IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 11, 2020, accepted November 28, 2020, date of publication December 3, 2020,
date of current version December 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3042213

K-Means Clustering Guided Generative Adversarial Networks for SAR-Optical Image Matching

WEN-LIANG DU[1], YONG ZHOU[1,3], JIAQI ZHAO[1], (Member, IEEE), AND XIAOLIN TIAN[2]
[1] School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China
[2] State Key Laboratory of Lunar and Planetary Sciences, Macau University of Science and Technology, Taipa, Macau
[3] Engineering Research Center of Mine Digitization, Ministry of Education of the People's Republic of China, Xuzhou 221116, China

Corresponding author: Yong Zhou (yzhou@cumt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61572505, Grant 62002360, Grant 61806206, Grant 61772530, and Grant U1610124; in part by the Six Talent Peaks Project in Jiangsu Province under Grant 2015-DZXX-010; in part by the Natural Science Foundation of Jiangsu Province under Grant BK20180639 and Grant BK20171192; in part by the China Postdoctoral Science Foundation under Grant 2018M642359; in part by the Science and Technology Development Fund of Macau under Grant 0038/2020/A1; in part by the Science and Technology Development Fund, Macu SAR under the open project of State Key Laboratory of Lunar and Planetary Science; and in part by the Fundamental Research Funds for the Central Universities under Grant 2020ZDPY0305; and in part by the Xuzhou Science and Technology Project under Grant KC18061.

ABSTRACT Synthetic Aperture Radar and optical (SAR-optical) image matching is a technique of finding correspondences between SAR and optical images. SAR-optical image matching can be simplified to single-mode image matching through image synthesis. However, the existing SAR-optical image synthesis methods are unable to provide qualified images for SAR-optical image matching. In this work, we present a K-means Clustering Guide Generative Adversarial Networks (KCG-GAN) to improve the image quality of synthesizing by constraining spatial information synthesis. KCG-GAN uses k-means segmentations as one of the image generator's inputs and introduces feature matching loss, segmentation loss, and L1 loss to the objective function. Meanwhile, to provide repeatable k-means segmentations, we develop a straightforward 1D k-means algorithm. We compare KCG-GAN with a leading image synthesis method—pix2pixHD. Qualitative results illustrate that KCG-GAN preserves more spatial structures than pix2pixHD. Quantitative results show that, compared with pix2pixHD, images synthesized by KCG-GAN are more similar to original optical images, and SAR-optical image matching based on KCG-GAN obtains at most 3.15 times more qualified matchings. Robustness tests demonstrate that SAR-optical image matching based on KCG-GAN is robust to rotation and scale changing. We also test three SIFT-like algorithms on matching original SAR-optical image pairs and matching KCG-GAN synthesized optical-optical image pairs. Experimental results show that our KCG-GAN significantly improves the performances of the three algorithms on SAR-optical image matching.

INDEX TERMS Image matching, image synthesis, synthetic aperture radar (SAR), generative adversarial networks (GANs).

I. INTRODUCTION

SAR-optical image matching is an important prerequisite for many Earth observation applications such as image fusion [1], image classification [2], [3], land-cover analysis [4], [5], land-use analysis [6], change detection [4], [7], and yield monitoring [8], [9]. SAR-optical image matching remains a challenging problem mainly because of SAR and optical sensors' different imaging mechanisms and principles. These differences lead to significant global geometric distortions and non-linear intensity differences, making it difficult to detect control points or correspondences by automatic algorithms or human eyes. [10], [11].

Recently, image synthesis emerges as a powerful tool to solve the above problem by simplifying the SAR-optical image matching to single-model image matching. In order to provide qualified synthesis results for SAR-optical matching, we present a K-means Clustering Guided Generative Adversarial Networks (KCG-GAN) motivated by semantic image synthesis methods [12]–[15]. Figure 1 illustrates an

The associate editor coordinating the review of this manuscript and approving it for publication was Qiangqiang Yuan.

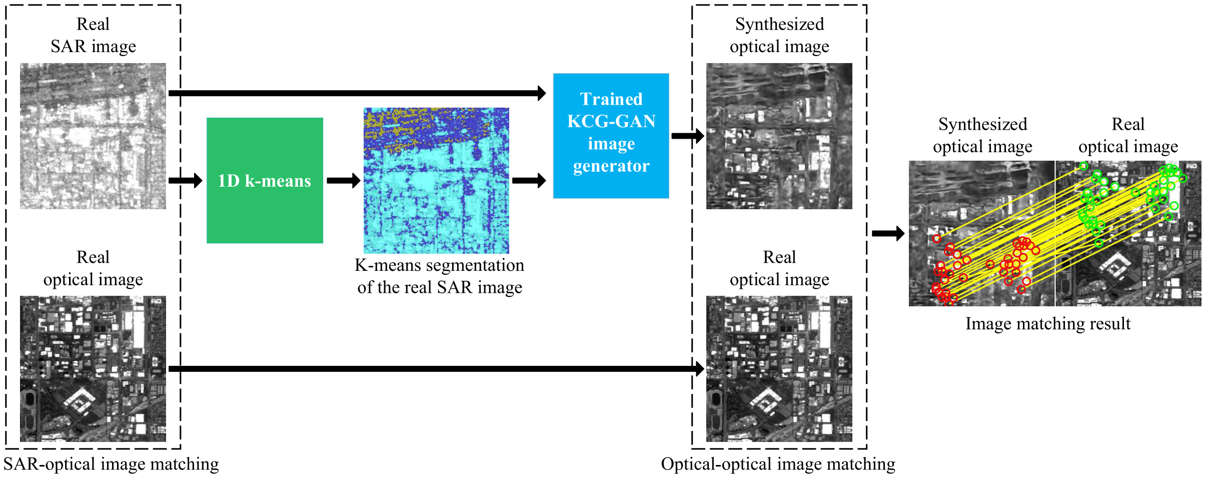

FIGURE 1. **An example of using the 1D k-means algorithm and a trained KCG-GAN generator to automatically match a pair of SAR and optical images.**

example of using KCG-GAN to match a pair of SAR and optical images. It can be seen that the SAR and optical images provide obviously different properties of targets. Specifically, the SAR image shows the physical properties of targets whereas the optical image shows more structural details [16]. We use KCG-GAN to translate the SAR image to the optical image, making the single-model image matching method can be performed directly on matching the synthesized and real optical images. To do this, we introduce k-means segmentations in KCG-GAN for controlling the spatial information in the image synthesis process. More specifically, we feed both SAR images and the corresponding k-means segmentations to its generator, and we introduce feature matching loss, L1 loss, and segmentation loss in the training process of KCG-GAN. **Note that the optical images used in KCG-GAN are grayscale, not RGB,** since the colorization of SAR images is expected to be an ill-posed problem [16], and most image matching methods only use grayscale information of images to be matched [17], [18].

Moreover, to provide unified synthesizing results, SAR and optical images in the training and test datasets should have unified segmentation centers respectively. Hence, we decide to obtain the clustering centers of SAR and optical images from the corresponding training dataset separately. However, it is impossible to use conventional k-means algorithms to cluster massive images simultaneously because of the high time or space complexities of the conventional algorithms. For instance, for a dataset that has 10,000 grayscale training images in the size of 256 × 256 pixels, the number of the corresponding samples to be clustered reaches 655,360,000. If clustering these samples by using conventional k-means algorithms, a regular personal computer may run out of memory. In addition, the conventional algorithms cannot provide repeatable segmentation, because they rely on initializing centers by random sampling. Many sophisticated segmentation methods [19]–[23] have been developed for segmenting remote sensing images. However, in this work, we focus on whether the segmentation information could benefit the SAR-optical image synthesis and matching results? Therefore, we develop a straightforward 1D k-means algorithm for efficiently obtaining repeatable segmentations from massive grayscale images. The 1D means that the massive grayscale images can be considered as a huge one-dimensional array.

Then the main contributions of this work are summarized as follows:

1) We employ k-means segmentations to control the spatial information in synthesizing optical images from SAR images.
2) The repeatable k-means segmentations are efficiently provided by our 1D k-means algorithm.
3) We discover that combining feature matching loss, L1 loss, and segmentation loss to the training progress can enhance the results of SAR-optical image matching, although L1 loss often leads to blurry images [12], [24].
4) There is a lack of quantitative analyses on the SAR-optical image matching based on image synthesis. In this work, we use 900 test image pairs to conduct a comprehensive quantitative comparison between our KCG-GAN and leading image synthesis method—pix2pixHD [12].
5) We conduct robust tests to our KCG-GAN, which shows that the SAR-optical image matching based on our KCG-GAN is robust to rotation and scale changing.

II. RELATED WORK

The SAR-optical matching methods can be classified into two categories: 1. matching based on similarity metrics; 2. transforming multi-modal matching into single-mode matching. The former tries to match multi-modal images based on designing intensity similarity metrics [25], [26], local feature descriptors [10], [11], [27], or learning similarity metrics [28]–[33]. The latter attempts to unify the textures of multi-modal images using image synthesis methods [34]–[36].

In this section, a literature review of SAR-optical image matching is briefly given, according to the two categories: A. Matching based on similarity metrics; B. Transforming multi-modal matching into single-mode matching.

A. MATCHING BASED ON SIMILARITY METRICS

Recently, learning-based methods have shown promising results in matching SAR and optical images [28]–[33], [37], [38], where Siamese and Pseudo-Siamese networks are the most popular network architectures. In [28], a Siamese network is proposed for learning the shift between SAR and optical patches. The geo-localization accuracy of optical images is improved by adjusting the optical sensor model parameters based on tie points generated by the Siamese network. To match very high-resolution SAR and optical images, a Pseudo-Siamese network is proposed to determine a point-wise similarity score [37]. In [29], hard negatives are generated for the training dataset of a variational-GAN, significantly decreasing the false positive rate of a Pseudo-Siamese network [37] in SAR-optical image matching. In [33], a Siamese fully convolutional network (SFcNet) is developed. The SFcNet is trained with a novel loss function for learning the descriptors of matching multi-modal patches.

Besides (Pseudo-)Siamese networks, many efforts have also been made in other network architectures. In [31], a Random Forest-based prediction framework is proposed to transform the matching problem into a classification task. It doubles the number of correspondences comparing to the Scale-Invariant Feature Transform (SIFT) [39] method. In order to improve the quality and diversity of the training, a generative matching network (GMN) is proposed. It applies generative adversarial networks (GANs) to generate coupled training data [32]. In [40], a deep metric based on a fully convolutional neural network (FCN) is proposed to predict whether SAR-optical image pairs are aligned or not. In [38], autoencoder-based matching techniques are extended to semi-supervised learning for SAR-optical image matching. To improve the geo-localization accuracy of optical images, [30] uses the HardNet [41] algorithm to classify matching and non-matching image pairs based on the Euclidean distance.

However, the corresponding image patches used in learning-based methods are mostly generated manually, resulting in time-consuming and cost-intensive [42]. For the learning-based methods that do not generate image patches manually, their image patches' centers are usually located by using feature detectors, such as Harris, DOG (Difference of Gaussians), etc. However, as these feature detectors are developed to match single-mode images, the large intensity and texture differences of the multi-modal images lead to low repeatability of the extracted features [11].

B. TRANSFORMING MULTI-MODAL MATCHING INTO SINGLE-MODE MATCHING

Transforming multi-modal matching into single-mode matching is based on minimizing the non-linear radiometric differences between multi-modal data. With the rapid development of GANs, much effort has been spent on SAR-to-optical or optical-to-SAR image synthesis methods [34], [35], [43], [44]. In [16], a comprehensive analysis of the optimization, opportunities, and limits of using conditional generative adversarial networks (cGANs)-based SAR-optical image translation for remote sensing tasks is presented. In addition, the potential of cGANs for SAR-optical image matching is explored in [34] and [35]. They use cGANs-based image synthesis techniques to transfer multi-modal matching into single-mode matching firstly. They then use well-designed single-modal matching methods (e.g., normalized cross-correlation (NCC), SIFT, and binary robust invariant scalable key (BRISK)) to match the synthesized and the corresponding real images. The experimental results showed that they both obtained better results than directly using single-mode matching methods for matching SAR and optical images, demonstrating this kind of approach's great potential. In [45], the pix2pix [24] is adopted to SAR-to-optical image synthesis. Reference [45] confirms that, based on pix2pix, a sufficient number of correspondences can be estimated. However, quantitative evaluations of image matching results are not provided in [45].

Another way to transform moti-modal matching into single-mode matching is using a style transfer technique to blend original structures with target textures. In [36], a pre-trained deep convolutional neural network (CNN)—VGG19 [46]—is introduced to extract deep pyramid features from SAR and optical images. Then, a bidirectional nearest neighbor field search (NNF) [47] is used to obtain the correct mapping relationship. Finally, the reconstructed images could obtain the correct texture features from other types of images. The experimental results showed that this method outperforms the histogram-of-orientated-phase-congruency (HOPC) method [10] in multi-modal image matching.

All of the above results have confirmed the potential of transforming multi-modal matching into single-mode matching. However, the existing SAR-optical image synthesis methods are still unable to work in a completely satisfying manner [42]. In this work, we aim to enhance image synthesis performance and quantitatively analyze the enhanced image synthesis method on SAR-optical image matching.

III. KCG-GAN

The goal of KCG-GAN is to learn a mapping from the combination of a SAR image and its k-means segmentation to the corresponding optical image: $G(x, S_1(x)) \to y$, where x, $S_1(x)$, and y are a real SAR image, the k-means segmentation of the SAR image, and the corresponding real optical image, respectively.

