

# A Fast Algorithm for SAR Image Segmentation Based on Key Pixels

Ronghua Shang<sup>1</sup>, Member, IEEE, Yijing Yuan, Licheng Jiao, Senior Member, IEEE, Biao Hou, Member, IEEE, Amir Masoud Ghalamzan Esfahani, Member, IEEE, and Rustam Stolkin, Member, IEEE

**Abstract**—Recent high-performance clustering methods process all pixels when segmenting an image, which results in a very large time complexity of these algorithms. Additionally, the performance of such algorithms can be severely affected by noise when dealing with highly polluted images. To address these problems, we propose a new unsupervised algorithm for segmenting synthetic aperture radar images based on a fuzzy clustering approach, called fast fuzzy C-means clustering based on key pixels. Our algorithm first selects a subset of special “key” pixels based on the rule of local extrema, and then performs image segmentation on only these key pixels using fuzzy clustering combined with nonlocal information. Next, the remaining non-key pixels can be rapidly segmented by combining the clustering results of the key pixels with a similarity metric rule which is robust to speckle noise. This approach greatly accelerates overall image segmentation because the time-consuming clustering operation is only performed on a small subset of pixels. We show the effectiveness of our proposed algorithm by a series of experiments including segmenting twelve simulated and four real synthetic aperture radar images. Moreover, to validate our results, we compare the segmentation results obtained by our algorithm with those obtained by seven other state-of-the-art segmentation algorithms from the literature. The experimental results suggest that our algorithm outperforms other state-of-the-art segmentation algorithms in both computational speed and speckle noise suppression.

**Index Terms**—Fuzzy clustering, image segmentation, key pixels, nonlocal information, synthetic aperture radar (SAR).

## I. INTRODUCTION

**R**ADAR machine vision is a new research field of electronics in which different processing-based applications are performed including synthetic aperture radar (SAR) image processing, remote sensing, geoscience, and so on [1]–[3]. SAR images have the advantage to generate more useful information than optical images due to robustness against different weather

or light conditions [4], so precise and fast processing of SAR images is important for a variety of different fields of study [5]. On the other hand, SAR images are usually affected by speckle noise, which is multiplicative noise [6], due to its imaging mechanism [3]. Hence, the processing of SAR images is a challenging problem [7]. One of the primary importance in the processing of SAR satellite images is SAR image segmentation [8], and this technique has many applications such as the military areas’ mapping [4], target recognition [9], natural disasters’ assessment [10], and so on. The goal of SAR image segmentation is to segment an SAR image into a number of nonoverlapping but connected regions [11], where pixels in the same region share similar qualities, i.e., pixels from different regions have different properties and good performance is obtained [12].

Segmentation is a fundamental task in SAR image processing [13] and many different methods have been proposed over several decades of research [14]. Thresholding methods [15] are easy to understand and implement but have low accuracy [16]. Yin and Yang [17] presented a level set method that is a powerful tool to depict the contours of regions. This method has become very popular in recent years, but it can perform poorly on images that are highly polluted with noise. Other approaches include Markov random fields, Stolkin *et al.* [18], and the classical machine learning methods such as the support vector learning and so forth are also effective for classification and clustering [19], [20]. Clustering methods are often used for segmenting SAR images [21] such as a coarse-to-fine clustering method to cluster near-duplicate images [22]. And clustering methods group pixels with similar properties and separate pixels with different properties [21]. For instance, fuzzy C-means (FCM) clustering [23] is a commonly used algorithm. This algorithm minimizes an objective function through updating a fuzzy membership matrix and clustering centers iteratively. It then hard divides each pixel according to the membership matrix. The spatial continuity of SAR images is a very useful form of *a priori* knowledge [24], which is often used to improve the accuracy of clustering [25]. It is essential to preserve image details while suppressing speckle noise when processing SAR images [26]. Ahmed *et al.* [27] proposed an FCM clustering algorithm with spatial constraints (FCM\_S) that added spatial local information to FCM for the first time to improve the robustness of the algorithm. However, this method requires calculating a spatial neighborhood term at each iteration resulting in a very long running time. Chen and Zhang [28] proposed FCM\_S1 and FCM\_S2 to overcome the large computation time of FCM\_S. These two algorithms used

Manuscript received November 27, 2016; revised March 26, 2017, July 1, 2017, and August 2, 2017; accepted August 15, 2017. Date of publication October 8, 2017; date of current version December 19, 2017. This work was supported by the National Natural Science Foundation of China under Grant 61371201. (Corresponding author: Ronghua Shang.)

R. Shang, Y. Yuan, L. Jiao, and B. Hou are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, Xidian University, Xi’an 710071, China (e-mail: rhshang@mail.xidian.edu.cn; 734093492@qq.com; 280331025@qq.com; avcodec@163.com).

A. M. Ghalamzan Esfahani and R. Stolkin are with the Extreme Robotics Lab, University of Birmingham, Birmingham B15 2TT, U.K. (e-mail: amir.ghalamzan@gmail.com; R.Stolkin@cs.bham.ac.uk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JSTARS.2017.2743338

local information to generate a mean filter image and a median filter image in advance to speed up subsequent clustering. Furthermore, the enhanced FCM algorithm and fast generalized FCM algorithm proposed by Szilagy *et al.* [29] and Cai *et al.* [30] used local information and image histograms to achieve fast segmentation. However, the common shortcoming of the above-mentioned algorithms is that they require manually setting an important parameter to achieve the balance between the suppression of noise and the preservation of details of the original image. Later, Krinidis and Chatzis [31] proposed a robust fuzzy local information C-means clustering algorithm (FLICM) which did not rely on any parameter settings. This algorithm adds a neighborhood term to the original objective function of FCM and automatically calculates a weight based on the spatial distance between pixels in the neighborhood. Gong *et al.* proposed the reformulated FLICM [32] and an FCM clustering algorithm with local information and kernel metric (KWFLICM) [33]. The former method added a local variance coefficient to the neighborhood term of FLICM whereas the latter method redefined the weight by using both spatial distance and gray values of pixels. In addition, a kernel metric was added to make the algorithm more efficient for image segmentation. Xiang *et al.* [34] proposed a kernel FCM algorithm with pixel intensity and location information (ILKFCM) for SAR image segmentation based on KWFLICM. The algorithm used the wavelet transform of a local neighborhood to generate new features and a new weight. Moreover, kernel metrics were added at each iteration.

The above-mentioned algorithms added local spatial information to FCM and improved the accuracy of clustering. However, achieving satisfactory segmentation using only local information is not feasible for SAR images affected by a large amount of speckle noise. Hence, many approaches have been proposed in recent years to use nonlocal information to suppress speckle noise more effectively [35]. For example, Ji and Wang [36] proposed a nonlocal fuzzy clustering algorithm with between-cluster separation measure (NS\_FCM), in which the nonlocal information was utilized in a modified version of FCM. This greatly improved the robustness of the algorithm to speckle noise. Shang *et al.* [37] also proposed a spatial fuzzy clustering algorithm with kernel metric based on immune clone (CKS\_FCM). CKS\_FCM uses a nonlocal mean method to filter the image to achieve better segmentation results. Liu *et al.* [38] proposed an FCM algorithm which added regional information to the objective function of FCM (ALFCM). It used both pixel-based and region-based information together to generate more useful spatial information.

Although promising segmentation results were achieved by the fuzzy clustering algorithms mentioned above, the computational expense of clustering all pixels of the image makes those algorithms very slow. Moreover, the use of local or nonlocal spatial information adds significant additional processing time. Clustering all pixels of an image is especially problematic for images with large size. In addition, for very noisy images, clustering process of all the pixels becomes even more challenging [39].

Recently, some proposed studies deal with the segmentation and classification of very high-resolution images using

point-wise methods [40]. These points are usually selected according to various rules so that they represent various kinds of important image information [41]. Pham *et al.* proposed two point-wise methods to deal with classification of very high-resolution multispectral image [42] and SAR image change detection [43]. Graph theory was also used in these studies. These algorithms only deal with a small number of selected pixels rather than all pixels in the image. The processing of reduced numbers of pixels significantly decreases the running time. However, while these subsampled pixels can represent the image to some extent, discarding a large number of other pixels risks losing important information of the original image.

In order to overcome the shortcomings of the fuzzy clustering algorithms and the point-wise methods mentioned above, we propose a fast algorithm for SAR image segmentation based on a set of key pixels. In contrast to the previous clustering algorithms used for SAR image segmentation, we perform a fuzzy clustering method, based on nonlocal information, only on key pixels. Clustering just a small number of key pixels can greatly reduce the impact of noise on the clustering iterations. Furthermore, it significantly reduces the computation time. On the other hand, our algorithm segments the remaining non-key pixels quickly by exploiting the clustering result obtained from the key pixels. This is the main difference between our proposed algorithm and the existing point-wise methods. Our method can segment both key and non-key pixels in a short time with high accuracy.

The contributions of this paper are as follows:

- 1) We perform clustering only on a few key pixels resulting in a significantly shorter running time of segmentation.
- 2) We use the key pixels, which are less affected by noise, and nonlocal spatial information together to execute clustering. Hence, a very accurate clustering result can be obtained.
- 3) We show how to use the clustering results of key pixels to segment large numbers of non-key pixels with high accuracy in a short time.

The remainder of this paper is structured as follows. Section II presents the details of the fast algorithm for SAR image segmentation based on key pixels. Section III analyzes the sensitivity of parameters used in the proposed algorithm. Section IV compares the segmentation results of our algorithm against the results of several other state-of-the-art algorithms, on both simulated and real SAR images. Section V provides concluding remarks.

## II. METHODOLOGY

In this paper, we propose a fast algorithm for SAR image segmentation. We call this algorithm fast FCM clustering based on key pixels (FKP\_FCM). First, our algorithm separates the whole image into two parts: 1) key pixels and 2) non-key pixels. FKP\_FCM processes these two parts separately. Second, we apply a fuzzy clustering algorithm based on nonlocal information to segment only key pixels. We finally segment the non-key pixels based on the clustering result of key pixels. A flow chart of our algorithm is depicted in Fig. 1, and this figure shows the main steps of FKP\_FCM.

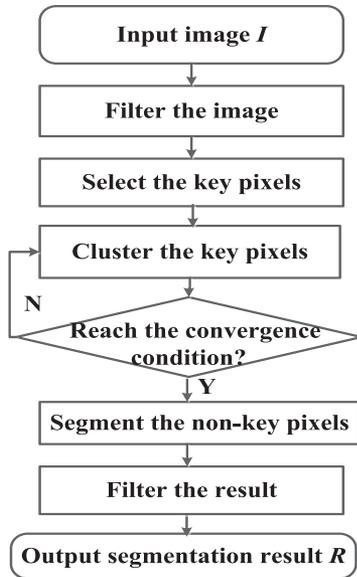


Fig. 1. Flow chart of FKP\_FCM.

### A. Key Pixels' Selection

Pham *et al.* [44] proposed a selection method of key points based on local extrema and proved that these points can represent basic information of an image without losing important information. In addition, Pham *et al.* proved that local extrema can result in a good performance on SAR image in [42] and [43]. They have also shown in [43] that using local maximum can produce similar results to the one by using local minimum. Hence, we use the rule of local maximum in our method to select key pixels.

Key pixels' selection is an important process to our approach since its result affects the segmentation of the entire image. We assume the characteristics of key pixels are

- 1) key pixels are uniformly distributed across the entire image;
- 2) they contain information about the corresponding areas which include the key pixels in the image; and
- 3) they are not often affected by noise.

Due to the influence of speckle noises, key pixels selected from the input image may have an uneven distribution based on the rule of local maximum. So, at first, we apply a low-pass filter on the SAR image, which can smooth the image homogeneously [43], in order to ensure the selected key pixels have a relatively uniform distribution over the whole image despite some losses of details. Although different simple filters are equally good, such as median filter or mean filter, we use Gaussian filter here.

Consider an input SAR image,  $I = \{I_{ab}, 0 \leq a \leq A, 0 \leq b \leq B\}$ , which can be segmented into  $c$  classes, where  $c \geq 2$  and  $I_{ab}$  is the gray value of the pixel in row  $a$  and column  $b$  of image  $I$ . Moreover,  $A \times B$  denotes the size of the image. In order to segment this image, we need to assign  $c$  classes by labeling each pixel of  $I$ . In addition,  $X = \{X_{ab}, 0 \leq a \leq A, 0 \leq b \leq B\}$  denotes the filtered image obtained after filtering image  $I$ . The pixel with the maximum intensity in its immediate neighborhood is called a local maximum.

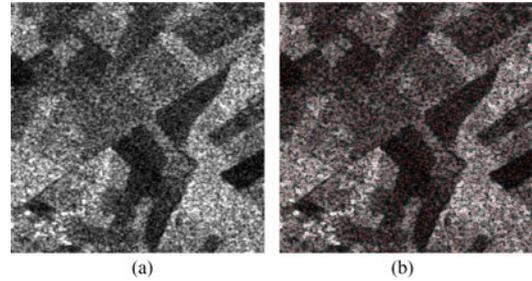


Fig. 2. Result of selecting key pixels on a real SAR image. (a) A real SAR image. (b) Result of selecting key pixels.

