SAR Image Change Detection Based on Joint Dictionary Learning With Iterative Adaptive Threshold Optimization

Qiuze Yu, Miao Zhang, Lijie Yu, Ruikai Wang, and Jinsheng Xiao

***Abstract*—Synthetic aperture radar (SAR) image change detection is still a challenge due to inherent speckle noise and scarce datasets. This article proposes a joint-related dictionary learning algorithm based on the k-singular value decomposition (K-SVD) algorithm called JR-KSVD and an iterative adaptive threshold optimization (IATO) algorithm for unsupervised change detection. The JR-KSVD algorithm adds dictionary correlation learning to the K-SVD algorithm to generate a uniform initial dictionary for dual-temporal SAR images, thereby reducing the instability of sparse representations due to atomic correlations and enhancing the extraction of image edges and details. The IATO approach employs thresholds obtained by the "difference-log ratio" fusion image for indefinite residual energy minimization iterations to gradually shrink the threshold variation range and finally generate the change images, which have a high degree of adaptivity and strong real-time performance. Finally, experiments on six real datasets demonstrate that the proposed algorithm exhibits superior detection performance and robustness against some state-of-the-art algorithms.**

***Index Terms*—Change detection, difference-log ratio image, iterative adaptive threshold, joint-related dictionary learning, synthetic aperture radar (SAR) image.**

Acronym	*Meaning*
SAR	Synthetic aperture radar.
K-SVD	K-singular value decomposition.
JR-KSVD	Joint-related k-singular value decomposition.
IATO	Iterative adaptive threshold optimization.
EKSVD	Efficient k-singular value decomposition.
GSR	Group sparse representation.
PCA	Principal component analysis.
MP	Matching pursuit.
OMP	Orthogonal matching pursuit.
BP	Basis pursuit.
FOCUSS	Focal underdetermined system solution.
1-D 2-D	One-dimensional Two-dimensional.
FN	False negative number.
FP	False positive number.
OE	Overall error numbers.
PCC	Percentage correct classification.
ROC	Receiver operating characteristic.
AUC	Area under the curve.
RFLICM	C-means clustering of trimmed fuzzy local information algorithm.
SAEFCM	Sparse autoencoder (SAE) and fuzzy c-means (FCM) clustering change detection algorithm.
SIFT	Scale invariant feature transform.
RUSACD	Robust unsupervised small area change detection.
SAFNet	Siamese adaptive fusion network.
GaborPCANet	Automatic change detection based on the principal component analysis network (PCANet).
NRELM	Neighborhood-based ratio and extreme learning machine.
GT	Ground truth.
SVM	Support vector machine.

Manuscript received 31 March 2022; revised 29 May 2022; accepted 26 June 2022. Date of publication 29 June 2022; date of current version 11 July 2022. This work was supported by the Open Project Program Foundation of the Key Laboratory of Opto-Electronics Information Processing, Chinese Academy of Sciences under Grant OEIP-O-202009. *(Corresponding author: Miao Zhang.)*

The authors are with the School of Electronic Information, Wuhan University, Wuhan 430072, China (e-mail: yuhenry007@whu.edu.cn; 2020202120095@whu.edu.cn; 707980725@qq.com; 2017301200198@whu.edu.cn; xiaojs@whu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3187108

I. Introduction

THE application of change detection technology in real life has increased over the years with the rapid development of urban construction and the frequent occurrence of various natural disasters, such as earthquakes, mudslides, and natural fires. Synthetic aperture radar (SAR) [1]–[4] is a useful ground imaging technology for seeing through masks and detecting camouflage. The greatest advantage of SAR is its all-weather capabilities, which give it a significant advantage over infrared and visible light sensors. As a result, using SAR images for change detection has become the current mainstream trend, and SAR images have many possible applications in environmental monitoring [5]–[7], wetland changes [8], [9], and land evolution [10].

Currently, there are two types of SAR image change detection approaches: 1) supervised learning and 2) unsupervised learning [11], [12]. The goal of supervised learning is to build a relevant classification network and then perform supervised training on multiple sets of input images [13]. Wang *et al.* [14] introduced a new joint change detection network based on similarity learning and built a C2-net and an intensity network to expose similarities between picture patches during training to detect a target, where the C2-net is a concatenated structure with a matrix composed of the diagonal elements of the covariance

TABLE I
RELEVANT REFERENCES AND THEIR MAIN CHARACTERISTICS

Methods	Main Characteristics	Articles
Supervised learning	Simple Image Detection	Wang *et al.* [14] , Li *et al.* [15]
	Convenient detection method	Gong *et al.* [16]
Unsupervised learning	Simple Image Detection	Majidi *et al.* [18]
	Complex detection methods and high detection costs	Saha *et al.* [19], Geng *et al.* [20]
	Solves the problem of speckle noise	Choi *et al.* [21], Chen *et al.* [22], Liu *et al.* [23]
	Poor detail and edge accuracy	Chen *et al.* [22], Zhan *et al.* [24]

of the second subnetwork of the dual-polarized complex SAR data as input; however, it shows good performance only in comparison with the "amplitude-based method" and for small SAR images. Li *et al.* [15] and others used deformation similarities to perform patch matching, which searches for the best set of atoms in a differential image dictionary, reconstructs patches for the new image, and maps from the differential image to the change detection image, but the test results in some areas occasionally have some shortcomings compared to several other advanced methods. Gong *et al.* [16] used trained deep neural networks to build change detection maps straight from two photos, eliminating the procedure of producing differential images between dual time phase SAR images. Unsupervised learning change detection is typically divided into three steps: 1) preprocessing; 2) obtaining a difference image; and 3) analyzing the difference image [17]. In a previous study [18], a saliency-guided method based on the Otsu threshold and a neighborhood ratio model based on the probability distribution were proposed, and the K-means method was applied to binarize the difference maps of SAR images, but this approach can perform change detection only for simple SAR images and cannot accurately extract smaller change regions. Other scholars [19] used pairs of unlabeled SAR images and optical images, transcoded the SAR images into optical images using the cycle generative adversarial network (CycleGAN), performed unsupervised training, and finally converted the optical images back into SAR images. However, this method uses both optical and SAR images, which increases the detection cost. Another study [20] combined unsupervised change detection methods and saliency-guided deep neural networks to obtain pseudotraining samples to achieve better binarization classification results.

The above discussion of supervised and unsupervised change detection indicates that most supervised change detection methods require more training data and that the computational cost of neural networks is high, but supervised methods are more effective in detecting complex terrain and require less manual adjustment of the parameters. For unsupervised change detection, the details of the images are not detected sufficiently well and are often applied to relatively simple terrain, but they can be detected directly without training samples, and the overall computational cost is relatively low. Considering the advantages and disadvantages of both supervised and unsupervised detection and the difficulty of obtaining high-resolution SAR images, this article adopts an unsupervised feature extraction method based on sparse representations for SAR image change detection.

There are significant issues in the field of unsupervised change detection based on SAR image change detection methods and SAR imaging properties. First, poor feature extraction results in a significant loss of image edge features. Second, these methods cannot automatically generate thresholds or generate thresholds that are not precise enough to provide accurate change images. Third, the reconstructed image contains more speckle noise. In regard to the problem of speckle noise, several researchers have explored the subject and achieved improved results [21]. Chen *et al.* [22] used the efficient k-singular value decomposition algorithm to train abstract and discriminative high-level features by using a block structure dictionary, which improves robustness against coherent speckle noise. Mean filtering was included in the modeling process by Liu *et al.* [23], resulting in despeckled SAR picture restoration based on the group sparse representation model. However, the first two difficulties of unsupervised change detection remain somewhat poorly understood, and thus, this article focuses on the two aspects of feature extraction and threshold selection. This approach can better extract image features more completely and preserve image edge information, as well as adaptively generate the image segmentation threshold, to achieve a more satisfactory segmentation effect. For the speckle noise problem, the sparse representation can suppress it to some extent since the feature signal is expressed in the dictionary with sparsity, while the noise signal is not sparse. In particular, since the image information is contained in a small number of coefficients, the corresponding feature signal coefficients are large, and the noise coefficients are small. However, although the complete signal is encoded, only the larger coefficients are retained, whereas the smaller coefficients are filtered, thus achieving the effect of filtering out noise.

To indicate the motivation and relevant contributions of this article, we have summarized the literature on supervised and unsupervised learning discussed above by their main features, as shown in Table I. We propose joint-related k-singular value decomposition (JR-KSVD), a joint dictionary learning algorithm, to address the problem of "poor detail and edge accuracy" indicated in the table, which is also one of the motivations of our article. By conducting correlation operations on the initial dictionaries in the sparse format, this technique minimizes the high similarity between atoms, reduces redundant features, and enhances edge and detail detection accuracy. Furthermore, most unsupervised change detection algorithms require human adjustment of segmentation thresholds to achieve better detection results, which greatly increases the labor cost. Therefore, this article combines the benefits of the top-down (TD) and bottom-up (BU) algorithms in the literature [25] to improve the Otsu algorithm [26] and proposes an iterative threshold optimization algorithm (IATO), which is the other contribution of our work. The algorithm fuses the sparsely reconstructed image's differential image and log-ratio image to adjust the reconstructed map,

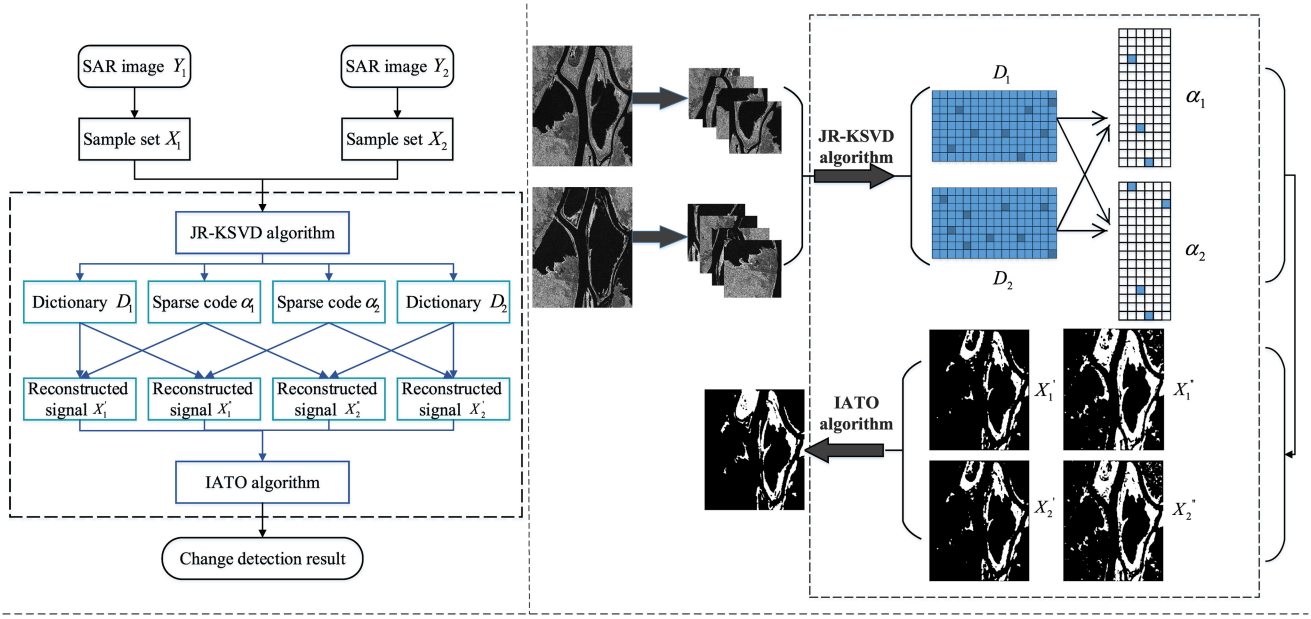

Fig. 1. Flowchart of SAR image change detection.

after which the image is iteratively constrained using one- and two-dimensional (1- and 2-D) segmentation algorithms until a suitable threshold is generated, and finally, the change image is generated. This approach significantly accelerates threshold selection and achieves adaptive image segmentation.