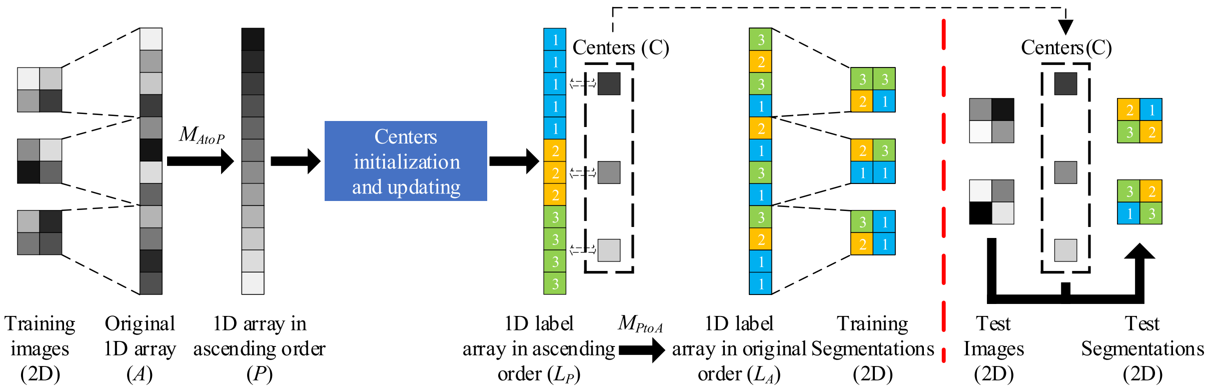
Training images (2D)
Original 1D array (A)
M_{AtoP}
1D array in ascending order (P)
Centers initialization and updating
Centers (C)
1D label array in ascending order (L_P)
M_{PtoA}
1D label array in original order (L_A)
Training Segmentations (2D)
Centers(C)
Test Images (2D)
Test Segmentations (2D)

FIGURE 2. An example of obtaining k-means segmentations of training and test images (either for a SAR dataset or an optical dataset) using the 1D k-means algorithm.

Hence, KCG-GAN model involves two parts: 1) k-means segmentor—1D k-means; 2) k-means based generator. We introduce these two parts in Section III-A and III-B respectively.

A. 1D K-MEANS ALGORITHM

Like other k-means-type algorithms, the clustering problem of the 1D k-means algorithm can be defined as: given an integer k and a set of n samples in $\mathbb{R}^1$, the goal is to choose k centers so as to minimize the objective function [48], [49]:

$$\phi = \min \sum_{a \in A} \|f(C, a) - a\|_2, \tag{1}$$

where A is the set of n samples, a is a sample in A, C is the set of k centers, $f(C, a)$ returns the nearest centers c ($c \in C$) to a, and $\|\cdot\|_2$ returns (L2 norm) Euclidean distance.

The difference between 1D k-means and other k-means algorithms is that 1D k-means algorithm is specifically developed for clustering the large 1D array. Utilizing 1D arrays' property that it can be easily sorted in ascending/descending order, the 1D k-means figures out a straightforward way of generating repeatable centers and reducing computational and space complexities. An example of using 1D k-means for generating segmentations from training and test images is illustrated in Figure 2. The dataset in this example stands for either a SAR or an optical dataset, and it only contains three training images and two test images (in the size of 2 × 2 pixels). **In other words, the cluster centers of SAR and optical images should be obtained separately.** As shown in Figure 2, the three 2D training images are first reshaped into a 1D array. Then the 1D array is sorted in ascending order. We denote the reshaped/original and sorted 1D arrays by A and P respectively. Hence the array A refers to the variable A in Equation 1. Then, after initializing and updating the centers, a 1D label array in ascending order (denoted by L_P) and the final centers (denoted by C) can be obtained. Finally, the k-means segmentations of training images are generated by restoring and reshaping L_P. The k-means segmentations of the two 2D test images are derived based on C. In summary, the 1D k-means algorithm contains three main steps: 1. data pre-processing; 2. repeatable center initialization; 3. cluster updating and generation.

1) DATA PREPROCESSING

Because the mapping relation from sorted array (P) to the original array (A) is needed to recover the original order of label array, we provide mapping functions as follows:

$$P = M_{AtoP}(A), \tag{2}$$

$$A = M_{PtoA}(P), \tag{3}$$

where M_{AtoP} and M_{PtoA} are mapping functions of A to P and P to A respectively.

2) REPEATABLE CENTER INITIALIZATION

The mean tree (MT) center initialization algorithm is presented to initialize repeatable centers. A toy example of using the MT algorithm for initializing five centers from the sorted array (P) is shown in Figure 3, where $C[1]$ to $C[5]$ are the five candidate centers, C_L and C_R are the starting and ending indices of the candidate centers, and $(\cdot)_L$ and $(\cdot)_R$ are the left and right parts of $(\cdot)$. More Specifically, for the left part of P, which is denoted by P_L, its candidate centers' starting and ending indices are $(C\_L)_L$ and $(C\_R)_L$. As can be seen from the figure, the MT algorithm's main idea is to keep dividing the sorted array (P) until there is only one candidate center in it. The midpoint of each division is determined by the corresponding array's mean value. Moreover, if the number of candidate centers is odd, we extract the middle candidate center and assign it by the corresponding array's mean value. We program the MT algorithm as a recursive function and describe it in Algorithm 1. Note that the number of clusters—k is equal to $C\_R - C\_L + 1$, where the C_R and C_L are the input values given before running Algorithm 1.

3) CLUSTER UPDATING AND GENERATION

In this step, we update the clusters only between adjacent centers because the array to-be-clustered has been sorted in ascending/descending order. Hence, the computational and

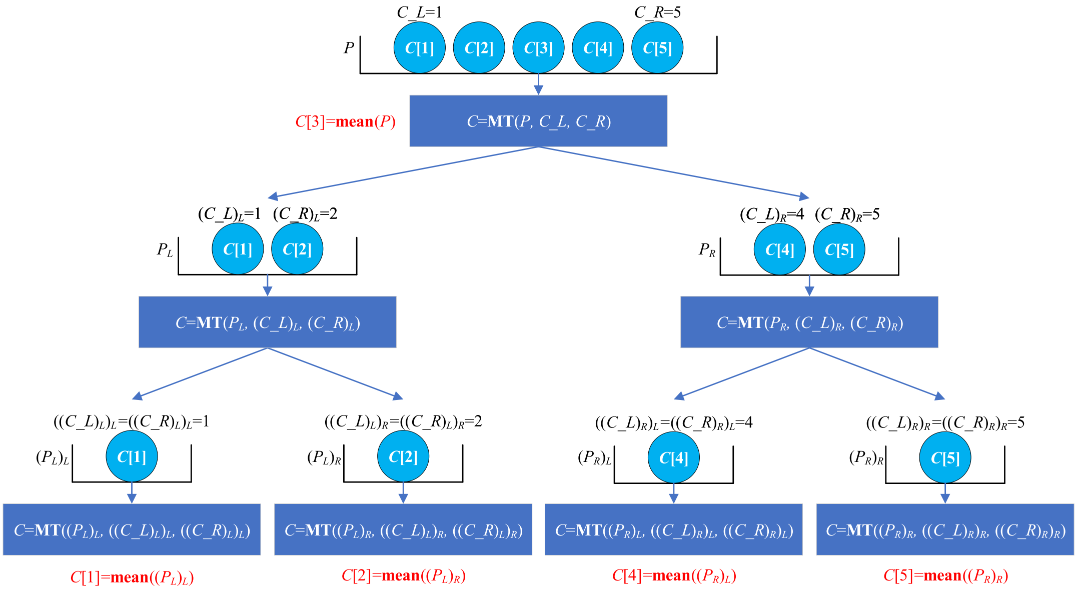
C_L=1
C_R=5
P
C[1]
C[2]
C[3]
C[4]
C[5]
C[3]=**mean**(P)
C=**MT**(P, C_L, C_R)
$(C\_L)_L$=1 $(C\_R)_L$=2
P_L
C[1]
C[2]
$(C\_L)_R$=4 $(C\_R)_R$=5
P_R
C[4]
C[5]
C=**MT**(P_L, $(C\_L)_L$, $(C\_R)_L$)
C=**MT**(P_R, $(C\_L)_R$, $(C\_R)_R$)
$((C\_L)_L)_L$=$((C\_R)_L)_L$=1
$(P_L)_L$
C[1]
$((C\_L)_L)_R$=$((C\_R)_L)_R$=2
$(P_L)_R$
C[2]
$((C\_L)_R)_L$=$((C\_R)_R)_L$=4
$(P_R)_L$
C[4]
$((C\_L)_R)_R$=$((C\_R)_R)_R$=5
$(P_R)_R$
C[5]
C=**MT**($(P_L)_L$, $((C\_L)_L)_L$, $((C\_R)_L)_L$)
C=**MT**($(P_L)_R$, $((C\_L)_L)_R$, $((C\_R)_L)_R$)
C=**MT**($(P_R)_L$, $((C\_L)_R)_L$, $((C\_R)_R)_L$)
C=**MT**($(P_R)_R$, $((C\_L)_R)_R$, $((C\_R)_R)_R$)
C[1]=**mean**($(P_L)_L$)
C[2]=**mean**($(P_L)_R$)
C[4]=**mean**($(P_R)_L$)
C[5]=**mean**($(P_R)_R$)

FIGURE 3. A toy example of the MT centers initialization for initializing five centers from an array P, where $C = \mathbf{MT}(\cdot)$ means the MT algorithm described in Algorithm 1.

Algorithm 1 The Mean Tree (MT) Center Initialization Algorithm

Input: P: a 1D pixel array sorted in ascending order; C_L: the starting index of candidate centers; C_R: the ending index of candidate centers; C: a 1-by-k array of zeros.

Output: C: k centers initialized by the algorithm.

```
function C = MT(P, C_L, C_R)
MeanValue = mean(P);
if C_L == C_R then % only one candidate center in the current P
    C(C_L) = MeanValue
    return
end if
w = C_R − C_L + 1;
if w is odd then
    MiddleCenter = C_L+floor((w/2);
    C(MiddleCenter) = MeanValue;
end if
MiddlePoint = the index of the first sample that is larger than MeanValue;
P_L = P(1 : MiddlePoint);
P_R = P(MiddlePoint + 1 : end);
(C_R)_L = C_L+floor(w/2) − 1;
(C_L)_R = C_R−floor(w/2) + 1;
C = MT(P_L, C_L, (C_R)_L);
C = MT(P_R, (C_L)_R, C_R);
end function
```

space complexities of 1D k-means can both be reduced to $O(n)$ (see Section IV-C.1), where n is the number of samples.

We keep updating the clusters until they no longer change. Finally, we obtain a 1D label array in the sorted order, and we can obtain the label array in the original order as follows:

$$L_A = M_{PtoA}(L_P), \tag{4}$$

where L_P is the label array in the sorted order, and L_A is the label array in the original order.

B. K-MEANS BASED GENERATOR

We use spatially aligned SAR and optical image pairs to train the generator and the discriminator of KCG-GAN, i.e., the KCG-GAN is fully supervised. To provide spatial guidance, the generator's inputs contain a SAR image and the SAR image's k-means segmentation. The objective function comprises four terms: *adversarial loss*, *segmentation loss*, *L1 loss*, and *feature matching loss*. Moreover, we apply cGANs in KCG-GAN since cGANs is more suitable than GANs for image synthesis tasks [24], and cGANs has already been adapted to tasks in multi-sensor remote sensing successfully [16]. Figure 4 shows the structure of KCG-GAN. It can be seen that the SAR and optical image pairs are spatially aligned, and an input image (the SAR image) is served as a condition of the discriminator.

1) ADVERSARIAL LOSS

The adversarial loss expresses the key idea of GAN—training a pair of networks (generator: G, and discriminator: D) in a minimax two-player game. The G learns to generate realistic images, and D learns to discriminate whether the image comes from G or training data. We apply the adversarial loss to the mapping of $G(x, S_1(x)) \rightarrow y$, where x and y are the SAR and optical images respectively, S_1 is the segmenter built by 1D k-means and the SAR training dataset, and $S_1(x)$ is the segmentation of x.

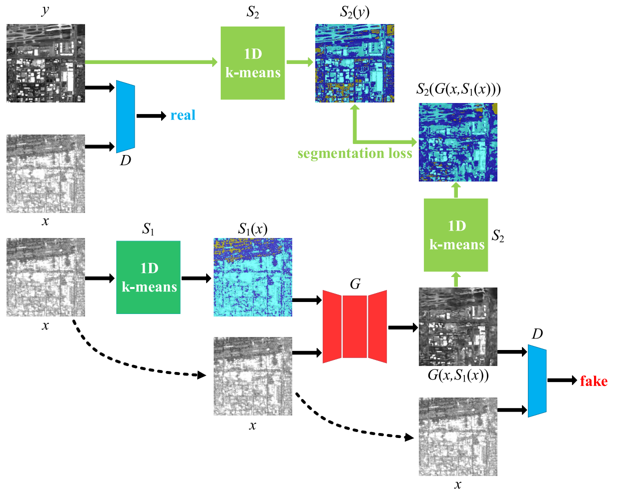

FIGURE 4. Training KCG-GAN to obtain the mapping of SAR(x)→optical(y). G is the generator that learns to synthesize optical images; D is the discriminator that learns to classify between synthesized image combinations and real image combinations; S_1 and S_2 are the segmenters that provides k-means segmentations of SAR and optical images, respectively.

