appearance of nonlinear illumination changes, shadows, and occlusions [19]. Unfortunately, the aforementioned issues are common in high-resolution remote sensing images with background variations. Traditional image matching methods do not work well for these kinds of images.

To improve matching performance in the context of image background variations, the multiscale edge features-based rotation and scale invariant shape context are proposed [5]. In the method, the multiscale morphological operator is used to detect local scale invariant features and the descriptor in the rotation-invariant shape context is designed to match the images. However, this method does not match with high-resolution remote sensing images because unreliable edge gradient information exists in these kinds of images. A line segment-based method is proposed to match remote sensing images with large background variations [20]. In this method, line segments are extracted by using an edge drawing line (EDLines [20]) detector. Line validation is performed to obtain the main shape contour, and histogram binning shape-based descriptors are used to match the line segments. The line segment-based matching methods are robust against global geometrical distortions and can achieve high accuracy. However, line segment-based methods strongly depend on relatively stable linear objects, such as coastlines, riverside lines, and mountain ridges. In actual scenarios, the shape of linear objects may be changed significantly for remote sensing images with complex background variations. For example, the shapes of coastlines are inconsistent in long-term multi-temporal images for human construction activities or sea inundation, and the corresponding lines extracted by edge detectors do not correspond to the conjugated locations. Therefore, complex background variations between two multi-temporal remote sensing images may significantly disrupt the feature detection and representation of conjugated regions and lead to unsatisfactory matching results using the traditional feature-based methods.

In recent years, several image matching methods based on deep learning have been proposed [19,21,22]. Deep networks determine the similarity between image patches, and this is achieved by learning directly from the training examples without having to consider manually designed features. High-level features, rather than the low-level point, line, and region features, are learned in matching. The Siamese-type architecture with two-stream network is commonly used to learn similarities in image matching [19,23–25]. The two-stream Siamese-type network is regarded an effective deep network because of its capability for high-level feature representation. It can improve performance in terms of viewpoint, illumination changes, and background variations. However, the Siamese-type architecture mainly focused on image matching in computer vision, and it does not work well for remote sensing images with complex background variations (complex spatial structures and severe intensity changes). In addition, the benefits of multiscale patch comparison and searching efficiency are difficult to balance due to the large size of remote sensing images.

In this study, we focus on learning how to match remote sensing images with complex background variations. This study aims to design a new technical framework based on the Siamese-type network to directly determine the similarity between remote sensing image patches without manually designed features and descriptors. Currently, no generic rule is applied when determining the architecture of deep learning. Generally, the architecture and parameters are determined through many repeated trials. Thus, the integrated multiple factors of nonlinear transformations between reference and sensed images are difficult to analyze. In this study, we considered each of the factors that may be involved in the remote sensing of images with complex background variations, such as geometric deformation and quality degradation. Then, the number of convolutional layers is determined on the basis of the factors rather than the architecture of deep network by blind repeated trials. In the proposed Siamese-type architecture, the convolutional layers with rectified linear unit (ReLU) activation and one max-pooling layer are arranged to learn abstract feature representations. Subsequently, two streams of convolutional layers share identical weights. In the output layer, the *sigmoid* function is employed to compute the positive and negative probabilities. The similarity of patch pairs is learned from the labeled training examples of matching and non-matching true color patches. To achieve high accuracy and efficiency of patch matching, sub-pixel Harris algorithm (S-Harris) and Gaussian pyramid coupling quadtree (GPCQ) are performed to obtain accurate localization. The searching space of candidate matches is narrowed, and multiscale patches are used to synchronously capture the matches. After initial matching, the false matches are removed by the random sample consensus (RANSAC) algorithm [26] and whole-to-local quadratic polynomial constraints.

The main contribution of this study centers on the design of the deep network for multiscale patch comparison, which, when combined with the S-Harris corner detector, can improve the matching performance for remote sensing images with complex background variations.

The remainder of this paper is organized as follows: Section 2 describes, in detail, the proposed method. Section 3 presents the comparative experimental results in combination with detailed analysis and discussion. Section 4 concludes this paper and provides our possible future work.

2. Methodology

The proposed matching framework mainly includes three steps: Siamese-type network training, S-Harris corner detection, and patch matching. At the training phase, the Siamese-type network (i.e., see Figure 1) is trained by the labeled examples of matching and non-matching true color patches. Then, in the matching phase with the trained network, the reference and sensed images are divided into grids with fixed sizes, and a number of sub-pixel Harris corners are extracted from each grid. Subsequently, GPCQ is established, deep features are extracted through the Siamese-type network, and multiscale similarity measure is synchronously performed. Unlike SIFT, multiscale image blocks are extracted directly to match via the pipeline of Siamese-type network in image Gaussian pyramid instead of three separate steps of multiscale feature detection, description, and matching. It is of importance that a known spatial resolution is used to resample the approximate scale for capturing the candidate conjugated patches. Despite the restriction of spatial

resolution of the reference and sensed images, it is suited to satellite images matching because of a known resolution and rough georeference obtained from the space-borne equipments, such as GPS for sensor position and star trackers for sensor attitude [27,28]. Finally, geometrical constraints are used to remove outliers based on whole-to-local quadratic polynomial functions. The detailed steps are presented in Figure 2.

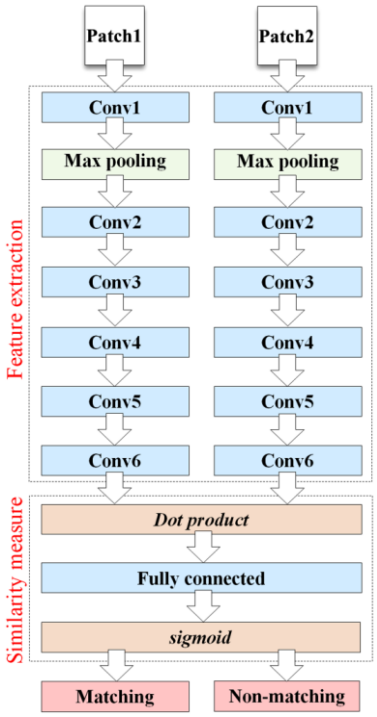


Figure 1. Architecture of Siamese convolutional neural network.

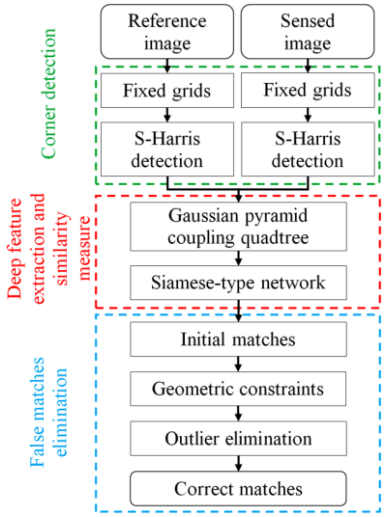
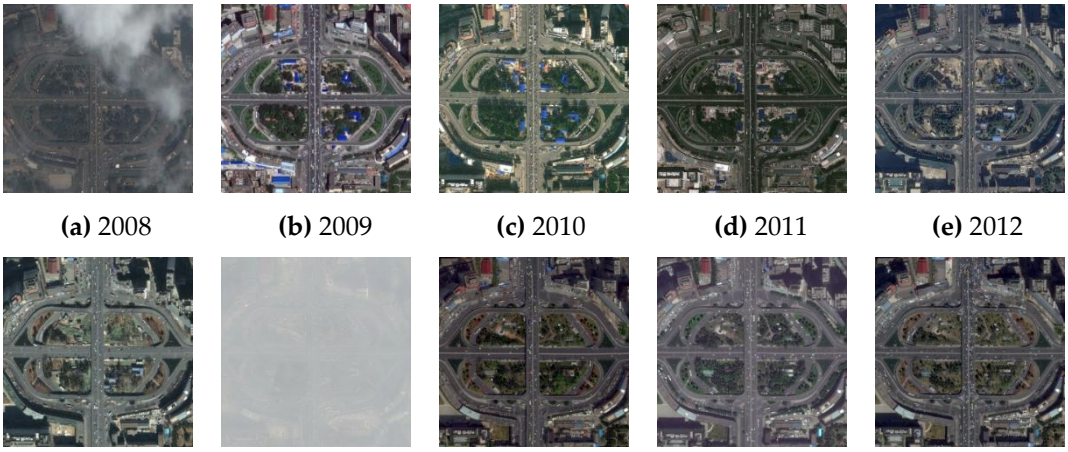


Figure 2. Schematic of the proposed matching framework.

As shown in Figure 1, input patches are extracted from reference and sensed images. Six convolutional layers are arranged for the extracted features in each stream. One max-pooling layer is wedged between the convolutional layers of Conv1 and Conv2, and two streams share identical weights. The similarity measure consists of one dot product layer, one fully connected layer, and one *sigmoid* layer. The matching and non-matching probabilities between 1 and 0 given by sigmoid function ($\frac{1}{1+e^{-x}}$) are used to define the similarity, in which 1 and 0 correspond to matching and non-matching target output values, respectively.

2.1. Siamese Convolutional Neural Network

The architecture of the Siamese convolutional neural network (SCNN) significantly affects the performance of similarity learning. However, except for repeated trials, no effective rule on determining the architecture of SCNN has been reported in the literature. In our study, the architecture is designed on the basis of the factors that affect matching performance. For the multi-temporal remote sensing images shown in Figure 3, the complex background variations can be simplified as certain types of factors, as listed below.



(a) 2008 **(b)** 2009 **(c)** 2010 **(d)** 2011 **(e)** 2012

(f) 2013 **(g)** 2014 **(h)** 2015 **(i)** 2016 **(j)** 2017

Figure 3. Multi-temporal Google Earth images of the same area from 2008 to 2017. Images are affected by complex background variations, including small rotation and translation, nonlinear geometric deformation, shadow, image quality degradation, and land cover changes.

Factor 1: Rotation and translation. These factors are the most basic problems and should be estimated between reference and sensed images. In satellite images, rotation and translation errors generally come from image distortion and navigation error. The transformation of rotation and translation can be represented as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} + t \quad (1)$$

in which (x, y) and (x', y') are the coordinates of the patch centers, θ is the rotation angle, and t denotes translation.

Factor 2: Nonlinear geometric deformation. Remote sensing images are generally affected by complex nonlinear geometric deformations, and they may be caused by topographic reliefs and earth curvature. Nonlinear geometric deformation between remote sensing images can be approximately described by polynomial transformation.