For selecting the key pixels, we use the principle of local maximum on the filtered image as follows:

$$\begin{cases} X_{ij} \in S, & \text{if } p = \arg \max(X(q)) \\ X_{ij} \in L, & \text{otherwise} \end{cases} \quad q \in \mathcal{N} \quad (1)$$

where  $p = (i, j)$  denotes the coordinate of current central pixel  $X_{ij}$ ,  $S$  is the set of key pixels,  $L$  is the set of non-key pixels, and  $\mathcal{N}$  represents the immediate neighborhood of  $X_{ij}$ , which is called the selection window. It is worth nothing that if  $X_{ij}$  is within a homogeneous area of the image, all the pixels in the immediate neighborhood of  $X_{ij}$  have the same intensity. Hence, no pixel can be selected as a key pixel within that limited area according to (1). Consequently, the final set  $S$  will be distributed unevenly across the whole image and it will lose information of these regions. Accordingly, wherever pixels within a certain neighborhood have the same gray values, we use a random selection of the central pixel. Fig. 2 shows the result of selecting key pixels on a real SAR image according to the approach described above.

A real SAR image is shown in Fig. 2(a), and the result of selecting key pixels in Fig. 2(b) shows 2105 key pixels with red dots, where the size of  $\mathcal{N}$  is  $3 \times 3$ . This figure shows the key pixels are almost uniformly distributed in the image, and within every limited area of the image in Fig. 2(a), some key pixels are selected. In addition, the speckle noise is distributed randomly in an SAR image and we only select small-amount local maximums as key pixels, which makes the selected key pixels suffering from less noise.

### B. Segmentation of Key Pixels

Existing clustering methods of image segmentation, such as FLICM [31], KWFLICM [33], ALFCM [38], use all pixels of an image for iterative computation, which makes these methods have very large computation time. To ensure a precise clustering result and a reduced computation time of image segmentation, we propose a new fuzzy clustering method based on nonlocal information only to segment the key pixels in SAR image. We are inspired from FLICM [31] to further suppress the effect of speckle noise by incorporating spatial nonlocal information, spatial local information, and gray values of pixels into the objective function as follows:

$$J = \sum_{i=1}^N \sum_{k=1}^c (u_{ki}^m \times \|S_i - V_k\|^2 + G_{ki}) \quad (2)$$

where  $N$  is the number of key pixels in set  $S$ ,  $c$  is the number of classes,  $m$  is the fuzzy coefficient, which is usually set to be 2,  $S_i$  represents the  $i$ th key pixel,  $V_k$  denotes the  $k$ th cluster center,  $u_{ki}$  is the membership degree of  $S_i$  to  $V_k$ , and  $G_{ki}$  is a fuzzy factor introduced newly with the form of

$$G_{ki} = \sum_{M_j \in M} w_{ij} (1 - u_{kj})^m \|M_j - V_k\|^2 \quad (3)$$

where  $M$  is the set of  $K$  key pixels which have the smallest spatial distances from  $S_i$ , and  $M_j$  is the  $j$ th key pixel in  $M$ . And the spatial distance  $d_{ij}$  between two pixels  $x_i$  and  $x_j$  in the image is calculated as follows:

$$d_{ij} = \sqrt{(r_i - r_j)^2 + (c_i - c_j)^2} \quad (4)$$

where  $r_i$  and  $c_i$  are row and column numbers of pixel  $x_i$  in  $X$ , respectively,  $r_j$  and  $c_j$  are those of pixel  $x_j$ . Since the key pixels are uniformly distributed across the whole image, the  $K$  nearest neighbors are not incorporated into the local neighborhood of  $S_i$ . This means that nonlocal spatial information is used here. In (3),  $w_{ij}$  denotes the weight of neighbor  $M_j$  on  $S_i$ , which is composed of two parts: 1) the spatial distance weight and 2) the intensity distance weight:

$$w_{ij} = w_s \cdot w_g \quad (5)$$

where  $w_s$  represents spatial distance weight, and  $w_g$  denotes intensity distance weight:

$$w_s = \frac{1}{d_{ij}^2 + 1} \quad (6)$$

$$w_g = \exp\left[-\left|\log \frac{\mu_i}{\mu_j}\right|\right]. \quad (7)$$

In (6),  $d_{ij}$  is the spatial Euclidean distance between  $S_i$  and  $M_j$  in the image, and  $\mu_i$  and  $\mu_j$  denote the average gray value of pixels which fall into  $S_i$ 's neighborhood  $O$  and average gray value of pixels which fall into  $M_j$ 's neighborhood  $O$  in image  $X$ , respectively. The log-ratio operator is used to further suppress speckle noise.

Neighbors in  $M$  with a close spatial position as well as a similar neighborhood to  $S_i$  yield large value of weights. This avoids the noisy key pixel located near the center pixel from generating a large interference in the clustering process. Furthermore, a non-noisy key pixel away from the center pixel may generate a larger weight. Finally, this method benefits from the nonlocal information so that it owes the robustness to speckle noise.

In order to find the minimum of the objective function  $J$ , we update the fuzzy membership matrix  $U$  and the clustering center matrix  $V$  iteratively based on the following two formulae:

$$u_{ki} = \frac{1}{\sum_{j=1}^c \left( \frac{\|S_i - V_k\| + G_{ki}}{\|S_i - V_j\| + G_{ji}} \right)^{\frac{1}{m-1}}}, \quad (8)$$

$$V_k = \frac{\sum_{i=1}^N u_{ki}^m S_i}{\sum_{i=1}^N u_{ki}^m}. \quad (9)$$

The iteration will be stopped if the change of matrix  $U$  in two successive iterations is smaller than a threshold  $\varepsilon$  or the number of iterations is equal or greater than a predefined value  $T_{\max}$ .

TABLE I  
PROCESS OF FUZZY CLUSTERING

---

**Algorithm 1:** Process of the fuzzy clustering method based on nonlocal information.

---

**Input:** key pixels set  $S$ , cluster number  $c$ , threshold  $\varepsilon$ , max iterative number  $T_{\max}$ , the number of nearest neighbors  $K$ .

**Output:** key pixels' clustering result  $C_S$ .

**Begin**

1. Find  $K$  nearest neighbors for every key pixel in  $S$ ;
2. Set the iterative counter  $t$  equal to 0, initialize the membership matrix  $U$  randomly;
3. Update the clustering center matrix  $V$  using (9);
4. Update the membership matrix  $U$  in (8);
5. If  $\max|U(t) - U(t-1)| < \varepsilon$  or  $t > T_{\max}$ , execute step 6; otherwise, execute step 3 and  $t = t + 1$ ;
6. Execute the defuzzification process using (10) and label every key pixel;
7. Output the clustering result  $C_S$ .

**End**

---

Then, we perform a defuzzification process to label every key pixel based on the fuzzy membership matrix  $U$ . The label of key pixel  $S_i$  is denoted by

$$C_{S_i} = \arg \max_k u_{ki} \quad k = 1, 2, \dots, c. \quad (10)$$

Pseudo-code of the fuzzy clustering method based on nonlocal information is presented in Table I.

### C. Segmentation of Non-Key Pixels

So far, we have shown how a small number of pixels, namely, key pixels, can be clustered. However, the remaining (non-key) pixels in the image must also be labeled to complete the segmentation. We segment the non-key pixels using the clustering results  $C_S$  obtained in the previous step. Specifically, the label of the  $i$ th non-key pixel  $L_i$  in  $L$  is considered to be identical to the label of  $E_n$ , which is the most similar key pixel to  $L_i$  in  $L_i$ 's spatial neighborhood  $H$ :

$$E_n = \arg \max_{E_j \in E} \omega_{ij} \quad (11)$$

where  $E$  is the set of key pixels incorporated into  $L_i$ 's spatial neighborhood  $H$ , and  $E_j$  is the  $j$ th key pixel in  $E$ . Although  $\omega_{ij}$  is similar to  $w_{ij}$ ,  $\omega_{ij}$  is the similarity between  $L_i$  and  $E_j$ , but  $w_{ij}$  is the weight of the  $j$ th neighbor  $M_j$  on  $S_i$ . Similar to  $w_{ij}$ ,  $\omega_{ij}$  is also composed of two parts: 1) spatial similarity and 2) intensity similarity:

$$\omega_{ij} = \omega_s \cdot \omega_g \quad (12)$$

where

$$\omega_s = \frac{1}{d_{ij}^2 + 1} \quad (13)$$

$$\omega_g = \exp\left[-\left|\log \frac{\mu_i}{\mu_j}\right|\right]. \quad (14)$$

Here,  $d_{ij}$  denotes the spatial Euclidean distance between the non-key pixel  $L_i$  and key pixel  $E_j$ , whereas in (6),  $d_{ij}$  is the spatial Euclidean distance between key pixel  $S_i$  and neighbor  $M_j$ . Moreover,  $\mu_i$  denotes the average gray value of pixels in

TABLE II  
PROCESS OF FKP\_FCM

---



---

**Algorithm 2:** Process of FKP\_FCM.

---

**Input:** SAR image  $I$ , segmentation class number  $c$ , the size of selection window  $\mathcal{N}$ , the number of nearest neighbors  $K$ , the sizes of neighborhood  $O$  and  $H$ , max iterative number  $Tmax$ , threshold  $\varepsilon$ .

**Output:** final segmentation map  $R$

**Begin**

1. Filter the SAR image  $I$  by Gaussian filter to obtain the filtered image  $X$ ;
2. Use (1) to obtain the key pixels set  $S$  and non-key pixels set  $L$ ;
3. Cluster the key pixels by method in Table I and obtain the segmentation result of key pixels  $C_S$ ;
4. Segment the non-key pixels by (11) and (15) and obtain the segmentation result of non-key pixels  $C_L$ ;
5. Smooth  $C$  using (16);
6. Output the final segmentation map  $R$ .

**End**

---



---

$L'_i$ 's spatial neighborhood  $O$ , and  $\mu_j$  denotes the average gray value of pixels in  $E'_j$ 's spatial neighborhood  $O$ . In addition, when there is no key pixel falling into the neighborhood  $H$  of  $L_i$ , we use the clustering center  $V$  obtained in the previous step to determine the label of  $L_i$ . Thus, the label of non-key pixel  $L_i$  is computed as follows:

$$C_{L_i} = \arg \min_{1 \leq k \leq c} |\mu_i - V_k| \quad (15)$$

where  $V_k$  is the  $k$ th clustering center obtained by the fuzzy clustering. Combining the segmentation result  $C_S$  of the key pixels and the segmentation result  $C_L$  of the non-key pixels, we can get the segmentation result  $C$  of the whole image. In contrast to other clustering algorithms, the very large-amount non-key pixels are not included in the clustering process but are segmented using the segmentation result of the key pixels. Avoiding using time-consuming clustering to segment majority pixels, i.e., non-key pixels, makes our algorithm very fast. Moreover, the results can effectively suppress effects of speckle noise, but it loses some detailed information of the image. So, we use a simple filter to smooth the results of segmentation. The final label of pixel  $I_{ij}$  in image  $I$  is

$$R_{ij} = \arg \max_{1 \leq k \leq c} n_k, \quad 1 \leq i \leq A, 1 \leq j \leq B \quad (16)$$

where  $n_k$  is the number of pixels whose labels are  $k$  based on the segmentation result  $C$  in  $I'_{ij}$ 's neighborhood  $P$ , and the size of  $P$  is set to be  $3 \times 3$  to preserve more details of the image.

#### D. Steps of FKP\_FCM

The main idea of FKP\_FCM is to divide the image into two parts and handle them separately. Fuzzy clustering based on nonlocal information is used to segment key pixels, and the clustering results of the key-pixels and similarity measure  $\omega_{ij}$  are used to segment non-key pixels. Pseudo-code of the segmentation of FKP\_FCM is presented in Table II.

### III. PARAMETER ANALYSIS

Now, we need to define the sizes of all windows involved in FKP\_FCM and the number of nearest neighbors  $K$  mentioned

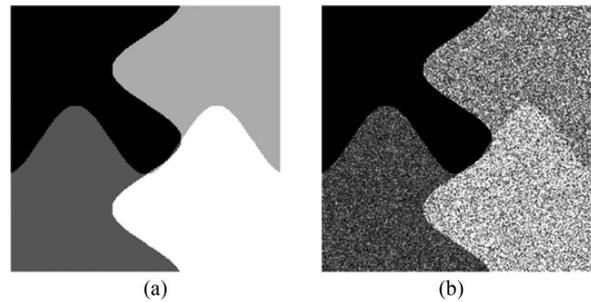


Fig. 3. Test image. (a) Ground truth. (b) Simulated SAR image.

in Section II. These sizes of the windows and the number  $K$  are important parameters because their values influence the segmentation result. These square windows are selection windows  $\mathcal{N}$  used for selecting the key pixels, the neighborhood window  $O$  used to calculate  $w_g$  and  $\omega_g$ , and the neighborhood window  $H$  used to segment non-key pixels. In this section, we will discuss how these parameters affect the segmentation result using a simulated SAR image shown in Fig. 3(b).