The traditional adversarial loss with the conditional setting is formulated as [24]:

$$\mathcal{L}_{adv} = \mathbb{E}_{(x,y)}\left[\log D(x,y)\right] + \mathbb{E}_x\left[\log\left(1 - D(x, G(x, S_1(x)))\right)\right], \tag{5}$$

where G learns to synthesize optical images $G(x, S_1(x))$ that fool the discriminator—D; D learns to classify between synthesized image combinations—$G(x, S_1(x))$ and y—and real image combinations—x and y; $D(\cdot)$ returns the classification/discrimination results; $\mathbb{E}_{(x,y)} \triangleq \mathbb{E}_{(x,y)\sim p_{data}(x,y)}$ and $\mathbb{E}_x \triangleq \mathbb{E}_{x\sim p_{data}(x)}$; $\mathbb{E}_{(\cdot)\sim p_{data}(\cdot)}[f(\cdot)]$ returns the expectation of $f(\cdot)$ with respect to the data-generating distribution $p_{data}(\cdot)$ [50].

In this work, instead of using the negative log-likelihood objective, we use LSGAN [51] for stable training. Therefore, Equation (5) becomes:

$$\mathcal{L}_{LSGAN-adv} = \mathbb{E}_{(x,y)}\left[(D(x,y)-1)^2\right] + \mathbb{E}_x\left[(D(x, G(x, S_1(x))))^2\right]. \tag{6}$$

2) SEGMENTATION LOSS

We introduce the segmentation loss [15] to maintain the spatial information of the synthesized images with respect to their corresponding real images. The segmentation loss is formulated as:

$$\mathcal{L}_{Seg} = \mathbb{E}_{(x,y)}\left[H\left(S_2(y), S_2(G(x, S_1(x)))\right)\right], \tag{7}$$

where $S_2(\cdot)$ returns optical k-means segmentations, and $H(\cdot)$ computes a pixel-wise cross-entropy by:

$$H(a,b) = -\sum_{i\in I}\sum_{j\in J} a_{i,j}\log b_{i,j}, \tag{8}$$

where a and b are two k-means segmentations whose image and label spaces are I and J.

3) L1 LOSS

We use L1 loss to guarantee that the synthesized images generate similar content to the corresponding real images. The L1 loss [24] is expressed as:

$$\mathcal{L}_{L1} = \mathbb{E}_{(x,y)}\left[\|y - G(x, S_1(x))\|_1\right]. \tag{9}$$

4) FEATURE MATCHING LOSS

We adopt feature matching loss [12], [52] to stabilize the training and produce a natural high-frequency structure. The feature matching loss minimizes the difference of features extracted from multiple layers of the discriminator while identifying the synthesized and real optical image combinations. The feature matching loss is defined as:

$$\mathcal{L}_{FM} = \mathbb{E}_{(x,y)}\sum_{m=1}^{M}\frac{1}{N_m}\left[\left\|\begin{array}{l} D_m(x,y) - \\ D_m(x, G(x, S_1(x)))\end{array}\right\|_1\right], \tag{10}$$

where D_m is the mth layer feature extractor of D, $D_m(\cdot)$ returns features extracted from the mth layer of D, M is the total number of feature layers, and N_m is the number of elements in the mth layer.

5) FINAL OBJECTIVE FUNCTION

The final objective function of KCG-GAN is:

$$\mathcal{L}_{KCG-GAN} = \mathcal{L}_{LSGAN-adv} + \lambda_1\mathcal{L}_{Seg} + \lambda_2\mathcal{L}_{L1} + \lambda_3\mathcal{L}_{FM}, \tag{11}$$

where λ_1, λ_2, and λ_3 are hyper-parameters that control the relative importance of the segmentation, L1 and, feature matching losses, respectively. The goal of the KCG-GAN is to solve:

$$G^* = \arg\min_G\max_D \mathcal{L}_{KCG-GAN}. \tag{12}$$

6) NETWORK ARCHITECTURE

We adopt the generative network architecture from Wang *et al.* [12], who has demonstrated remarkable results for high-resolution image synthesis. The original architecture in [12] is constructed with a multi-scale generator and discriminator, but in this work, we only use the global generator and discriminator in KCG-GAN.

The generator architecture consists of: `c7s1-64`, `d128`, `d256`, `d512`, `d1024`, `R1024`, `R1024`, `R1024`, `R1024`, `R1024`, `R1024`, `R1024`, `R1024`, `R1024`, `u512`, `u256`, `u128`, `u64`, `c7s1-3`, where `c7s-`k is a 7×7 Convolution-InstanceNorm-ReLU layer with k filters and stride 1; `d`k is a 3×3 Convolution-InstanceNorm-ReLU layer with k filters and stride 2; `R`k is a residual block that contains two 3×3 convolutional layers with the same number of filters on both layers; and `u`k is a 3×3 fractional-strided-Convolution-InstanceNorm-ReLU layer with k filters and stride $1/2$.

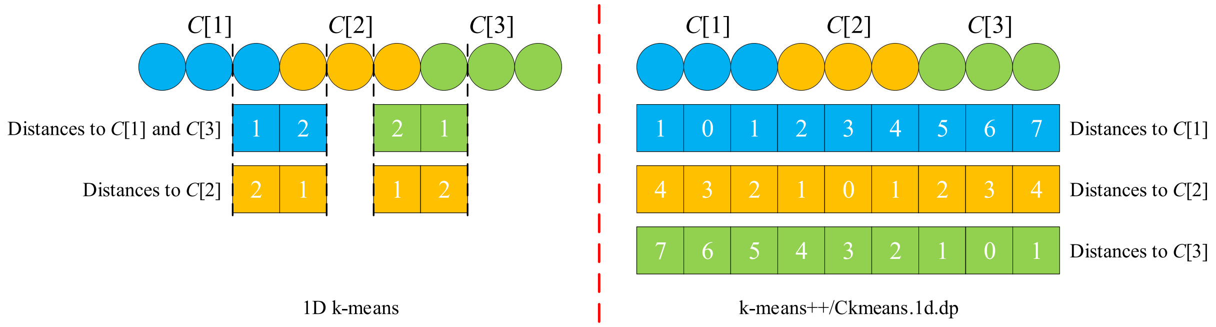

FIGURE 5. A toy example of the space occupying of 1D k-means, k-means++ and Ckmeans.1d.dp algorithms for clustering nine samples into three classes. The circles represent the samples sorted in ascending order, and we assume that the distance between adjacent samples is one; the blocks represent the distance arrays used by the three algorithms, and the numbers are the distances from samples to corresponding centers.

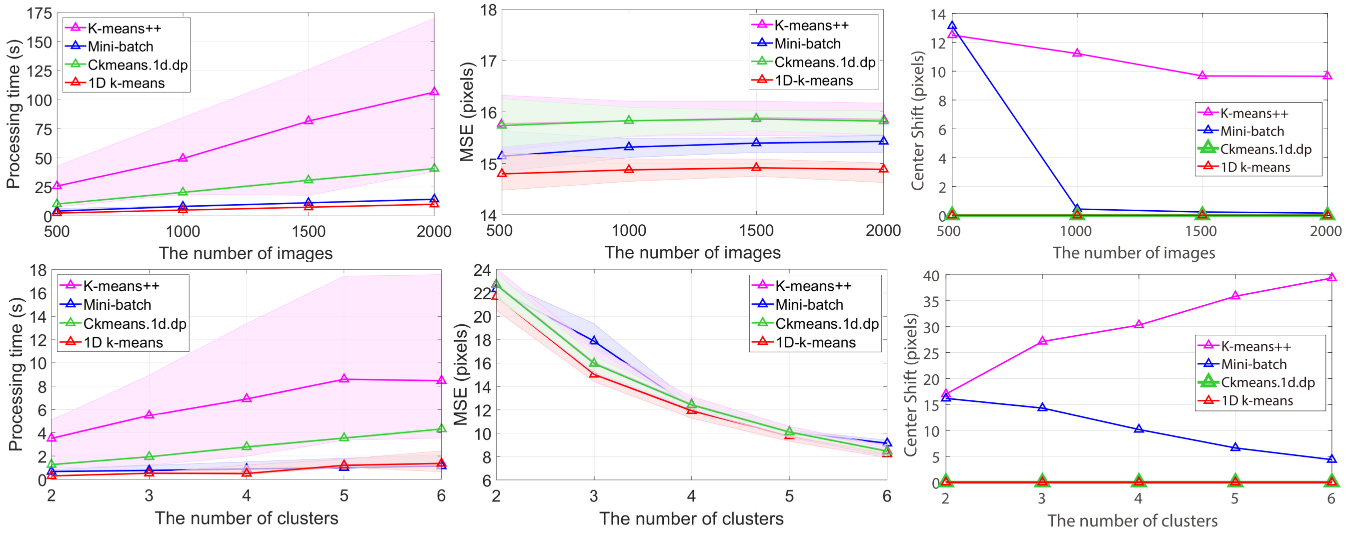

FIGURE 6. Processing time of the four algorithms on datasets with various image numbers (left) and cluster numbers (right). The lines and shadows represent the average and range of processing time of the four algorithms in each experiment.

We use 70 × 70 PatchGANs [12], [24], [53], [54] for the discriminator networks. The discriminator architecture is: `C64-C128-C256-C512`, where Ck is a 4×4 Convolution-InstanceNorm-LeakyReLU layer with k filters and stride 2. The leaky ReLUs are used with a slope of 0.2.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, the implementation details about KCG-GAN are shown in Section IV-A; the datasets and set-up of 1D k-means and KCG-GAN are introduced in Section IV-B; a quantitative comparison of the presented 1D k-means algorithm against three k-means-type algorithms is provided in Section IV-C; the image synthesis and matching results of the presented KCG-GAN and pix2pixHD are given in Section IV-D.

A. IMPLEMENTATION DETAILS

We use the Adam solver [55] to train the KCG-GAN and pix2pixHD from scratch.

In order to determine the number of clusters in KCG-GAN, we random sample one-hundred optical images from the SEN-12 dataset and cluster these images in different numbers of clusters by using the 1D k-means algorithm. Then, we use the Elbow method to determine the number of clusters based on clustering results' MSE values (see the bottom row and middle column of Figure 6). The Elbow method returns the "Elbow-point" of the MSE values, i.e., the second derivative's minimum. In the case of MSE values in Figure 6, the "Elbow-point" is at three clusters. Hence, the number of clusters is chosen as three in this work. The source codes of 1D k-means and KCG-GAN are available at https://github.com/WenliangDu/KCGGAN.

B. DATASETS AND SET-UP

We gather the training and test images from SEN1-2 dataset [56] (256 × 256 pixels for each image) that contains 282,384 pairs of aligned SAR-optical images. Specifically, in order to evaluate image synthesis methods on rural, semi-urban, and urban scenarios, we select image pairs from s1(2)_0, s1(2)_3, s1(2)_6, s1(2)_7, s1(2)_9, s1(2)_10, s1(2)_14, s1(2)_15, s1(2)_17, s1(2)_18, s1(2)_26 and s1(2)_35 folders of the ROIs1158 spring subgroup. Moreover, since image pairs that mostly contain sea and forest scenarios are naturally hard to be matched, these kinds of image pairs are removed from the dataset. Finally, 9,459 image pairs are selected. Specifically, 8, 559 pairs are used for training, and 900 pairs are used for testing (300 pairs for each scenario), and there is no test image overlapping training images.

To discuss the generalization of pix2pixHD and KCG-GAN, we select image pairs from s1(2)_2, s1(2)_5, s1(2)_20, s1(2)_25, and s1(2)_27 folders of the ROIs1158 spring subgroup to construct a generalization dataset. We remove the image pairs that mostly contain sea and forest scenarios from the generalization dataset, just like what we have done to the training and test image pairs. Finally, we have 5, 665 SAR-optical image pairs in the generalization dataset.

We coarse-tune the values of the three hyper-parameters in a small dataset because optimizing them in Equation (11) requires a huge computational power and time. The small dataset consists of 1,025 SAR-optical image pairs collected from the $s1\_3$ (SAR) and $s2\_3$ (optical) folders of the spring season of the SEN1-2 dataset [56], where 936 pairs for training, and 89 pairs for testing. We found that the weights of feature matching (λ_3) loss, segmentation (λ_1) loss, and L1 (λ_2) loss can be used to adjust the details, structures, and textures of the synthesized images, respectively. We set the weight of feature matching loss (λ_3 in Equation (11)) to 10, which is also the default setting in pix2pixHD [12]. We then set weights of the segmentation and content (L1) losses (λ_1 and λ_2) to 10 and 50 for balancing the matching and synthesis results of KCG-GAN. Note that this setting is for reference only, not the optimal setting for all kinds of SAR-optical images. If applying KCG-GAN to other types of SAR-optical images, users should re-tune the weights of losses according to the properties of corresponding SAR-optical images for obtaining better image synthesis and matching results.

The experiments of k-means-type algorithms are performed on a personal computer with an Intel i5-6600 CPU and 20 GB RAM. The training and tests of image synthesis are conducted on a server with 2 Intel Xeon Gold 5117 CPU processors, 24 GB RAM, and an NVIDIA Tesla P100 with 16 GB HBM2 memory. For the image synthesis, we set the number of epochs to 200, and we set the number of channels of input and output to one since the training and test images used in this work are grayscale.