Factor 3: Shadow. Shadows are common in high-resolution remote sensing images, especially in areas with significant topography. The influence of shadow can be expressed as [29]

$$G_{nonshadow} = a_k \cdot G_{shadow} + b_k \quad (2)$$

in which $G_{nonshadow}$ and G_{shadow} indicate the grayscale of pixels without shadow and with shadow, respectively. Furthermore, a_k and b_k are two coefficients.

Factor 4: Image quality degradation. This factor is a widespread problem when surface radiations pass through the atmosphere. The grayscale degradation model can be written as [30]

$$f = A \cdot u + \varepsilon \quad (3)$$

in which u and f are the true and degradation images, respectively; A is a diagonal matrix with diagonal elements composed of 0 and 1; and ε represents the noise vector.

Factor 5: Land cover changes. This factor is considered a widely existing complex nonlinear problem in multi-temporal remote sensing images. Nonlinear grayscale changes are difficult to describe using a generic mathematical model.

Apart from the aforementioned factors, other changes (i.e., sensor settings) may be found but are rather difficult to model. Moreover, although the relationship of pairs after decoupling is simple to determine, many uncertainties are found in the coupled factors. Consequently, the integration of factors is difficult to analyze, and the coefficients of each transformation are difficult to compute simultaneously. In this study, the combination of these factors is represented in multiple hidden layers to avoid explicit solutions. The multiple transformations between output O and input X of the deep network can be expressed as

$$O = f_n(\dots f_2(f_1(XW_1)W_2 \dots)W_n) \tag{4}$$

in which $f_1, f_2, \dots, f_n$ denotes the various transformations caused by the involved factors of image matching (e.g., f_1 reflects the factor in Equation (1)), while $W_1, W_2, \dots, W_n$ denotes the related weights. In the proposed method, five convolutional layers are arranged to describe the five aforementioned factors. An additional layer is also added to describe the unknown factors. The architecture of the proposed Siamese-type network includes three types of layers: convolutional, pooling, and fully connected layers (Figure 1). Batch normalization [31] is wedged into each convolutional layer before the activation of neurons. In this network, two streams of convolutional and pooling layers sharing the weights are arranged without assuming any feature extraction and description. ReLU activation is employed for feature sparse representation in the convolutional layers. Max-pooling is used for feature map compression and complexity simplification. Subsequently, one fully connected layer is concatenated to the decision network. The sigmoid function is employed to define the similarity in the fully connected layer. Output f_j^l of the j^{th} feature map in the l^{th} layer via convolution can be written as

$$f_j^l = \sigma(z_l) = \sigma\left(\sum_{i \in s_{l-1}} f_i^{l-1} * w_{ij}^l + b_j^l\right) \tag{5}$$

where f_i^{l-1} is the i^{th} feature map in the $(l-1)^{th}$ layer; s_{l-1} is the number of feature maps in the $(l-1)^{th}$ layer; w and b represent the convolution kernels (weights) and biases, respectively; $*$ is the convolution operator; and $\sigma(.)$ denotes the activation function. ReLU $\sigma(x) = \max(0, x)$ is applied in our method. Unlike the activation function used in the output layer of deep networks [19], in the proposed Siamese-type network, sigmoid function instead of ReLU is adopted to compute matching and non-matching probabilities, which are restricted between 0 and 1. Hence, the hinge-based loss function may be unsuitable for computing the loss in terms of the output values, while the global cost function is an alternative function with regard to sigmoid output. Therefore, the proposed Siamese-type network is trained in a supervised manner by minimizing cost function$J$.

$$J(w, b) = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{2}\left\|h\left(x^{(i)}\right) - y^{(i)}\right\|^2\right) + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}\left(w_{ji}^l\right)^2, \tag{6}$$

in which *h(x)* are the trained results of the output layer; y are the expected output values given in a supervised manner; n and n_l are the number of trained data and layers, respectively; λ is the weight decay parameter; and s_l and s_{l+1} are the number of feature maps in layers l and $l+1$, respectively. Back propagation, which is used to update the weights and biases from one layer to the next via stochastic gradient descent, can be written as

$$w_{ij}^{(l)} = w_{ij}^{(l)} - \eta\frac{\partial J(w, b)}{\partial w_{ij}^{(l)}} = w_{ij}^{(l)} - \eta\left(f_j^{(l)}\delta_i^{(l+1)}\right) \tag{7}$$

$$b_i^{(l)} = b_i^{(l)} - \eta \frac{\partial J(W,b)}{\partial b_i^{(l)}} = b_i^{(l)} - \eta \delta_i^{(l+1)} \tag{8}$$

$$\frac{\partial J(w,b)}{\partial w_{ij}^{(l)}} = \frac{\partial J(W,b)}{\partial z_i^{(l+1)}} \frac{\partial z_i^{(l+1)}}{\partial w_{ij}^{(l)}} \tag{9}$$

$$\frac{\partial J(w,b)}{\partial b_i^{(l)}} = \frac{\partial J(W,b)}{\partial z_i^{(l+1)}} \frac{\partial z_i^{(l+1)}}{\partial b_i^{(l)}} \tag{10}$$

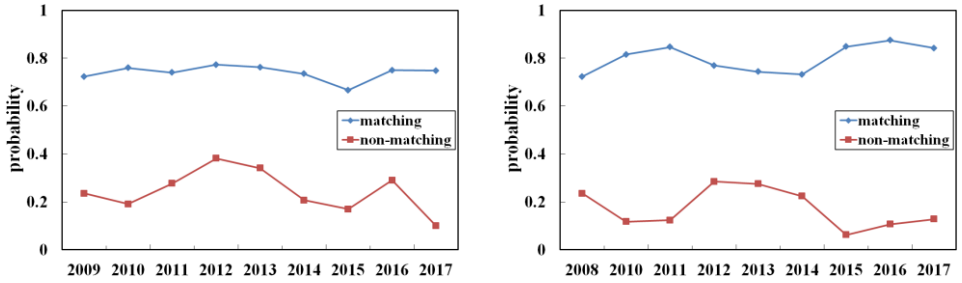
in which η is the learning rate. The residual error $\delta_i^{(n_l)}$ of the output layer can be computed by

$$\delta_i^{(n_l)} = \frac{\partial J(W,b;x,y)}{\partial z_i^{(n_l)}} = -\left(y_i - f\left(z_i^{(n_l)}\right)\right) \cdot f'\left(z_i^{(n_l)}\right) \tag{11}$$

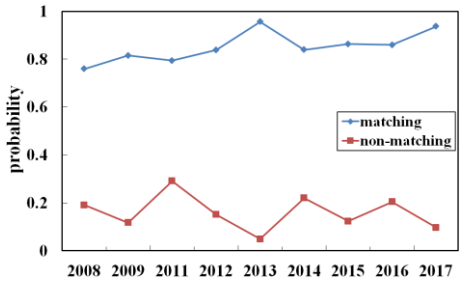
The residual error of back propagation in the i^{th} feature map of l^{th} convolutional layer is computed as

$$\delta_i^{(l)} = \left(\sum_{j=1}^{s_{i+1}} w_{ji}^{(l)} \delta_i^{(l+1)}\right) f'\left(z_i^{(l)}\right) \tag{12}$$

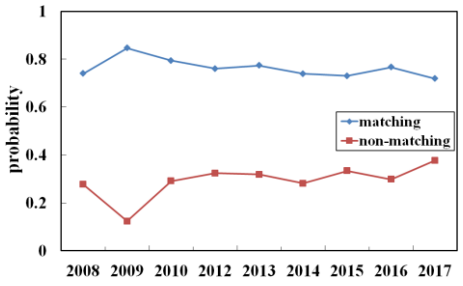
The main advantage of the proposed SCNN is its strong capability to deal with nonlinear problems using the multilayer network. Figure 4 shows the pairwise matching and non-matching probabilities computed by the proposed SCNN for all the patch pairs in Figure 3. The image patch of each year is compared with other years. The architecture of the proposed SCNN in this test is expressed as follows: The two streams initially consist of the same branch $C(64,7,3)$–ReLU–$P(2,2)$–$C(128,5,1)$–ReLU–$C(128,5,1)$–ReLU–$C(128,5,1)$–ReLU–$C(256,5,1)$–ReLU–$C(256,5,1)$–ReLU. The concatenated part $F(512)$-*Sigmoid*-$F(2)$. $C(n,k,m)$ denotes the convolutional layer with n filters of spatial size $k \times k$ of band number m. $P(k,s)$ represents a max-pooling layer with size $k \times k$ of stride s. $F(n)$ is a fully connected layer with n output units. Training dataset with true-color patches is generated from Google Earth images. Multi-temporal remote sensing images on conjugated areas are divided as image patches (size 96×96) to be used as inputs. Overfitting is avoided with a data augmentation strategy [19], Gaussian filtering with standard deviation $\sigma = 1.6$ and $\sigma = 3.2$ respectively. Weights are initialized by Gaussian random distribution. The initial learning rate of 0.01 and momentum of 0.9 are used to train the proposed SCNN. In the results, the matching probabilities are considered high-level, whereas the non-matching probabilities are considered low-level. This indicates that the images of the same scene can be considered to be highly similar by the proposed SCNN in spite of severe intensity changes (as described in Figure 3) between images. We can thus infer that the proposed SCNN is robust to linear and nonlinear transformations, such as illumination, cloud cover, and land cover changes.



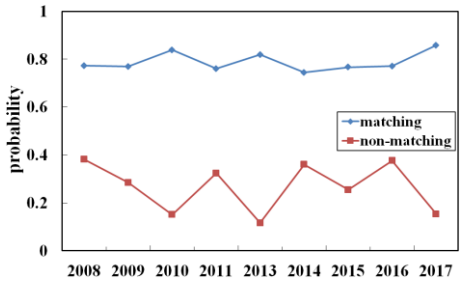
(a) Image in 2008 as reference image. **(b)** Image in 2009 as reference image.



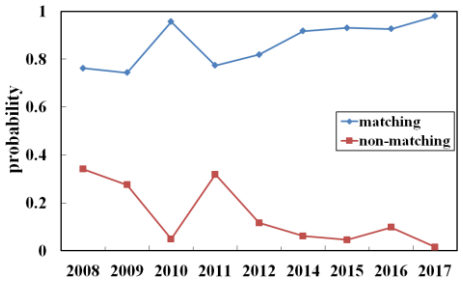
(c) Image in 2010 as reference image.



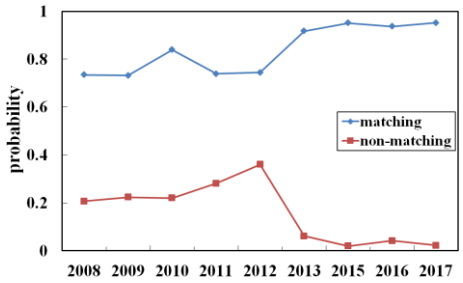
(d) Image in 2011 as reference image.