We artificially pollute Fig. 3(a) and generate a 1-look simulated SAR image shown in Fig. 3(b). We use Fig. 3(b) to test the performance of FKP\_FCM with different values of these parameters. In order to quantitatively evaluate the segmentation results, we utilize the segmentation accuracy (SA) proposed in [45] often used in image segmentation. SA represents the ratio of pixels correctly segmented as follows:

$$SA = \frac{\sum_{i=1}^c P_i \cap C_i}{\sum_{j=1}^c C_j} \quad (17)$$

where  $c$  is the number of classes,  $P_i$  is the set of pixels whose labels are determined to be  $i$  by the algorithm, and  $C_i$  is the set of pixels whose labels are  $i$  in the ground truth. SA belongs to the range  $[0, 1]$ . The larger the SA, the better the result.

#### A. Selection Window $\mathcal{N}$

We consider the local maximum to be the key pixel, as shown in (1). Thus, the selection window, denoted by  $\mathcal{N}$ , determines the generation of key pixels. Because the quality of key pixels directly determines the quality of final segmentation result, we set the size of  $\mathcal{N}$  to be 3, 5, 6, 9, 11, and 13 to test the effects of  $\mathcal{N}$  on the segmentation results of Fig. 3(b). We consider the sizes of  $O$ ,  $H$ , and  $K$  to be 5, 7, and 20, respectively, in all the experiments. Fig. 4 shows the effect of different  $\mathcal{N}$  on the segmentation results.

Fig. 4(a) shows that the increased size of  $\mathcal{N}$  results in the decreased number of key pixels. If the size of  $\mathcal{N}$  is  $3 \times 3$ , the number of key pixels is 2648 which is the largest but only account for 4.45% of total pixels in the image. Consequently, even in an SAR image with large size, the number of key pixels is also very small. In addition, Fig. 4(b) shows that the increased size, i.e., the decreased number of key pixels, results in decreased SA of segmentation, where the maximum value of SA is obtained when the size of  $\mathcal{N}$  is  $3 \times 3$ . The logical reason confirms that the increased size causes increasingly sparser distribution of

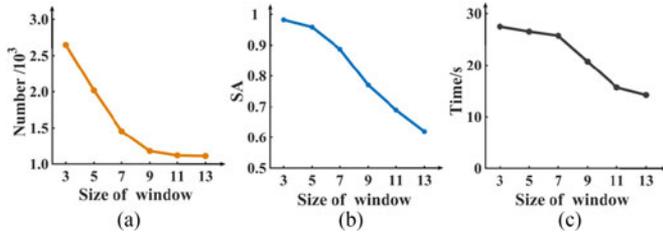


Fig. 4. Selection window  $\mathcal{N}$  affects the segmentation result: (a) effect on the number of key pixels; (b) effect on SA; and (c) effect on running time.

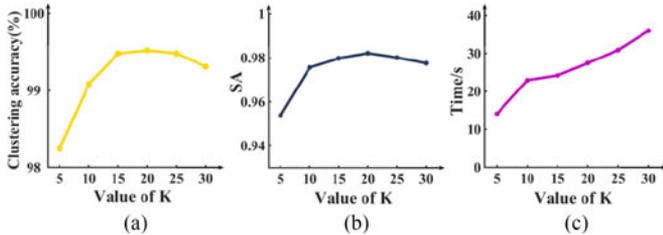


Fig. 5. Value of  $K$  affects the segmentation results of the key pixels and the whole image: (a) the clustering accuracy of the key pixels versus the value of  $K$ ; (b) SA versus the value of  $K$ ; and (c) running time of the algorithm versus the value of  $K$ .

the key pixels across the entire image. The fewer the number of the key pixels, the less information they convey, so the selection of key pixels highly influences the final result of the segmentation.

Fig. 4(c) shows that the running time decreases significantly by increasing the size of  $\mathcal{N}$ . A larger  $\mathcal{N}$  can generate a smaller number of key pixels, which results in a shorter clustering time and consequently a shortened time of computation will be needed to achieve the total segmentation. If the number of key pixels is the largest, i.e., the size of  $\mathcal{N}$  is  $3 \times 3$ , the time used by FKP\_FCM to segment all pixels in the image is not more than 30 s. This shows the superiority of FKP\_FCM in terms of computation time required for SAR image segmentation. If the size of the image to be segmented is large, the number of selected key pixels increases. Nonetheless, in comparison with the existing methods processing all the pixels across the image iteratively, FKP\_FCM can achieve a much faster segmentation result.

### B. Number of Nearest Neighbors $K$

$K$  nearest neighbors make the clustering of key pixels very robust to speckle noise. As  $K$  determines the clustering result of the key pixels, we consider  $K$  to be 5, 10, 15, 20, 25, and 30 to test the effect of value of  $K$  on the clustering accuracy of the key pixels. The clustering accuracy represents the proportion of correctly clustered key pixels to all key pixels. The result of the clustering plays an important role in the final segmentation of the image. Hence, the effect of the value of  $K$  on the whole segmentation result is also tested. We consider the size of  $\mathcal{N}$ ,  $O$ , and  $H$  to be 3, 5, and 7, respectively. Fig. 5 shows the effects of the value of  $K$  on the segmentation results of key pixels and image.

Fig. 5(a) shows that the clustering uses only a small amount of nonlocal information if  $K$  is very small. This results in a lower accuracy of the clustering. That is, the clustering accuracy improves significantly if  $K$  increases. Nonetheless, the clustering accuracy does not change a lot when  $K$  is larger than 15. The reason is that the calculation of weight  $w$  in the clustering process consists of two items: 1) the spatial distance and 2) the intensity distance. The larger the  $K$  value, the smaller the impact of some key pixels that are far from the central key pixel. Hence, the effect of these far pixels can be ignored. Besides, the clustering process is robust to speckle noise by using sufficient nonlocal information when  $K$  is a little large.

Fig. 5(b) shows the change of SA versus different  $K$  has a trend similar to the one of the clustering accuracy versus different  $K$ . This illustrates that the segmentation result of the key pixels directly determines the segmentation of the whole image, so to achieve a good segmentation result, the clustering of the key pixels must have high accuracy. Fig. 5(c) shows that the running time rapidly increases for increased  $K$ , but the SA does not increase significantly for increased  $K$  when it is larger than 15. And the algorithm reaches the highest SA at  $K = 20$ .

### C. Neighborhood Window $H$

The clustering results of the key pixels within the neighborhood window  $H$  are used to segment the non-key pixels. That is to say, the segmentation result of non-key pixels is determined by the size of  $H$ , the clustering results, and distribution of key pixels. Since non-key pixels form the majority pixels of the image, a promising segmentation of image can be obtained if the non-key pixels are segmented accurately and vice versa. The size of  $H$  is a crucial parameter which determines the segmentation of non-key pixels. If the window size is small, neighborhood window  $H$  of most non-key pixels may not contain key pixels. Hence, the segmentation of these pixels will depend on the cluster  $V$ , as (15), which will make the segmentation highly suffer noise. If the window size is larger, for many non-key pixels, there will be some key pixels involved in their neighborhood window  $H$ , then (11) will be used to segment these non-key pixels. With the accurate clustering results of key pixels, many non-key pixels can be correctly segmented. But the segmentation of non-key pixels may lose some details with larger window size and tend to be stable when the window size continues to increase. The reason is that some key pixels whose locations are far from the center pixel in  $H$  have little effect on the segmentation.

We consider the size of  $H$  to be 3, 5, 7, 9, 11, 13, 15, 17, and 19 to test the effects of  $H$  on the segmentation of the tested image. The sizes of  $\mathcal{N}$  and  $O$  are 3 and 5, respectively, and  $K$  is 20. Fig. 6 shows that  $H$  influences the segmentation result of Fig. 3(b).

The results presented in this figure confirm our analysis. In specific, it shows that SA is very low with a value of 0.78 with the size of window  $H$  equal to 3. But as the size of window  $H$  increases, the SA rapidly increases and reaches a maximum at the size of window  $H$  equal to 7. When the size exceeds 7, SA tends to be constant. Fig. 6(b) shows the segmentation time increases significantly by increasing the size of  $H$ . Larger  $H$

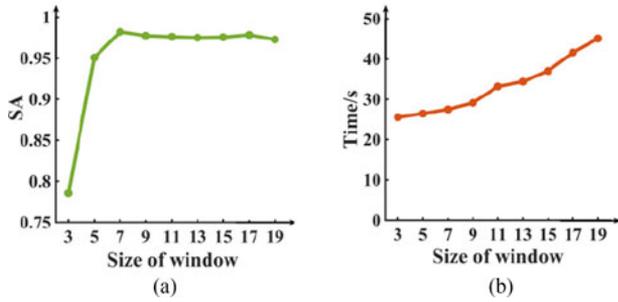


Fig. 6. Effect of neighborhood window  $H$  on the segmentation: (a) the size of  $H$  versus SA; and (b) the size of  $H$  versus the running time.

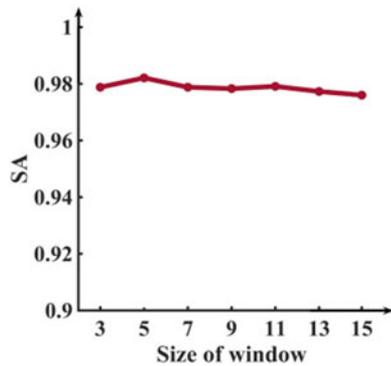


Fig. 7. Effect of neighborhood window  $O$  on SA.

contains more key pixels, so the algorithm needs to take more time to calculate similarity between the key pixels involved in  $H$  and central non-key pixel.

The size of the window  $H$  also determines the ability of our approach to suppress speckle noise. If the image to be segmented contains small-number noise, we can use a small size such as 3 or 5 to retain more important details. If the image is highly polluted with noise, noise' suppression is more important. Hence, we consider the size of window  $H$  for this case to be large, e.g., 7 or 9. In this experiment, the tested image is a 1-look simulated SAR image with serious noise. So, the best segmentation is achieved when size of  $H$  is 7. This also guarantees a balance between the suppression of noise and the preservation of details of the image.

#### D. Neighborhood Window $O$

Window  $O$  is used both in calculating  $w_g$  when segmenting key pixels and  $\omega_g$  when segmenting non-key pixels, as shown in (7) and (14). In FKP\_FCM, the sizes of window  $O$  in the calculations of  $w_g$  and  $\omega_g$  are the same. We set the size of window  $O$  to be 3, 5, 7, 9, 11, 13, and 15 for testing the effect of  $O$  on the segmentation of Fig. 3(b), and the result is shown in Fig. 7. The size of  $\mathcal{N}$  and  $H$  are set to be 3 and 7, where  $K$  is 20. Fig. 7 shows SA is almost unchanged with increasing the size of  $O$  from 3 to 15. This illustrates that the final segmentation is not sensitive to  $O$ . The weight  $w$  and similarity  $\omega$  used in this paper are both composed of two parts: 1) the spatial distance and

2) the intensity distance. The algorithm only uses the window  $O$  and gray values of pixels located in  $O$  for calculating the intensity distance. The log-ratio operator is also used to restrain the speckle noise as shown in (7) and (14), which further weakens the influence of the size of window  $O$  on the final result. Therefore, the final segmentation is stable with different size of window  $O$ .

#### IV. EXPERIMENTAL PERFORMANCE EVALUATION

To show the effectiveness of FKP\_FCM, we compare the results obtained by FKP\_FCM against four clustering-based algorithms which are CKS\_FCM [37], ILKFCM [34], NS\_FCM [36], and ALFCM [38]. In addition, we compare the results obtained by two powerful level-set-based algorithms including the multiregion level-set partitioning algorithm (MLSP) [46] and the two-phase algorithm based on kurtosis curvelet energy and unsupervised spectral regression (KCUSR) [47] with the results obtained by our proposed algorithm to show the efficiency of FKP\_FCM. Because FKP\_FCM is based on clustering, algorithms which combine fuzzy clustering and active contour methods are also compared. Following the idea in [48], we first apply SFCM [49] to segment the SAR image and initialize the contours in MLSP based on the result, and MLSP is then executed to realize the final segmentation. In this paper, this comparison algorithm is referred to as SFCM + MLSP. Further information about the parameters used in comparison algorithms refers to the original papers. In the experiments, we segment 12 simulated noisy SAR images generated from three simulated images as well as four real SAR images. Then, the segmentation results and running times of all algorithms are compared. The results of two level-set based algorithms and SFCM + MLSP shown in this section are the best results achieved over ten runs.