C. EXPERIMENTS OF 1D K-MEANS ALGORITHM

We compare 1D k-means algorithm with three k-means type algorithms: k-means++ [48], mini-batch [49], and Ckmeans.1d.dp [57] algorithms. We implement our 1D k-means algorithm with MATLAB code. In contrast, we implement the three algorithms based on the corresponding library or package since we want to reproduce their best performance. Specifically, the k-means++ algorithm is implemented by using the function—`kmeans` from the MATLAB Library; the mini-batch algorithm is implemented by using python code based on the `sklearn` package [58], and the batch size of the mini-batch algorithm is set to one-hundred for better results; the Ckmeans.1d.dp algorithm is implemented by using R code based on the `Ckmeans.1d.dp` package [59]. Then, we discuss space complexities of the four algorithms in Section IV-C.1, and we evaluate the four algorithms on datasets with various image numbers and cluster numbers (i.e., the value of k) in Section IV-C.2.

1) SPACE COMPLEXITY

Figure 5 shows a toy example of the spaces occupying of the 1D k-means, k-means++ and Ckmeans.1d.dp algorithms. It can be seen that for updating the nearest centers of each sample, k-means++ and Ckmeans.1d.dp algorithms should compute the distances of all samples to the centers. Therefore, their space complexities are $O(nk)$, where n is the number of all the samples, and k is the number of clusters. In contrast, the 1D k-means algorithm only computes the distance between the samples and their adjacent centers (see Figure 5) since the array to be clustered has been sorted in ascending order. Hence, the space complexity of 1D k-means algorithm is only $O(n)$.

As for the mini-batch algorithm, its space complexity should be $O(km)$ based on its original article [49], where m is the batch size, and m is usually much smaller than n. However, in practice, running mini-batch, k-means++ and Ckmeans.1d.dp algorithms occupied an unacceptable amount of memory, which caused our personal computer (see Section IV-B) to run out of memory when we clustered more than 7,000 images (256 × 256).

2) COMPARISONS OF THE FOUR K-MEANS-TYPE ALGORITHMS

We evaluate the four k-means-type algorithms on datasets with various image and cluster numbers. For the experiments of various image numbers, we randomly sample different numbers of optical images from the 9,459 image pairs (see Section IV-B) to construct the datasets. Then, we use the four algorithms to cluster each dataset into three classes. For the experiments of various cluster numbers, we randomly sample one-hundred optical images from the 9,459 image pairs, and use the four algorithms to cluster each dataset in different classes. Note that ten independent datasets are prepared for each experiment to obtain reliable experimental results. In addition, we test k-means++ and mini-batch algorithms for ten times on each dataset because their results are random. Furthermore, we have sorted all the datasets of each experiment in ascending order before they were clustered by the four algorithms.

The left column of Figure 6 shows the four algorithms' processing times on various image and cluster numbers.

TABLE 1. NOQMs of different variants of pix2pixHD and KCG-GAN on 900 test image pairs. Feat and L1 are feature matching and L1 losses, respectively; G_G and G_M are global and multi-scale generators, respectively; D_S and D_M are single and multi-scale discriminators, respectively; SegIn represnets using segmentations as one of the generator's inputs; Seg is segmentation loss.

Methods	Variants	Rural (300)	Semi-urban (300)	Urban (300)
pix2pixHD	GAN+Feat+G_G+D_S (pix2pixHD basic version)	18	9	14
	GAN+Feat+G_G+D_S+VGG	23	22	38
	GAN+Feat+G$_\mathrm{G}$+D$_\mathrm{M}$ (pix2pixHD best version)	35	**32**	**46**
	GAN+Feat+G_G+D_M + VGG	22	23	28
	GAN+Feat+G_M+D_M	22	7	0
	GAN+Feat+G_M+D_M+VGG (pix2pixHD full version)	**46**	25	0
KCG-GAN	GAN+Feat+G_G+D_S+SegIn	0	0	0
	GAN+Feat+G_G+D_S+Seg	0	0	0
	GAN+Feat+G_G+D_S+Seg+SegIn (KCG-GAN w/o L1)	14	11	19
	GAN+Feat+G_G+D_S+L1 (KCG-GAN w/o Seg)	60	**79**	143
	GAN+Feat+G$_\mathrm{G}$+D$_\mathrm{S}$+L1+SegIn+Seg (KCG-GAN full version)	**63**	78	**145**

It is obvious that the processing time of k-means++ increased dramatically when increasing the number of images or clusters, while the processing time of mini-batch, Ckmeans.1d.dp, and 1D k-means is much more stable. More importantly, the processing time of 1D k-means is always less than 11 seconds, even when the number of images is reached 2,000.

The middle column of Figure 6 shows the four algorithms' mean square errors (MSEs) in various image and cluster numbers. The MSE is derived from $\sqrt{\phi}$, where ϕ is obtained by Equation (1). As we can see that 1D k-means obtains the lowest MSE in all experiments, which means 1D k-means could provide more reasonable centers than the other algorithms.

We evaluate the stability of four algorithms by using the center shift. The right column of Figure 6 shows the four algorithms' center shift values in different image and cluster numbers. Remember that ten independent datasets are prepared for each experiment, and the k-means++ and mini-batch algorithms are tested ten times for each dataset. Hence, for a certain experiment, the value is obtained by the maximum center shift among the ten datasets. It can be seen that only Ckmeans.1d.dp and 1D k-means could attain a zero center shift since they do not rely on initializing the centers randomly. In summary, the presented 1D k-means algorithm can provide repeatable segmentations and are much more efficient than the other three algorithms.

D. EXPERIMENTS OF KCG-GAN

We firstly introduce evaluation criteria in Section IV-D.1. Then, we compare the image synthesis and matching results of our KCG-GAN against a leading method—pix2pixHD [12]—in Section IV-D.2. Then, we discuss the generalization and robustness of KCG-GAN and pix2pixHD in Section IV-D.3 and IV-D.4. Finally, in Section IV-D.5, we test three SIFT-like algorithms—SIFT, SAR-SIFT [60] and PSO-SIFT [61] on directly matching SAR-optical image pairs and KCG-GAN synthesized optical-optical image pairs.

1) EVALUATION CRITERIA

We chose six criteria to conduct quantitative evaluations of image synthesizing and matching. Two of them are used for evaluating the quality of image synthesizing: peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [62].

The rest four criteria are used for evaluating image matching. Specifically, the first criterion is the **number of correct correspondences (NOCCs)**, where the correspondences are obtained from the coarse matching [63] of SIFT descriptors. Since the SAR and optical images are spatially aligned in the test dataset, a correct correspondence is chosen as the correspondence whose Euclidean distance is less than three pixels [34], [35]. The second criterion is the **number of qualified matchings (NOQMs)**, which is defined as the matching whose NOCCs is equal or greater than eight because the fundamental matrix can be derived by at least eight correct correspondences [64], [65]. The remaining two criteria are **outlier ratio** (*a.k.a.* false-matching ratio), and **root mean square error (RMSE)** [66], where the RMSE values are obtained by matching results of the locally linear transforming (LLT) method [67]. The RMSE of LLT ($RMSE_{LLT}$) is derived as follows:

$$RMSE_{LLT} = \sqrt{\mathbb{E}\left(\|u_{LLT} - v_{LLT}\|_2^2\right)}, \tag{13}$$

where u_{LLT} and v_{LLT} are the correspondences of synthesized and transformed images preserved by the LLT method.

2) RESULTS OF IMAGE SYNTHESIS AND MATCHING

COMPARISON OF FULL VERSIONS AGAINST VARIANTS

The full version of pix2pixHD involves a global or multi-scale generator (G_G or G_M), a single or multi-scale discriminator (D_S or D_M), and adversarial (GAN), feature matching (Feat), VGG losses. Specifically, the number of a multi-scale discriminator in D_M is two in this work. We test six variants of pix2pixHD and show their quantitative comparisons in Table 1. Obviously, **GAN+Feat+G_G+D_M** is the best variant of pix2pixHD. This result indicates that although feature matching and VGG losses, and the multi-scale generator and discriminator are all used for expressing more natural details, their effects are not simply superimposed. In other words, if too many actions of generating details were taken in image

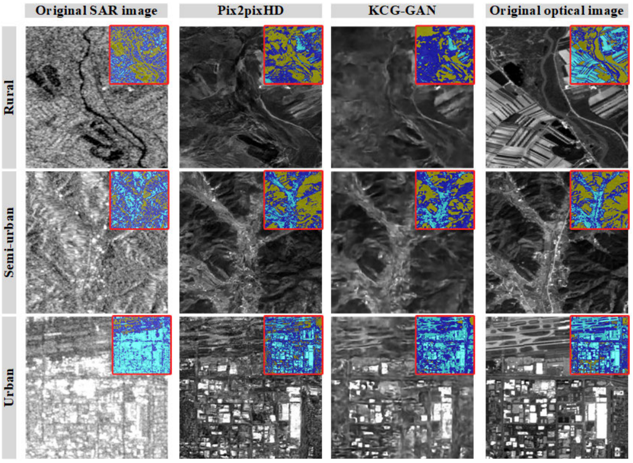

FIGURE 7. Image synthesis results of pix2pixHD and our KCG-GAN on three typical images pairs. The top right of each image shows its segmentation obtained by 1D k-means algorithm.

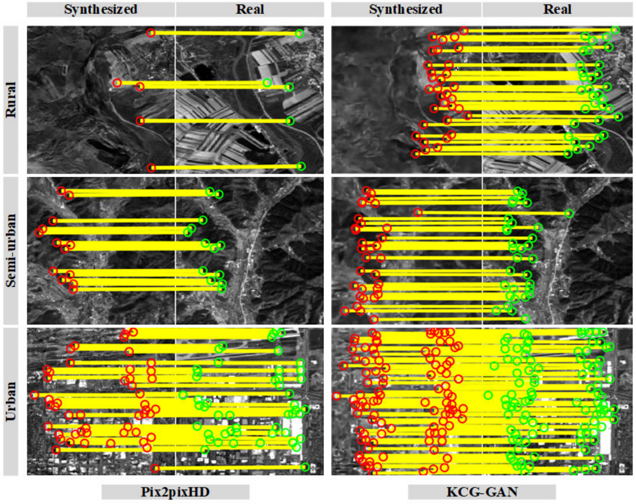

FIGURE 8. Correct correspondences obtained by pix2pixHD and our KCG-GAN from the three typical images pairs (red cycles: correct matched points in synthesized images; green cycles: correct matched points in real images; and yellow lines: lines between correct matched points).

synthesis, needless details may be generated, which harms the subsequent SAR-optical matching.

The full version of our KCG-GAN involves a global generator, a single discriminator, a segmentation component, GAN loss, feature matching loss, and content (L1) loss. **The segmentation component consists of the segmentation loss and using k-means segmentations as one of the inputs of the generator.** One reason for not using the multi-scale generator and discriminator is that the size of the images in the SEN1-2 dataset is only 256 × 256. Another reason is that we only want to build the general architecture of KCG-GAN in this work. The quantitative results of three variants of KCG-GAN are shown at the bottom of Table 1. It can be seen that, based on the pix2pixHD basic version, if we only add segmentations as the generator's inputs or only add segmentation loss, none of the qualified matchings could be obtained. If simultaneously adding them as the segmentation component, we could obtain more qualified matchings than the basic structure of pix2pixHD for the semi-urban and urban scenarios. Moreover, the L1 loss improved the matching results substantially, and the L1 loss combined with the segmentation component can further enhance the matching results in rural and urban scenarios. Overall, for the 900 test image pairs, the SAR-optical image matching based on the full version of KCG-GAN can obtain 1.86, 2.43, and 3.15 times more qualified matchings than those based on the best variant of pix2pixHD in rural, semi-urban, and urban scenarios, respectively. Note that the following comparisons are based on the full version of KCG-GAN and the best variant of pix2pixHD.

QUALITATIVE ILLUSTRATION OF IMAGE SYNTHESIS RESULTS

Figure 7 presents three examples of image synthesis results obtained by pix2pixHD and our KCG-GAN. The corresponding k-means segmentation results are illustrated at the top-right of each image. The three image pairs are picked from rural, semi-urban, and urban scenarios of the 900 test image pairs. It can be seen that the synthesizing results of pix2pixHD are more visually appealing than the synthesizing results of KCG-GAN since more details are synthesized based on the multi-scale discriminator of pix2pixHD. Nevertheless, KCG-GAN preserves more spatial structures than pix2pixHD since the content and structure information constrained by the L1 loss and segmentation component of KCG-GAN. Moreover, we enlarge one typical part of each of the three optical images and their corresponding synthesized images (see Figure 9). Obviously, KCG-GAN can preserve more spatial structures than pix2pixHD in the three examples. Figure 8 shows the correct correspondences obtained by pix2pixHD and KCG-GAN from the three typical image pairs. It can be seen that, based on KCG-GAN, more correct

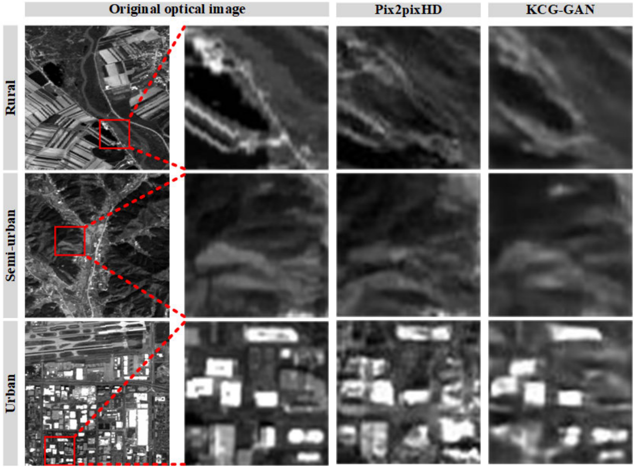

FIGURE 9. Enlarged optical images and the corresponding enlarged synthesized images.