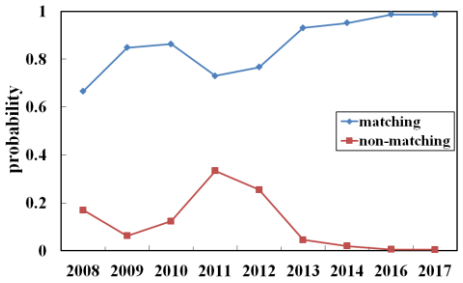
(e) Image in 2012 as reference image.



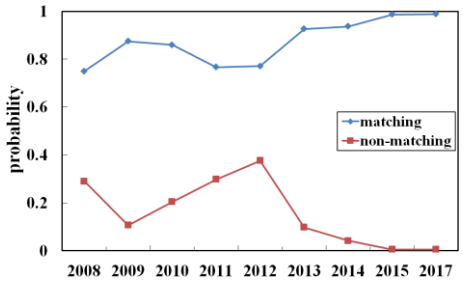
(f) Image in 2013 as reference image.



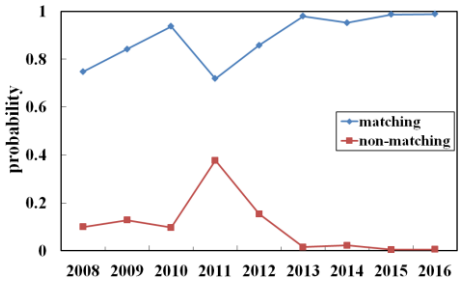
(g) Image in 2014 as reference image.



(h) Image in 2015 as reference image.



(i) Image in 2016 as reference image.



(j) Image in 2017 as reference image.

Figure 4. Matching and non-matching probabilities between multi-temporal remote image patches in Figure 3. (**a**–**j**) show the statistical results from 2008 to 2017.

2.2. Coarse-Conjugated Patch Decision

Considering that many similar objects or patterns may exist in different image areas in remote sensing images, traditional matching methods that use exhaustive searching strategies may cause matching ambiguity and generate false matches. To limit the search area and improve efficiency, a GPCQ-based coarse-to-fine method is exploited to find the conjugated area in this study. Image patches with fixed sizes are extracted from the reference and sensed images. Patch comparison is performed from the top of the Gaussian image pyramid to the bottom, as shown in Figure 5. First, a

Gaussian image pyramid is established based on the given size of patches. If the given size of patches is $d \times d$, then the image size in the top pyramid is $2d \times 2d$ and the size of next layer is $4d \times 4d$. Thus, one upper layer has half spatial resolution of a pixel, so that we can cover wider area with a fixed image size in a higher layer and conduct coarse searching. In this paper, the minimum number of layers is set as 5. The bottom layer is the original image. If the number of layers established is less than 5, the original image will be enlarged to reach the minimum number of layers. Second, the Gaussian image pyramid is split into a series of patches with fixed size $d \times d$ based on the quadtree principle. Third, SCNN is used to compare the similarity of the reference and the sensed patches in the current above-and-below scales of the Gaussian image pyramid in the range of the conjugated patches obtained from one upper layer. The patch similarity is defined based on the difference of matching probability $p^{(m)}$ and non-matching probability $p^{(nm)}$, to which $\left(p_{i,j}^{(m)} - p_{i,j}^{(nm)}\right)_{max}$ denotes the maximum difference of the i^{th} patch in the sensed image to the j^{th} patch in the reference image. This definition satisfies the constraint in Equation (13), and patches i and j are regarded as a pair of conjugated regions.

$$\frac{\left(p_{i,j}^{(m)} - p_{i,j}^{(nm)}\right)_{\substack{2nd \\ max}}}{\left(p_{i,j}^{(m)} - p_{i,j}^{(nm)}\right)_{max}} < RA \tag{13}$$

in which $2nd\ max$ is the second maximum difference on the i^{th} patch in the sensed image while RA is the ratio threshold set to 0.6.

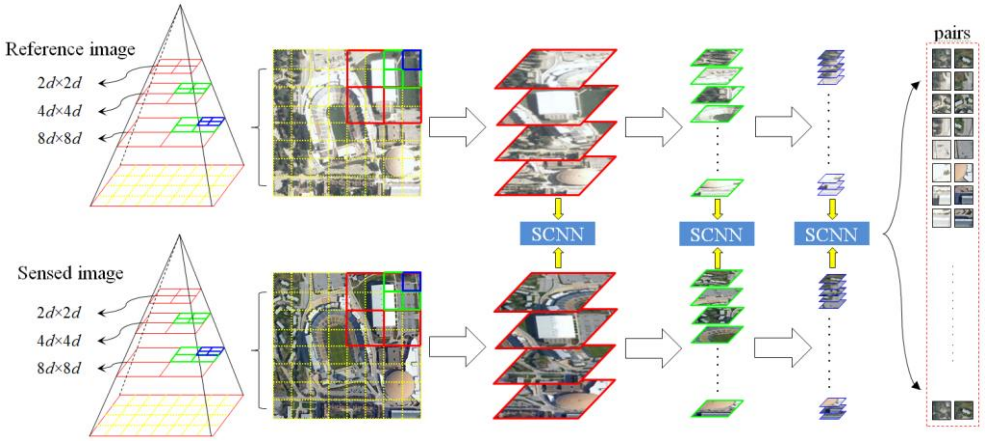


Figure 5. Patch comparison via GPCQ. The red rectangles are the patches, which are located at the top layer of Gaussian image pyramid. For example, four patches with sizes $d \times d$ are found in the top pyramid layer with size $2d \times 2d$, in which d is set to 96 pixels. The green and blue rectangles are the patches in the second and third layers, respectively. SCNN is used to compare the similarity between the patches in the reference and sensed images.

2.3. Multiscale-Conjugated Point Decision

Coarse-conjugated patches are found using the method described in the previous section. However, the conjugated patches from the comparison of grids with fixed sizes are difficult to locate precisely, and a 2D shift between coarse-conjugated patches exists. Furthermore, the center of the coarse patches cannot be easily used to obtain the accurate localization of the matches. In Ref. [32], normalized cross-correlation (NCC) is adopted to determine the precise conjugated points of optical and synthetic aperture radar images in Siamese-type networks. Score maps for the searching space with 51 × 51 pixels are generated from the reference and sensed images, and the high peak in the score map is considered a conjugated point location. However, the NCC is a time-consuming method.

In this study, we use a fast-point localization method based on the Harris algorithm [33], which is regarded as a simple and efficient approach. Only the corners of the coarse-conjugated

patches, rather than every pixel in the patch, are selected as candidates to find the precise point locations. The algorithm is expressed below.

Algorithm:

Input: P_s and P_r are the coarse-conjugated patches in the sensed and reference images, respectively; $C_{i=1}^{n}$ are the Harris corners in P_s and P_r, and n is the number of Harris corners

Parameters: matching probability p_m, non-matching probability p_{nm}, and similarity index $Si = \frac{p_m - p_{nm}}{p_{nm}}$.

Compute the distance of center to Harris corners $d(C_{i=1}^{n})_s$ and $d(C_{i=1}^{n})_r$ in the patches.

Traverse the Harris corners based on $d(C_{i=1}^{n})$ from min to max.

for $i = 1$ to n_s **do**

  for $j = 1$ to n_r **do**

    Compute $Si_{sr} \leftarrow \frac{p_m - p_{nm}}{p_{nm}}$

    if $(Si_{sr})_{max} < Si_{sr}$ **then**

      Update $(Si_{sr})_{max} \leftarrow Si_{sr}$

  end for

end for

Record the coordinates of Harris corners with $(Si_{sr})_{max}$.

The location accuracy of the original Harris algorithm is expressed at the pixel level. To achieve sub-pixel accuracy, we utilize the improved sub-pixel level Harris operator (S-Harris). The neighbors of the Harris corner, rather than by using local non-maximum suppression of a response function, are considered for corner detection. The least square method is adopted to refine the location of the Harris corner at the sub-pixel level.

$$J = V^{\mathrm{T}} P V \tag{14}$$

in which $V = [\hat{x} - x_1, \hat{y} - y_1, \hat{x} - x_2, \hat{y} - y_2, \cdots, \hat{x} - x_n, \hat{y} - y_n]^{\mathrm{T}}$; $(\hat{x}, \hat{y})$ and (x_n, y_n) are the coordinates of the refined corner and neighbors, respectively; n is the number of neighbors; and P is the diagonal weight matrix $diag\,[p_{w1}, p_{w1}, p_{w2}, p_{w2}, \cdots, p_{wn}, p_{wn}]$, which is computed with a corner response value. To ensure that the corners are evenly distributed, the reference and sensed images are divided into fixed grid sizes (i.e., 96×96 pixels in this paper). After non-maximum suppression, the pixels in each grid are sorted in descending order based on their corner response values. The first 100 pixels (the given number) are considered as the corners of each grid. If the total number of S-Harris corners is less than 3000 in an image, the given number of each grid is set to 200. In this paper, the S-Harris with grids is named gridding S-Harris, while the corners detected in the whole image are named non-gridding S-Harris.

Matching S-Harris corners with a fixed window is difficult to accomplish for remote sensing images with changed scales. To find the scale-invariant matches, the multiscale patches are compared synchronously to capture the S-Harris corners for the initial matches.

2.4. Outlier Elimination

Outliers are inevitable in initial matching. Thus, polynomials and the RANSAC algorithm are usually combined to eliminate outliers. However, local geometric distortions may be inconsistent for different terrains or land covers. To overcome this problem, a whole-to-local outlier elimination is implemented from the top to bottom layers of the Gaussian pyramid. The main steps of outlier elimination are as follows:

Step 1: Find the correct match set S_{CM} for the top layer of the Gaussian pyramid by using geometric transformation and RANSAC.

Step 2: In the next Gaussian pyramid layer, validate all initial matches by using local polynomials. As shown in Figure 6, six correct matches for the initial match (P_1, P_2) are selected to solve the local polynomial coefficients, and point P_2' is estimated with P_1. If the residual error of

P_2 and P_2' is less than the triple standard deviation, then (P_1, P_2) is regarded as a pair of correct matches and is saved in set S_{CM}.

Step 3: Repeat Step 2 until the validation task is completed for all the Gaussian pyramid layers.