To generate simulated SAR images, we deliberately contaminate original simulated images with various different levels of multiplicative speckle noise [50] to generate 1-look, 2-look, 4-look, and 6-look simulated SAR images of SI1, SI2, and SI3. Fig. 8 shows three simulated images with and without simulated multiplicative speckle noise.

The three simulated images SI1, SI2, and SI3 can be used as ground truth maps to evaluate the segmentation results of noisy images generated from them. The detailed information of these three simulated images is shown in Table III.

In addition, we also apply FKP\_FCM and other comparison algorithms on four real SAR images, as shown in Fig. 9. Fig. 9(a) shows a  $256 \times 256$  SAR image called FARMLAND. This 4-look image was captured by the European remote sensing satellite viewing a farm in Italy with VV polarization and a resolution of 12.5 m. This image can be segmented into three different types of farmlands. The second real SAR image is called XIAN, which was captured by TerraSAR in X-band, viewing the region of Xi'an, China, with a resolution of 1 m, as shown in Fig. 9(b). XIAN image is 8-look and has a size of  $256 \times 256$ , and it can be segmented into three different types of farmlands and water regions. Fig. 9(c) shows the third real SAR image, called MARICOPA, which was imaged by an airborne SAR in Ku-band over the Maricopa Agricultural Center near Arizona.

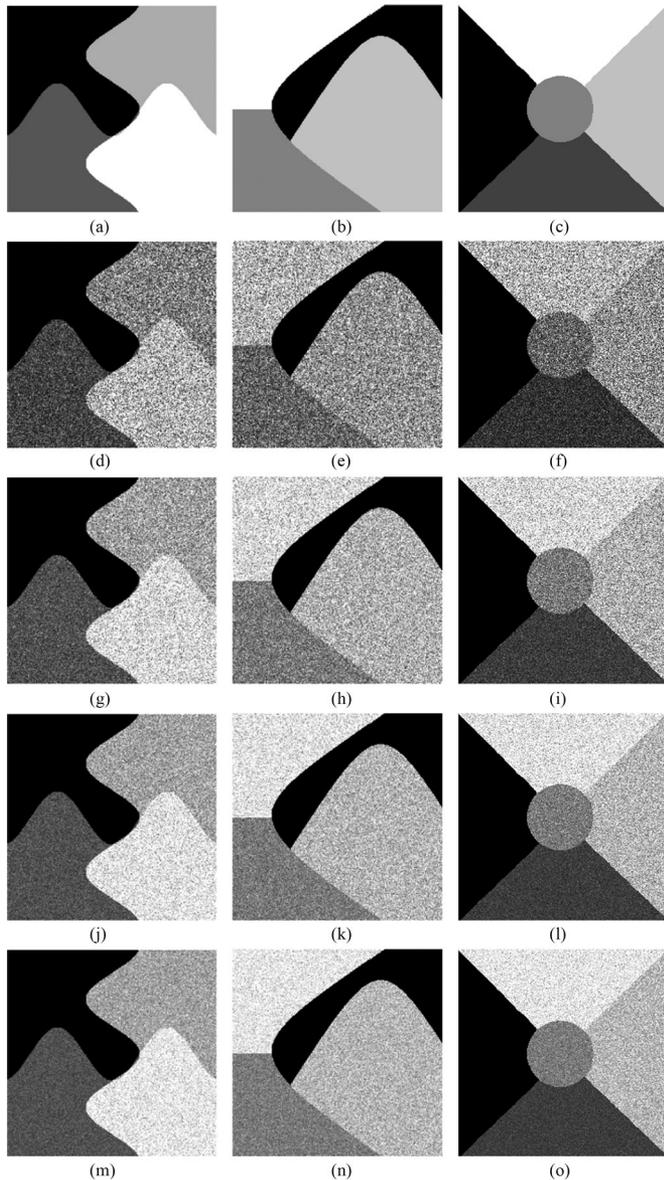


Fig. 8. Simulated images and simulated SAR images: (a) S11; (b) S12; (c) S13; (d), (g), (j), and (m) 1-, 2-, 4-, and 6-look simulated SAR images generated from S11; (e), (h), (k), and (n) 1-, 2-, 4-, and 6-look simulated SAR images generated from S12; and (f), (i), (l), and (o) 1-, 2-, 4-, and 6-look simulated SAR images generated from S13.

TABLE III  
DETAILED INFORMATION OF SIMULATED IMAGES

Image	Size	Class number	Gray value in each region
S11	$244 \times 244$	4	0, 85, 170, and 255
S12	$256 \times 256$	4	0, 128, 192, and 255
S13	$283 \times 283$	5	0, 64, 128, 192, and 255

Its size is  $350 \times 350$  and its resolution is 1 m. This image can be segmented into four regions with three different types of farmlands and water. Finally, Fig. 9(d) shows the real SAR image called TRAUNSTEIN, which was captured by F-SAR in X-band over Traunstein in Bavaria, Germany. The size of this image is

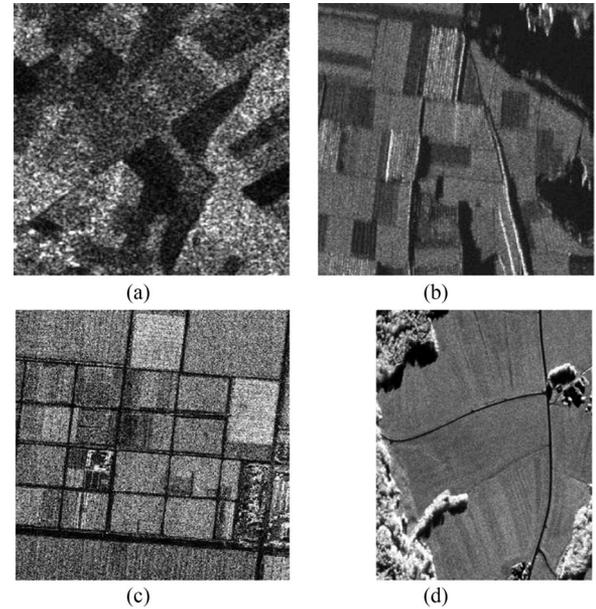


Fig. 9. Real SAR images: (a) FARMLAND image; (b) XIAN image; (c) MARICOPA image; and (d) TRAUNSTEIN image.

$1001 \times 779$  with HH polarization and a resolution of 1 m. Three classes can be obtained after segmenting TRAUNSTEIN, which are trees, farmlands, and shadows together with water.

When segmenting the images mentioned above, the parameters of FKP\_FCM were set as follows: the size of selection window  $\mathcal{N}$  is  $3 \times 3$ , the number of nearest neighbors  $K$  is 20, and the size of neighborhood window  $O$  is  $5 \times 5$  for all images. Furthermore, for 1-look and 2-look simulated SAR images of S11, all simulated SAR images of S12 and S13, MARICOPA image, and TRAUNSTEIN image, the size of  $H$  is  $7 \times 7$ ; for 4-look and 6-look simulated SAR images of S11 and XIAN image, the size of  $H$  is  $5 \times 5$ ; and for FARMLAND image, the size of  $H$  is  $11 \times 11$ . Our experiments were executed in MALAB R2014b environment using a computer with an Intel core i5 2.60 GHz CPU and 8 GB RAM. Our experimental results confirm that our algorithm generates good segmentation results by using these choices of parameters.

#### A. Results on Simulated SAR Images

1) *Results on Simulated SAR Images Generated From S11:* We segment four SAR images of S11 shown in Fig. 8 by comparison algorithms and FKP\_FCM, where the simulated image shown in Fig. 8(a) is used as the ground truth.

SAs and computation times of comparison and our proposed algorithms are presented in Table IV. These results show that FKP\_FCM outperforms other algorithms in terms of computation time and SA for 1- and 2-look simulated SAR images. In addition, FKP\_FCM successfully segments all the images with the minimum computation time. On the other hand, the SAs yielded by KCUSR [47] are just slightly better than the ones obtained by FKP\_FCM only for 4- and 6-look images. In summary, FKP\_FCM is more efficient than other clustering algorithms because 1) it clusters only key pixels

TABLE IV  
RESULTS ON SIMULATED SAR IMAGES OF S11

Algorithm	1 look		2 looks		4 looks		6 looks	
	SA%	<i>t/s</i>	SA%	<i>t/s</i>	SA%	<i>t/s</i>	SA%	<i>t/s</i>
ILKFCM [34]	97.36	182	97.59	175	97.68	162	97.75	179
NS_FCM [36]	96.05	105	96.94	89	97.23	85	97.32	82
CKS_FCM [37]	91.74	389	95.96	223	98.00	167	98.69	143
ALFCM [38]	73.82	102	84.85	189	95.45	166	96.24	59
MLSP [46]	97.58	88	97.20	86	98.57	78	98.79	78
KCUSR [47]	97.75	98	98.25	96	<b>99.00</b>	93	<b>99.49</b>	90
SFCM + MLSP	97.90	114	97.53	107	98.69	101	99.09	105
FKP_FCM	<b>98.21</b>	<b>29</b>	<b>98.60</b>	<b>26</b>	98.80	<b>23</b>	99.22	<b>22</b>

(this substantially reduces the total clustering time) and 2) it segments non-key pixels based on the key pixels (this results in an increased robustness to noise and a fast segmentation). Next, ILKFCM uses features in the wavelet domain and the kernel function to reduce the influence of speckle noise, yielding SAs around 97% for all simulated SAR images. Nonetheless, ILKFCM does not perform very well on images with less noise, such as 4- or 6-look images and it results in large computation times.

A nonlocal mean filtering is required by NS-FCM to generate a filtered image in advance. Nevertheless, because the objective function of NS\_FCM does not include a neighborhood item similar to other clustering methods, the computation time is greatly reduced. This algorithm still uses all the pixels for clustering resulting in a running time longer than the one needed by FKP\_FCM. Moreover, NS-FCM falls behind FKP\_FCM in terms of SA. On the other hand, CKS\_FCM obtains high values of SA for images with high looks, whereas the SAs obtained by this algorithm are small for images that are heavily polluted by noise. CKS\_FCM uses both a nonlocal mean filtering process and an immune clone algorithm to determine initial cluster centers. Hence, its corresponding computation time for segmentation is longer than others.

Table IV shows that ALFCM yields the lowest SAs for the images with severe noise, but its running times are similar to NS\_FCM because the regional information is calculated in advance, i.e., it is not iterative computation. To increase the accuracy of the segmentation, ALFCM utilizes regional/local spatial information; nevertheless, the performance of mean shift algorithm used in this algorithm is highly affected by the noise. This makes the regional information inaccurate such that an ultimate segmentation outcome suffers significantly from noise. From Table IV, we can conclude that two level-set based algorithms yield reasonably good results for all tested images; in addition, KCUSR has highest SAs for 4- and 6-look images. Although the level-set-based algorithms can accurately depict contours of regions, their ability to precisely recognize contours decreases with increased noise. The results on images with high noise level show that FKP\_FCM outperforms MLSP and KCUSR. Furthermore, MLSP and KCUSR need longer time to compute than the one needed by FKP\_FCM because the contours need to evolve iteratively. Besides, they have longer computation time for the segmentation of images with severe noise. KCUSR

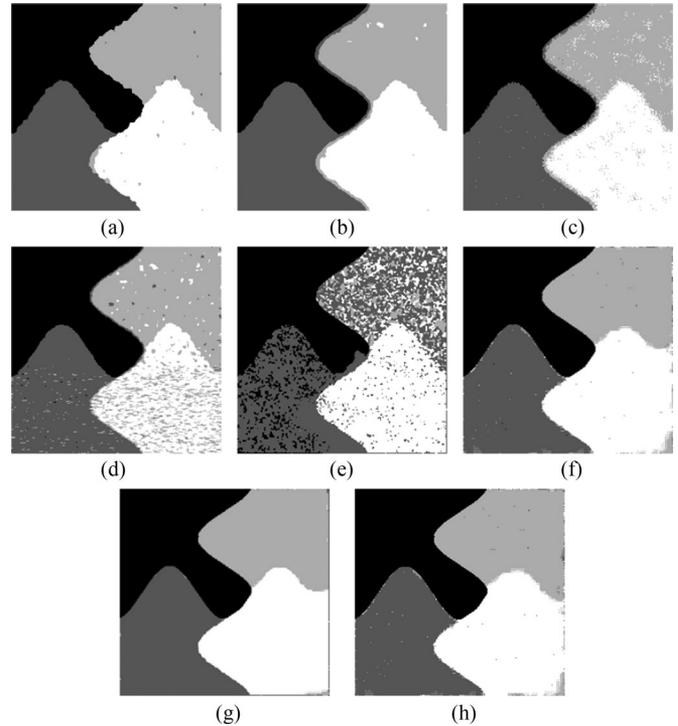


Fig. 10. Segmentation maps on 1-look simulated SAR image of S11: (a) FKP\_FCM; (b) ILKFCM [34]; (c) NS\_FCM [36]; (d) CKS\_FCM [37]; (e) ALFCM [38]; (f) MLSP [46]; (g) KCUSR [47]; and (h) SFCM + MLSP.

uses more time because it applies both the Gabor filter bank and spectral regression to obtain more useful features. Table IV also shows that SFCM + MLSP has better performances than MLSP on all tested images because the initialization based on the result of SFCM prevents, to some extent, contours' evolution of MLSP from getting stuck into local optimal.