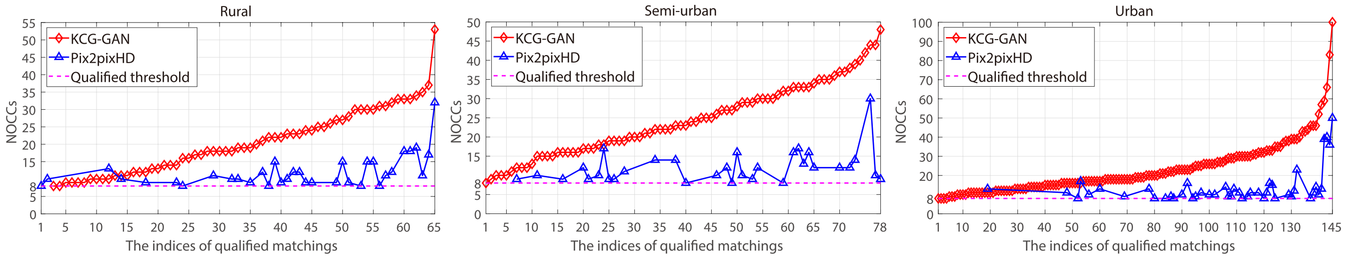

FIGURE 10. NOCCs of the qualified matchings obtained by the two methods.

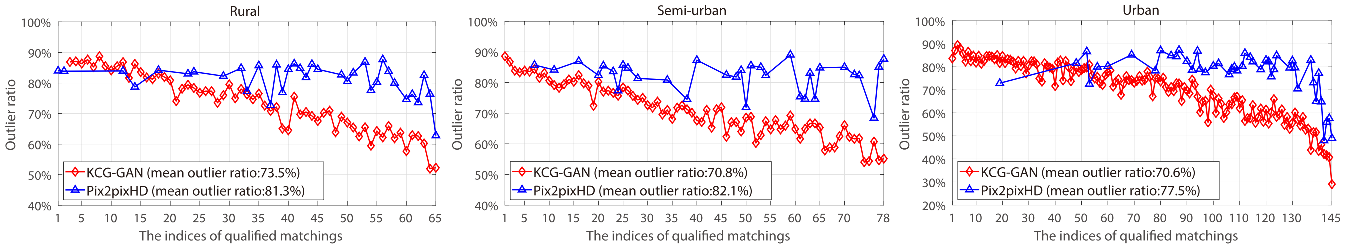

FIGURE 11. Outlier ratios of the qualified matchings obtained by the two methods.

TABLE 2. PSNRs and SSIM indexes of the two methods in three scenarios.

Scenarios(Methods)	PSNR		SSIM	
	Maximum	Average	Maximum	Average
Rural(KCG-GAN)	20.74	**16.11**	**0.59**	**0.33**
Rural(pix2pixHD)	**20.81**	15.86	0.57	0.29
Semi-urban(KCG-GAN)	**22.09**	14.97	**0.53**	**0.24**
Semi-urban(pix2pixHD)	21.03	**15.06**	0.47	0.22
Urban(KCG-GAN)	**21.00**	**16.12**	**0.56**	**0.26**
Urban(pix2pixHD)	20.38	15.36	0.51	0.22

correspondences are obtained. It reveals that without constraining the synthesis of spatial information, the features can not be precisely synthesized to the demand for image matching.

QUANTITATIVE COMPARISONS OF IMAGE SYNTHESIS AND MATCHING RESULTS

We show quantitative comparisons of image synthesizing results of KCG-GAN and pix2pixHD in Table 2. It can be seen that pix2pixHD obtains slightly higher maximum and average PSNRs than KCG-GAN in rural and semi-urban scenarios. In contrast, KCG-GAN achieves the best SSIM results for all three scenarios and obtains significantly higher PSNRs than pix2pixHD in the urban scenario. Hence, in general, KCG-GAN synthesized higher-quality images than pix2pixHD. Taken together with image synthesis results shown in Figures 7 and 9, although KCG-GAN synthesized blur images due to the L1 loss [12], images synthesized by KCG-GAN are more similar to the real optical images than the images synthesized by pix2pixHD.

We use NOCCs, outlier ratio, and RMSE [66] of LLT to provide the quantitative evaluation of image matching results. Figure 10 shows the NOCCs of the qualified matchings obtained by the two methods from 900 test image pairs. Note that the NOCCs obtained by KCG-GAN are ranked in ascending order, and the NOCCs obtained by pix2pixHD are arranged in terms of the same image-pairs of KCG-GAN. We see that there are only a few cases that the pix2pixHD obtains more NOCCs, and, in these cases, the differences between NOCCs obtained by the two methods are small. While, in most cases, our KCG-GAN obtains much more NOCCs than pix2pixHD. Specifically, eighty percent of qualified matchings obtained by KCG-GAN have more than twelve, sixteen, and thirteen NOCCs in rural, semi-urban, and urban scenarios, respectively. In contrast, eighty percent of qualified matchings obtained by pix2pixHD only have more than nine NOCCs in all the three scenarios.

On the other hand, NOCCs obtained by the pix2pixHD in Figure 10 show rise trends except for the last two cases in the semi-urban scenario. It means that the pix2pixHD can obtain a qualified matching with a higher probability for the image pair that KCG-GAN already obtained a qualified matching.

Next, we show the corresponding outlier ratios and $RMSE_{LLT}$ in Figure 11 and Figure 12. The two methods' outlier ratios show declining trends with increased NOCCs (except for the last two qualified matchings of pix2pixHD in the semi-urban scenario). In contrast, there is a relatively low correlation between RMSE and NOCCs. For instance, for the twenty-third qualified matching obtained by KCG-GAN in the urban scenario, the outlier ratio is 77.2% (Figure 11), and the NOCCs is 13 (Figure 10). Still, the LLT can not preserve more than seven correspondences to derive the fundamental matrix (Figure 12). We name this kind of experiment as the failed experiment and set the corresponding RMSE to NaN (not a number). It is because even the correct correspondences

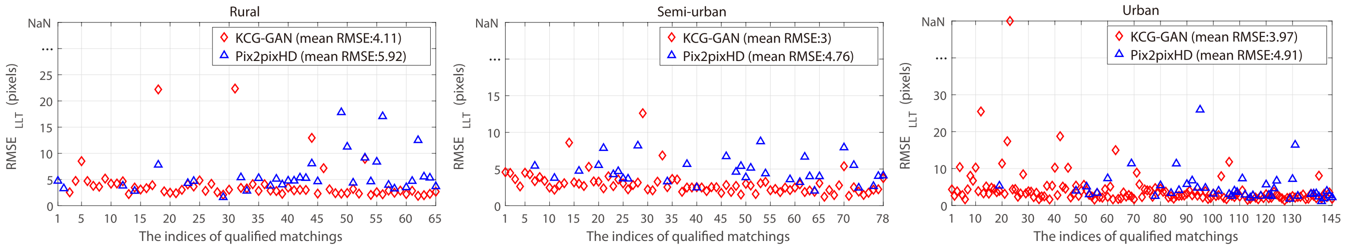

FIGURE 12. RMSEs of LLT obtained by the two methods from the qualified matchings, where NaN represents failed experiments.

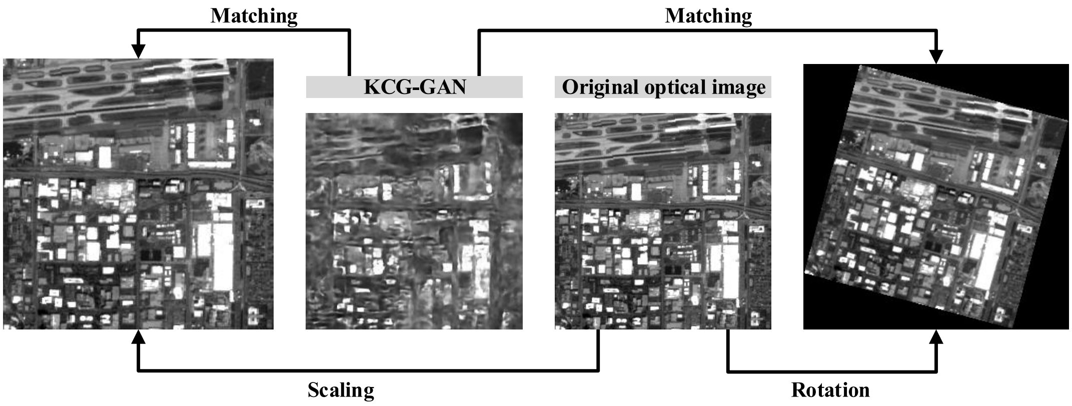

FIGURE 13. Overview of robustness tests of KCG-GAN.

TABLE 3. PSNRs and SSIM indexes of the two methods on matching image pairs of other domains.

Methods	PSNR		SSIM	
	Maximum	Average	Maximum	Average
KCG-GAN	**23.26**	**13.81**	**0.80**	**0.20**
pix2pixHD	21.89	13.20	0..74	0.17

are good enough for the subsequent processing, but it may also be hard for image matching methods to discriminate them from outliers.

The average values of the outlier ratio and $RMSE_{LLT}$ are shown at the legends of Figure 11 and Figure 12. Obviously, KCG-GAN provides a lower outlier ratio and $RMSE_{LLT}$ than pix2pixHD.

In addition, both of the two methods obtain the best NOCCs, NOQMs (see Table 1), and outlier ratio in the urban scenario. This may be because the urban images naturally contain more features than images from the other two scenarios.

3) GENERALIZATION DISCUSSION

We test KCG-GAN and pix2pixHD on five other domains—s1(2)_2, s1(2)_5, s1(2)_20, s1(2)_25, and s1(2)_27 folders of the ROIs1158 spring sub-group. As shown in Table 3, KCG-GAN obtains better PSNR and SSIM results than pix2pixHD, which indicates L1 loss and segmentation component improve the generalization of SAR-to-optical image synthesis.

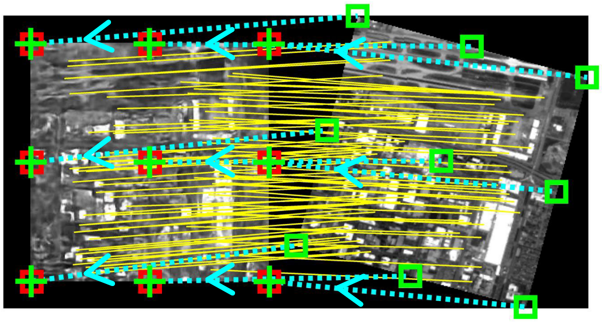

FIGURE 14. Calculating $RMSE_{CP}$ based on nine pairs of checkpoints, where the red and green squares are the checkpoints, the green crosses are the projected checkpoints, the yellow lines represent correct correspondences, and the cyan dashed lines and arrows represent the projection.

However, neither of the two methods acquires qualified matchings from image pairs in the five domains. Hence, same to other data-driven deep learning models in remote sensing [68], improving generalization of image synthesis based SAR-optical image matching is still a challenge.

4) ROBUSTNESS TESTS

In this section, we conduct robustness tests of our KCG-GAN with pix2pixHD in the cases of rotation and scale changing (see Figure 13). The correct correspondence of a matching can be expressed as:

$$\|u_l T - v_l\|_2 \leq \varepsilon, \tag{14}$$

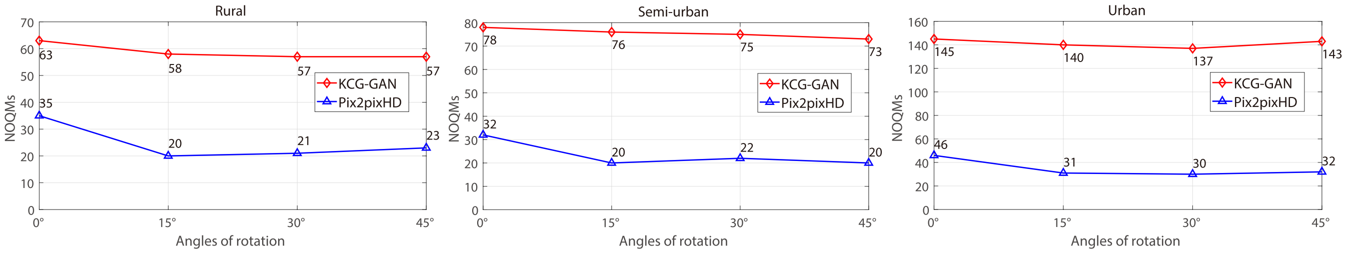

FIGURE 15. The number of qualified matchings of rotation tests.