In the validation, if more than six matches are found around a point, then the quadratic polynomial of Equation (15) is selected to fit the local geometric transformation. Otherwise, the affine transformation of Equation (16) is used to describe the local geometric transformation.

$$\begin{cases} x_1 = a_0 + a_1 x_2 + a_2 y_2 + a_3 x_2^2 + a_4 x_2 y_2 + a_5 y_2^2 \\ y_1 = b_0 + b_1 x_2 + b_2 y_2 + b_3 x_2^2 + b_4 x_2 y_2 + b_5 y_2^2 \end{cases}, \tag{15}$$

in which (x_1, y_1) and (x_2, y_2) are the coordinates of the matches in the reference and sensed images, respectively, while $a_0, a_1, \ldots, a_5$ and $b_0, b_1, \ldots, b_5$ are the polynomial coefficients.

$$\begin{cases} x_1 = c_0 + c_1 x_2 + c_2 y_2 \\ y_1 = d_0 + d_1 x_2 + d_2 y_2 \end{cases}. \tag{16}$$

in which (x_1, y_1) and (x_2, y_2) are the coordinates of the matches in the reference and sensed images, respectively, while c_0, c_1, c_2 and d_0, d_1, d_2 are the coefficients of the affine transformation.

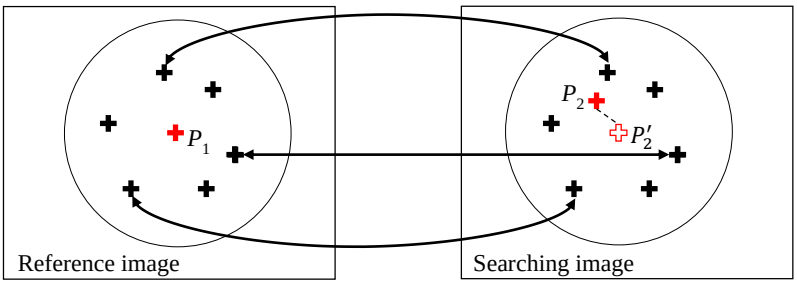


Figure 6. Local outlier elimination. (P_1, P_2) denotes the initial matches of the reference and sensed images. P_2' is estimated from P_1 based on local polynomial coefficients.

3. Experimental Evaluation and Discussion

3.1. Experimental Datasets

The training datasets are generated from Google Earth historical images. The datasets include rivers, coastlines, roads, farmlands, forests, mountains, and buildings in urban and rural areas. The datasets also contain patches with different periods, scales, illuminations, shadows, and land cover changes. A total of 80,000 pairs of patches (half matching and half non-matching patches) with a fixed size of 96 × 96 pixels are extracted in a supervised manner from Google Earth historical images. The non-matching patches examples were generated by two ways: firstly, we randomly select patches from different matching patch pairs to construct non-matching patches; besides, some examples are produced by cropping similar objects (e.g., two different buildings) from Google Earth historical images. Furthermore, 240,000 pairs of patches are extended to avoid overfitting by image rotation, Gaussian blur, and affine transformation. Therefore, the total number of patch pairs is 320,000, in which 312,000 and 8000 pairs of patches are randomly selected as training and test datasets, respectively.

ZY3, GF1, IKONOS, and Google Earth high-resolution remote sensing images with complex background variations are selected to construct the image pairs shown in Figure 7. Then, the proposed matching framework is evaluated. Different objects (buildings, rivers, coastlines, and forests) and different types of terrain are found in the images. A detailed description of each image pair is shown in Table 1.

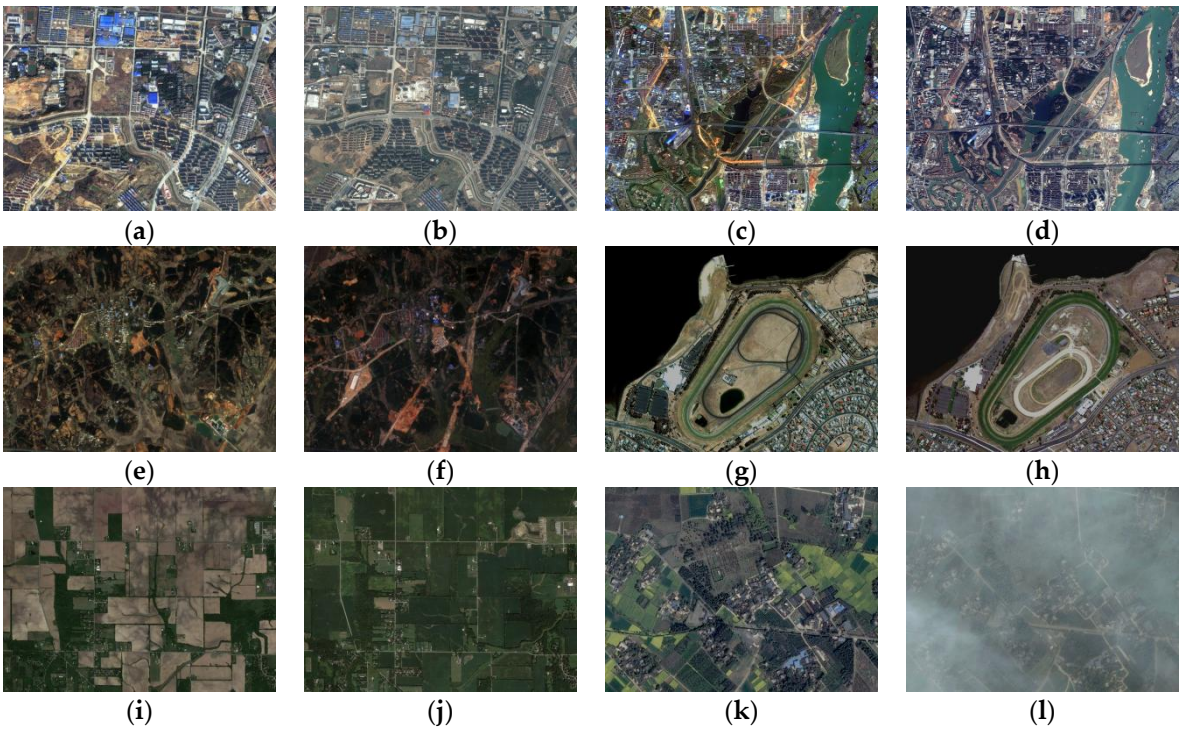


Figure 7. Experimental image pairs. (**a**,**b**) is a pair of ZY3 (fusion image obtained from multispectral and panchromatic images) and Google Earth images in an urban area in China. (**c**,**d**) is a pair of GF1 (fusion image obtained from multispectral and panchromatic images) and Google Earth images in China. (**e**,**f**) is a pair of ZY3 and GF1 images with large background variations in a mountain area in China. (**g**,**h**) is a pair of IKONOS and Google Earth images with coastline in Australia. The images in (**i**,**j**) are a pair of Google Earth images with farmlands in different seasons in the United States. (**k**,**l**) is a pair of Google Earth images in China, in which (**l**) is contaminated by cloud and haze.

Table 1. Description for each pair of images in the experiments.

Pairs	Image Number	Year	Image Source	Image Size (Unit: Pixel)	Spatial Resolution (Unit: Meter)
Pair 1	(a)	2013	ZY3	1000 × 750	2.10
	(b)	2017	Google Earth	1765 × 1324	1.19
Pair 2	(c)	2015	GF1	1972 × 1479	2.00
	(d)	2017	Google Earth	3314 × 2485	1.19
Pair 3	(e)	2013	ZY3	780 × 585	5.80
	(f)	2015	GF1	565 × 424	8.00
Pair 4	(g)	2003	IKONOS	1190 × 893	1.00
	(h)	2017	Google Earth	1000 × 750	1.19
Pair 5	(i)	2016	Google Earth	1936 × 1452	1.19
	(j)	2016	Google Earth	1936 × 1452	1.19
Pair 6	(k)	2015	Google Earth	1686 × 1264	1.19
	(l)	2016	Google Earth	1686 × 1264	1.19

3.2. SCNN Training

To improve the reliability of the training samples derived from multi-temporal remote sensing images with background variations, a batch size of 200 is used, which is larger than the batch size of 128 in Ref. [19]. Therefore, 1560 iterations exist in each round. The SCNN is trained in parallel on Nvidia GPUs within 100 rounds, and the training is forced to terminate when the average value of the loss function is less than 0.001. Weights are initialized for training by random Gaussian distributions [34]. The momentum and weight decay are set to 0.9 and 0.0005, respectively. Then,

the learning rate is reduced to accelerate the training and obtain good performance. In this study, a piecewise function is adopted to adjust the learning rate. The initial learning rate is set to 0.01 then decreased gradually with the following formula:

$$\eta_{iter} = \begin{cases} \alpha * \eta_{iter-1}, & if\ (iter\ \%\ 100 = 0) \\ else & \dfrac{\eta_{iter-1}}{1 + 2.5 * \eta_{iter-1}} \end{cases} \quad (17)$$

in which $iter$ denotes the number of iterations; η_{iter} denotes the learning rate of the $iter^{th}$ iteration, which is updated based on previous learning rate η_{iter-1}; % is an operator for computing the remainder; the optimal convergence can be achieved by decreasing the learning rate at about every 100 iterations based on the observation of our experiments; and α is a constant, which is set to 0.75.

3.3. Feature Visualization

Figure 8 shows the visualization of features at each convolutional layer (Conv1–Conv6) of the SCNN after ReLU activation. The feature maps show one of the features in the convolutional layer, e.g., one of 64 features is shown in the feature maps labeled Conv1. We can see that low-level texture information is captured in Conv1, and many high-level semantic features are extracted in deep convolutional layers. For example, the feature maps labeled as Conv3 in Figure 8a highlight high-rise residential community regions, and Conv4 in Figure 8b highlight the regions with bodies of water. The two compared feature maps in each convolutional layer for all pairs are similar, which demonstrates that the SCNN is suitable for the feature extraction of images with complex background variations.