The methodologies and ideas used in this investigation are detailed in the second half of this publication. The dependability and applicability of the research approach are proven in the third section by displaying the experimental data. Finally, in the conclusion, the work is summarized, and the study's limitations and potential improvements are discussed.

In the scope of this article, the following notations are used: $\mathbb{R}$ is the set of real vectors. x is an $M \times N$-dimensional matrix used to represent the set of signals. D is an $M \times K$-dimensional matrix, and d_i is a column vector in D with dimension M. α is a $K \times N$-dimensional matrix. $\| \bullet \|_0$ is used to calculate l^0. ε denotes the maximum reconstruction error. L, l denotes the sparsity, and Q, q denotes the number of iterations. $\| \bullet \|_F^2$ is used to calculate l^2. T_0 denotes the upper limit of the zero norm of α. Ω denotes the contraction matrix. Δ denotes the diagonal matrix. $T_r(\bullet)$ denotes the trace finding operation.

II. Methodology

We employ dictionary learning and sparse representation to reconstruct gray images for SAR image change detection and an adaptive threshold segmentation method to generate change images. The first step is to determine how to obtain a pair of adaptive sparse dictionaries. Dictionary learning methods [27] that are commonly used include the K-SVD dictionary learning method [28], the principal component analysis (PCA) dictionary learning method [29], and the online learning dictionary learning method [30], [31]. Because of its fast learning speed and high capacity to learn features, the K-SVD algorithm is frequently utilized for SAR image training. By upgrading the K-SVD algorithm, the JR-KSVD algorithm is obtained in this study. In comparison to the K-SVD technique, the JR-KSVD approach uses correlation learning and dictionary rearrangement to initialize a dictionary, which enhances dictionary learning stability and uniformity in SAR image change detection and improves edge retention. Furthermore, the rearranged dictionary atoms are exceptionally representative and may more effectively represent a changing section in a pair of dual time phase SAR images, which improves the detection accuracy. The second step is to design the adaptive threshold segmentation function. This study proposes the IATO algorithm, which uses an initial differential image to determine the initial threshold and then computes the threshold as a parameter of the "differential-log ratio" fusion function to build a new threshold. The final change image is generated by continuous optimization iterations. The IATO algorithm is highly adaptive and effective in segmentation, which substantially simplifies the process of determining the best segmentation threshold and makes change detection much easier compared with previous articles. The SAR image change detection flowchart is shown in Fig. 1, which clearly depicts the experimental design concept.

A. Joint Dictionary Learning and Reconstruction of Synthetic Aperture Radar Images

1) Joint-Related Dictionary Learning Based on an Overcomplete Dictionary: The JR-KSVD algorithm proposed in this article is based on the overcomplete sparse representation technique. Sparse representation theory states that in the representation of complex signals, the majority of natural signals can be thought of as linear superpositions of a sequence of atomic signals, with the plurality of these linear coefficients having zero values; hence, they are referred to as sparse coefficients. In general,

the signal matrix $x \in \mathbb{R}^{M\times N}$ denotes a sparsely represented signal in the experiment (M and N represent the number of rows and columns of the matrix, respectively), and the vector set $d_i \in \mathbb{R}^M$, where $D = \{d_i\}, i = 1, 2 \ldots K, K > M$ denotes the overcomplete basis, i.e., the atoms mentioned in sparse representation theory, where a series of atoms is joined into a dictionary. Matrix D with dimensions $M \times K$ is defined and used as a dictionary. When $K > M$, that is, when the number of columns in the dictionary exceeds the number of rows, the dictionary is said to be an overcomplete dictionary, and the atom d_i is the dictionary's basic unit that symbolizes the dictionary's column vector. The optimization process of (1) is, in general, the exploration of a sparse expression for the signal x

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_0 \text{ s.t. } \|x - D\alpha\|_F^2 \leq \varepsilon. \tag{1}$$

In the above equation, D and α represent the dictionary matrix and the coefficient matrix used to represent the signal x, respectively ($x \in \mathbb{R}^{M\times N}$, $D \in \mathbb{R}^{M\times K}$, and $\alpha \in \mathbb{R}^{K\times N}$). The symbol $\| \bullet \|_0$ is used to calculate l^0, while ε represents the maximum permissible reconstruction error. Our goal is to find the dictionary D that allows α to be the sparsest, which is a process of finding the optimal solution, i.e., an NP-hard problem, which was initially solved by Mallat *et al.* [32] with the matching pursuit (MP) algorithm. As a result, some new algorithms, such as the orthogonal MP (OMP) algorithm [33], basis pursuit algorithm [34], and focal underdetermined system solution methods [35], have been proposed to solve this problem. In this article, the mature OMP algorithm is used to perform sparse signal decomposition. The OMP algorithm must run through dictionary atoms one by one, project the signal orthogonally onto the selected atoms, and choose the atom with the highest projection value as the best match for signal x; then, the signal residuals are computed. The residual signal is utilized as the new signal, and the best-matching atoms are chosen; this process is repeated until signal x can be represented by a linear combination of these best-matching atoms plus the final residual value. When the residuals are small and negligible, signal x is a linear combination of these atoms. The overall computational flow of the OMP algorithm is described in Algorithm 1.

Algorithm 1: OMP algorithm steps.

input: Test signal x, dictionary D, sparsity L, maximum allowable reconstruction error ε, residual signal $r_0 = x$, sparsity $l = 0$, and atomic index set $I_0 = \varnothing$;

output: The sparse matrix of signal x, $\alpha = (D_{I_k}^T D_{I_k})^{-1} D_{I_k}^T x$;

1: Look for the atom with the largest projection: $\hat{k} = \arg\max_k |d_k^T r_{k-1}|$. $\hat{k}$ represents the serial number of the selected atom; that is, the atomic index value d_k^T represents the selected atom, and r_{k} represents the residual after matching;

2: Update the atomic index set: $I_{\mathrm{k}} = (I_{\mathrm{k-1}}, \hat{k})$;

3: Update the residual signal: $\alpha_{i_k} = (D_{I_k}^T D_{I_k})^{-1} D_{I_k}^T x \quad , \quad r_{\mathrm{k}} = x - D_{I_k}\alpha_{I_k}, l = l + 1$, where α_{I_k} denotes the sparse matrix;

4: Determine the end-of-cycle condition. If $l \geq L$ or $r_k < \varepsilon$, then stop the iteration; otherwise, continue the loop.

In addition to the superiority of the sparse representation algorithm, the accuracy of the sparse representation of signals is determined by the dictionary's reasonableness and completeness. Additionally, the dictionary needs to contain the main structural information to express the signal so that the signal can be represented completely by a sparse representation. In general, we can obtain dictionaries from original signal samples by training, or we can compose dictionaries by function transformations. The latter dictionaries are primarily derived from particular transforms, such as the short-time Fourier transform [36], overcomplete wavelet transform [37], and contourlet transform [38]; they are simple to obtain but less adaptive and cannot be used to adequately represent SAR images. Therefore, to ensure the accuracy and sparsity of the sparse representation, we must utilize a learning dictionary to generate an adaptive dictionary. According to a significant quantity of data, the K-SVD algorithm, which is derived from the K-means clustering technique, can make good use of the properties of SAR images and may be used with any of the matching tracking algorithms to build a dictionary that fits the requirements. We proposed the JR-KSVD algorithm by combining the similarities of dual time phase images with the differences in the images to be generated. The JR-KSVD algorithm runs a joint correlation calculation on the two initial dictionaries and selects the combination of atoms with the highest degree of difference among the many atoms to generate the new dictionary. The new dictionary is then utilized as the initial dictionary of the K-SVD algorithm, which is iteratively updated on the two signals, and finally, two groups of dictionaries and sparse matrices are formed to accurately describe the input signals. The high degree of difference among the atoms shown in the JR-KSVD algorithm accelerates the subsequent algorithm's convergence and improves the stability of the iteratively created dictionaries. The JR-KSVD algorithm is separated into three steps: 1) calculate the joint correlation and define the new initial dictionary; 2) locate the sparse matrix using the OMP technique; and 3) utilize the SVD decomposition algorithm to update the dictionary and sparse matrix. The three steps of the method are discussed separately in the next sections.

a) *Calculate the joint correlation:* We initially segment the dual time phase SAR images with a sliding window of size $r \times r$ to obtain several patches (atoms) of size r^2 and then choose 256 of them at random to build the initial dictionaries D_A and D_B. The dataset and initial dictionary generation process are depicted in Fig. 2. Equation (2) obtains a new dictionary *D* determined by the initial dictionaries D_A and D_B by designing the correlation function Corr() and setting constraints. Equation (3) takes the new dictionary D and the sparse matrices α_1 and α_2 as mutual adjustment variables to obtain the dictionary and sparse matrices that can best represent the input signals x_1 and x_2 in preparation for the next image reconstruction and threshold segmentation, which are expressed as D_1, D_2,

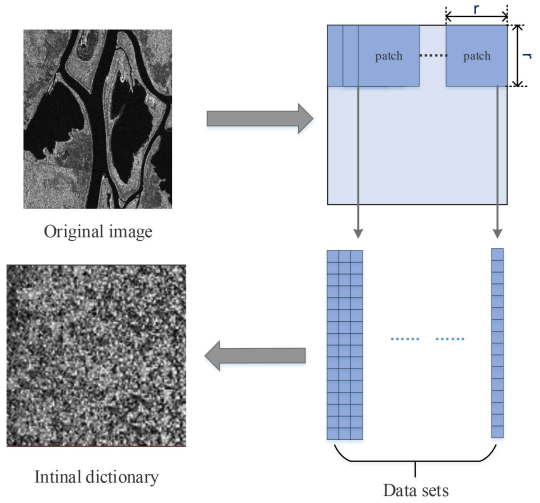

Fig. 2. Dataset and initial dictionary generation process.

and α_1, α_2.

$$D = \text{Corr}(D_A, D_B) \text{ s.t.} \max\left(\|D_A - D_B\|_2^2\right) \tag{2}$$

$$[D_1, D_2, \alpha_1, \alpha_2] = KSVD\left(\alpha_1, \alpha_2, D, X_1, X_2\right). \tag{3}$$

b) *Locate the sparse matrix:* The method for solving the dictionary and sparse matrix using K-SVD on the initial dataset $X = \{x_i\}_{i=1,2}, x \in \mathbb{R}^{M\times N}$ can be described by

$$\min_{D,\alpha}\left\{\|X - D\alpha\|_F^2\right\}, \text{ s.t. } \forall_i, \|\alpha_i\|_0 \le T_0. \tag{4}$$