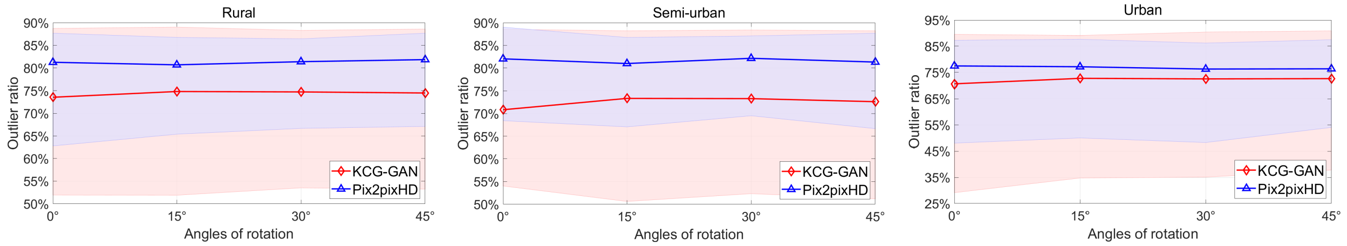

FIGURE 16. Outlier ratios of rotation tests, where markers and shadows represent averages and ranges of outlier ratios.

where u_l and v_l are the lth correspondences obtained from synthesized and transformed (rotated or scaled) optical images, T is the transformation matrix, which is known since the angles of rotation and scale factors are setting manually, and ε is the threshold of discriminating correct correspondence. We set ε to three pixels, which is the same as the threshold of correct correspondence in Section IV-D.1.

In order to evaluate the reliability of correct correspondences, we introduce another evaluation criterion—the RMSE of checkpoints, denoted by $RMSE_{CP}$. In particular, we sample **one-hundred** pairs of gridded checkpoints from the images to be matched. Then, we project the checkpoints from transformed images to the synthesized images according to T_{CC}. T_{CC} is the transformation matrix constructed by the correct correspondences from synthesized images, denoted by u_{CC}, and the correct correspondences from transformed images, denoted by v_{CC}. T_{CC} is expressed as $T_{CC} = u_{CC}/v_{CC}$. Finally, we can obtain $RMSE_{CP}$ by obtaining the RMSE between the projected and the original checkpoints in the synthesized image. Figure 14 gives an example of projecting **nine** gridded checkpoints from a rotated optical image to a synthesized optical image. The projected (red squares) and original checkpoints (green crosses) are almost identical. It implies that the correct correspondences (yellow lines) derived by this matching are reliable to the subsequent processing. The quantitative criterion—$RMSE_{CP}$ is expressed as:

$$RMSE_{CP} = \sqrt{\mathbb{E}\left(\|u_{CP} - v_{CP}T_{CC}\|_2^2\right)}, \tag{15}$$

where $\mathbb{E}(\cdot)$ returns the mean value, $\|\cdot\|_2^2$ returns squared Euclidean distance, u_{CP} and v_{CP} are the gridded checkpoints from synthesized and transformed images, respectively.

We evaluate the robustness on the same 900 test image pairs from Section IV-D.2. We use NOQMs, outlier ratio, $RMSE_{LLT}$, and $RMSE_{CP}$ to evaluate the robustness quantitatively. Specifically, for each case, the outlier ratio, $RMSE_{LLT}$, and $RMSE_{CP}$ are presented by the corresponding average, maximum, and minimum values of qualified matchings. In particular, for the $RMSE_{LLT}$, the results of failed experiments are not taken into account. The failed experiments have been introduced in the *Quantitative Comparison* of Section IV-D.2.

CASES OF ROTATION CHANGING

We rotate the original optical images by 15, 30, and 45 degrees. Then we evaluate the matching results of synthesized and rotated optical images. We also give the matching results of synthesized and original images. Figures 15, 16, and 17 show the NOQMs, outlier ratio, and $RMSE_{CP}$ of the two methods in various rotation angles. It can be seen that the NOQMs of the two methods only decrease slightly with increased angle of rotation. Meanwhile, for the three scenarios, the average outlier ratios of KCG-GAN and pix2pixHD are distributed among 70.6% $\sim$ 74.8% and 76.3% $\sim$ 82.2% respectively, and the average $RMSE_{CP}$s of KCG-GAN and pix2pixHD are distributed among 2 $\sim$ 2.8 and 2.4 $\sim$ 3.3 respectively. It reveals that the NOQMs of the two methods are both robust to rotation changing, and the correct correspondences derived by the two methods can provide reliable matchings for subsequent processing. Moreover, KCG-GAN significantly outperforms pix2pixHD in NOQMs and outlier ratio comparisons of rotation tests. It implies that spatial information constraints of KCG-GAN keep the superior in generating correct correspondences in varying rotation angles.

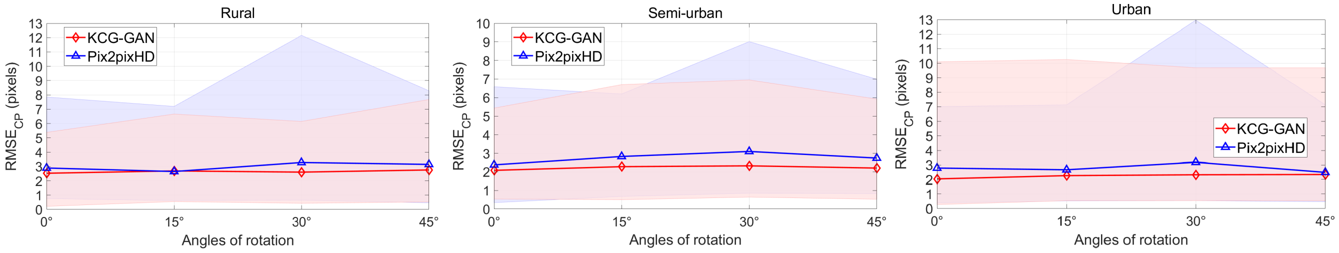

FIGURE 17. ***$RMSE_{CP}$*s of rotation tests, where markers and shadows represent averages and ranges of *$RMSE_{CP}$*s.**

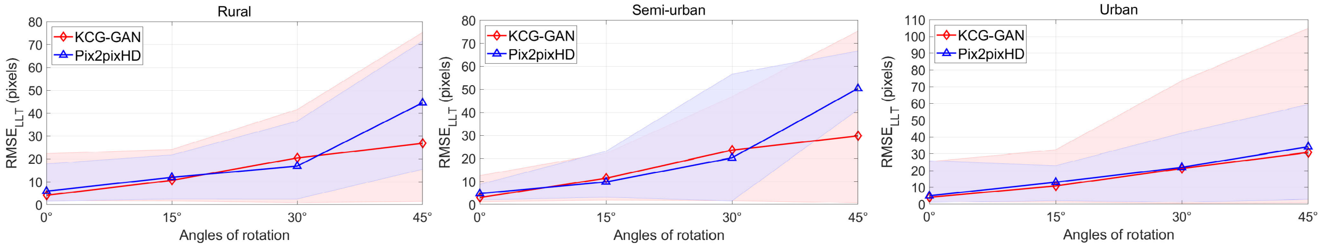

FIGURE 18. ***$RMSE_{LLT}$*s of rotation tests, where markers and shadows represent averages and ranges of *RMSE*s.**

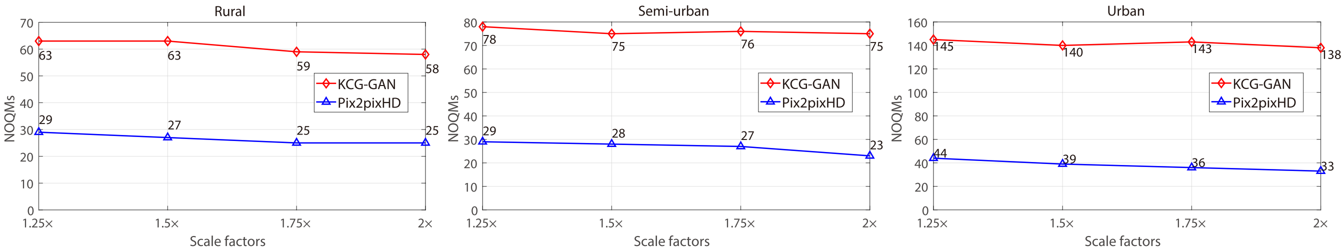

FIGURE 19. **The number of qualified matchings of where scale tests.**

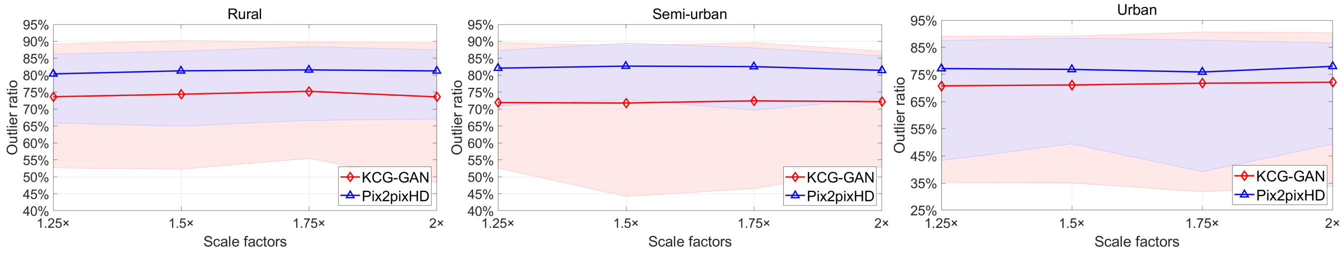

FIGURE 20. **Outlier ratios of scale tests, where markers and shadows represent averages and ranges of outlier ratios.**

On the other hand, the values of $RMSE_{LLT}$s show rising trends with increased angles of rotation (see Figure 18). It is because that although the SIFT descriptors of correct correspondences are rotation-invariant, the ability to remove outliers of the LLT method declines with increased angles of rotation.

CASES OF SCALE CHANGING

We use bicubic interpolation to scale up the original optical images to 1.25, 1.5, 1.75, and 2 times. Then we evaluate the matching results of synthesized and scaled optical images. Figures 19, 20, 21, and 22 show NOQMs, outlier ratio, $RMSE_{CP}$, and $RMSE_{LLT}$s of the two methods in different scale factors, respectively. It can be seen that only NOQMs of the two methods slightly decline with increased scale factors. In contrast, the average values of outlier ratio, $RMSE_{CP}$, and $RMSE_{LLT}$s of the two methods are stable with respect to the different scale factors. Specifically, for the three scenarios, the average outlier ratios of KCG-GAN and pix2pixHD are distributed among 71% $\sim$ 75.3% and 76% $\sim$ 82.6%, respectively; the average $RMSE_{CP}$s of KCG-GAN and pix2pixHD are distributed among 2 $\sim$ 2.6 and 2.3 $\sim$ 3.1, respectively;

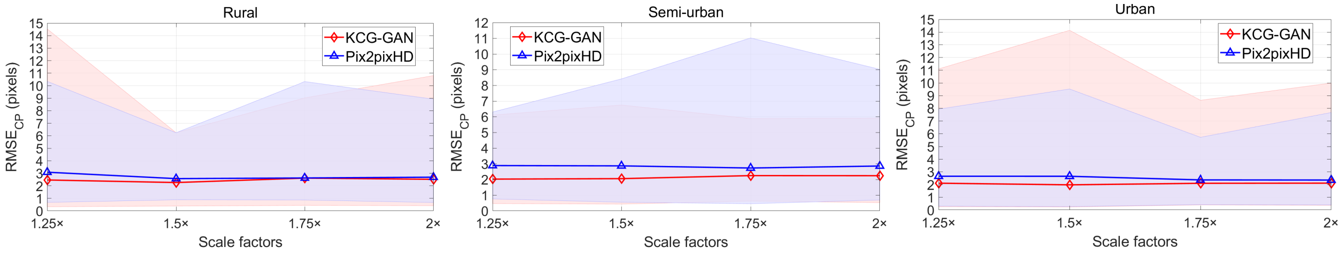

FIGURE 21. ***$RMSE_{CP}$*s of scale tests, where markers and shadows represent averages and ranges of *$RMSE_{CP}$*s.**

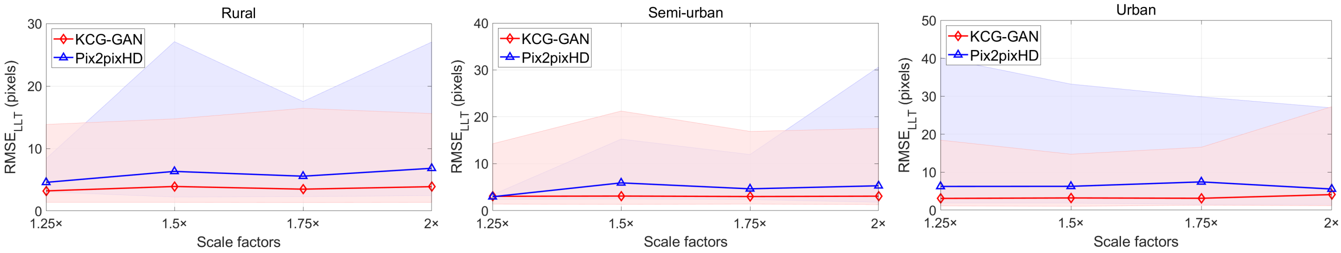

FIGURE 22. ***$RMSE_{LLT}$*s of scale tests, where markers and shadows represent averages and ranges of *RMSE*s.**

the average $RMSE_{LLT}$s of KCG-GAN and pix2pixHD are distributed among $3 \sim 4.1$ and $2.9 \sim 7.4$, respectively. It reveals that SAR-optical image matchings based on KCG-GAN and pix2pixHD are both robust to scale changing, and the correct correspondences are also easy to be identified by the LLT method. Moreover, KCG-GAN continues to outperform pix2pixHD in scale tests significantly.