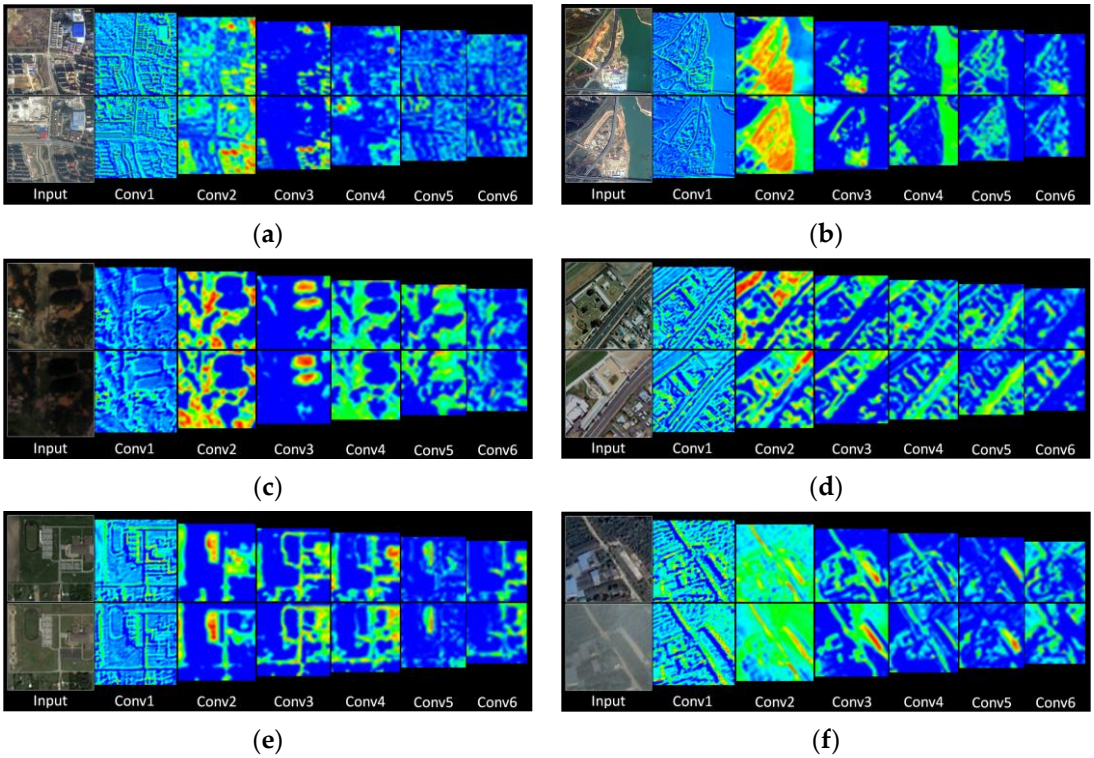


Figure 8. Examples of feature visualization learned by the proposed SCNN. (**a**), (**b**), (**c**), (**d**), (**e**) and (**f**) are the visual features in Pair1, Pair2, Pair3, Pair4, Pair5 and Pair6 respectively.

3.4. Evaluation Criteria of Matching Performance

Three indicators, namely, the number of correct matches (NCM), matching precision (MP), and root mean square error ($RMSE$), are used to evaluate the proposed method in our experiments. MP and $RMSE$ can be computed as follows:

$$MP = \frac{NCM}{NTM} \times 100\% \tag{18}$$

in which NTM is the number of total matches (initial matches).

$$RMSE = \sqrt{\frac{1}{NCM}\sum_{i=1}^{NCM}[(x - x')^2 + (y - y')^2]} \tag{19}$$

in which (x, y) is the coordinate of the correct matches in the sensed image, while (x', y') is the transformed coordinate of the correct matches in the reference image. The NCM is counted and manually checked in our experiments.

3.5. Comparison of SCNNs with Different Numbers of Layers

To evaluate the effects of layer number in our method, we reduced (layer−) and added (layer+) one convolutional layer to train and test the datasets. The performance is evaluated by determining average accuracy (AA), which is computed as follows:

$$AA = \frac{1}{iters}\sum_{i=1}^{iters}\frac{PMN_i}{TMN}, \tag{20}$$

in which $iters$ is the number of iteration in each round; PMN_i is the number of positives for matching and non-matching pairs in the i^{th} iteration; and TMN is the total number of pairs.

Figure 9 shows the average accuracy of each round for the training and test data with layer− and layer+. Our network and the deep network (layer+) achieved higher accuracy by nearly 3% compared with layer−. Layer+ converged slower than our network. Layer− and our network converged at nearly 10 rounds (1.56×10^4 iterations), whereas layer+ converged at nearly 18 rounds (2.808×10^4 iterations). In addition, our network and layer+ performed better than layer− in terms of NCM, MP, and $RMSE$, as shown in Figure 10. The experimental results demonstrate the effective performance of our network given the tradeoff between accuracy and network complexity.

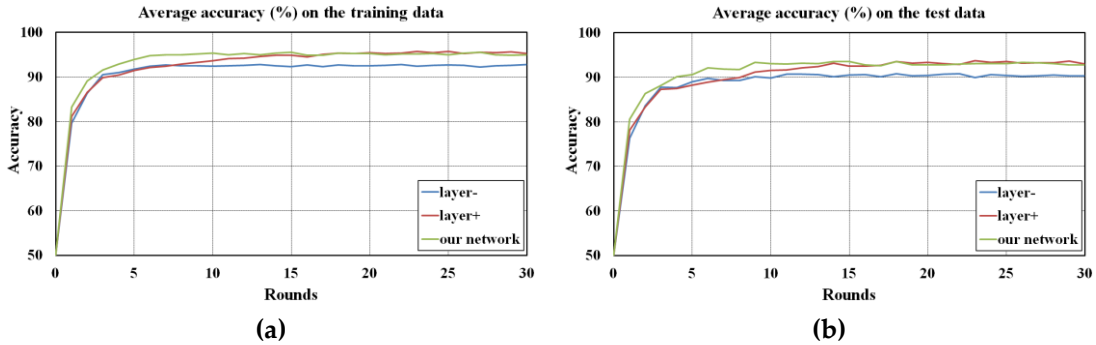


Figure 9. Comparison of average accuracies for each round between training (**a**) and test (**b**) data with layer−, layer+, and our network.

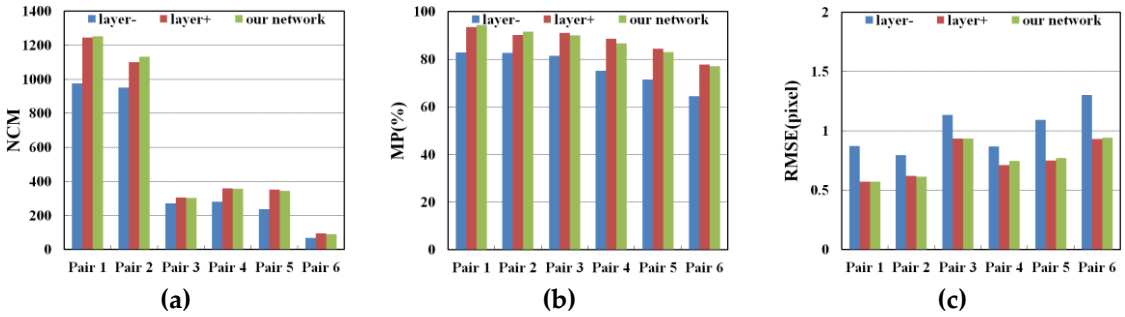


Figure 10. Comparison of (**a**) NCM, (**b**) MP, and (**c**) RMSE values with different deep SCNNs.

3.6. Comparison between Gridding and Non-Gridding S-Harris

The gridding S-Harris detector is compared with the non-gridding S-Harris for the proposed matching framework. The filtering radius $r = 5$ pixels and standard deviation $\sigma = 0.8$ are used for Gaussian filtering. The corner response value R is computed as $R = \det M - k * (\text{trace}M)^2$, in which M is a Harris matrix and k is set as 0.04. The radius of the local window is set to 7 pixels. The computed NCM and RMSE of the gridding S-Harris and non-gridding S-Harris are shown in Figure 11.

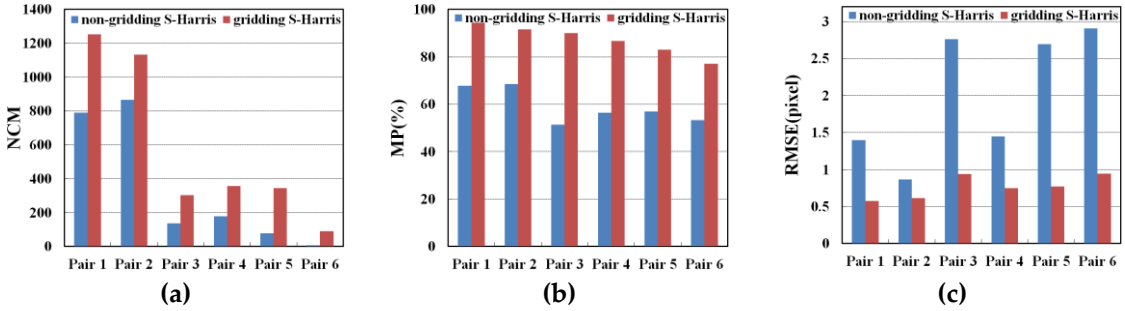


Figure 11. Comparison of (**a**) NCM, (**b**) MP, and (**c**) RMSE between gridding S-Harris and non-gridding S-Harris.

As shown Figure 11, the gridding S-Harris performs better than the non-gridding S-Harris in terms of NCM, MP, and RMSE. This finding is attributed to the detection of a fixed number of corners in each grid that is not easily affected by image grayscale variations. Then, the evenly distributed corners are obtained to accurately compute the geometric transformation (as shown in Section 3.8). Many homogenous or weakly textured areas can be seen in the images of Pairs 5 and 6, while large background variations exist in Pairs 3 and 4. In those images, unevenly distributed corners and matching ambiguity caused by non-gridding S-Harris significantly reduced the number of matches and accuracy.

3.7. Evaluation of GPCQ

GPCQ is adopted in our method to narrow down the searching space of conjugated points and reduce matching ambiguity. Then, the proposed matching frameworks with and without GPCQs are compared for the performance evaluation of the GPCQ. In the test, the patch size is set to 96×96 pixels. The SCNN-based pairwise similarity measure and the quadratic polynomial constraint are used to obtain the matches through a bi-directional matching strategy. Figure 12 shows the comparative results in terms of the NCM, MP, and RMSE.