To solve for the other variable α, we must assume that D is fixed. Equation (4) can be changed into the following equation at this stage:

$$\|X - D\alpha\|_F^2 = \sum_{i=1}^{M}\|x_i - D\alpha_i\|_F^2. \tag{5}$$

Hence, solving (4) is transformed into solving *M* repeated decomposition steps, which is expressed in (5) above. Therefore, (4) can be simplified as

$$\min_{\alpha_i}\left\{\|x_i - D\alpha_i\|_F^2\right\}, \text{ s.t. } \forall_i, \|\alpha_i\|_0 \le T_0, i = 1, 2, .., M. \tag{6}$$

This equates to finding the sparse representation matrix of the input signal x_i on the dictionary D with sparsity T_0, which can be solved using the previously mentioned OMP approach.

c) *Update the dictionary:* The dictionary is updated using atoms, with just one column of atoms being updated at a time. Simultaneously, the sparse coefficient related to this atom is modified, i.e., the atom's contribution to the error is cleared, and the remaining error matrix is deconstructed using SVD. In light of the preceding considerations, (5) is rewritten as follows:

$$\begin{aligned}\|X - D\alpha\|_F^2 &= \left\|X - \sum_{j=1}^{K} d_j\alpha_T^j\right\|_F^2 \\ &= \left\|\left(X - \sum_{j\ne k}^{K} d_j\alpha_T^j\right) - d_k\alpha_T^k\right\|_F^2 \\ &= \left\|E_k - d_k\alpha_T^k\right\|_F^2\end{aligned} \tag{7}$$

where d_j represents the jth column of atoms in the dictionary, α_T^j represents the jth row of the sparse matrix (the jth column of the transpose matrix of the sparse matrix), and when updating the kth atom, $X - \sum_{j\ne k}^{K} d_j\alpha_T^j$ is the overall error that represents the sample signal when atom d_k is not involved in the sparse representation. Thus, the objective function is transformed to fix the other columns of the dictionary D and locate the d_k that minimizes the overall error. To solve d_k and α_T^k, we must decompose E_k using the SVD method. We simply need to discover the nonzero elements in α_T^k and the corresponding columns of the original sample signal using d_k to meet the decomposition sparsity condition. By specifying the index group $\omega_k = \{i \mid 1 \le i \le M, \alpha_T^k(i) \ne 0\}$, the K-SVD method discovers the nonzero elements in α_T^k. ω_k is the index of the nonzero elements in α_T^k, and the corresponding signal is the sample signal that uses d_k for the sparse representation. The contraction matrix Ω_k is defined as

$$\Omega_k(a,b) = \begin{cases} 1 & (a,b) = (\omega_k(i), i) \\ 0 & \text{else.}\end{cases} \tag{8}$$

The reduced row vector, error matrix, and dataset matrix are denoted by $\alpha_R^k = \alpha_T^k\Omega_k$, $E_k^R = E_k\Omega_k$, and $X_k^R = X\Omega_k$, respectively, and (7) can be rewritten as

$$\left\|E_k\Omega_k - d_k\alpha_T^k\Omega_k\right\|_F^2 = \left\|E_k^{\mathrm{R}} - d_k\alpha_R^k\right\|_F^2. \tag{9}$$

For E_k^R, SVD is used as $E_k^R = U\Delta V^T$. $\tilde{d}_k = U(:,1)$ and $\tilde{\alpha}_R^k = V(:,1) * \Delta(1,1)$ are the updated atoms and their sparse representation coefficients, respectively.

The JR-KSVD algorithm's flowchart, which includes the three steps listed above, is shown in Fig. 3, and the exact steps are shown in Algorithm 2.

2) Difference Map Based on Cross-Reconstruction: The dictionaries D_1 and D_2 obtained from the JR-KSVD algorithm contain many textures and local features corresponding to the original image, respectively; thus, we can use $D_1\alpha_1$ and $D_2\alpha_2$ to represent the reconstructed image, and when the error between the reconstructed image and the original image is small, i.e., $\|X_1 - D_1\alpha_1\|_2^2 \le \varepsilon$ and $\|X_2 - D_2\alpha_2\|_2^2 \le \varepsilon$, the reconstructed image can be approximated as the original image to be tested. The reconstruction map can then depict the difference as the change region of the two images. For the above idea, we use cross-reconstruction to derive the change region of the two images. If the reconstructed signal $X_1' = D_1\alpha_1$ is a close approximation of the original image Y_1, then the reconstructed signal

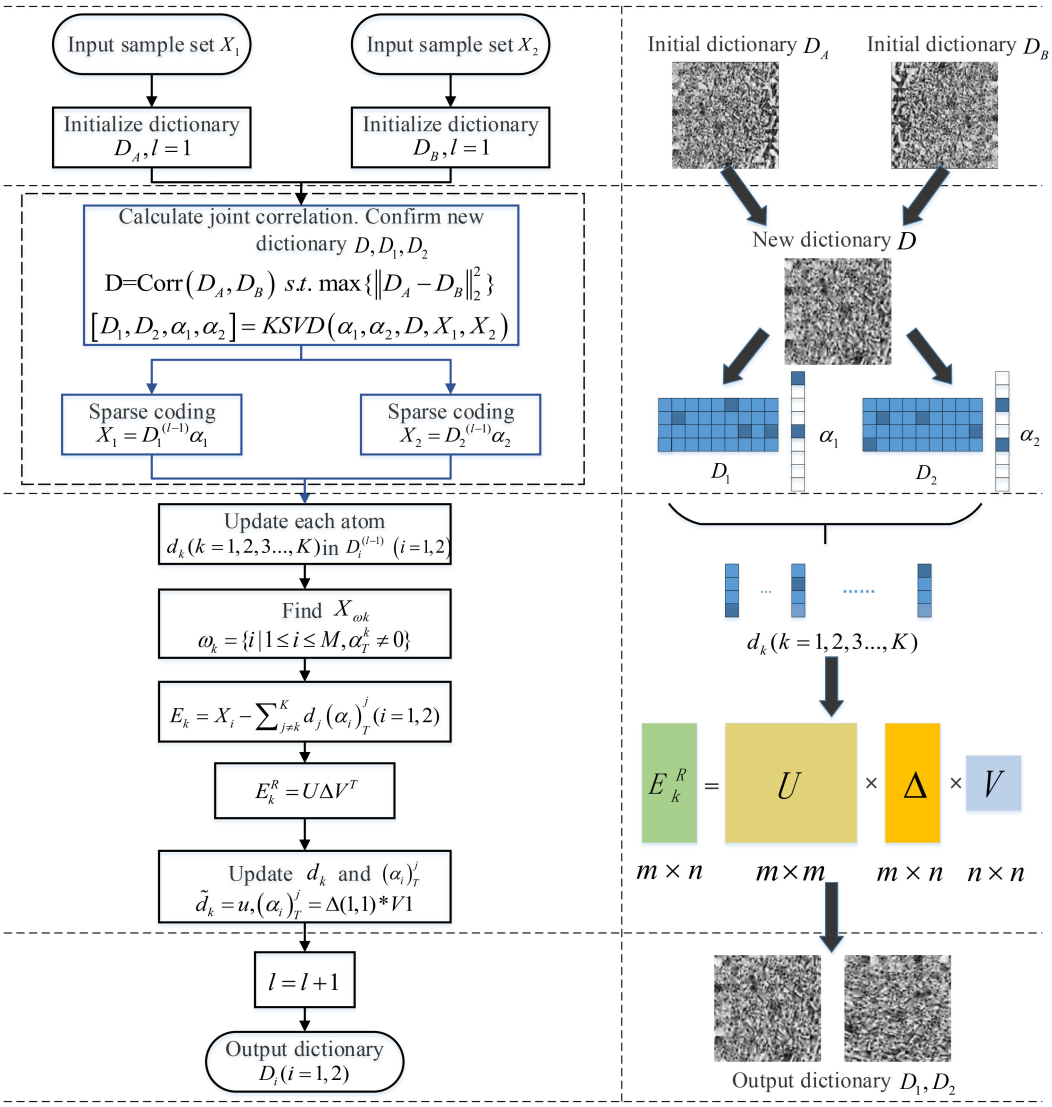

Fig. 3. Flowchart of the JR-KSVD algorithm.

$X_1'' = D_1\alpha_2$ is an unmodified image based on the original image Y_1. The difference between the two is the changed area of the image. Similarly, $X_2' = D_2\alpha_2$ is a close approximation of the original image Y_2, and $X_2'' = D_2\alpha_1$ can be considered an unchanged image based on the original image Y_2. The difference maps of the two photos can then be represented using both $X_1' - X_1'' = D_1\alpha_1 - D_1\alpha_2$ and $X_2' - X_2'' = D_2\alpha_2 - D_2\alpha_1$. To reduce the interference caused by random mistakes, the difference images DI_image are represented by the average of the two

$$\text{DI\_image} = \frac{(X_1' - X_1'') + (X_2' - X_2'')}{2}. \tag{10}$$

Because the difference map obtained by subtraction can better describe the variation region, while the ratio map obtained by the ratio can better suppress disturbances in the invariant region, we can obtain the ratio signals $\frac{X_1'}{X_1''} = D_1\alpha_1/D_1\alpha_2$ and $\frac{X_2'}{X_2''} = D_2\alpha_2/D_2\alpha_1$ as well as the corresponding ratio map DR_image for the reconstructed signal

$$\text{DR\_image} = \frac{\left(\frac{X_1'}{X_1''}\right) + \left(\frac{X_2'}{X_2''}\right)}{2}. \tag{11}$$

B. Iterative Adaptive Threshold Optimization of Differential Images

The ultimate result of change detection is provided as a binary image, while the difference and ratio maps acquired in the previous section are grayscale images, which are reviewed in this section along with how to convert them into binary images. When converting a grayscale image to a binary map, a threshold value is used to segment the grayscale range, and the threshold value chosen has a direct impact on the detection effect. Human adjustment is frequently used to compare the changing image created by different thresholds with the true value image to find and filter the optimum threshold. This, however, necessitates prior knowledge, i.e., the truth map, and can be time-consuming.

Algorithm 2: JR-KSVD algorithm steps.

input: Original dataset $X = \{x_i\}_{i=1,2}$; Initialize the number of iterations Q and dictionary D_1 and D_2; The number of iterations $q = 1$;

output: Dictionary D and sparse code α_1, α_2;

1: Compute the joint correlation and solve the differentiated structure dictionary D from the initial dictionary with (2);
2: Sparse coding using the OMP algorithm to find the approximate solution of X with (3);
3: Find the index set $\omega_k = \{i \mid 1 \le i \le M, \alpha_T^k \ne 0\}$ that uses atom d_k to be updated;
4: Calculate the overall signal error $E_k = X - \sum_{j \ne k} d_j \alpha_T^j$ and convert it to $E_k^R = E_k \Omega_k$, where $\Omega_k(a,b) = \begin{cases} 1 & (a,b) = (\omega_k(i), i) \\ 0 & \text{else} \end{cases}$;
5: Decompose matrix E_k^R by the SVD algorithm, $E_k^{\mathrm{R}} = U \Delta V^T$;
6: Update the current atom and its corresponding sparse representation coefficient $\tilde{d}_k = U(:,1), \quad \tilde{\alpha}_R^k = V(:,1) * \Delta(1,1)$;
7: Loop through Steps 3– 6 until all atoms have been updated;
8: Loop through the operations in Steps 2–7 until the number of iterations $q = Q$.