Next, to further highlight the strengths and weaknesses of the algorithms, eight segmentation maps of 1-look simulated SAR image are used and shown in Fig. 10. 1-look simulated SAR image has the highest level of noise, so the segmentation results of eight algorithms on this image vary greatly. Fig. 10(a)–(e) show that the result of FKP\_FCM generates rough edges, but the other four clustering-based algorithms, which are used for comparing, generate smooth edges. Because our algorithm segments non-key pixels based on the clustering results of key pixels, details of the images, especially edges, are not well preserved by the algorithm; however, the segmentation map of FKP\_FCM has the fewest points wrongly segmented, as shown in Fig. 10(a). The segmentation result obtained by ILKFCM has smooth edges and a few limited areas wrongly segmented [see Fig. 10(b)], but its performance is worse than FKP\_FCM in terms of SA because the use of wavelet makes some pixels in the edge regions wrongly segmented. In addition, Fig. 10(c) shows that the map of NS\_FCM has poor region uniformity and many discrete parts are wrongly segmented by NSFCM. On the other hand, the map in Fig. 10(d) shows that CKS\_FCM suffers much from speckle noise. Fig. 10(e) shows that ALFCM yields the worst segmentation because the simple mean-shift algorithm cannot obtain the accurate region information from images with serious

TABLE V  
RESULTS ON SIMULATED SAR IMAGES OF SI2

Algorithm	1 look		2 looks		4 looks		6 looks	
	SA%	t/s	SA%	t/s	SA%	t/s	SA%	t/s
ILKFCM [34]	94.24	286	96.42	260	96.60	216	96.55	255
NS_FCM [36]	84.25	240	90.28	125	95.57	100	96.53	84
CKS_FCM [37]	73.79	614	84.65	419	92.59	279	95.80	221
ALFCM [38]	69.24	232	81.34	123	92.18	124	95.74	81
MLSP [46]	96.89	91	97.30	86	98.27	85	98.42	87
KCUSR [47]	96.60	102	98.12	99	98.26	95	98.66	95
SFCM + MLSP	96.29	132	97.58	131	98.03	128	<b>99.08</b>	122
FKP_FCM	<b>97.05</b>	<b>42</b>	<b>98.48</b>	<b>34</b>	<b>98.65</b>	<b>29</b>	98.69	28

noise. Fig. 10(f) and (g) shows that the maps obtained by MLSP and KCUSR are less affected by the noise (these algorithms are robust to noise more than clustering-based ones) and edges of the obtained clustering are reasonably smooth and clear. Actually, they design to detect the contours of regions by building a statistic model of the image, but they cannot yield accurate contours on the images polluted with high level of noise. Fig. 10(h) shows that the result of SFCM + MLSP is similar to (but slightly better than) the one yielded by MLSP because of incorporating the fuzzy clustering method.

2) *Results on Simulated SAR Images Generated From SI2:* In addition to the previous experiments, we also use the eight algorithms to segment four simulated SAR images of SI2. We consider the number of classes to be 4 and use the image in Fig. 8(b) as the ground truth. The SAs and computation times obtained by the algorithms are presented in Table V.

The true clustering centers of four simulated SAR images from SI2 are close and its size is larger than SI1, as shown in Table III. Hence, SAs and running times of all algorithms on these images are a bit worse than the ones obtained on images generated from SI1, as evidenced by Table V. FKP\_FCM results in the highest SA and the shortest times on 1-, 2-, and 4-look images. Table V shows that SFCM + MLSP has the highest SA on 6-look image but longer running time, which fully proves that initial contours are crucial in level-set-based algorithms. The results in Table V show that ILKFCM segments images well but it needs much longer computation time for clustering all pixels in the segmentation process. Results obtained by CKS\_FCM on four images have lower SAs and very long running times because of the use of the immune clone algorithm. NS\_FCM needs shorter computation times for segmenting those images, but its running times and SAs are still much worse than the ones of FKP\_FCM. ALFCM results in good SAs on 4-look and 6-look simulated SAR images while it results in the lowest SAs on other images. ALFCM uses FCM to initialize the membership matrix, which accelerates the convergence of the algorithm and results in shorter running times. MLSP and KCUSR yield promising results which are even better than the ones of four comparison clustering-based algorithms in terms of both SA and the running time. However, the results are not as good as the ones obtained by FKP\_FCM. Fig. 11 shows eight segmentation maps on 1-look simulated SAR image to more clearly illustrate/compare their performances.

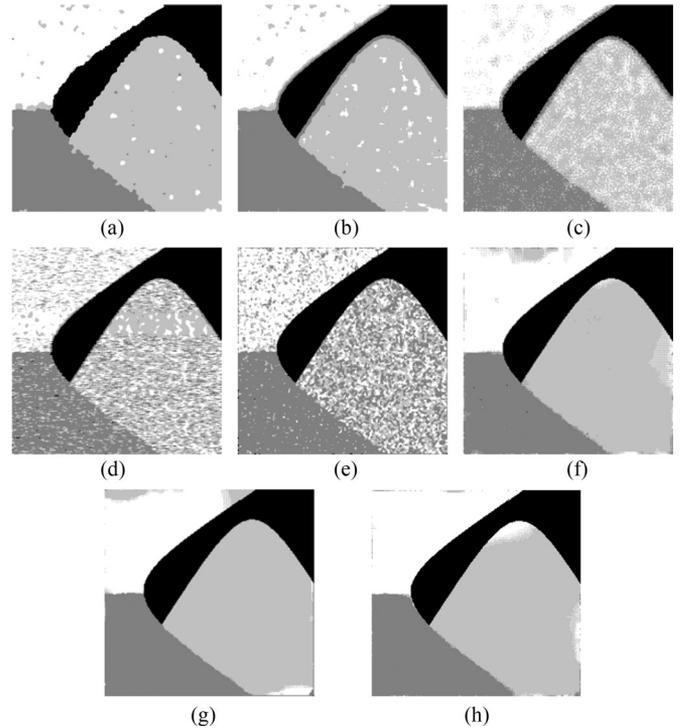


Fig. 11. Segmentation maps of 1-look simulated SAR image of SI2: (a) FKP\_FCM; (b) ILKFCM [34]; (c) NS\_FCM [36]; (d) CKS\_FCM [37]; (e) ALFCM [38]; (f) MLSP [46]; (g) KCUSR [47]; and (h) SFCM + MLSP.

The segmentation results of NS\_FCM and CKS\_FCM have many points which are wrongly segmented [see Fig. 11(c) and (d)]. The segmentation map of ILKFCM is a little worse than the one obtained by FKP\_FCM [see Fig. 11(a) and (b)]. Furthermore, ILKFCM wrongly segments many pixels in the areas close to edges. On the other hand, ALFCM achieves the worst segmentation result because the mean shift used in this algorithm is not robust to noise [see Fig. 11(e)]. Although the maps of MLSP and KCUSR have better regional uniformity [see Fig. 11(f) and (g)], the obtained maps have inaccurate contours of regions. Despite the help of SFCM [see Fig. 11(h)], SFCM + MLSP still results in inaccurate contours because of severe noise. FKP\_FCM achieves a better segmentation map with fewer wrongly segmented points than the ones obtained by clustering-based algorithms, whereas it detects more accurate contours than level-set-based algorithms.

3) *Results on Simulated SAR Images Generated From SI3:* Simulated SAR images of SI3 are segmented into five regions (see Table III), and each region has close gray value to others. Hence, segmenting these simulated SAR images is very challenging. We use all the eight algorithms to segment these images, and Table VI represents the corresponding SAs and computation times.

Table VI shows that FKP\_FCM yields the best results on all simulated SAR images in terms of SAs and computation times. The results demonstrate FKP\_FCM's superiority over other algorithms (especially on highly polluted images) because it utilizes local and nonlocal spatial information for segmentation.

TABLE VI  
SIMULATED SAR IMAGES OF SI3

Algorithm	1 look		2 looks		4 looks		6 looks	
	SA%	<i>t</i> /s						
ILKFCM [34]	95.77	493	97.18	264	97.31	276	97.27	245
NS_FCM [36]	89.90	77	95.41	65	96.57	62	96.99	67
CKS_FCM [37]	71.26	764	77.62	432	81.63	413	87.24	387
ALFCM [38]	79.41	182	90.33	179	93.84	127	96.35	79
MLSP [46]	92.07	158	94.78	159	94.86	157	95.76	161
KCUSR [47]	95.55	174	95.13	171	95.72	166	95.96	164
SFCM + MLSP	92.51	227	94.45	209	95.54	205	94.63	197
FKP_FCM	<b>97.50</b>	<b>49</b>	<b>98.38</b>	<b>46</b>	<b>98.27</b>	<b>42</b>	<b>98.58</b>	<b>49</b>

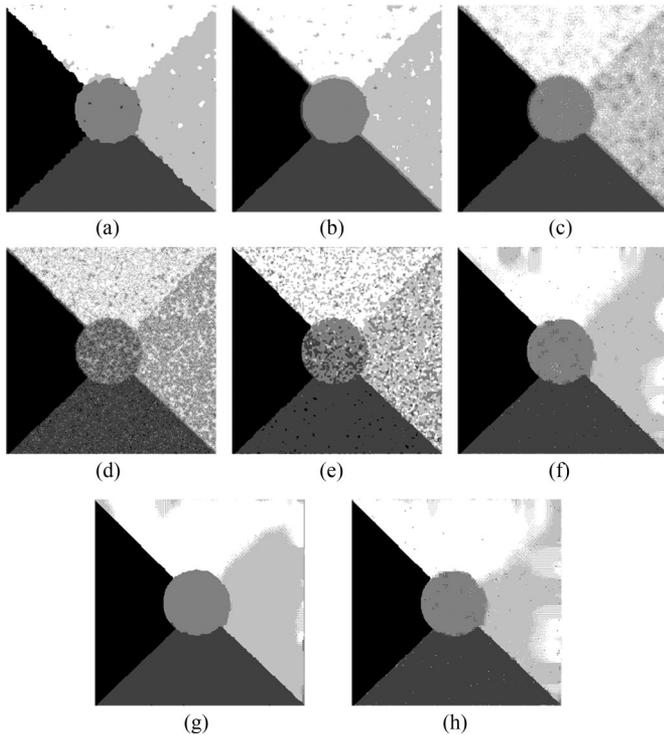


Fig. 12. Segmentation maps of 1-look simulated SAR image of SI3: (a) FKP\_FCM; (b) ILKFCM [34]; (c) NS\_FCM [36]; (d) CKS\_FCM [37]; (e) ALFCM [38]; (f) MLSP [46]; (g) KCUSR [47]; and (h) SFCM + MLSP.

ILKFCM yields good results; nonetheless, it completes the segmentation in times longer than the ones needed by FKP\_FCM. The main factor causing these long computation times is the use of kernel method and wavelet domain information in ILKFCM to suppress the effects of speckle noise. NSFCM has promising segmentation results on all images except 1-look image. CKS\_FCM yields the worst results. Moreover, ALFCM obtains very poor segmentation of images with serious noise because the process of generating superpixels is sensitive to noise. Two level-set algorithms, namely, MLSP and KCUSR, obtained similar results; but KCUSR uses more information of pixels to achieve better results. On the other hand, SFCM + MLSP yields SAs similar to MLSP, but consumes longer times. Fig. 12 shows the segmentation maps on 1-look simulated SAR image obtained by all eight algorithms illustrating the performance.

The maps achieved by clustering-based algorithms show that these algorithms suffer much from noise (see Fig. 12); on the other hand, maps achieved by level-set algorithms have inaccurate contours. This demonstrates these algorithms, i.e., clustering-based and level-set algorithms, are only capable of partially addressing issues of SAR image segmentation. In contrast, the map achieved by FKP\_FCM has fewer points wrongly segmented than maps achieved by other clustering-based algorithms [see Fig. 12(b)–(e)], and it has more accurate contours than maps achieved by level-set algorithms [see Fig. 12(f)–(h)].

The results presented above demonstrate that FKP\_FCM outperforms other seven algorithms in terms of SA and computation time. For instance, FKP\_FCM has better segmentation of SAR images with severe multiplicative noise because: 1) in the key pixels' clustering process, both local and nonlocal spatial information are used to reduce the influence of multiplicative noise; and 2) the multiplicative noise is further suppressed using the reasonably precise clustering results of key pixels and a robust similarity metric for segmenting the remaining non-key pixels. Nevertheless, FKP\_FCM and other algorithms yield similar segmentation results for SAR images with only a low level of multiplicative noise, namely, 4-look and 6-look images.