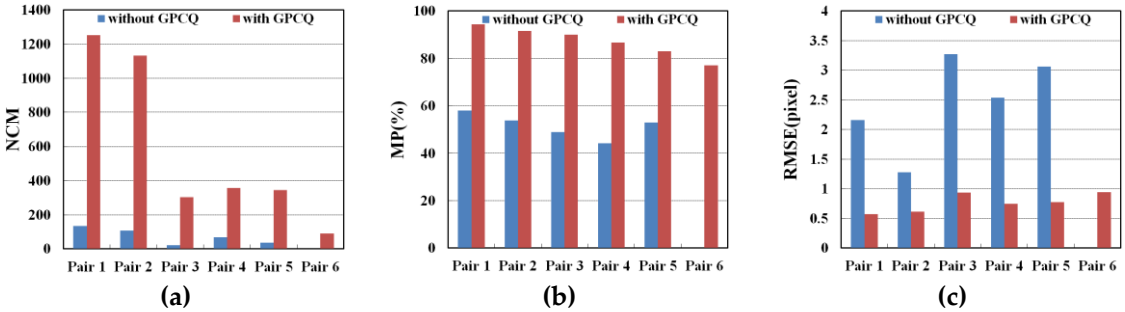


Figure 12. Comparison of (**a**) NCM, (**b**) MP, and (**c**) RMSE with and without GPCQs.

On the basis of the experimental results, the proposed matching framework with GPCQ performed better than the framework without GPCQ. This finding is attributed to two reasons. First, many local image contents may be similar in the different areas of the same image, which result in many patches with low distinctiveness. The images in Figure 7i,j,l contain more highly similar local regions compared with the other experimental images. Thus, the exhaustive searching strategy may have produced matching ambiguity in the proposed framework without GPCQ, such that only a few correct matches are obtained. Second, the limited, unevenly distributed matches hindered the implementation of an accurate polynomial geometric registration.

3.8. Performance Evaluation of the Proposed Matching Framework

To evaluate its performance, the complete framework is compared with SIFT and other three state-of-the-art matching methods, namely, two matching methods for remote sensing images with background variations (i.e., see Jiang [5] and Shi [20]) and a 2-channel deep network-based method (i.e., see Zagoruyko [19]). The comparative NCM, MP, and RMSE values are shown in Tables 2–4, respectively. In the experiments, the initial matches of RMSE over 3 pixels are considered to be false matches for all comparative methods. The visualization results of the proposed matching framework and the comparative methods are shown in Figures 13–17. The proposed matching framework can obtain many correct and regularly distributed matches for all experimental pairs, and it can achieve relatively higher accuracy of less than 1 pixel, as shown in Table 4.

Table 2. NCM values of each method for remote sensing images with complex background variations.

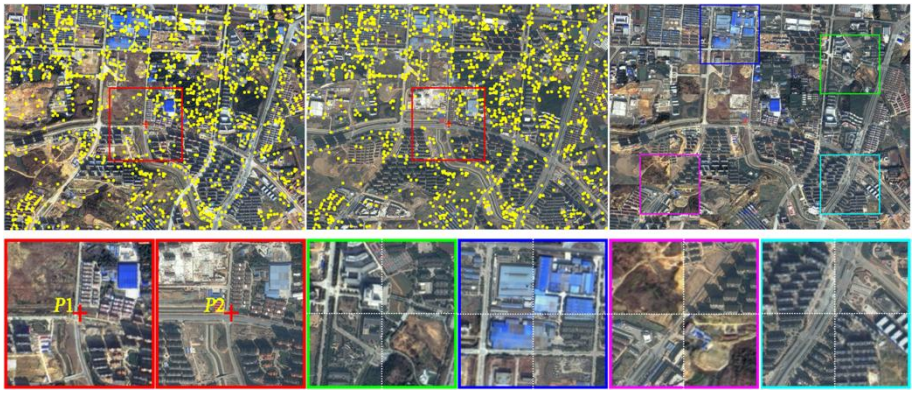
Image Pair	Matching Methods				
	SIFT	**Jiang**	**Shi**	**Zagoruyko**	**Proposed**
Pair 1	93	13	24	39	**1253**
Pair 2	69	19	25	71	**1132**
Pair 3	10	9	13	0	**303**
Pair 4	0	0	9	0	**356**
Pair 5	14	0	7	0	**345**
Pair 6	0	0	0	0	**91**

Table 3. MP values of each method for remote sensing images with complex background variations.

Image Pair	Matching Methods				
	SIFT	**Jiang**	**Shi**	**Zagoruyko**	**Proposed**
Pair 1	51.8%	68.3%	73.6%	85.7%	**94.3**%
Pair 2	54.2%	76.2%	80.4%	84.6%	**91.6**%
Pair 3	42.6%	71.5%	77.8%	0.0%	**89.9**%
Pair 4	0.0%	0.0%	79.1%	0.0%	**86.7**%
Pair 5	60.4%	0.0%	66.2%	0.0%	**82.9**%
Pair 6	0.0%	0.0%	0.0%	0.0%	**77.1**%

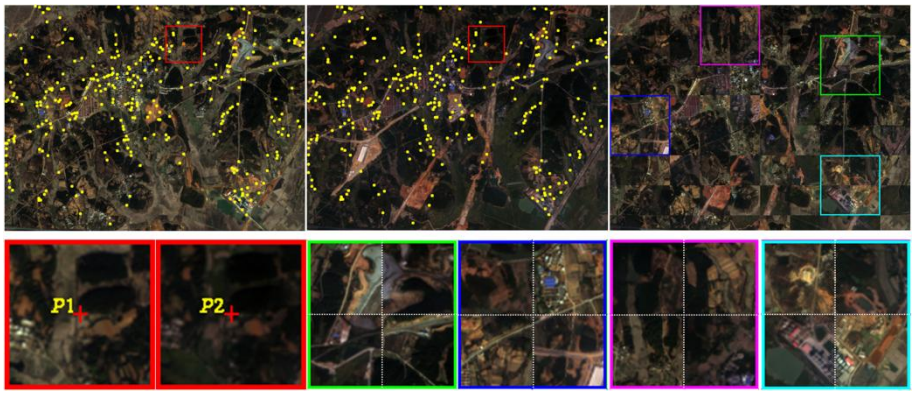
Table 4. RMSE values of each method for remote sensing images with complex background variations.

Image Pair	Matching Methods				
	SIFT	Jiang	Shi	Zagoruyko	Proposed
Pair 1	0.8732	2.9674	2.1536	0.9657	**0.5736**
Pair 2	0.9485	2.1453	2.4665	0.9054	**0.6143**
Pair 3	2.4153	2.7478	2.5833	Null	**0.9372**
Pair 4	Null	Null	2.0751	Null	**0.7476**
Pair 5	1.7834	Null	2.9887	Null	**0.7732**
Pair 6	Null	Null	Null	Null	**0.9426**

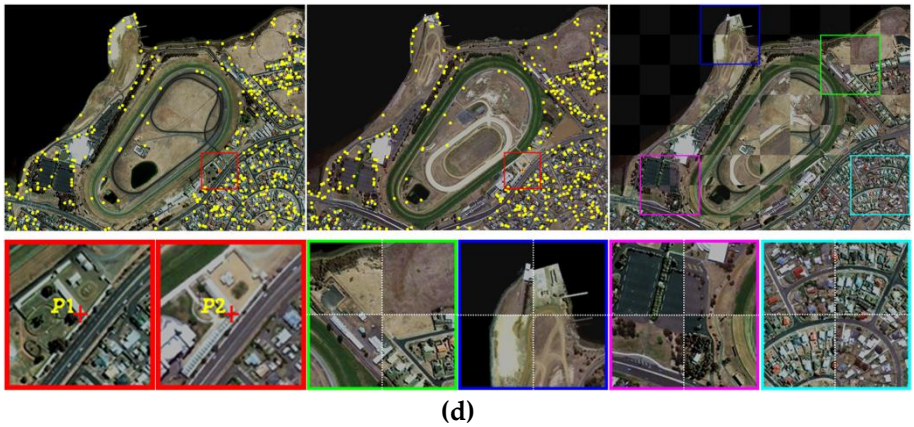


(a)

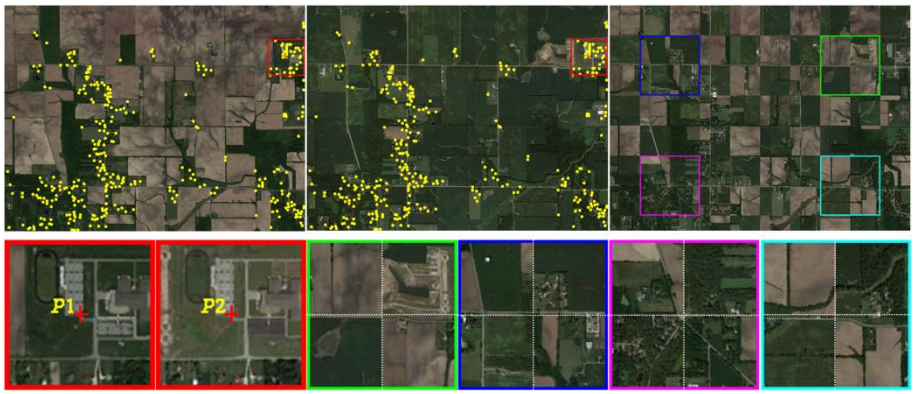
(b)



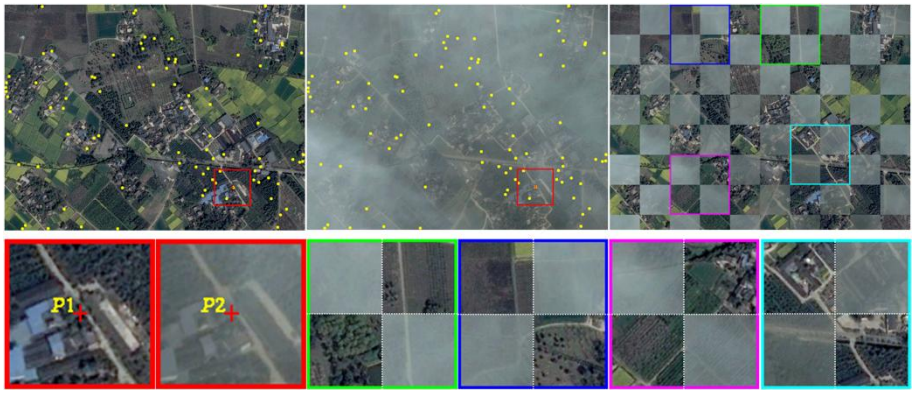
(c)



(d)

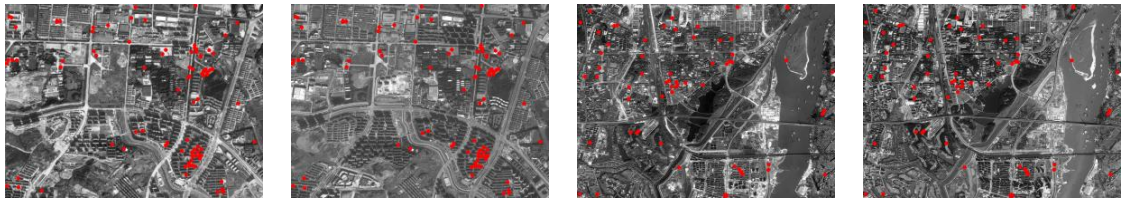


(e)



(f)

Figure 13. Matching and registration results of the proposed matching framework. The matches of Pairs 1–6 are pinned to the top-left two images of (**a**–**f**) using yellow dots. The two small sub-regions marked by red boxes correspond to the two conjugated patches P1 and P2. The top-right image shows the registration result of the checkerboard overlay of the image pair. The four small sub-regions marked by green, blue, magenta, and cyan are enlarged to show the registration details.