As a result, we offer the IATO method to solve this problem. The algorithm creates a fused image based on the properties of differential and ratiometric images, applies the classic 1-D and 2-D Otsu thresholding segmentation algorithm to iteratively optimize the threshold values within a given range, and finally generates a segmented change detection image. The overall flowchart of the IATO algorithm is shown in Fig. 4.

1) Design and Segmentation of Fusion Images: In the previous section, we discussed how to obtain a difference image and a ratio image from the reconstructed signals, i.e., the DI_image and DR_image signals in the flowchart, and we can now use these two signals to build a fusion function to obtain a fused image for iterative threshold segmentation. The gray value of each pixel in a difference image represents the difference between the corresponding pixels in the two original images, and the difference range is $(-255, 255)$. The ratio range between the values of the pixels in a ratio image is $(0, +\infty)$. The absolute value of the gray value of the difference image is taken, and the larger the absolute value is, the more likely the region will change; similarly, the closer the elemental value of the ratio image is to 0 or $+\infty$, the more often the associated region will change. To take advantage of this nonlinear relationship in the ratio image for threshold selection, we constrained the logarithmic function for pixels with elemental values in the range (0,1) and the exponential function for pixels with elemental values in the range $(1, +\infty)$. This increases the grayscale value difference between the shifting and invariant regions, making threshold selection easier while lowering threshold selection sensitivity

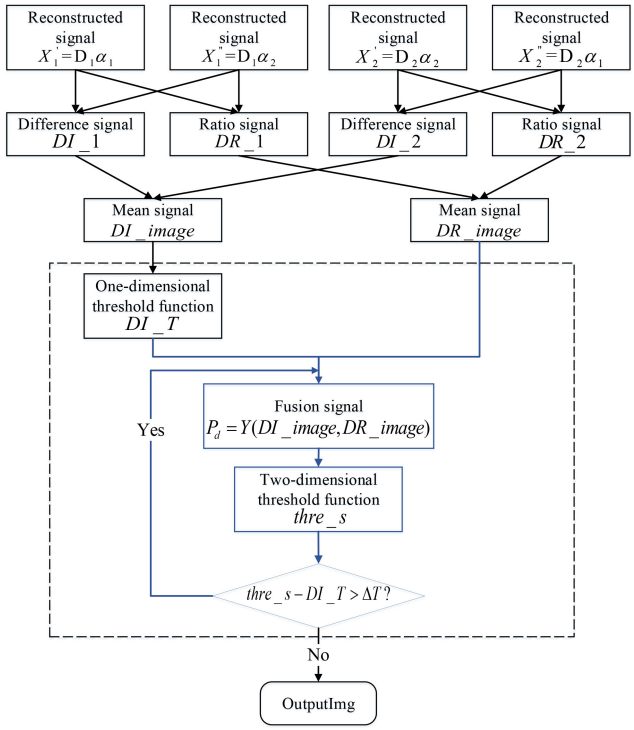

Fig. 4. IATO algorithm flowchart.

and enhancing fault tolerance. The following equations describe differential and ratiometric image processing, with DI_sig and DR_sig representing the processed differential and ratiometric signal maps, respectively

$$\mathrm{DI\_sig} = \mid \mathrm{DI\_image} \mid \tag{12}$$

$$\mathrm{DR\_sig} = \begin{cases} \mid \lg(\mathrm{DR\_image}) \mid, & \mathrm{DR\_image} \subseteq (0,1) \\ e^{\mathrm{DR\_image}-1} - 1, & \mathrm{DR\_image} \subseteq (1, +\infty). \end{cases} \tag{13}$$

As shown in Fig. 4, the differential signal DI_sig must be fed into the 1-D Otsu threshold function to obtain the initial threshold DI_T, which is the fusion function's initial parameter. For the fusion function, we first roughly categorize pixels in the differential image with gray values greater than DI_T as change regions and then execute differential-ratio fusion on pixels with gray values less than DI_T to generate a new gray value corresponding to that pixel's location. P_d denotes the fused image. The fusion function is shown as follows:

$$P_d = \mathrm{DI\_sig} \times \mathrm{DR\_sig}^3 \quad \text{s.t.} \mid P_d \mid \le 255. \tag{14}$$

2) Iterative Adaptive Threshold Optimization: The fused image generated by the fusion function is used as a new grayscale image and requires threshold selection and binarization segmentation to obtain the change detection image. To acquire the threshold for iteration and the evolving image segmented by the threshold, the first-generation fusion image is fed into the 2-D Otsu threshold segmentation method. The 2-D Otsu method and

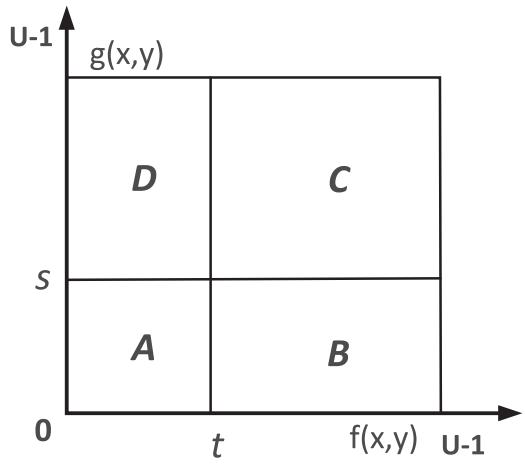

Fig. 5. Two-dimensional statistical histogram.

its implementation in the IATO algorithm are described below. The 2-D Otsu algorithm for noise suppression is more powerful than the 1-D Otsu algorithm because it considers the distribution of pixel gray values as well as the average gray values of neighboring pixels to form a 2-D vector, and the optimal threshold is set when the trace of the interclass dispersion matrix is at its maximum. Assume $f(x,y)$ is the original image's gray value for pixel (x,y), $g(x,y)$ is the gray average of pixel (x,y) after $K \times K$ mean filtering, U is the total number of gray levels, and N is the sum of all pixel points. Let $f(x,y) = i$, $g(x,y) = j$, and (i,j) refer to the number of pixel points N_{ij} in the image. The probability that the pixel point appears is

$$P_{ij} = N_{ij}/N, \sum_{i=0}^{U-1}\sum_{j=0}^{U-1} p_{ij} = 1. \tag{15}$$

Given an array (t,s) and a 2-D statistical histogram, let t be the horizontal coordinate representing a pixel's gray value, and let s be the vertical coordinate indicating the average gray value of a pixel's neighborhood. Parts A and C represent the background and foreground, respectively, in the 2-D histogram, which is divided into four parts: A, B, C, and D. The gray value of the pixel at B is significantly larger than the neighborhood's average gray value, suggesting that it is a noise point, whereas the gray value of the pixel at D is much lower, indicating that it is an edge point. Fig. 5 is a 2-D statistical histogram. The probabilities of the background and target classes occurring, along with their associated mean vectors, are ω_0, ω_1, μ_0, and μ_1. The overall grayscale mean vector of the image is μ_T, where $\omega_0 + \omega_1 \approx 1$; then

$$\omega_0 = \sum_{i=0}^{t-1}\sum_{j=0}^{s-1} p_{ij}, \omega_1 = \sum_{i=t}^{U-1}\sum_{j=s}^{U-1} p_{ij} \tag{16}$$

$$\mu_0 = (\mu_{0i}, \mu_{0j})^T = \left(\sum_{i=0}^{t-1}\sum_{j=0}^{s-1} \frac{ip_{ij}}{\omega_0}, \sum_{i=0}^{t-1}\sum_{j=0}^{s-1} \frac{jp_{ij}}{\omega_0} \right)^T \tag{17}$$

$$\mu_1 = (\mu_{1i}, \mu_{1j})^T = \left(\sum_{i=t}^{U-1}\sum_{j=s}^{U-1} \frac{ip_{ij}}{\omega_1}, \sum_{i=t}^{L-1}\sum_{j=s}^{L-1} \frac{jp_{ij}}{\omega_1} \right)^T \tag{18}$$

$$\mu_T = (\mu_{Ti}, \mu_{Tj})^T = \left(\sum_{i=0}^{L-1}\sum_{j=0}^{U-1} ip_{ij}, \sum_{i=0}^{U-1}\sum_{j=0}^{L-1} jp_{ij} \right)^T. \tag{19}$$

The interclass dispersion matrix is defined as follows:

$$\begin{aligned} V(s,t) &= \omega_0 \left[(\mu_0 - \mu_T) \cdot (\mu_0 - \mu_T)^T \right] \\ &\quad + \omega_1 \left[(\mu_1 - \mu_T) \cdot (\mu_1 - \mu_T)^T \right] \\ &= \omega_0 (\mu_0 - \mu_T)^2 + \omega_1 (\mu_1 - \mu_T)^2. \end{aligned} \tag{20}$$

Interclass dispersion is gauged by the dispersion matrix trace, and when the trace reaches its maximum value $(\hat{t}, \hat{s})$, it indicates an optimal segmentation threshold

$$\begin{aligned} \mathrm{Tr}(V(t,s)) &= \omega_0 \left((\mu_{0i} - \mu_{Ti})^2 + (\mu_{0j} - \mu_{Tj})^2 \right) \\ &\quad + \omega_1 \left((\mu_{1i} - \mu_{Ti})^2 + (\mu_{1j} - \mu_{T_j})^2 \right) \\ &= \frac{(\mu_i - \omega_0 \mu_{Ti})^2 + (\mu_j - \omega_0 \mu_{Tj})^2}{\omega_0 (1 - \omega_0)} \end{aligned} \tag{21}$$

$$\mathrm{Tr}(V(\hat{t},\hat{s})) = \max(\mathrm{Tr}(V(t,s))), 0 \le s \le U-1, 0 \le t \le U-1. \tag{22}$$

In general, the threshold values $\hat{t}$ and $\hat{s}$ are near each other and can be used as the selected thresholds for image segmentation. The threshold value $\hat{t}$ indicates the gray value of a pixel at a location, and the threshold value $\hat{s}$ represents the gray value in the vicinity of that pixel. We stretch and normalize the values of the 2-D Otsu threshold segmentation function using the fused image P_d created in the previous section, which stretches the pixel values of the points that were originally considered coarse change regions and then normalizes all of the pixel values to the interval range of $(-255, 255)$. Following the calculation of the 2-D thresholding function, we construct a new threshold that may differ from the initial threshold discussed in the previous section, where a larger difference indicates a higher risk of false positives (FPs) in image segmentation. Thus, the new threshold is utilized as an input to the fusion function to cycle the segmentation once more, and the threshold change range is gradually reduced until it is stable. The change detection image obtained by using stabilized threshold segmentation is the final result map.