In summary, the correct correspondences generated by the matchings based on KCG-GAN and pix2pixHD are robust to the rotation and scale changing. This is because the two image synthesis methods transform the SAR-optical image matching into optical-optical image matching. Then, the scale, rotation, and illumination invariant single-mode feature descriptor could be applied to achieve robust image matching, e.g., SIFT [39], Speeded-Up Robust Features (SURF) [69], etc.

TABLE 4. NOQMs of coarse matching results of the three SIFT-like algorithms on matching original and KCG-GAN synthesized SAR-optical images.

Methods	Rural (300)	Semi-urban (300)	Urban (300)
SIFT	0	0	0
PSO-SIFT	**10**	**17**	**19**
SAR-SIFT	2	1	7
SIFT + KCG-GAN (Ours)	63	78	145
PSO-SIFT + KCG-GAN	**65**	**82**	**152**
SAR-SIFT + KCG-GAN	61	78	132

5) COMPARISONS OF THE THREE SIFT-LIKE ALGORITHMS

We firstly show the NOQMs of coarse matching results of the three SIFT-like algorithms on original SAR-optical matching and KCG-GAN based SAR-optical matching in Table 4. The SAR-optical images are the 900 test image pairs introduced in Section IV-B. Obviously, based on KCG-GAN, the three algorithms obtain at least 4.8 times more qualified matchings (PSO-SIFT tests in the semi-urban scenario) than those used to match the original SAR-optical images directly. It means that our KCG-GAN significantly improves the performances of the three algorithms on matching SAR-optical images.

We then use another three criteria—NOCCs (Figure 23), outlier ratios (Figure 24), and $RMSE_{LLT}$s (Figure 25) to show more details of the comparisons. Two things are worth noting about the three criteria. The first one is that the three criteria are obtained from matching KCG-GAN synthesized SAR-optical images. The second one is that the NOCCs of SIFT in Figure 23 are ranked in ascending order, and the NOCCs obtained by PSO-SIFT and SAR-SIFT are arranged in terms of the same image-pairs of SIFT. As a result, the outlier ratios and RMSEs of the three algorithms shown in Figure 24 and 25 are arranged in terms of the same image-pairs of NOCCs in Figure 23.

Among the three algorithms, PSO-SIFT obtains the most qualified matchings from both original and KCG-GAN synthesized SAR-optical images (see Table 4). This is because PSO-SIFT increases the number of correct correspondences by combining the position, scale, and orientation of each keypoint [61] (see Figure 23). However, PSO-SIFT also preserves much more outliers than other two algorithms (see Figure 24), making the outliers removing algorithm—LLT hard to discriminate correct correspondences from so many outliers (see Figure 25). On the other hand, although SAR-SIFT preserves the lowest outlier ratios (see Figure 24), it doesn't preserve the lowest RMSEs. This is because the SAR-Harris detector of SAR-SIFT generates some correspondences that are too close to each other [60], resulting in

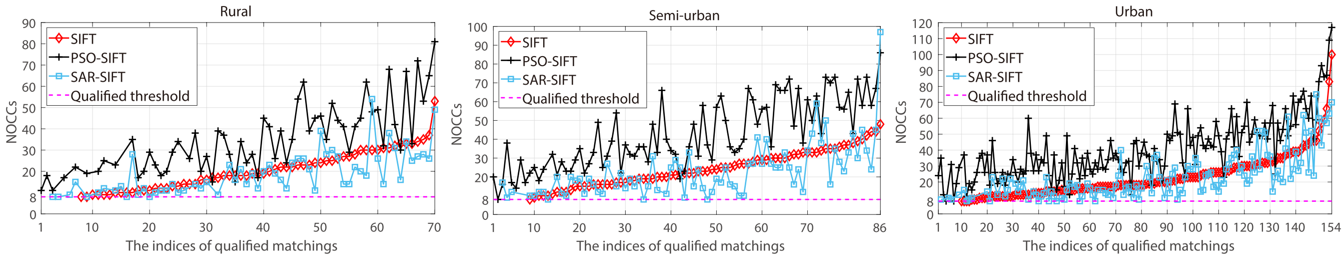

FIGURE 23. NOCCs of the qualified matchings obtained by the three SIFT-like algorithms based on KCG-GAN.

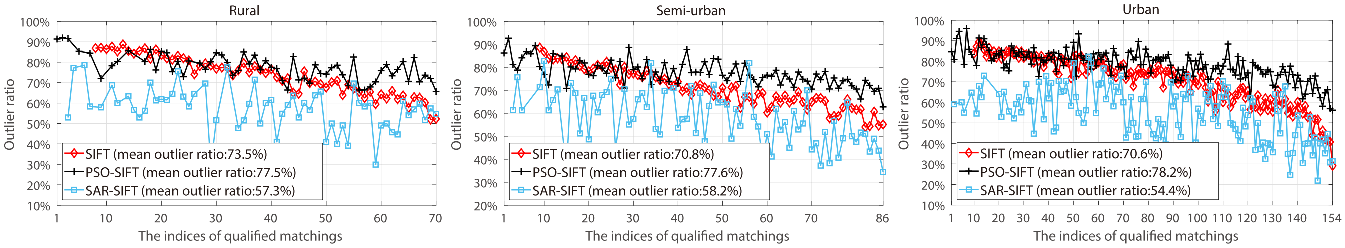

FIGURE 24. Outlier ratios of the qualified matchings obtained by the three SIFT-like algorithms based on KCG-GAN.

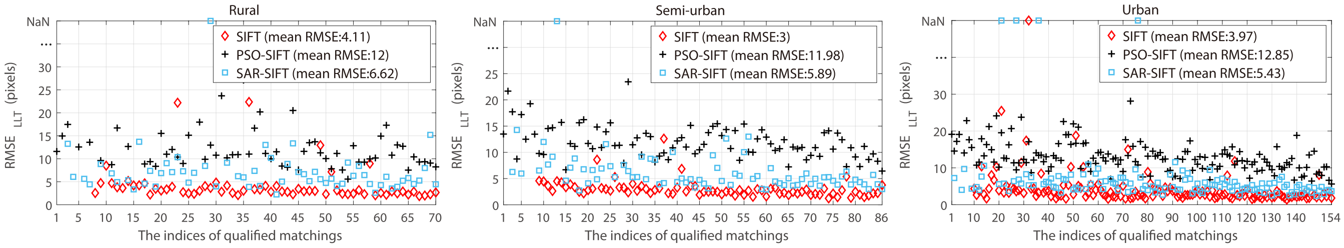

FIGURE 25. $RMSE_{LLT}$s of qualified matchings obtained by the three SIFT-like algorithms based on KCG-GAN, where NaN represents failed experiments.

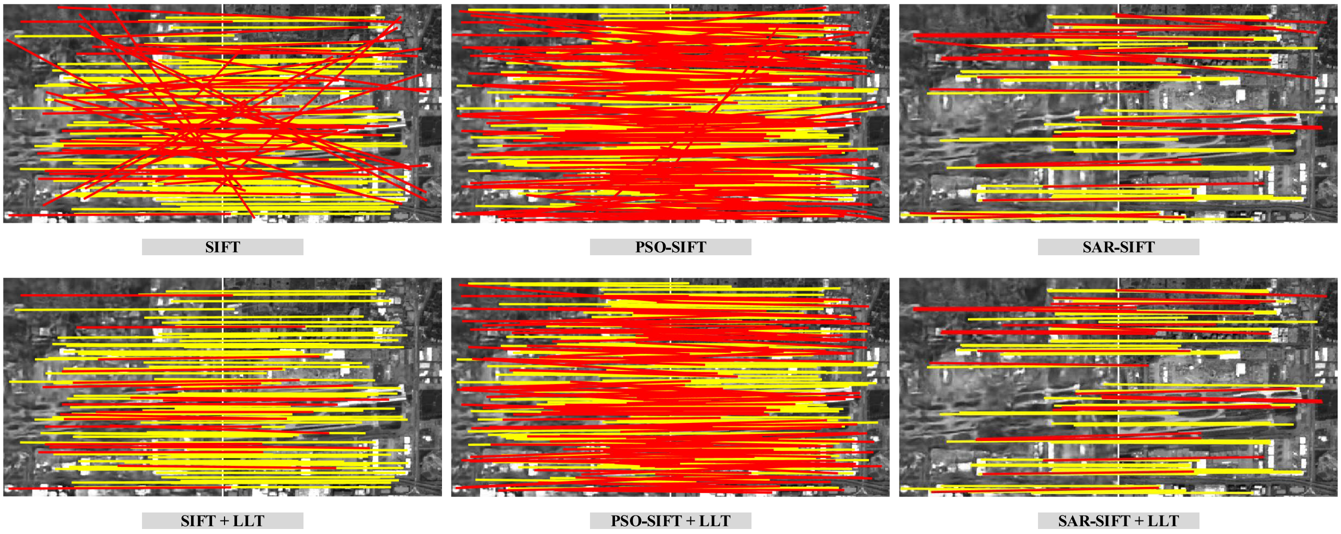

FIGURE 26. Applying the LLT algorithm on removing outliers of image matching results of three SIFT-like algorithms based on KCG-GAN (yellow lines: lines between correct matched points; blue lines: lines between incorrect matched points).

an imprecise transformation matrix generation. SIFT obtains the lowest $RMSE_{LLT}$s and preserves the medium NOCCs and outlier ratios compared to PSO-SIFT and SAR-SIFT.

We show a typical example of image matching results of the three SIFT-like algorithms with and without the LLT algorithm in Figure 26. In this example, NOCCs preserved by

SIFT, PSO-SIFT, and SAR-SIFT are 83, 109, and 63, respectively; outlier ratios of SIFT, PSO-SIFT, and SAR-SIFT are 41%, 57%, and 31%, respectively. Hence, PSO-SIFT preserves the most correct correspondences, and SAR-SIFT obtains the lowest outlier ratio. However, after removing outliers by the LLT algorithm, outliers ratios of SIFT, PSO-SIFT, and SAR-SIFT decline to 21%, 50%, and 31%, respectively; $RMSE_{LLT}$s obtained by SIFT, PSO-SIFT, and SAR-SIFT are 3.51, 6.62, and 4.01, respectively. It reveals that outliers generated by SIFT are much easier to be removed by LLT than outliers generated by PSO-SIFT and SAR-SIFT.

Therefore, in this work, we use the original SIFT algorithm for achieving the low RMSEs. Meanwhile, the original SIFT matching results are more general than the other SIFT-like algorithms' matching results because the original SIFT is not designed for any specific kinds of images.

V. CONCLUSION

In this work, we presented a KCG-GAN to improve the image quality of synthesizing by controlling the spatial information. We used k-means segmentations as one of the inputs of KCG-GAN and applied feature matching loss, segmentation loss, and L1 loss to the training of KCG-GAN. Moreover, we developed a straightforward 1D k-means algorithm to obtain the repeatable k-means segmentations from massive grayscale images, which is more efficient than the state-of-the-art k-means algorithms. We conducted qualitative, quantitative, generalization, and robustness tests for KCG-GAN compared with pix2pixHD. Qualitative results indicated that KCG-GAN could preserve more spatial structures than pix2pixHD. Quantitative results showed that, compared with pix2pixHD, KCG-GAN synthesized higher-quality optical images and obtained 1.86, 2.43, and 3.15 times more qualified matchings in rural, semi-urban, and urban scenarios, respectively. Generalization testing results showed that KCG-GAN obtains better generalization than pix2pixHD in SAR-to-optical image synthesis. Robustness testing results showed that KCG-GAN is robust to rotation and scale changing. We also tested three SIFT-like algorithms on original SAR-optical matching and KCG-GAN based SAR-optical matching. Experimental results showed that, based on our KCG-GAN, the three algorithms obtained at least 4.8 times more qualified matchings than those directly used to match original SAR-optical images.

Future work will focus on applying sophisticated segmentation methods [19]–[23] to SAR-to-optical image synthesis for improving the generalization and accuracy of the SAR-optical image matching.

REFERENCES

[1] K. Sprohnle, E.-M. Fuchs, and P. Aravena Pelizari, "Object-based analysis and fusion of optical and SAR satellite data for dwelling detection in refugee camps," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 5, pp. 1780–1791, May 2017.

[2] L. Gomez-Chova, D. Tuia, G. Moser, and G. Camps-Valls, "Multimodal classification of remote sensing images: A review and future directions," *Proc. IEEE*, vol. 103, no. 9, pp. 1560–1584, Sep. 2015.

[3] A. Ley, O. Dhondt, S. Valade, R. Haensch, and O. Hellwich, "Exploiting GAN-based SAR to optical image transcoding for improved classification via deep learning," in *Proc. 12th Eur. Conf. Synth. Aperture Radar (EUSAR)*, Jun. 2018, pp. 1–6.

[4] J. De Alban, G. Connette, P. Oswald, and E. Webb, "Combined landsat and L-Band SAR data improves land cover classification and change detection in dynamic tropical landscapes," *Remote Sens.*, vol. 10, no. 2, p. 306, Feb. 2018.