(a) **(b)**

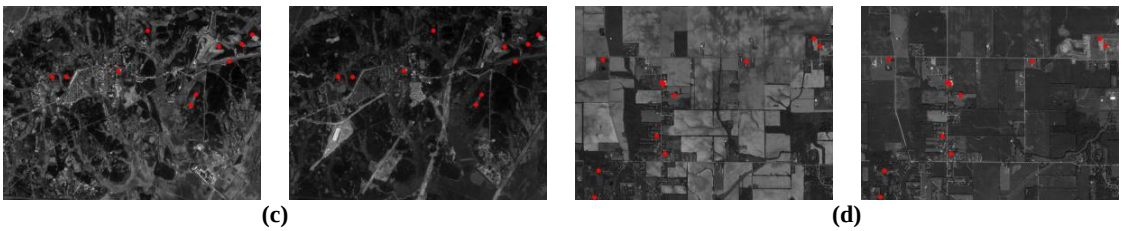


Figure 14. Matching results of SIFT. The matches of Pairs 1, 2, 3, and 5 are shown in (**a**), (**b**), (**c**), and (**d**), respectively. No correct match is obtained for the images of Pairs 4 and 6 (i.e., see Table 2).

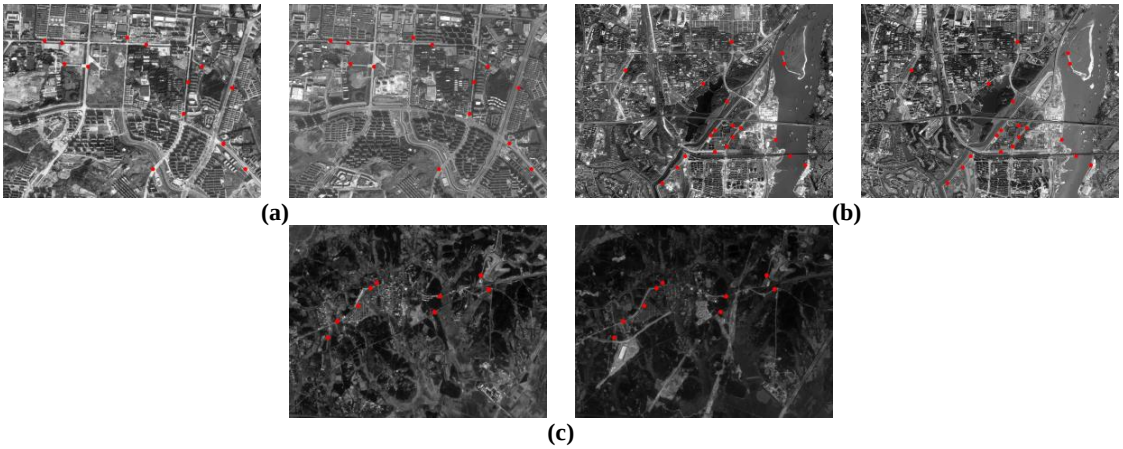


Figure 15. Matching results using Jiang's method [5]. (**a**–**c**) are the matching results of Pairs 1–3. No correct match is obtained for the images of Pairs 4–6 (i.e., see Table 2).

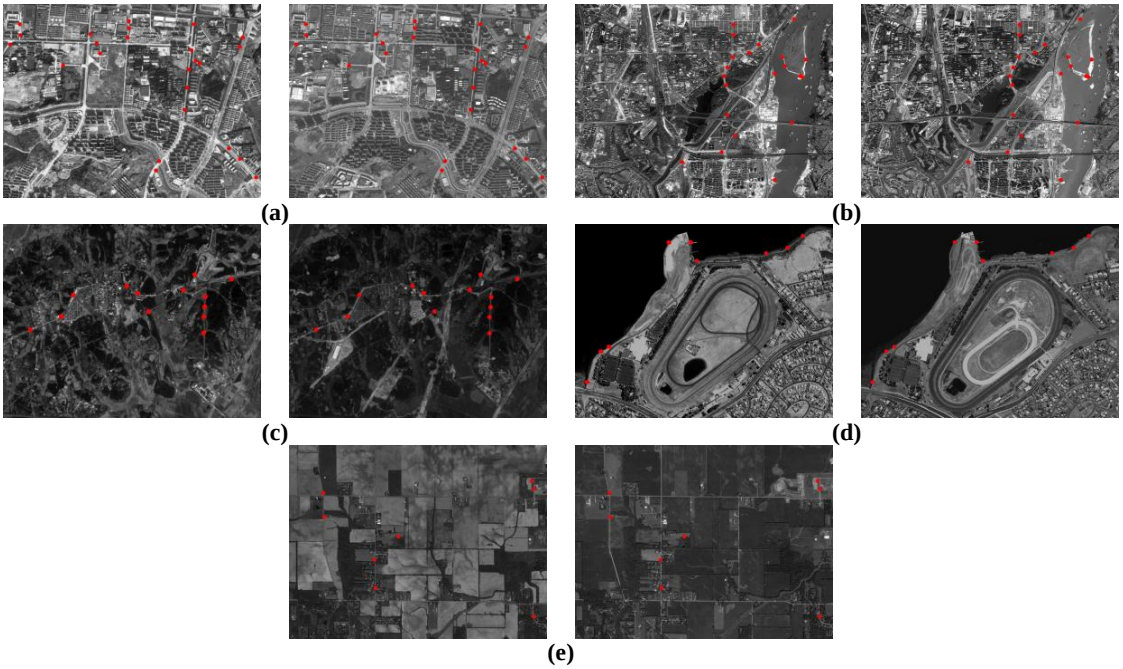
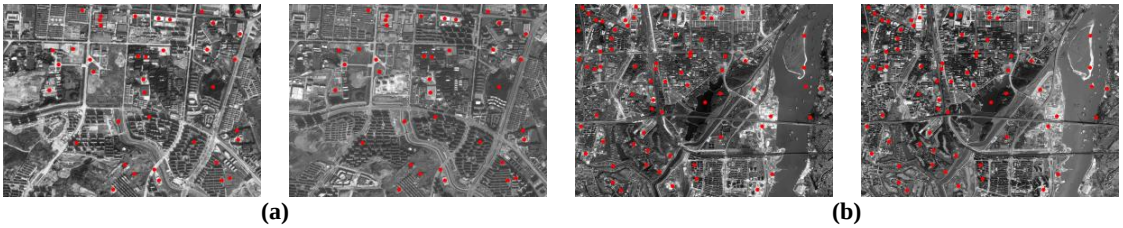


Figure 16. Matching results using Shi's method [20]. (**a**–**e**) are the matching results of Pairs 1–5. No correct match is obtained for the image of Pair 6 (i.e., see Table 2).



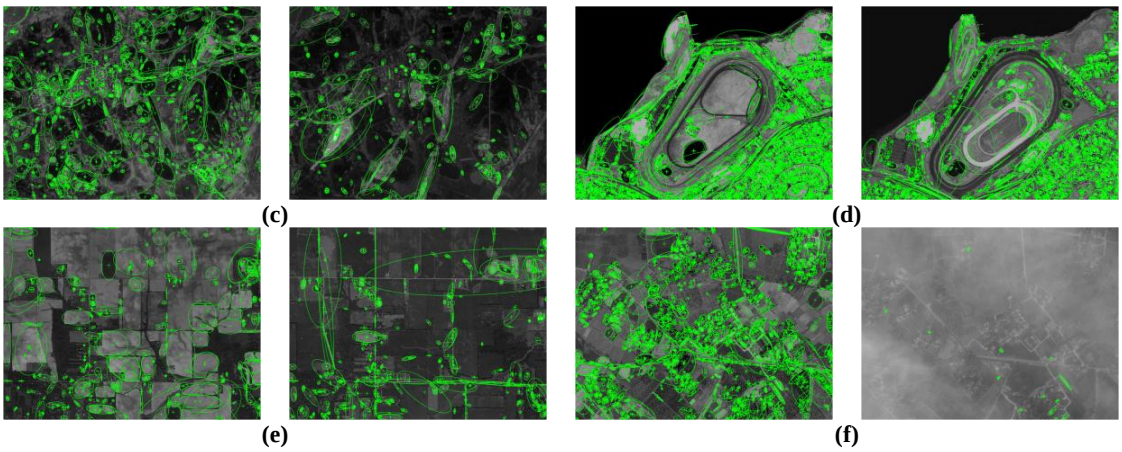


Figure 17. Matching results using Zagoruyko's method [19]. (**a**,**b**) are the matching results of Pairs 1 and 2, respectively. No correct match is obtained for the images of Pairs 3–6 (i.e., see Table 2). (**c**–**f**) highlights the ellipse and centroids of MSER of Pairs 3–6.

Table 2 presents the comparative results of the five methods in terms of NCM. Except for our method, the methods used for the comparison are sensitive to image quality degradation and failed in Pair 6. The matches obtained by using the methods of Jiang and Shi are mainly distributed on the line objects, as shown in Figures 15 and 16, respectively. The method of Zagoruyko, a deep learning-based method that combined 2-channel deep network with MSER [35], worked well for Pairs 1 and 2, as shown in Figure 17a,b, respectively. However, the method of Zagoruyko is less effective than the four other methods, because few corresponding MSERs are detected, as shown in Figure 17c–f. The MP values of the methods of Jiang and Shi are higher than those of the SIFT method, and their corresponding correct matches are less. Moreover, as shown in Table 4, the RMSEs of the methods of Jiang and Shi are over 2 pixels. These results can be attributed to the line location determined by the edge detector, which is easily affected by complex background variations. The methods of Jiang and Shi were less effective than SIFT in terms of RMSE.