## B. Results on Real SAR Images

1) *Results on FARMLAND Image:* In this section, we use eight algorithms to segment FARMLAND image where the number of clusters is considered to be 3. The final segmentation map is demonstrated by three colors, namely, blue, red, and yellow; every color indicates one type of farmland.

Fig. 13(a) shows the ground truth map of the FARMLAND; the final segmentation maps obtained by the algorithms are also shown in Fig. 13(b)–(i). These maps show that FKP\_FCM achieves the best segmentation result on the FARMLAND image. The segmentation map of NS\_FCM shows that many pixels are wrongly segmented by NSFCM [see Fig. 13(d)]; the result of CKSFCM is even worse [see Fig. 13(e)]. In contrast, the noisy limited areas of the map obtained by ILKFCM [see Fig. 13(c)] are less than the map obtained by CKS\_FCM, NS\_FCM, and ALFCM [see Fig. 13(d)–(f)]. The map of ALFCM wrongly segments many pixels in red and blue regions and contains a high level of noise [see Fig. 13(f)].

Although the map obtained by MLSP has smooth edges [see Fig. 13(g)], some contours shown in the ground truth map [see Fig. 13(a)] are undetected. And it is incapable of successfully detecting, especially, the red and blue regions. The map obtained by KCUSR has clear contours as well as promising regional uniformity [see Fig. 13(h)]. In addition, it detects the blue regions better than MLSP because KCUSR executes an accurate clustering after determining the boundaries [47]. However, KCUSR wrongly assigns yellow segmentation label to many pixels belonging to the red regions because of noise. The map obtained by SFCM + MLSP is much better than the one of MLSP [see Fig. 13(i)] for more accurate contours. Initializing the contour using results of SFCM improves segmentation ability of MLSP, but some regions are still wrongly segmented [see Fig. 13(i)]. Using only Fig. 13 as a reference is very hard

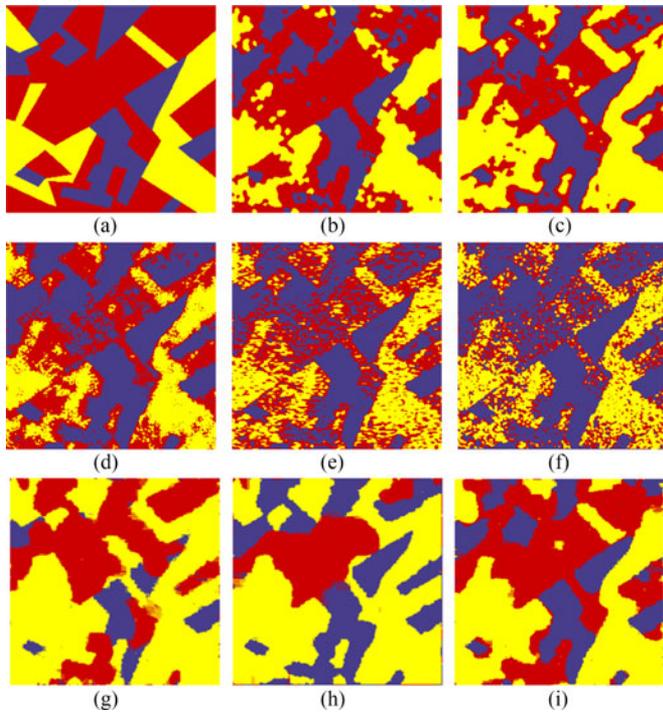


Fig. 13. Segmentation maps of FARMLAND image: (a) Ground truth map; (b) FKP\_FCM; (c) ILKFCM [34]; (d) NS\_FCM [36]; (e) CKS\_FCM [37]; (f) ALFCM [38]; (g) MLSP [46]; (h) KCUSR [47]; and (i) SFCM + MLSP.

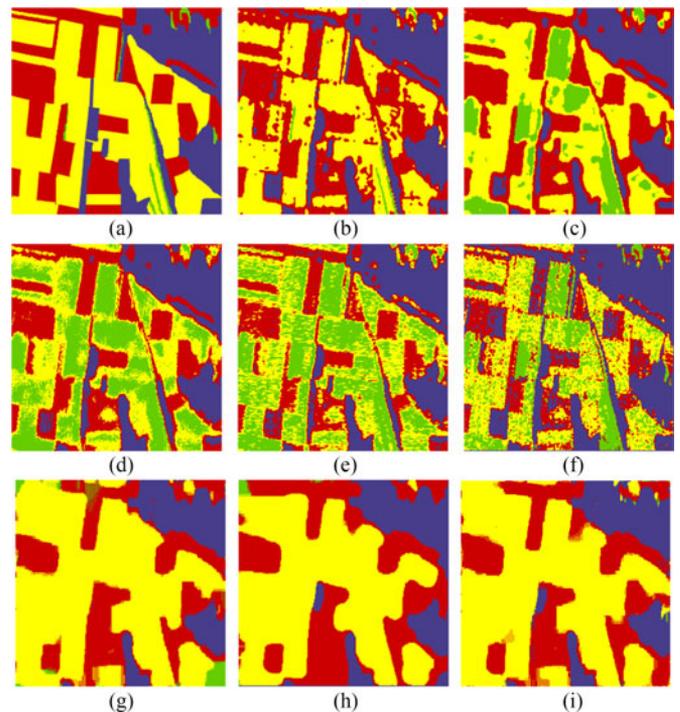


Fig. 14. Segmentation maps of XIAN image: (a) Ground truth map; (b) FKP\_FCM; (c) ILKFCM [34]; (d) NS\_FCM [36]; (e) CKS\_FCM [37]; (f) ALFCM [38]; (g) MLSP [46]; (h) KCUSR [47]; and (i) SFCM + MLSP.

TABLE VII  
SEGMENTATION ACCURACY ON REAL SAR IMAGES

Algorithm	FARMLAND	XIAN	MARICOPA	TRAUNSTEIN
ILKFCM [34]	0.7049	0.6891	0.3751	0.6003
NS_FCM [36]	0.6851	0.5641	0.5305	0.8782
CKS_FCM [37]	0.6485	0.5229	0.4637	0.4625
ALFCM [38]	0.5648	0.5845	0.4062	0.7317
MLSP [46]	0.5824	0.7151	0.5090	0.8956
KCUSR [47]	0.6290	0.7250	0.5204	0.8943
SFCM + MLSP	0.7536	0.7245	0.5416	<b>0.8984</b>
FKP_FCM	<b>0.7973</b>	<b>0.7595</b>	<b>0.6034</b>	0.8973

to identify the best segmentation map. Therefore, we will later represent the corresponding SAs for each map in Table VII.

In summary, FKP\_FCM has stronger robustness to speckle noise and it accurately segments the interwoven limited areas [see Fig. 13(b)], namely, red and blue regions. Nevertheless, the map of FKP\_FCM has rough edges because we segment non-key pixels based on its neighborhood  $H$  rather than clustering.

2) *Results on XIAN Image:* Next, we use the XIAN image having four regions shown in the ground truth map in Fig. 14(a) with four colors: red, yellow, green, and blue. The blue color indicates the water area, and the other three colors represent three different types of farmlands. The segmentation maps of XIAN image obtained by different algorithms are shown in Fig. 14.

Fig. 14(b) shows that some wrongly segmented points exist in the segmentation map obtained FKP\_FCM. However, FKP\_FCM yields segmentation of the most regions of the XIAN

image much more accurately than other algorithms. The white slender limited area in the lower region of the original SAR image [see Fig. 9(b)], which is supposed to be marked green, and its surrounding farmland, which is supposed to be marked yellow, should be segmented into two classes. However, only FKP\_FCM successfully segments them and others do not segment them correctly. As another example, in comparison, clustering-based algorithms almost wrongly assign green label to the regions that are supposed to be segmented with yellow color. These evidences indicate that these algorithms, which are used for validation of our algorithm, have poor ability to distinguish these two regions. ILKFCM yields a map with better regional uniformity than FKP\_FCM [see Fig. 14(c)] because it uses wavelet decomposition and kernel function. Nonetheless, it loses many details of the image. Although NS\_FCM retains many details of the original image [see Fig. 14(d)], it wrongly assigns green label to some pixels which are supposed to be yellow. It can be seen in Fig. 14(e) that the result of CKS\_FCM algorithm is severely affected by noise [see Fig. 14(e)]; in this segmented map, most regions that are supposed to be marked yellow are green. The segmented map by ALFCM is the worst [see Fig. 14(f)]. In addition, this algorithm, similar to CKS\_FCM, wrongly segments many regions. MLSP does not yield good segmentation [see Fig. 14(g)] in comparison with the ground truth [see Fig. 14(a)] because the energy function of MLSP is not robust to noise. So, some contours of XIAN image are undetected in the segmented map [see Fig. 14(g)]. KCUSR also yields a poor segmentation [see Fig. 14(h)]; it results in inaccurate contours because noise causes false segmentation of many pixels in the red and yellow

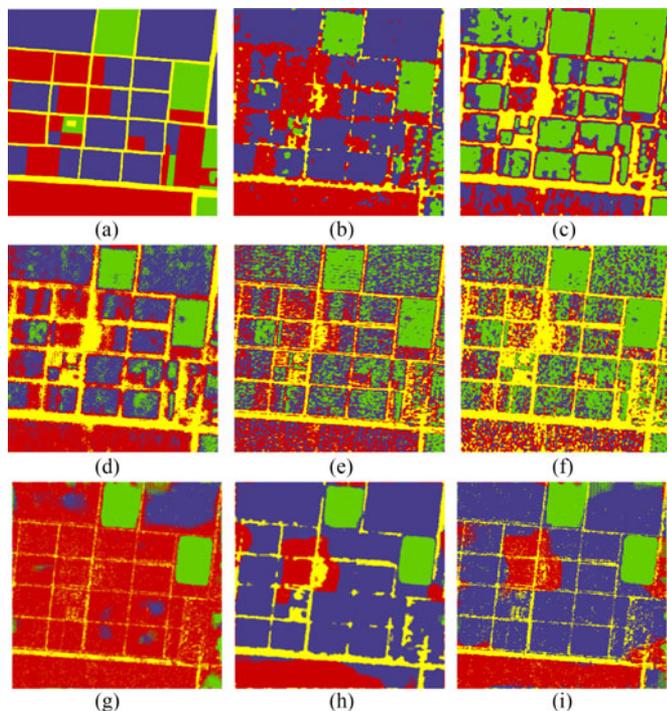


Fig. 15. Segmentation maps of MARICOPA image: (a) Ground truth map; (b) FKP\_FCM; (c) ILKFCM [34]; (d) NS\_FCM [36]; (e) CKS\_FCM [37]; (f) ALFCM [38]; (g) MLSP [46]; (h) KCUSR [47]; and (i) SFCM + MLSP.

regions. The poor performance of SFCM + MLSP is also similar to MLSP [see Fig. 14(i)]; nonetheless, it successfully detects some red regions in the middle upper part of the image unlike MLSP.

3) *Results on MARICOPA Image:* The number of classes for segmenting MARICOPA image is 4. This image consists of three kinds of farmlands, shown with yellow, blue, and green colors, and water pool, shown with blue, in the ground truth segmented map [see Fig. 15(a)].