[5] H. Zhang and R. Xu, "Exploring the optimal integration levels between SAR and optical data for better urban land cover mapping in the pearl river delta," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 64, pp. 87–95, Feb. 2018.

[6] B. Mishra and J. Susaki, "SAR and optical data fusion for land use and cover change detection," in *Proc. IEEE Geosci. Remote Sens. Symp.*, Jul. 2014, pp. 4691–4694.

[7] X. Niu, M. Gong, T. Zhan, and Y. Yang, "A conditional adversarial network for change detection in heterogeneous images," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 1, pp. 45–49, Jan. 2019.

[8] M. Campos-Taberner, F. García-Haro, G. Camps-Valls, G. Grau-Muedra, F. Nutini, L. Busetto, D. Katsantonis, D. Stavrakoudis, C. Minakou, L. Gatti, M. Barbieri, F. Holecz, D. Stroppiana, and M. Boschetti, "Exploitation of SAR and optical sentinel data to detect rice crop and estimate seasonal dynamics of leaf area index," *Remote Sens.*, vol. 9, no. 3, p. 248, Mar. 2017.

[9] K. Van Tricht, A. Gobin, S. Gilliams, and I. Piccard, "Synergistic use of radar Sentinel-1 and optical Sentinel-2 imagery for crop mapping: A case study for Belgium," *Remote Sens.*, vol. 10, no. 10, p. 1642, Oct. 2018.

[10] Y. Ye, J. Shan, L. Bruzzone, and L. Shen, "Robust registration of multimodal remote sensing images based on structural similarity," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2941–2958, May 2017.

[11] Y. Ye, L. Bruzzone, J. Shan, F. Bovolo, and Q. Zhu, "Fast and robust matching for multimodal remote sensing image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9059–9070, Nov. 2019.

[12] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8798–8807.

[13] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2337–2346.

[14] S. W. Huang, C. T. Lin, S. P. Chen, Y. Y. Wu, P. H. Hsu, and S. H. Lai, "Auggan: Cross domain adaptation with gan-based data augmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Munich, Germany, Sep. 2018, pp. 731–744.

[15] S. Jiang, Z. Tao, and Y. Fu, "Segmentation guided image-to-image translation with adversarial networks," in *Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, May 2019, pp. 1–7.

[16] M. Fuentes Reyes, S. Auer, N. Merkle, C. Henry, and M. Schmitt, "SAR-to-optical image translation based on conditional generative adversarial networks—Optimization, opportunities and limits," *Remote Sens.*, vol. 11, no. 17, p. 2067, 2019.

[17] X. Jiang, J. Ma, J. Jiang, and X. Guo, "Robust feature matching using spatial clustering with heavy outliers," *IEEE Trans. Image Process.*, vol. 29, pp. 736–746, 2020.

[18] J. Ma, X. Jiang, J. Jiang, and Y. Gao, "Feature-guided Gaussian mixture model for image matching," *Pattern Recognit.*, vol. 92, pp. 231–245, Aug. 2019.

[19] Y. Duan, F. Liu, L. Jiao, P. Zhao, and L. Zhang, "SAR image segmentation based on convolutional-wavelet neural network and Markov random field," *Pattern Recognit.*, vol. 64, pp. 255–267, Apr. 2017.

[20] F. Liu, Y. Duan, L. Li, L. Jiao, J. Wu, S. Yang, X. Zhang, and J. Yuan, "SAR image segmentation based on hierarchical visual semantic and adaptive neighborhood multinomial latent model," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 7, pp. 4287–4301, Jul. 2016.

[21] F. Liu, J. Shi, L. Jiao, H. Liu, S. Yang, J. Wu, H. Hao, and J. Yuan, "Hierarchical semantic model and scattering mechanism based PolSAR image classification," *Pattern Recognit.*, vol. 59, pp. 325–342, Nov. 2016.

[22] L. Jiao, M. Zhang, F. Liu, W. Ma, and L. Li, "A two-stage evolutionary fuzzy clustering framework for noisy image segmentation," *IEEE Access*, vol. 8, pp. 186663–186678, 2020.

[23] S. Liu, Q. Shi, and L. Zhang, "Few-shot hyperspectral image classification with unknown classes using multitask deep learning," *IEEE Trans. Geosci. Remote Sens.*, early access, Sep. 4, 2020, doi: 10.1109/TGRS.2020.3018879.

[24] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5967–5976.

[25] Y. Xiang, R. Tao, F. Wang, and H. You, "Automatic registration of optical and SAR images VIA improved phase congruency," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2019, pp. 931–934.

[26] X. Xie, Y. Zhang, X. Ling, and X. Wang, "A novel extended phase correlation algorithm based on log-Gabor filtering for multimodal remote sensing image registration," *Int. J. Remote Sens.*, vol. 40, no. 14, pp. 5429–5453, Jul. 2019.

[27] J. Markiewicz, K. Abratkiewicz, A. Gromek, W. Ostrowski, P. Samczyński, and D. Gromek, "Geometrical matching of SAR and optical images utilizing ASIFT features for SAR-based navigation aided systems," *Sensors*, vol. 19, no. 24, p. 5500, Dec. 2019.

[28] N. Merkle, W. Luo, S. Auer, R. Müller, and R. Urtasun, "Exploiting deep matching and SAR data for the geo-localization accuracy improvement of optical satellite images," *Remote Sens.*, vol. 9, no. 6, p. 586, Jun. 2017.

[29] L. Hughes, M. Schmitt, and X. Zhu, "Mining hard negative samples for SAR-optical image matching using generative adversarial networks," *Remote Sens.*, vol. 10, no. 10, p. 1552, Sep. 2018.

[30] T. Bürgmann, W. Koppe, and M. Schmitt, "Matching of TerraSAR-X derived ground control points to optical image patches using deep learning," *ISPRS J. Photogramm. Remote Sens.*, vol. 158, pp. 241–248, Dec. 2019.

[31] R. Hansch, O. Hellwich, and X. Tu, "Machine-learning based detection of corresponding interest points in optical and SAR images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2016, pp. 1492–1495.

[32] D. Quan, S. Wang, X. Liang, R. Wang, S. Fang, B. Hou, and L. Jiao, "Deep generative matching network for optical and SAR image registration," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2018, pp. 6215–6218.

[33] H. Zhang, W. Ni, W. Yan, D. Xiang, J. Wu, X. Yang, and H. Bian, "Registration of multimodal remote sensing image based on deep fully convolutional neural network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 3028–3042, Aug. 2019.

[34] N. Merkle, P. Fischer, S. Auer, and R. Muller, "On the possibility of conditional adversarial networks for multi-sensor image matching," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 2633–2636.

[35] N. Merkle, S. Auer, R. Muller, and P. Reinartz, "Exploring the potential of conditional adversarial networks for optical and SAR image matching," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 6, pp. 1811–1820, Jun. 2018.

[36] J. Zhang, W. Ma, Y. Wu, and L. Jiao, "Multimodal remote sensing image registration based on image transfer and local features," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 8, pp. 1210–1214, Aug. 2019.

[37] L. H. Hughes, M. Schmitt, L. Mou, Y. Wang, and X. X. Zhu, "Identifying corresponding patches in SAR and optical images with a pseudo-siamese CNN," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 784–788, May 2018.

[38] L. H. Hughes and M. Schmitt, "A semi-supervised approach to SAR-optical image matching," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. IV-2/W7, pp. 71–78, Sep. 2019.

[39] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[40] S. Hoffmann, C.-A. Brust, M. Shadaydeh, and J. Denzler, "Registration of high resolution SAR and optical satellite imagery using fully convolutional networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2019, pp. 5152–5155.

[41] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss," in *Proc. Conf. Neural Inf. Process. Syst. (NeurIPS)*, Long Beach, CA, USA, Dec. 2017, pp. 4826–4837.

[42] L. H. Hughes, N. Merkle, T. Burgmann, S. Auer, and M. Schmitt, "Deep learning for SAR-optical image matching," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2019, pp. 4877–4880.

[43] J. Zhang, J. Zhou, and X. Lu, "Feature-guided SAR-to-optical image translation," *IEEE Access*, vol. 8, pp. 70925–70937, 2020.

[44] Y. Li, R. Fu, X. Meng, W. Jin, and F. Shao, "A SAR-to-Optical image translation method based on conditional generation adversarial network (cGAN)," *IEEE Access*, vol. 8, pp. 60338–60343, 2020.

[45] H. Toriya, A. Dewan, and I. Kitahara, "SAR2OPT: Image alignment between multi-modal images using generative adversarial networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. IGARSS*, Jul. 2019, pp. 923–926.

[46] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, May 2015. [Online]. Available: https://www.robots.ox.ac.uk/~vgg/research/very_deep/#pub and https://iclr.cc/archive/www/2015.html

[47] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "PatchMatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.*, vol. 28, no. 3, p. 24, 2009.

[48] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, New Orleans, LA, USA, Jan. 2007, pp. 1027–1035.

[49] D. Sculley, "Web-scale k-means clustering," in *Proc. 19th Int. Conf. World Wide Web - WWW*, 2010, pp. 1177–1178.

[50] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[51] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2794–2802.

[52] D. Lin, K. Fu, Y. Wang, G. Xu, and X. Sun, "MARTA GANs: Unsupervised representation learning for remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 2092–2096, Nov. 2017.

[53] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.

[54] C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," in *Proc. 2016 Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands, Oct. 2016, pp. 702–716.

[55] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, May 2015. [Online]. Available: https://iclr.cc/archive/www/2015.html and https://arxiv.org/abs/1412.6980

[56] M. Schmitt, L. H. Hughes, and X. X. Zhu, "The sen1-2 dataset for deep learning in sar-optical data fusion," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vols. IV–1, pp. 141–146, Sep. 2018.

[57] H. Wang and M. Song, "Ckmeans.1d.dp: Optimal k-means clustering in one dimension by dynamic programming," *R J.*, vol. 3, no. 2, pp. 29–33, 2011.

[58] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[59] *Cran—Package Ckmeans.1d.dp*. Accessed: Dec. 3, 2020. [Online]. Available: https://cran.r-project.org/web/packages/Ckmeans.1d.dp/

[60] F. Dellinger, J. Delon, Y. Gousseau, J. Michel, and F. Tupin, "SAR-SIFT: A SIFT-like algorithm for SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 1, pp. 453–466, Jan. 2015.

[61] W. Ma, Z. Wen, Y. Wu, L. Jiao, M. Gong, Y. Zheng, and L. Liu, "Remote sensing image registration with modified SIFT and enhanced feature matching," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 1, pp. 3–7, Jan. 2017.

[62] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[63] A. Vedaldi and B. Fulkerson .(2008). *VLFeat: An Open and Portable Library of Computer Vision Algorithms*. [Online]. Available: http://www.vlfeat.org/

[64] R. I. Hartley, "In defense of the eight-point algorithm," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 6, pp. 580–593, Jun. 1997.

[65] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," *Nature*, vol. 293, no. 5828, pp. 133–135, Sep. 1981.

[66] H. Goncalves, J. A. Goncalves, and L. Corte-Real, "Measures for an objective evaluation of the geometric correction process quality," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 2, pp. 292–296, Apr. 2009.

[67] J. Ma, H. Zhou, J. Zhao, Y. Gao, J. Jiang, and J. Tian, "Robust feature matching for remote sensing image registration via locally linear transforming," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 12, pp. 6469–6481, Dec. 2015.

[68] M. Reichstein, G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais, and Prabhat, "Deep learning and process understanding for data-driven Earth system science," *Nature*, vol. 566, no. 7743, pp. 195–204, Feb. 2019.

[69] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (surf)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, p. 346—359, 2008.

WEN-LIANG DU received the B.S., M.S., and Ph.D. degrees in computer science and technology from the Macau University of Science and Technology, Taipa, Macau, in 2011, 2014, and 2018, respectively.

He is currently a Postdoctoral Research Associate with the School of Computer Science and Technology, China University of Mining and Technology. His research interests include computer vision, image processing, and pattern recognition.

YONG ZHOU received the M.S. and Ph.D. degrees in control theory and control engineering from the China University of Mining and Technology, in 2003 and 2006, respectively. He is currently a Professor with the School of Computer Science and Technology, China University of Mining and Technology. His research interests include machine learning, intelligence optimization, and data mining.

JIAQI ZHAO (Member, IEEE) received the B.Eng. degree in intelligence science and technology and the Ph.D. degree in circuits and systems from Xidian University, Xi'an, China, in 2010 and 2017, respectively. From 2013 to 2014, he was an exchange Ph.D. Student with the Leiden Institute for Advanced Computer Science (LIACS), University of Leiden, The Netherlands. He is currently with the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou, China. His current research interests include multiobjective optimization, machine learning, deep learning, and image processing.

XIAOLIN TIAN graduated from Peking University, Beijing, China.

She was a Lecturer and an Associate Professor with Peking University, in 1982 and 1989, respectively. She was a Visiting Scholar with the Artificial Intelligence Laboratory, Computer Vision Group, EECS, University of Michigan, Ann Arbor, MI, USA, in 1989; and a Research Associate with the Institute for Advanced Computer Studies, University of Maryland, Collage Park, MA, USA, in 1990. She is currently a Professor with the Faculty of Information Technology and the Space Science Institute/Lunar and Planetary Science Laboratory, Macau University of Science and Technology, Taipa, Macau. Her research interests include image processing and pattern recognition. She is also working on medical image processing and the intelligent computing and automatic processing for huge data of Change.