III. Experimental Results and Discussion

A. Dataset Description

A total of six sets of dual-temporal SAR images were used in the experimental part of this study, and preprocessing, such as alignment and correction, was performed before testing. We organized the six sets of experimental data into a table for convenience, as shown in Table II. Figs. 6–11 show these six sets of experimental data, where (a) and (b) are images taken at

TABLE II
DATASET DESCRIPTION

Dataset	Sensor	Time1	Time2	Pixel
Ottawa	Radarsat-1	July 1997	August 1998	290 × 350
Bern	ERS-2	April 1999	May 1999	301 × 301
Yellow River	Radarsat-2	June 2008	June 2009	256 × 288
Mexico	Landsat-7	April 2000	May 2002	512 × 512
Guangdong	TerraSAR-X	May 2010	December 2010	1180 × 1080
Shanghai	TerraSAR-X	July 2020	August 2020	1050 × 1050

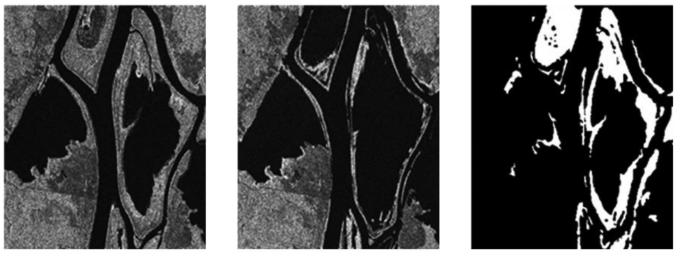

Fig. 6. Dual time (a), (b) phase images and (c) the reference image from the Ottawa area. (a) In July 1997. (b) In August 1998. (c) Ground truth.

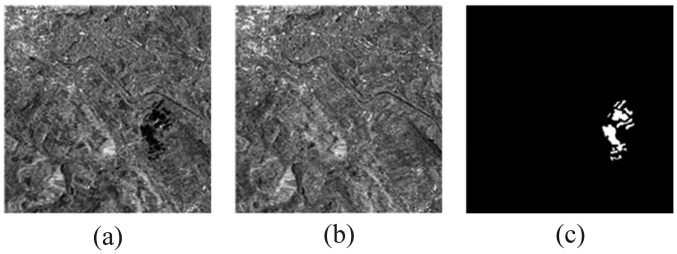

Fig. 7. Dual time (a), (b) phase images and (c) the reference image from the Bern area. (a) In April 1999. (b) In May 1999. (c) Ground truth.

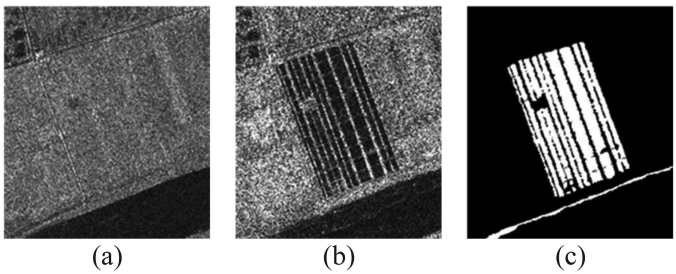

Fig. 8. Dual time (a), (b) phase images and (c) the reference image from the Yellow River area. (a) In June 2008. (b) June 2009. (c) Ground truth.

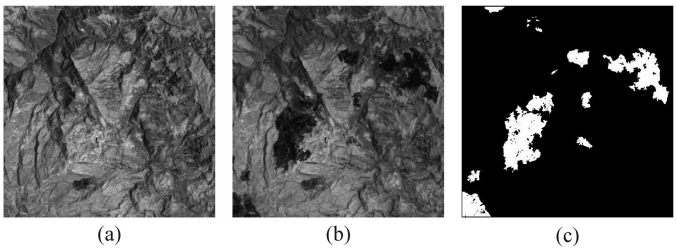

Fig. 9. Dual time (a), (b) phase images and (c) the reference image from the Mexico area. (a) In April 2002. (b) June 2003. (c) Ground truth.

Fig. 10. Dual time (a), (b) phase images and (c) the reference image from the GuangDong area. (a) In May 2010. (b) In December 2010. (c) Ground truth.

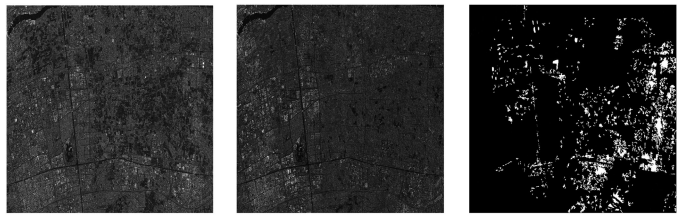

Fig. 11. Dual time (a), (b) phase images and (c) the reference image from the Shanghai area. (a) In July 2020. (b) In August 2020. (c) Ground truth.

TABLE III
RELATIONSHIPS BETWEEN THE DETECTION ACCURACY AND PARAMETERS R AND S

Dataset	r	s	PCC	Kappa	Time (s)
Ottawa	3	1	0.9815	0.9298	2,952
	4	2	0.9814	0.9294	209
	5	1	0.9698	0.8892	2,876
	6	2	0.9521	0.8151	213
Yellow River	3	1	0.9650	0.8735	1,521
	4	2	0.9647	0.8730	116
	5	1	0.9557	0.8505	1,362
	6	2	0.9401	0.7864	117

different periods, and (c) is the corresponding change reference image [ground truth (GT) image].

B. Parameter Settings and Evaluation Metrics

1) Analysis of Parameters: There are five adjustable parameters in this experiment, namely, the sliding window size r (atomic feature number r^2), the sliding distance s, the number of atoms (dictionary size) K, the number of iterations, and the coefficient of cooperation, of which r and s are the most important factors affecting the detection effect. Therefore, in this article, these two parameters are tested to select the value with the best-combined effect. To ensure the suitability of the sliding window and sliding distance, the experiments are set to have both odd and even numbers. The detection findings and times for the four sets of sliding windows and sliding distances corresponding to $r = 3$ $s = 1, r = 4$ $s = 2, r = 5$ $s = 1$, and $r = 6$ $s = 2$ on the Ottawa and Yellow River datasets are shown in Table III.

The table shows that the detection effect decreases as the sliding window and sliding distance increase. Although the detection accuracy is slightly lower when $r = 4$ and $s = 2$ is used instead of $r = 3$ and $s = 1$, the time consumed is substantially reduced, even less than one-tenth of the latter. Taking both time and accuracy into account, this article recommends $r = 4$ and

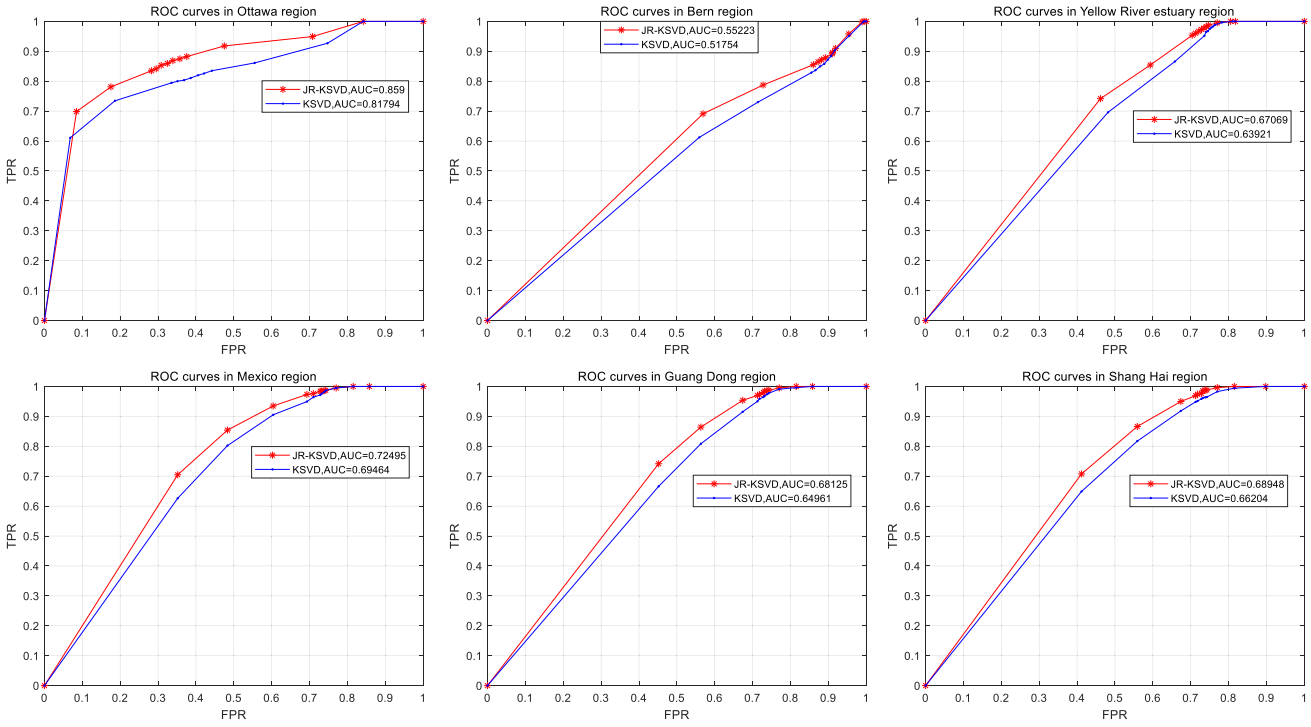

Fig. 12. ROC curves for the six datasets.

$s = 2$. For the dictionary size K, if it is set too small, such as $K = 128$, it cannot capture all the features in the image, resulting in a significant reduction in the detection of edges and details; if it is set too large, such as $K = 512$, feature redundancy becomes a problem, and the detection time greatly increases. Thus, the dictionary size is set to $K = 256$ in this experiment. The number of iterations and the coefficient of cooperation are adjusted at 1 and 5, respectively, based on experience from a prior study to balance the detection accuracy and detection time.

2) Evaluation Metrics: The false negative number (FN), FP, total number of mistakes (OE), percentage correct classification (PCC), and kappa coefficient are all evaluation indices for the SAR image change detection performance.

1) The FN metric indicates the number of samples that are judged to be negative but are in fact positive.
2) The FP metric indicates the number of samples that are judged to be positive but are in fact negative.
3) The OE metric indicates the total number of pixels with detection errors, i.e., the sum of FN and FP.

$$OE = FP + FN. \tag{23}$$

4) The PCC metric shows the ratio of the number of correctly detected pixels to the total number of pixels, where N represents the total number of pixels; i.e.,

$$\text{PCC} = \frac{TP + TN}{TP + FP + TN + FN} = \frac{TP + TN}{N}. \tag{24}$$

5) The kappa coefficient is a consistency test and a measurement of classification accuracy that is usually bounded between $(-1, 1)$ or (0,1). The better the classification accuracy is, the closer the kappa coefficient is to 1. The kappa coefficient is computed as follows: ($Nu =$ number of true unaltered pixels and $Nc =$ number of true changed pixels)

$$Nu = FP + TN, Nc = FN + TP \tag{25}$$

$$\text{PRE} = \frac{(TP + FP)Nc + (FN + TN)Nu}{N^2} \tag{26}$$

$$\text{Kappa} = \frac{\text{PCC} - \text{PRE}}{1 - \text{PRE}}. \tag{27}$$

C. Comparison and Discussion of the Experimental Results

For the change detection of SAR images, our work primarily presents the JR-KSVD algorithm for dictionary initialization and the IATO algorithm for threshold selection. The effects of these two factors on the final detection findings are discussed as follows, along with the experimental result graphs and data comparison tables to illustrate the efficacy of the approach described in this work.