FKP\_FCM yields nonsmooth edges; meanwhile, a few wrongly segmented points are contained in the red regions at the bottom of the map [see Fig. 15(b)]. On the other hand, this algorithm successfully segments blue and green regions. Next, ILKFCM assigns wrongly green color to the areas supposed to be blue [see Fig. 15(c)]; likewise, it marks wrongly the regions supposed to be red with blue. Although NS\_FCM retains many details of the original image [see Fig. 15(d)], the resulting segmentation map contains many points wrongly segmented; this makes the segmentation map blurred. CKS\_FCM also suffers from speckle noise seriously, and it wrongly segments the blue and green regions [see Fig. 15(e)]. The map of ALFCM with high noise level is similar to that of CKS\_FCM [see Fig. 15(f)]. Similarly, ALFCM wrongly segments the red and yellow regions. We can see that the ability of MLSP [see Fig. 15(g)] to segment red and blue regions is very poor. KCUSR performs better than MLSP, as shown in Fig. 15(h), due to the use of Gabor filter; contours of the map, however, are not very accurate because of the heavy noise of MARICOPA image. In Fig. 15(i), SFCM + MLSP has much better result than the one of MLSP

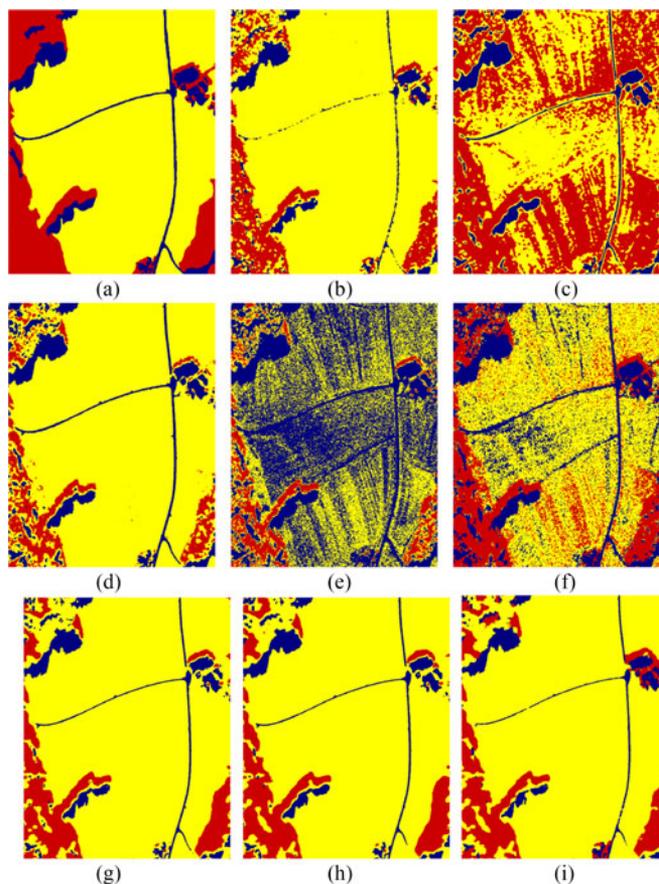


Fig. 16. Segmentation maps of TRAUNSTEIN image: (a) Ground truth map; (b) FKP\_FCM; (c) ILKFCM [34]; (d) NS\_FCM [36]; (e) CKS\_FCM [37]; (f) ALFCM [38]; (g) MLSP [46]; (h) KCUSR [47]; and (i) SFCM + MLSP.

due to the initial contours obtained by SFCM, but it still wrongly segments some blue and red regions.

4) *Results on TRAUNSTEIN Image:* The size of TRAUNSTEIN image is  $1001 \times 779$ , which is larger than other three real SAR images; hence, segmenting this image involves much longer computation time. This image is especially used to further confirm the efficiency of our proposed algorithm, namely, FKP\_FCM. TRAUNSTEIN image consists of three different areas, which are marked with red, blue, and yellow in the ground truth map [see Fig. 16(a)]. The segmentation results yielded by different algorithms are presented in Fig. 16.

Fig. 16 illustrates that NS\_FCM, MLSP, KCUSR, SFCM + MLSP, and our algorithm yield reasonably good segmentation maps. On the other hand, other three algorithms, namely, ILKFCM, CKS\_FCM, and ALFCM, show poor performances because of the speckle noise [in Fig. 16(c), (e), and (f)]. We can see in Fig. 16(g), (h), and (i) that MLSP, KCUSR, and SFCM + MLSP have similar results with clear contours because they benefit from the level-set methods. NS\_FCM achieves a promising segmentation map with some wrongly segmented points [see Fig. 16(d)]. In contrast, Fig. 16(b) shows that FKP\_FCM results in a number of wrongly segmented red pixels less than the corresponding map obtained by NS\_FCM in the cost of losing some details.

TABLE VIII  
RUNNING TIMES ON REAL SAR IMAGES

Algorithm	FARMLAND	XIAN	MARICOPA	TRAUNSTEIN
ILKFCM [34]	298.3544 s	366.9614 s	652.3885 s	3921.3 s
NS_FCM [36]	168.5366 s	219.2620 s	553.6243 s	4128.6 s
CKS_FCM [37]	817.4360 s	1102.5247 s	2729.5188 s	8467.1 s
ALFCM [38]	112.0980 s	209.7888 s	403.2569 s	4536.9 s
MLSP [46]	74.6680 s	181.7670 s	333.0439 s	1010.1 s
KCUSR [47]	82.7761 s	198.5236 s	355.6721 s	1326.7 s
SFCM + MLSP	98.2117 s	221.6559 s	408.9597 s	1112.4 s
FKP_FCM	<b>49.6627 s</b>	<b>56.4470 s</b>	<b>120.9367 s</b>	<b>642.3369s</b>

5) *Comparisons of Segmentation Accuracy and Running Time:* Although the segmentation maps of four real SAR images obtained by all algorithms have been presented in the previous parts of this section, it is not feasible to quantitatively determine which one performs best; or one may not identify the algorithm outperforming others if the obtained segmentation maps are similar. Hence, we computed the SA of the obtained maps using the ground truth maps; the resulting accuracy values are presented in Table VII.

Table VII shows that FKP\_FCM either achieved the best SAs in most of the cases or yield SAs very close to the best SAs for SAR images. This demonstrates the superiority of FKP\_FCM for segmenting SAR images. On the other hand, SFCM + MLSP has only the best SA on TRAUNSTEIN image because 1) this algorithm benefits from fuzzy clustering and active contour methods and 2) this image has high-resolution and less noise level. So, this algorithm is the most suitable for segmenting this image. Other clustering-based methods have little lower SAs because they are sensitive to noise. Next, two level-set-based algorithms have SAs close to the ones of clustering-based algorithms for segmenting these images because increased noise results in decreased ability to detect contours and boundaries.

In order to verify the efficiency of our proposed algorithm for segmenting real SAR images, we represent the computation times of eight algorithms in Table VIII. These four real SAR images have different sizes and are subject to different levels of speckle noise, so necessary computation times for segmenting them are very different.

The lesson we learn from Table VIII is that CKS\_FCM requires the longest computation time because of the use of the immune clone to find initial cluster centers. The next most time consuming algorithm for segmenting SAR images is ILKFCM because it computes features in a wavelet domain. ALFCM and NS\_FCM are faster than CKS\_FCM and ILKFCM because regional information in ALFCM and nonlocal mean filtered image in NS\_FCM are computed in advance (i.e., this computation is not iterative). MLSP and KCUSR are faster than other algorithms except FKP\_FCM. They evolve contours iteratively without considering the spatial neighborhood information of each pixel, which is often utilized in clustering algorithms. SFCM + MLSP needs longer time to achieve segmentation because this algorithm needs to execute SFCM to first segment the image to initialize the contours. In summary, our algorithm is much faster

TABLE IX  
CLUSTERING RESULTS ON KEY PIXELS

SI1			SI2		
look	Number of key pixels	Number/ratio of correctly clustered pixels	look	Number of key pixels	Number/ratio of correctly clustered pixels
1	2511	2499/0.9952	1	2855	2798/0.9800
2	2558	2549/0.9965	2	2893	2883/0.9965
4	2549	2544/0.9980	4	2888	2883/0.9983
6	2507	2507/1.0000	6	2814	2811/0.9989

than other seven algorithms for segmenting different real SAR images because it does not use all the pixels of an image for a time-consuming clustering process.

The comparison of the segmentation of simulated and real SAR images obtained by state-of-the-art clustering and level-set based algorithms demonstrates that our proposed algorithm, i.e., FKP\_FCM, yields high-quality segmentation in a reasonable time shorter than other algorithms. This shows the effectiveness of our approach and its superiority over the state-of-the-art algorithms.

### C. Clustering Result on Key Pixels

In this paper, we only cluster key pixels selected according to the local maximum rule. The clustering result of key pixels determines the final segmentation of the whole image as described above. Here, we discuss the effectiveness of the fuzzy clustering method. Eight simulated SAR images generated from SI1 and SI2 (see Fig. 8) are chosen to test and the results are presented in Table IX.

It can be seen from Table IX that a single image with different looks has similar but different key pixels. The reason is that different looks indicate different level of speckle noise; four simulated noisy images generated from the same image (SI1 or SI2) are similar but different from each other as shown in Fig. 8. So, when we select key pixels on these simulated noisy images, different results will be obtained despite using the same selection window and the same Gaussian filter. Table IX also shows that the fuzzy clustering method proposed in this paper yields very high accuracy on key pixels for segmenting all eight images. In the 6-look simulated SAR image of SI1, all key pixels are clustered correctly where the clustering accuracy is equal to or more than 0.98 for every image of SI2. If the clustering result of key pixels has high accuracy, the whole image has high SA and vice versa. The proposed fuzzy clustering method has such a high accuracy because

- 1) the number of key pixels is small and they include small number of noise;
- 2) nonlocal spatial information can improve the robustness of clustering to speckle noise; and
- 3) the influence of speckle noise can be further suppressed by the log-ratio operator when calculating the weight  $w$ .

And if key pixels are properly segmented, the majority of non-key pixels can also be correctly segmented.

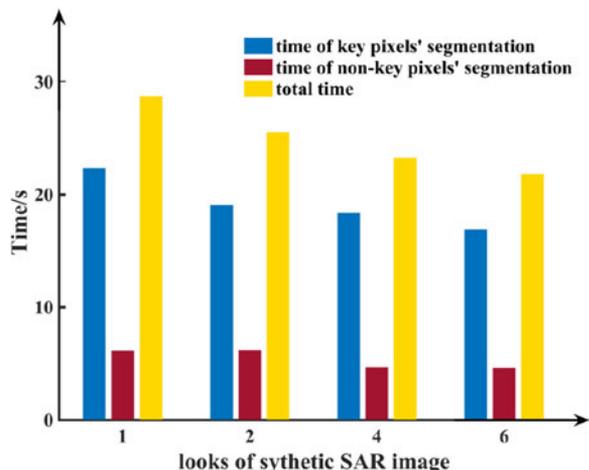


Fig. 17. Times used in FKP\_FCM.

#### D. Study on Time Complexity

According to the argument presented in the previous sections, FKP\_FCM achieves good segmentation of SAR images in a reasonable short time mainly because we apply the time-consuming clustering only on a small number of key pixels. Hence, a large number of non-key pixels are segmented quickly based on the clustering result of key pixels. To further analyze the computation time of FKP\_FCM, four simulated SAR images generated from SII are used for testing. Fig. 17 shows the times used for segmenting key and non-key pixels as well as the total running time.

The time used for clustering the key pixels accounts for about 80% of the total running time (see Fig. 17) and the time taken for segmenting the non-key pixels is about 20%. If the size of selection window  $\mathcal{N}$  is  $3 \times 3$ , the number of key pixels is about 5% of the number of total pixels. So, we use a time-consuming method to accurately segment a small number of key pixels and use a fast method to segment a large number of non-key pixels. Then, an accurate segmentation result on SAR image can be quickly obtained.

The experimental results show the effectiveness of our approach for segmenting different SAR images. Moreover, they illustrate the superiority of our approach over state-of-the-art algorithms.

## V. CONCLUSION

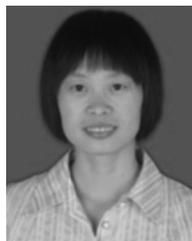
This paper has presented a fast and unsupervised algorithm for segmenting SAR images by only clustering a small number of key pixels. In contrast to the state-of-art clustering methods for segmenting SAR images, our proposed algorithm, called FKP\_FCM, at first selects a small number of special pixels as key pixels which are then clustered at low-computational cost using fuzzy clustering based on nonlocal information. Next, the segmentation of all remaining non-key pixels is achieved using both the segmentation results of the key pixels and a robust similarity metric. We have demonstrated the effectiveness of our algorithm via a variety of experiments using 12 simulated noisy SAR images (generated from three synthetic images) and

four real SAR images. We validated our proposed approach by comparing the results obtained by our algorithm with the results obtained by seven other state-of-the-art segmentation algorithms from recent literature. The experimental results show that our algorithm outperforms the state-of-the-art segmentation algorithms in terms of SA and computation time. Although our algorithm is very efficient in segmenting SAR images, the size of neighborhood window  $H$  must be set manually in order to achieve a good balance between suppressing noise and preserving details of the original image. Future work will focus on improving the algorithm in terms of preserving edges and other details of the original image, and the level-set methods may offer a potential way forward.