On the basis of experimental results, the proposed framework presented more significant improvements than the other methods in terms of matching performance when remote sensing images with complex background variations were used. The effectiveness of the proposed matching framework can be explained by a number of reasons. First, the deep and abstract features obtained by training are more salient than the manually designed features. As shown by the feature maps of the conjugated patches in Figure 8, the conjugated features are highly similar despite the existence of significant background variations. Second, the gridding S-Harris operator can find evenly distributed corners with high-location accuracy. Third, the GPCQ-based searching strategy can improve the comparison reliability of conjugated patches. In the GPCQ algorithm, the Gaussian pyramid-based multiscale patch similarity comparison and quadtree can reduce the searching space of S-Harris corners and improve the robustness of comparison. Finally, the quadratic polynomial-based whole-to-local outlier elimination method can remove false matches and improve matching precision.

In the SIFT method, the difference-of-Gaussian detector and the gradient-based SIFT descriptor are both sensitive to significant changes in intensity caused by complex background variations. However, background variations may result in local support regions with different image contents for each SIFT feature. Subsequently, the SIFT method generates descriptors with low similarity, thereby resulting in many outliers. In the methods of Jiang and Shi, the matches are produced from relatively stable shapes and structures, such as coastlines and roads. However, these kinds of line objects seldom present rich and regularly distributed contents in remote sensing images, and thus, they are unsuitable for obtaining satisfactory point matching results. In addition, the edge detector is sensitive to nonlinear grayscale changes. If changes occur between the line objects (e.g., inconsistent shape of coastlines in Figure 7g,h), then the location accuracy of these lines will be low. In the method of Zagoruyko, the MSER detector is sensitive to local noise and structure component

changes caused by complex background variations. A highly similar shape context for conjugated regions is difficult to detect. The failed examples in Figure 17c–f suggest that the MSER is unstable for complex background variations. As evidenced by the experimental results, the line- and region-based methods cannot fully guarantee good matching results when using remote sensing images with complex background variations, because the intensity information is unreliable.

4. Conclusions

In this paper, we present the SCNN-based matching framework to effectively find matches between remote sensing images with complex background variations. First, a Siamese-type network is designed to directly learn the similarity of conjugated patches. Second, a gridding S-Harris algorithm is developed to determine the coordinates of point matches, and the GPCQ-based searching strategy is used to narrow down the searching space and achieve multiscale comparison of conjugated patches. Finally, a whole-to-local quadratic polynomial constraint is used to remove the false matches. The deep features detected by the SCNN are more distinctive and unambiguous compared with the manually designed features. Furthermore, the statistical and visualization results indicate that our framework can obtain more well-distributed matches and higher matching precision and accuracy for all experimental image pairs compared with the state-of-the-art matching methods. The results proved the high capability of the proposed framework in matching remote sensing images with complex background variations.

However, the proposed matching framework is constrained by the unknown spatial resolutions of the reference and sensed images. The reference and sensed images should be resampled by approximating the same spatial resolution before matching. In addition, the image matching framework is proposed to improve matching performance in the context of image background variations, in which there should be significant texture changes, so some images with discriminative textures were collected to generate the training and test datasets. As we know, the accuracy of supervised machine learning highly depends on the quality and variety of the training data set. As a result, a lot of correct matches can be obtained in the images with discriminative textures (i.e., Figure 13a,b), while only a few correct matches were obtained in some areas with homogeneous or weak textures (i.e., the farmland areas with homogeneous textures in Figure 13e, the weakly textured areas covered by thick cloud in Figure 13f). Therefore, the proposed image matching framework may not be suitable for the images fully covered by homogeneous or weak textures.

Our future work may focus on improving the matching performance of the proposed method for images without known spatial resolutions or with several homogenous areas.

Acknowledgments: This research was supported by the National Key Research and Development Program of China (2016YFB0502603), National Natural Science Foundation of China (41401526 and 41501492), and Jiangxi Natural Science Foundation of China (20171BAB213025). The authors would like to thank the editor-in-chief, the anonymous associate editor, and the reviewers for their systematic review and valuable comments.

Author Contributions: Haiqing He was primarily responsible for conceiving the method and writing the source code and the paper. Min Chen designed the experiments and revised the paper. Ting Chen generated the datasets and performed the experiments. Dajun Li improved the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Brown, L.G. A survey of image registration techniques. *ACM Comput. Surv.* **1992**, *24*, 325–376.
2. Maintz, J.B.A; Viergever, M.A. A survey of medical image registration. *Med. Image Anal.* **1998**, *2*, 1–36.
3. Zitová, B.; Flusser, J. Image registration methods: A survey. *Image Vis. Comput.* **2003**, *21*, 977–1000.
4. Dawn, S.; Saxena, V.; Sharma B. Remote sensing image registration techniques: A survey. In *International Conference on Image and Signal Processing*; Elmoataz, A., Lezoray, O., Nouboud, F., Mammass, D., Meunier, J., Eds.; Springer-Berlin: Heidelberg, Germany, 2010; Volume 6134, pp. 103–112.
5. Jiang, J.; Zhang, S.; Cao, S. Rotation and scale invariant shape context registration for remote sensing images with background variations. *J. Appl. Remote Sens.* **2015**, *9*, 92–110.

6. Yang, K.; Pan A.; Yang, Y.; Zhang, S.; Ong, S.H.; Tang, H. Remote sensing image registration using multiple image features. *Remote Sens.* **2017**, *9*, 1–21.
7. Chen M.; Habib A.; He H.; Zhu Q.; Zhang W. Robust feature matching method for SAR and optical images by using Gaussian-Gamma-shaped bi-windows-based descriptor and geometric constraint. *Remote Sens.* **2017**, *9*, 882.
8. Lowe, D. Object recognition from local scale-invariant features. In Proceedings of the 7th IEEE International Conference on Computer Vision, Kerkyra, Greece, 20–27 September 1999; p. 1150.
9. Bay, H.; Ess, A.; Tuytelaars, T.; Gool, L.C. Speeded-up robust features (SURF). *Comput. Vis. Image Und.* **2008**, *110*, 346–359.
10. Bradley, P.E.; Jutzi, B. Improved feature detection in fused intensity-range image with complex SIFT. *Remote Sens.* **2011**, *3*, 2076–2088.
11. Ke, Y.; Sukthankar, R. PCA-SIFT: A more distinctive representation for local image descriptors. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 27 June–2 July 2004; pp. 506–513.
12. Mikolajczyk, K.; Schmid, C. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal.* **2005**, *27*, 1615–1630.
13. Morel, J.M.; Yu, G. ASIFT: A new framework for fully affine invariant image comparison. *SIAM J. Imaging Sci.* **2009**, *2*, 438–469.
14. Li, Q.; Wang, G.; Liu, J.; Chen S. Robust scale-invariant feature matching for remote sensing image registration. *IEEE Geosci. Remote Sens. Lett.* **2009**, *6*, 287–291.
15. Brook, A.; Bendor, E. Automatic registration of airborne and spaceborne images by topology map matching with SURF processor algorithm. *Remote Sens.* **2011**, *3*, 65–82.
16. Chen, Q.; Wang, S.; Wang, B.; Sun, M. Automatic registration method for fusion of ZY-1-02C satellite images. *Remote Sens.* **2013**, *6*, 157–279.
17. Cai, G.R.; Jodoin, P.M.; Li, S.Z.; Wu, Y.D.; Su, S.Z. Perspective-SIFT: An efficient tool for low-altitude remote sensing image registration. *Signal Process.* **2013**, *93*, 3088–3110.
18. Li, J.; Hu, Q.; Ai, M. Robust feature matching for remote sensing image registration based on l_q-estimator. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1989–1993.
19. Zagoruyko, S.; Komodakis, N. Learning to compare image patches via convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 4353–4361.
20. Shi, X.; Jiang, J. Automatic registration method for optical remote sensing images with large background variations using line segments. *Remote Sens.* **2016**, *8*, 426.
21. Altwaijry, H.; Trulls, E.; Hays, J.; Fua, P.; Belongie, S. Learning to match aerial images with deep attentive architecture. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 27–30 June 2016; pp. 3539–3547.
22. Chen, L.; Rottensteiner, F.; Heipke, C. Invariant descriptor learning using a Siamese convolutional neural network. In Proceedings of the ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Prague, Czech Republic, 12–19 July 2016; pp. 11–18.
23. Simo-Serra, E.; Trulls, E.; Ferraz, L.; Kokkinos, I.; Fua, P.; Moreno-Noguer, F. Discriminative learning of deep convolutional feature point descriptors. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 118–126.
24. Ahmed, E.; Jones, M.; Marks, T.K. An improved deep learning architecture for person re-identificaiton. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3908–3916.
25. Melekhov, I.; Kannala J.; Rahtu E. Siamese Network Features for Image Matching. In Proceedings of the 23rd International Conference on Pattern Recognition, Cancun, Mexico, 4–8 December 2016; pp. 378–383.
26. Fischler, M.A.; Bolles, R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *ACM* **1981**, *24*, 381–395.
27. Brum, A.G.V.; Pilchowski, H.U.; Faria, S.D. Attitude determination of spacecraft with use of surface imaging. In Proceedings of the 9th Brazilian Conference on Dynamics Control and their Applications (DICON'10), Serra Negra, Brazil, 7–11 June 2010; pp. 1205–1212.

28. Kouyama, T.; Kanemura, A.; Kato, S.; Imamoglu, N.; Fukuhara, T.; Nakamura R. Satellite attitude determination and map projection based on robust image matching. *Remote Sens*. **2017**, *9*, 90; doi:10.3390/rs9010090
29. Zhang, H.; Sun, K.; Li, W. Object-oriented shadow detection and removal from urban high-resolution remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 6972–6982.
30. Cheng, Q.; Shen, H.; Zhang, L.; Li, P. Inpainting for remotely sensed images with a multichannel nonlocal total variation model. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 175–187.
31. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning (ICML–15), Lille, France, 6–11 July 2015.
32. Merkle, N.; Luo, W.; Auer, S.; Müller, R.; Urtasun, R. Exploiting deep matching and SAR data for the geo-localization accuracy improvement of optical satellite images. *Remote Sens.* **2017**, *9*, 586.
33. Harris, C. A combined corner and edge detector. In Proceedings of the 4th Alvey Vision Conference, Manchester, UK, 31 August–2 September 1988; pp. 147–151.
34. Brown, M.; Hua, G.; Winder, S. Discriminative learning of local image descriptors. *IEEE Trans. Pattern Anal*. **2011**, *33*, 43–57.
35. Matas, J.; Chum, O.; Urban, M.; Pajdla, T. Robust wide baseline stereo from maximally stable extremal regions. In Proceedings of the British Machine Vision Conference, Cardiff, UK, 2–5 September 2002; pp. 384–396.