1) Analysis of the JR-KSVD Ablation Experiment: First, we compare and describe the performance of the K-SVD and JR-KSVD methods at the same threshold using receiver operating characteristic (ROC) curves. The higher the detection accuracy is, the closer the area under the curve (AUC) of the ROC curve is to 1. As shown in Fig. 12, the AUCs for the JR-KSVD method, represented by the red circle, are all greater than those for the K-SVD algorithm, represented by the blue circle, indicating that JR-KSVD has better detection performance at the same threshold setting.

In addition, we verify the performance of the JR-KSVD algorithm from another aspect and briefly illustrate it using the

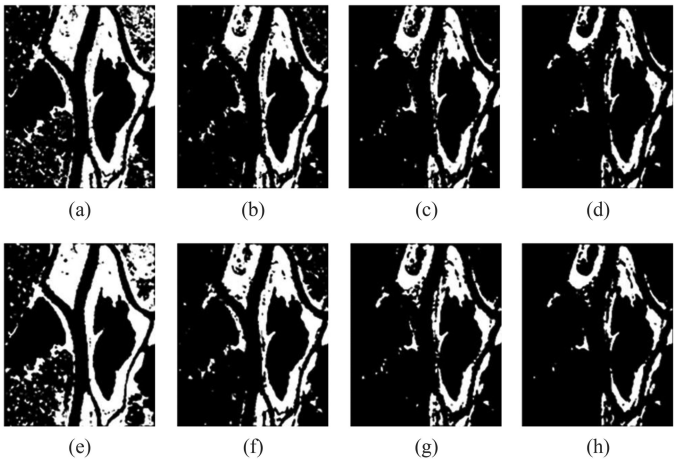

Fig. 13. (a)–(d) Image segmentation results corresponding to the JR-KSVD algorithm. (e)–(h) Image segmentation results corresponding to the K-SVD algorithm. (The thresholds set are 100, 150, 200, and 250.).

TABLE IV
CHANGE DETECTION RESULTS OF JR-KSVD AND K-SVD

Methods	Ottawa		Bern	
	PCC(%)	Kappa(%)	PCC(%)	Kappa(%)
JR-KSVD	98.14	92.94	99.67	87.21
KSVD	97.86	92.72	99.41	87.03

Ottawa region as an example. Fig. 13 shows the reconstructed images obtained using the JR-KSVD and K-SVD algorithms and thresholding them to detect images (segmentation thresholds of 100, 150, 200, and 250), where the two images in each column correspond to the same threshold. We can see that when the threshold value is in the (100, 150) interval, the region that is mistaken for a changing area is smaller when detected by the JR-KSVD algorithm than that detected by the K-SVD method, and the spot noise is also less. The area detected by the JR-KSVD algorithm is broader when the threshold value is in the interval of (150, 250), suggesting that the detection effect is better for some edge points or areas with tiny range changes, and the area image is closer to the real changing image. Therefore, we infer that the true segmentation threshold of the Ottawa area should be approximately from 100 to 150 based on the above four groups of comparison images.

Next, we discuss the effectiveness of the JR-KSVD algorithm. As demonstrated in Table IV, the JR-KSVD algorithm improves PCC on the Ottawa and Bern datasets by 0.28% and 0.26%, respectively, whereas kappa values improve to varying degrees. This indicates that the JR-KSVD algorithm, which improves on the K-SVD technique, performs better in detecting SAR image changes.

2) Analysis of the IATO Ablation Experiment: Now, we discuss how well the IATO algorithm works. The IATO method described in this research may be used to establish segmentation thresholds and, eventually, modify images without human intervention. The correct segmentation threshold for the Ottawa region, based on the foregoing, should be in the range of (100,150). To test the IATO algorithm's performance, we set

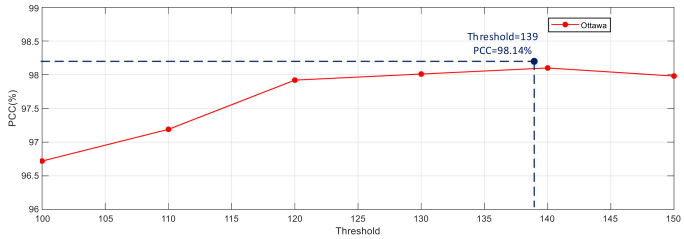

Fig. 14. Relationship between the PCC and threshold on the Ottawa dataset.

the segmentation thresholds to 100, 110, 120, 130, 140, and 150 to ascertain the real threshold values.

As shown in Fig. 14, the PCC corresponding to the threshold value set to 140 is 98.10%, which is the closest to the detection result of the IATO algorithm whose segmentation threshold is 139 and the PCC is 98.14%, with almost no difference between the two, demonstrating the reasonableness and accuracy of the threshold value generated by the IATO automation.

3) Analysis and Discussion of All Experimental Results: A binarized image is the end result of change detection; hence, an appropriate threshold for segmenting the sparsely represented image should be selected, as the threshold selection accuracy has a direct impact on the change detection results. In this section, the IATO algorithm results based on the JR-KSVD algorithm are shown, and various techniques of change detection are compared to assess their benefits and drawbacks. A K-means-based change detection algorithm for SAR images (K-means) [39], [40], change detection method based on C-means clustering of trimmed fuzzy local information (RFLICM) [41], sparse autoencoder (SAE) and fuzzy c-means (FCM) clustering change detection algorithm (SAEFCM) [42], scale invariant feature transform (SIFT) feature point-based change detection algorithm (SIFT) [43], robust unsupervised small area change detection from SAR imagery using deep learning (RUSACD) [44], SAR image change detection via a Siamese adaptive fusion network (SAFNet) [45], automatic change detection in synthetic aperture radar images based on a PCA network (PCANet) (GaborPCANet) [46], and change detection from SAR images based on neighborhood-based ratio and extreme learning machine (NRELM) [47] are compared with the methods proposed in this article.

In this experiment, the six datasets are divided into two categories according to their size, where images 6–8 represent the smaller pixel datasets and are also the three most commonly used experimental datasets in the change detection field, which we call Class A datasets, while images 9–11 represent the larger pixel datasets and are also the newer datasets, which we call Class B datasets. To reflect the experimental rigor, we performed eight sets of comparative experiments on the Class A dataset, and the experimental methods cover different types of supervised and unsupervised learning. Based on this, we selected four groups of better algorithms evaluated on the Class B dataset to verify the strengths, weaknesses, and robustness of the algorithms. To show the detection performance of different algorithms more intuitively, the detection images with better results in the comparison experiments are compared with the

TABLE V
COMPARISON OF PERFORMANCE INDICATORS IN THE OTTAWA AREA

Method	FN	FP	OE	PCC	Kappa
K-means	2,682	272	2,954	0.9709	0.8884
RFLICM	2,530	**166**	2,696	0.9734	0.8939
SAEFCM	1,410	1,044	2,454	0.9760	0.9109
SIFT	3,437	612	4,049	0.9601	0.8465
RUSACD	5,213	240	5,453	0.9659	0.8611
SAFNet	**834**	819	**1,653**	**0.9837**	**0.9388**
GaborPCANet	984	996	1,980	0.9812	0.9252
NRELM	1,092	812	1,904	0.9813	0.9264
Proposed	1,168	719	1,887	0.9814	0.9294

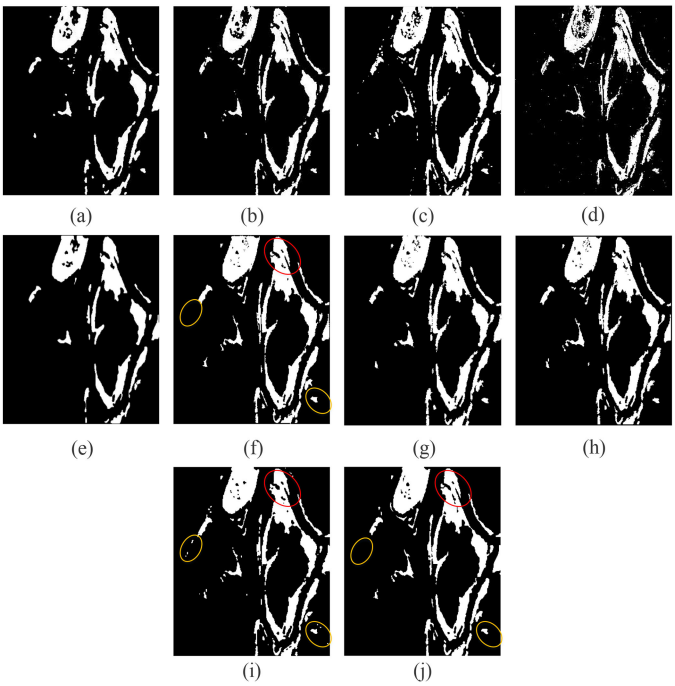

Fig. 15. Change detection results on the Ottawa dataset. (a) K-means. (b) RFLICM. (c) SAEFCM. (d) SIFT. (e) RUSACD. (f) SAFNet. (g) GaborPCANet. (h) NRELM. (i) Proposed. (j) GT.

method proposed in this article and the GT in detail, where the red circles outline the places with large discrepancies from the GT, and the yellow circles outline the noise points.

For the Class A dataset, we discuss the detection results for the Ottawa, Bern, and Yellow River regions. The detection metrics of the nine experimental methods in the Ottawa region are given in Table V. From the table, we can see that the SAFNet [Fig. 15(f)] algorithm corresponds to the best detection metrics, i.e., it is the closest to the GT, and the detection effect has been improved compared with the algorithm proposed in this article. From Fig. 15, we can see that the red circle in Fig. 15(f) is closer to the GT map than that in Fig. 15(i), with better detail and edge processing; in addition, the yellow circle in Fig.15(i) has more speckle noise than that in Fig. 15(f), which is not found in Fig. 15(j). As a result of these two findings, the detection results of the proposed algorithm are somewhat worse than those of the SAFNet algorithm, but compared with the remaining seven algorithms, our algorithm still has some advantages.