## REFERENCES

- [1] M. Poodanchi, G. Akbarizadeh, E. Sobhanifar, and K. Ansari-Asl, "SAR image segmentation using morphological thresholding," in *Proc. 2014 6th Inf. Knowl. Technol.*, Iran, 2014, pp. 27–29.
- [2] D. Karimi, K. Rangzan, G. Akbarizadeh, and M. Kabolizadeh, "Combined algorithm for improvement of fused radar and optical data classification accuracy," *J. Electron. Imag.*, vol. 26, no. 1, pp. 1–10, 2017.
- [3] S. Gou, X. Zhuang, H. Zhu, and T. Yu, "Parallel sparse spectral clustering for SAR image segmentation," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 6, no. 4, pp. 1949–1963, Aug. 2013.
- [4] Y. Huang and Y. C. Huang, "Segmentation SAR satellite images with the multilayer level set approach," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 4, no. 3, pp. 632–642, Sep. 2011.
- [5] Y. Ding, Y. Li, and W. Yu, "SAR image classification based on CRFs with integration of local label context and pairwise label compatibility," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 7, no. 1, pp. 300–306, Jan. 2014.
- [6] H. Yu, X. Zhang, S. Wang, and B. Hou, "Context-based hierarchical unequal merging for SAR image segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 995–1009, Feb. 2013.
- [7] J. Feng, Z. Cao, and Y. Pi, "Multiphase SAR image segmentation with G0-statistical-model-based active contours," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 7, pp. 4190–4199, Jul. 2013.
- [8] Z. Tirandaz and G. Akbarizadeh, "Unsupervised texture-based SAR image segmentation using spectral regression and Gabor filter bank," *J. Indian Soc. Remote Sens.*, vol. 44, pp. 177–186, Apr. 2016.
- [9] M. Rahmani and G. Akbarizadeh, "Unsupervised feature learning based on sparse coding and spectral clustering for segmentation of synthetic aperture radar images," *IET Comput. Vision*, vol. 9, no. 5, pp. 629–638, Sep. 2015.
- [10] D. Karimi, G. Akbarizadeh, K. Rangzan, and M. Kabolizadeh, "Effective supervised multiple-feature learning for fused radar and optical data classification," *IET Radar Sonar Navig.*, vol. 11, no. 5, pp. 768–777, May 2017.
- [11] F. Wang, Y. Wu, M. Li, P. Zhang, and Q. Zhang, "Adaptive hybrid conditional random field model for SAR images segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 1, pp. 537–550, Jan. 2017.
- [12] J. Li, X. Li, B. Yang, and X. Sun, "Segmentation-based Image copy-move forgery detection scheme," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 507–518, Mar. 2015.
- [13] G. Akbarizadeh and Z. Tirandaz, "SAR image segmentation using unsupervised spectral regression and Gabor filter bank," in *Proc. 2015 7th Inf. Knowl. Technol.*, Iran, 2015, pp. 1–4.
- [14] G. Akbarizadeh, "A new statistical-based kurtosis wavelet energy feature for texture recognition of SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 11, pp. 4358–4368, Nov. 2012.
- [15] G. Moser and S. B. Serpico, "Generalized minimum-error thresholding for unsupervised change detection from SAR amplitude imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 10, pp. 2972–2982, Oct. 2006.
- [16] J. Gu, L. Jiao, S. Yang, F. Liu, B. Hou, and Z. Zhao, "A multi-kernel joint sparse graph for SAR image segmentation," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 9, no. 3, pp. 1265–1285, Mar. 2016.
- [17] J. Yin and J. Yang, "A modified level set approach for segmentation of multiband polarimetric SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 11, pp. 7222–7232, Nov. 2014.

- [18] R. Stolkin, M. Hodgetts, A. Greig, and J. Gilby, "Extended Markov random fields for predictive image segmentation," in *Proc. Adv. Pattern Recog.*, India, 2007, pp. 208–214.
- [19] B. Gu and V. S. Sheng, "A robust regularization path algorithm for  $\nu$ -support vector classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 5, pp. 1241–1248, May 2017.
- [20] B. Gu, V. S. Sheng, K. Y. Tay, W. Romano, and S. Li, "Incremental support vector learning for ordinal regression," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 7, pp. 1403–1416, Jul. 2015.
- [21] Y. Zhong, A. Ma, and L. Zhang, "An adaptive memetic fuzzy clustering algorithm with spatial information for remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 7, no. 4, pp. 1235–1248, Apr. 2014.
- [22] Z. Zhou, Q. J. Wu, F. Huang, and X. Sun, "Fast and accurate near-duplicate image elimination for visual sensor networks," *Int. J. Distrib. Sens. Netw.*, vol. 13, no. 2, pp. 1–12, 2017.
- [23] D. L. Pham and J. L. Prince, "An adaptive fuzzy C-means algorithm for image segmentation in the presence of intensity inhomogeneities," *Pattern Recog. Lett.*, vol. 20, no. 1, pp. 57–68, Jan. 1999.
- [24] J. Fan, Y. Wu, M. Li, W. Liang, and Q. Zhang, "SAR image registration using multiscale image patch features with sparse representation," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 10, no. 4, pp. 1483–1493, Apr. 2017.
- [25] M. Gong, L. Su, M. Jia, and W. Chen, "Fuzzy clustering with a modified MRF energy function for change detection in synthetic aperture radar images," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 1, pp. 98–109, Feb. 2014.
- [26] P. Zhang, M. Li, Y. Wu, and H. Li, "Hierarchical conditional random fields model for semisupervised SAR image segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 9, pp. 4933–4951, Sep. 2015.
- [27] M. N. Ahmed, S. M. Yamany, N. Mohamed, A. A. Farag, and T. Moriarty, "A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data," *IEEE Trans. Med. Imag.*, vol. 21, no. 3, pp. 193–199, Mar. 2002.
- [28] S. Chen and D. Zhang, "Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure," *IEEE Trans. Syst. Man Cybern.*, vol. 34, no. 4, pp. 1907–1916, Aug. 2004.
- [29] L. Szilagyi, Z. Benyo, S. M. Szilagyi, and H. S. Adam, "MR brain image segmentation using an enhanced fuzzy c-means algorithm," in *Proc. 25th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Mexico, 2003, pp. 724–726.
- [30] W. Cai, S. Chen, and D. Zhang, "Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation," *Pattern Recog.*, vol. 40, no. 3, pp. 825–838, Mar. 2007.
- [31] S. Krinidis and V. Chatzis, "A robust fuzzy local information C-means clustering algorithm," *IEEE Trans. Image Process.*, vol. 19, no. 5, pp. 1328–1337, May 2010.
- [32] M. Gong, Z. Zhou, and J. Ma, "Change detection in synthetic aperture radar images based on image fusion and fuzzy clustering," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 2141–2151, Apr. 2012.
- [33] M. Gong, Y. Liang, J. Shi, W. Ma, and J. Ma, "Fuzzy C-means clustering with local information and kernel metric for image segmentation," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 573–584, Feb. 2013.
- [34] D. Xiang, T. Tang, C. Hu, Y. Li, and Y. Su, "A kernel clustering algorithm with fuzzy factor: Application to SAR image segmentation," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 7, pp. 1290–1294, Jul. 2014.
- [35] S. Wang, L. Jiao, and S. Yang, "SAR images change detection based on spatial coding and nonlocal similarity pooling," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 9, no. 8, pp. 3452–3466, Aug. 2016.
- [36] J. Ji and K. Wang, "A robust nonlocal fuzzy clustering algorithm with between-cluster separation measure for SAR image segmentation," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 7, no. 12, pp. 4929–4936, Dec. 2014.
- [37] R. Shang *et al.*, "A spatial fuzzy clustering algorithm with kernel metric based on immune clone for SAR image segmentation," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 9, no. 4, pp. 1640–1652, Apr. 2016.
- [38] G. Liu, Y. Zhang, and A. Wang, "Incorporating adaptive local information into fuzzy clustering for image segmentation," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3990–4000, Nov. 2015.
- [39] J. Feng, L. Jiao, X. Zhang, M. Gong, and T. Sun, "Robust non-local fuzzy c-means algorithm with edge reservation for SAR image segmentation," *Signal Process.*, vol. 93, no. 2, pp. 487–499, Feb. 2013.
- [40] M. T. Pham, G. Mercier, and J. Michel, "Textural features from wavelets on graphs for very high resolution panchromatic Pléiades image classification," *Revue Française Photographie. Télédétection*, no. 208, pp. 131–136, Oct. 2014.
- [41] C. Yuan, Z. Xia, and X. Sun, "Coverless image steganography based on SIFT and BOF," *J. Internet Technol.*, vol. 18, no. 2, pp. 435–442, Mar. 2017.
- [42] M. T. Pham, G. Mercier, and J. Michel, "Pointwise graph-based local texture characterization for very high resolution multispectral image classification," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 8, no. 5, pp. 1962–1973, May 2015.
- [43] M. T. Pham, G. Mercier, and J. Michel, "Change detection between SAR images using a pointwise approach and graph theory," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 4, pp. 2020–2032, Apr. 2016.
- [44] M. T. Pham, G. Mercier, and J. Michel, "Wavelets on graphs for very high resolution multispectral image segmentation," in *Proc. 2014 IEEE Geosci. Remote Sens. Symp.*, Canada, 2014, pp. 2273–2276.
- [45] C. Li, R. Huang, Z. Ding, J. C. Gatenby, D. N. Metaxas, and J. C. Gore, "A level set method for image segmentation in the presence of intensity inhomogeneities with application to MRI," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 2007–2016, Jul. 2011.
- [46] I. B. Ayed, A. Mitiche, and Z. Belhadj, "Multiregion level-set partitioning on synthetic aperture radar images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 793–800, May 2005.
- [47] Z. Tirandaz and G. Akbarizadeh, "A two-phase algorithm based on Kurtosis curvelet energy and unsupervised spectral regression for segmentation of SAR images," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 9, no. 3, pp. 1244–1264, Mar. 2016.
- [48] M. Modava and G. Akbarizadeh, "Coastline extraction from SAR images using spatial fuzzy clustering and the active contour method," *Int. J. Remote Sens.*, vol. 38, no. 2, pp. 355–370, 2017.
- [49] K. Chuang, H. Tzeng, S. Chen, J. Wu, and T. Chen, "Fuzzy c-means clustering with spatial information for image segmentation," *Comput. Med. Imag. Graph.*, vol. 30, no. 1, pp. 9–15, Jan. 2006.
- [50] B. Hou, X. Zhang, X. Bu, and H. Feng, "SAR image despeckling based on nonsubsampled shearlet transform," *IEEE J. Sel. Topics Appl. Earth Obs. Remote Sens.*, vol. 5, no. 3, pp. 809–823, Jun. 2012.



**Ronghua Shang** (M'09) received the B.S. degree in information and computation science and the Ph.D. degree in pattern recognition and intelligent systems from Xidian University, Xi'an, China, in 2003 and 2008, respectively.

She is currently a Professor with Xidian University. Her current research interests include evolutionary computation, image processing, and data mining.



**Yijing Yuan** received the B.E. degree in electrical engineering from the School of Computer Science and Control Engineering, North University of China, Taiyuan, China. She is currently working toward the M.S. degree in the School of Electronic Engineering, Xidian University, Xi'an, China.

Her current research interests include image processing and machine learning.



**Licheng Jiao** (SM'89) received the B.S. degree in electrical engineering from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees in signal, circuit and system from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively.

From 1990 to 1991, he was a Postdoctoral Fellow in the National Key Laboratory for Radar Signal Processing, Xidian University, Xi'an, China. Since 1992, he has been a Professor in the School of Electronic Engineering, Xidian University. Currently, he is the Director of the Key Lab of Intelligent Perception and Image Understanding of Ministry of Education of China, Xidian University. He has charge of about 40 important scientific research projects, and published more than 20 monographs and a hundred papers in international journals and conferences. His research interests include image processing, natural computation, machine learning, and intelligent information processing.

Dr. Jiao is a member of the IEEE Xi'an Section Execution Committee, the Chairman of Awards and Recognition Committee, a Vice Board Chairperson of the Chinese Association of Artificial Intelligence, a Councilor of the Chinese Institute of Electronics, a committee member of the Chinese Committee of Neural Networks, and an Expert of the Academic Degrees Committee of the State Council.



**Biao Hou** (M'07) received the B.S. and M.S. degrees in mathematics from Northwest University, Xi'an, China, in 1996 and 1999, respectively, and the Ph.D. degree in circuits and systems from Xidian University, Xi'an, China, in 2003.

Since 2003, he has been in the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education, Xidian University, where he is currently a Professor. His research interests include multiscale geometric analysis and synthetic aperture radar image processing.



**Rustam Stolkin** (M'xx) received the M.Eng. degree in engineering science from the University of Oxford, Oxford, U.K., in 1998, and the Ph.D. degree in robotic vision from the University College London, London, U.K., in 2004.

He is currently the Royal Society Industry Fellow for Nuclear Robotics, and a Professor of robotics with the University of Birmingham, Birmingham, U.K., where he is the Founder and the Director of the Extreme Robotics Lab. He is also the Director of spinout company A.R.M Robotics Ltd.



**Amir Masoud Ghalamzan Esfahani** (M'xx) received the B.S. and M.S. degrees in mechanical engineering from Urmia University and Iran University of Science and Technology, Iran, in 2004 and 2009, respectively, a second-level specialization degree in automatic control engineering from the Politecnico di Torino, Turin, Italy, in 2010, and the Ph.D. degree in robotics from the Politecnico di Milano, Milan, Italy, in 2015.

Since 2015, he has been a Post-Doctoral Research Fellow with the University of Birmingham, Birmingham, U.K. His research interests include robotic manipulation, robot learning from demonstration, machine learning, and computer version.