Similarly, the detection metrics of nine experimental methods in the Bern region are given in Table VI, from which it can be seen

TABLE VI
COMPARISON OF PERFORMANCE INDICATORS IN THE BERN AREA

Method	FN	FP	OE	PCC	Kappa
K-means	482	**9**	491	0.9946	0.7691
RFLICM	257	113	370	0.9951	0.8009
SAEFCM	171	145	316	0.9961	0.8530
SIFT	237	371	605	0.9933	0.7398
RUSACD	402	93	495	0.9945	0.7499
SAFNet	**110**	259	369	0.9959	0.8478
GaborPCANet	496	23	519	0.9942	0.7148
NRELM	177	128	305	0.9965	0.8634
Proposed	173	124	**297**	**0.9967**	**0.8721**

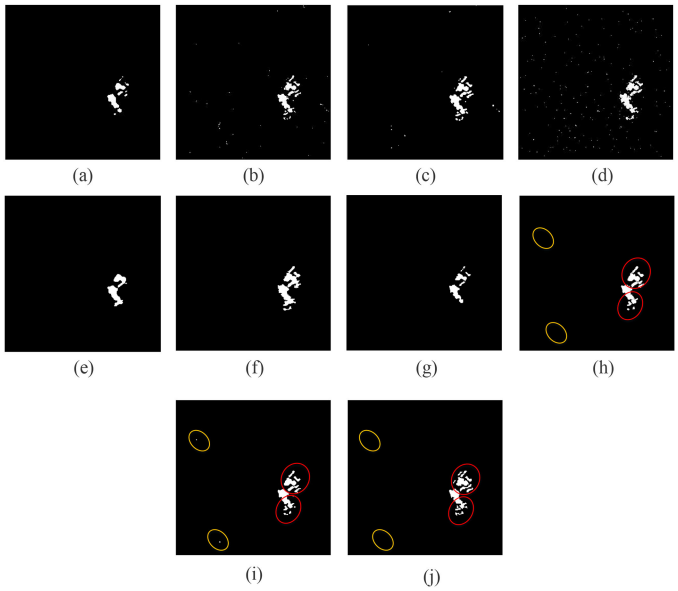

Fig. 16. Change detection results on the Bern dataset. (a) K-means. (b) RFLICM. (c) SAEFCM. (d) SIFT. (e) RUSACD. (f) SAFNet. (g) GaborPCANet. (h) NRELM. (i) Proposed. (j) GT.

TABLE VII
COMPARISON OF PERFORMANCE INDICATORS IN THE YELLOW RIVER AREA

Method	FN	FP	OE	PCC	Kappa
K-means	3,498	1,688	5,186	0.9302	0.7601
RFLICM	1,335	9,765	11,100	0.8506	0.6766
SAEFCM	11,863	840	12,703	0.8290	0.5626
SIFT	5,029	**5**	5,034	0.9322	0.7322
RUSACD	2,292	181	2,473	0.9381	0.7677
SAFNet	3,245	602	3,847	0.9478	0.8102
GaborPCANet	1,707	1,842	3,549	0.9519	0.8390
NRELM	3,687	755	4,442	0.9398	0.7789
Proposed	**1,038**	1,809	**2,847**	**0.9646**	**0.8730**

that the NRELM [Fig. 16(h)] algorithm has the best detection results, except for the algorithm proposed in this article. Graph 16 shows that the red circle in Fig. 16(h) is more similar to that in Fig. 16(i), but the edge in Fig. 16(i) is closer to the GT map; however, it is apparent that there is some speckle noise in the yellow circle in Fig. 16(i) but not in Fig. 16(h) and (j). Combining both of these findings, the speckle noise has less influence on the overall result; thus, the detection result of this article's algorithm is slightly improved over that of the NRELM algorithm.

Finally, we analyze the third set of data in the Class A dataset. The nine detection results on the Yellow River dataset are given in Table VII, from which the best two sets of experimental results

TABLE VIII
COMPARISON OF PERFORMANCE INDICATORS ON THE CLASS B DATASET

Dataset	Method	FN	FP	OE	PCC	Kappa
Mexico	RUSACD	6,642	**402**	7,044	0.9731	0.8289
	SAFNet	**1,495**	7,504	11,100	0.9575	0.7199
	GaborPCANet	4,918	699	5,617	0.9786	0.8687
	NRELM	2,620	1,217	3,837	0.9854	0.9148
	Proposed	1,607	1,696	**3,033**	**0.9870**	**0.9152**
Guangdong	RUSACD	20,426	12,119	32,545	0.9757	0.7082
	SAFNet	23,942	9,902	33,844	0.9734	0.6819
	GaborPCANet	1,749	13,611	15,360	0.9767	0.8701
	NRELM	29,969	**7,668**	37,637	0.9705	0.6196
	Proposed	**1,578**	11,154	**12,741**	**0.9788**	**0.8714**
Shanghai	RUSACD	31,601	25,596	67,197	0.9209	0.6585
	SAFNet	7,698	54,684	62,382	0.9434	0.7643
	GaborPCANet	10,638	103,841	114,479	0.8599	0.5055
	NRELM	23,924	68,245	92,169	0.9164	0.5979
	Proposed	**6537**	**20243**	**36600**	**0.9601**	**0.8419**

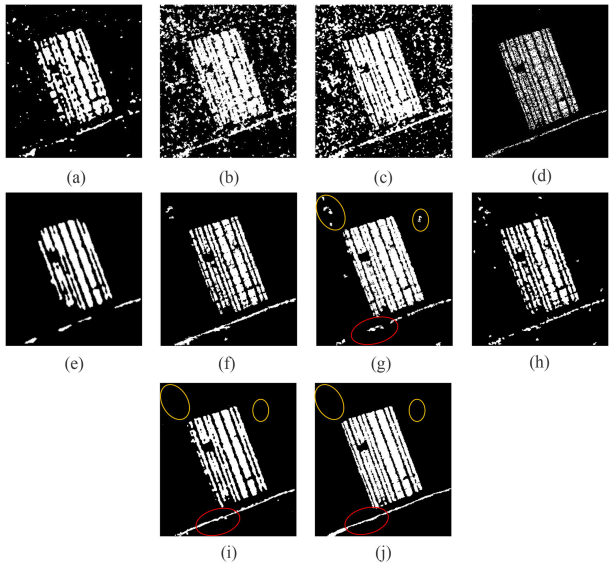

Fig. 17. Change detection results on the Yellow River dataset. (a) K-means. (b) RFLICM. (c) SAEFCM. (d) SIFT. (e) RUSACD. (f) SAFNet. (g) GaborPCANet. (h) NRELM. (i) Proposed. (j) GT.

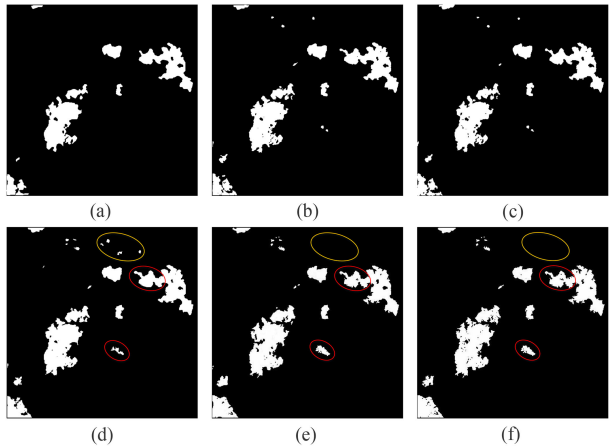

Fig. 18. Change detection results on the Mexico dataset. (a) RUSACD. (b) SAFNet. (c) GaborPCANet. (d) NRELM. (e) Proposed. (f) GT.

correspond to the GaborPCANet [Fig. 17(g)] algorithm and the proposed [Fig. 17(i)] algorithm in this article. Observing Fig. 17(g) and (i), we find that the red circle in Fig. 17(g) is disconnected, while it is continuous in Fig. 17(i) and (j); additionally, there is more speckle noise in the yellow circle part in Fig. 17(g), while there is almost nothing in Fig. 17(i). Thus, based on the advantages of these two aspects, the algorithm in this article has a relatively large improvement over the GaborPCANet algorithm.

Combining the eight comparison experiments for the above three datasets, we find that the better detection algorithms corresponding to each dataset are different, which indicates that the algorithms only have superior results for one or a few datasets. However, in general, the detection results of the RUSACD, SAFNet, GaborPCANet, and NRELM algorithms are better than those of the other four algorithms. Therefore, we use these four sets of comparative experiments to conduct experimental tests on the Class B dataset for further comparison with the algorithm proposed in the article.

Figs. 18–20 are the detection results and GT maps of the Class B dataset under five experiments, and Table VIII shows the detection indexes that correspond to them. As seen in the table, in the Mexico region, the NRELM [Fig. 18(d)] algorithm corresponds to a better detection effect and is closest to the detection index of this article. The main shortcomings are reflected in the lack of clear texture detection in the red circle in the figure and the higher amount of speckle noise in the yellow circle. For the detection results in Guangdong and Shanghai, we can see in Figs. 19 and 20 that both have the problem of "more change areas, smaller areas, and scattered distributions," which lead some algorithms to mistake smaller change areas as speckles [Figs. 19(a) and 20(a)] or to detect smaller unchanged areas as changed areas together with the changed areas around them, resulting in patches of changed areas [e.g., Fig. 20(c) and (d)]. However, similar to the Class A dataset, the corresponding superior detection algorithms (other

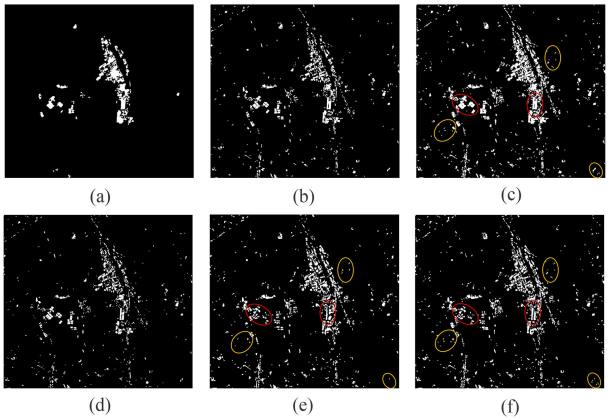

Fig. 19. Change detection results on the Guangdong dataset. (a) RUSACD. (b) SAFNet. (c) GaborPCANet. (d) NRELM. (e) Proposed. (f) GT.

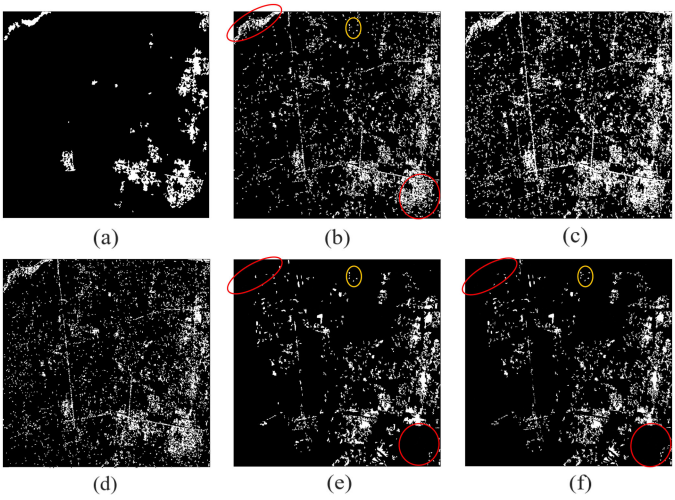

Fig. 20. Change detection results on the Shanghai dataset. (a) RUSACD. (b) SAFNet. (c) GaborPCANet. (d) NRELM. (e) Proposed. (f) GT.

than the algorithm in this article) are different for the Guangdong and Shanghai regions, as indicated by the GaborPCANet algorithm and the SAFNet algorithm, respectively, but both have slight shortcomings compared to the algorithm proposed herein.

Combining the above two types of datasets, A and B, the algorithm proposed in this article has a better detection effect and robustness in the rest of the datasets, except that its detection effect on the Ottawa dataset is slightly less than that of the SAFNet algorithm, which also indicates that the proposed algorithm has a more stable detection effect for SAR images of different sizes and compositions and is more universal.

4) Computational Complexity Comparisons: Finally, we discuss the time consumed by each technique and utilize the GaborPCANet approach, which has received much attention in recent years, as a baseline for comparison. The detection times of the algorithm proposed in this research and the other four algorithms are shown in Table IX, and the findings show that the GaborPCANet approach takes the longest time but has higher accuracy. This is because the PCANet and later the SVM classifier are trained using more samples. The NRELM method, on the other hand, is less effective in detection despite taking the least amount of time due to its lack of training. Overall, the unsupervised learning algorithm described in this study outperforms better performance in terms of both detection accuracy and detection time.

TABLE IX
COMPUTING TIME OF DIFFERENT METHODS ON OTTAWA, BERN AND YELLOW RIVER DATASETS

Method	Runtime(s)		
	Ottawa	Bern	Yellow River
GaborPCANet [46]	2974.61	1901.73	2899.18
RUSACD [44]	890.78	458.53	508.63
SAFNet [45]	1120.35	843.10	799.24
NRELM [47]	11.71	6.79	5.23
Proposed	209.12	171.26	117.58

IV. CONCLUSION

We proposed the JR-KSVD algorithm to increase the quality of sparse representation and the IATO technique to automatically obtain the best segmentation thresholds in our investigation of SAR image change detection, and we integrated these two techniques with cross-reconstruction theory. The JR-KSVD algorithm, for example, focuses on improving the dictionary used in the K-SVD algorithm's initial iteration, which reduces the instability of sparse coding caused by the high correlation of atoms and sharpens the edges and details of reconstructed images. In addition, the IATO algorithm, through several iterations of the reconstructed image, continuously reduces the threshold range, while still retaining the accuracy of the reconstructed image and finally binarizes the reconstructed map by the automatically generated threshold to obtain a detection result map close to the GT. From the experimental results, it can be concluded that with our algorithm, increased detection accuracy and consistency results are achieved when compared to the results of previous algorithms, and our results are also less affected by speckle noise. Although this study obtained strong results in SAR image change detection, it still has several flaws, the most notable of which are as follows.

1) The JR-KSVD algorithm suggested in this article is an improved version of the K-SVD algorithm, but its efficiency is not much improved. Since learning and training a new initialization dictionary for each sample set is still needed, the JR-KSVD algorithm does not improve the detection rate. However, it decreases the detection rate to a certain extent when the image to be detected is quite large.
2) The IATO algorithm achieves accurate thresholding by continuously generating thresholds and constraining the reconstructed image. However, for each iteration, an image constraint and a threshold estimation are performed. As a result, the iterative process is relatively tedious, and the detection accuracy decreases with the increase in the number of iterations.
3) Due to the lack of a large quantity of experimental data, only three common datasets and three new datasets were tested in this article for comparison experiments.

For future research, we believe that we can start by improving the efficiency of threshold iteration, continuously optimizing the threshold constraint function, and reducing the number of iterations as much as possible, greatly shortening the time used for image segmentation. Although the algorithm proposed in this article still has much improvement potential, it nevertheless succeeds at reasonably improving the quality of SAR image change detection and motivates some suggestions and possible research directions for future research in the area of unsupervised change detection.

REFERENCES

[1] Y. K. Chan and V. Koo, "An introduction to synthetic aperture radar (SAR)," *Prog. Electromagnetics Res. B*, vol. 2, pp. 27–60, 2008.

[2] P. Berens, "Introduction to synthetic aperture radar (SAR)," in *Proc. Adv. Radar Signal Data Process.*, 2006, pp. 3-1–3-14.

[3] A. Mian, G. Ginolhac, J.-P. Ovarlez, and A. M. Atto, "New robust statistics for change detection in time series of multivariate SAR images," *IEEE Trans. Signal Process.*, vol. 67, no. 2, pp. 520–534, Jan. 2019.

[4] D. Ciuonzo, V. Carotenuto, and A. De Maio, "On multiple covariance equality testing with application to SAR change detection," *IEEE Trans. Signal Process.*, vol. 65, no. 19, pp. 5078–5091, Oct. 2017.

[5] P. Addabbo, F. Biondi, C. Clemente, D. Orlando, and L. Pallotta, "Classification of covariance matrix eigenvalues in polarimetric SAR for environmental monitoring applications," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 34, no. 6, pp. 28–43, Jun. 2019.

[6] B. Brisco, M. Mahdianpari, and F. Mohammadimanesh, "Hybrid compact polarimetric SAR for environmental monitoring with the radarsat constellation mission," *Remote Sens.*, vol. 12, no. 20, 2020, Art. no. 3283.

[7] A. Garioud, S. Valero, S. Giordano, and C. Mallet, "On the joint exploitation of optical and SAR imagery for Grassland monitoring," *Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. XLIII-B3-2020, pp. 591–598, 2020.

[8] L. White, B. Brisco, M. Dabboor, A. Schmitt, and A. Pratt, "A collection of SAR methodologies for monitoring wetlands," *Remote Sens.*, vol. 7, no. 6, pp. 7615–7645, 2015.

[9] S. Wdowinski, S.-W. Kim, F. Amelung, T. H. Dixon, F. Miralles-Wilhelm, and R. Sonenshein, "Space-based detection of wetlands' surface water level changes from l-band SAR interferometry," *Remote Sens. Environ.*, vol. 112, no. 3, pp. 681–696, 2008.

[10] C. Zhou, H. Gong, Y. Zhang, T. A. Warner, and C. Wang, "Spatiotemporal evolution of land subsidence in the Beijing plain 2003–2015 using persistent scatterer interferometry (PSI) with multi-source SAR data," *Remote Sens.*, vol. 10, no. 4, 2018, Art. no. 552.

[11] L. Du, H. Dai, Y. Wang, W. Xie, and Z. Wang, "Target discrimination based on weakly supervised learning for high-resolution SAR images in complex scenes," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 461–472, Jan. 2020.

[12] X. Huang, J. Ding, and Q. Guo, "Unsupervised image registration for video SAR," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1075–1083, 2020.

[13] S. Chen and H. Wang, "SAR target recognition based on deep learning," in *Proc. Int. Conf. Data Sci. Adv. Anal.*, 2014, pp. 541–547.

[14] C. Wang, W. Su, and H. Gu, "A joint change detection method on complex-valued polarimetric synthetic aperture radar images based on feature fusion and similarity learning," *Int. J. Remote Sens.*, vol. 42, no. 13, pp. 4864–4881, 2021.

[15] L. Li *et al.*, "Deformable dictionary learning for SAR image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4605–4617, Aug. 2018.

[16] M. Gong, J. Zhao, J. Liu, Q. Miao, and L. Jiao, "Change detection in synthetic aperture radar images based on deep neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 125–138, Jan. 2016.

[17] M. Elsaadouny, J. Barowski, and I. Rolfes, "Unsupervised learning implementation for SAR images clustering," in *Proc. Int. Conf. Electromagnetics Adv. Appl.*, 2021, pp. 104–104.

[18] M. Majidi, S. Ahmadi, and R. Shah-Hosseini, "A saliency-guided neighbourhood ratio model for automatic change detection of SAR images," *Int. J. Remote Sens.*, vol. 41, no. 24, pp. 9606–9627, 2020.

[19] S. Saha, F. Bovolo, and L. Bruzzone, "Building change detection in VHR SAR images via unsupervised deep transcoding," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 1917–1929, Mar. 2020.

[20] J. Geng, X. Ma, X. Zhou, and H. Wang, "Saliency-guided deep neural networks for SAR image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7365–7377, 2019.

[21] H. Choi and J. Jeong, "Speckle noise reduction technique for SAR images using statistical characteristics of speckle noise and discrete wavelet transform," *Remote Sens.*, vol. 11, no. 10, 2019, Art. no. 1184.

[22] J. Chen, L. Jiao, and Z. Wen, "High-level feature selection with dictionary learning for unsupervised SAR imagery terrain classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 1, pp. 145–160, Jan. 2017.

[23] S. Liu, G. Zhang, and W. Liu, "Group sparse representation based dictionary learning for SAR image despeckling," *IEEE Access*, vol. 7, pp. 30809–30817, 2019.

[24] Y. Zhan, T. Dai, J. Huang, Y. Dong, F. Ye, and C. Tang, "Double iterative optimal dictionary learning-based SAR image filtering method," *Signal, Image, Video Process.*, vol. 12, no. 4, pp. 783–790, 2018.

[25] M. M. S. Youssef, C. Mallet, N. Chehata, A. Le Bris, and A. Gressin, "Combining top-down and bottom-up approaches for building detection in a single very high resolution satellite image," in *Proc. IEEE Geosci. Remote Sens. Symp.*, 2014, pp. 4820–4823.

[26] Z. Qu and L. Zhang, "Research on image segmentation based on the improved OTSU algorithm," in *Proc. 2nd Int. Conf. Intell. Hum.- Mach. Syst. Cybern.*, 2010, vol. 2, pp. 228–231.

[27] I. Tošić and P. Frossard, "Dictionary learning," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 27–38, Mar. 2011.

[28] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[29] Y. Yang *et al.*, "Multi-focus image fusion via clustering PCA based joint dictionary learning," *IEEE Access*, vol. 5, pp. 16985–16997, 2017.

[30] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, 2009, pp. 689–696.

[31] C. Lu, J. Shi, and J. Jia, "Online robust dictionary learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 415–422.

[32] S. G. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[33] K. Hwang *et al.*, "OMP: A RISC-based multiprocessor using orthogonal-access memories and multiple spanning buses," in *Proc. 4th Int. Conf. Supercomput.*, 1990, pp. 7–22.

[34] S. Chen and D. Donoho, "Basis pursuit," in *Proc. 28th Asilomar Conf. Signals, Syst, Comput.*, vol. 1, 1994, pp. 41–44.

[35] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm," *IEEE Trans. Signal Process.*, vol. 45, no. 3, pp. 600–616, Mar. 1997.

[36] L. Zhou, Z. Sun, and W. Wang, "Learning to short-time fourier transform in spectrum sensing," *Phys. Commun.*, vol. 25, pp. 420–425, 2017.

[37] R. Yan, L. Shao, and Y. Liu, "Nonlocal hierarchical dictionary learning using wavelets for image denoising," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4689–4698, Dec. 2013.

[38] J. Cai, Q. Cheng, M. Peng, and Y. Song, "Fusion of infrared and visible images based on nonsubsampled contourlet transform and sparse K-SVD dictionary learning," *Infrared Phys. Technol.*, vol. 82, pp. 85–95, 2017.

[39] Y. Zheng, X. Zhang, B. Hou, and G. Liu, "Using combined difference image and k-means clustering for SAR image change detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 3, pp. 691–695, Mar. 2014.

[40] L. Jia, M. Li, P. Zhang, Y. Wu, and H. Zhu, "SAR image change detection based on multiple kernel k-means clustering with local-neighborhood information," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 6, pp. 856–860, Jun. 2016.

[41] 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.

[42] M. Gong, X. Niu, P. Zhang, and Z. Li, "Generative adversarial networks for change detection in multispectral imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2310–2314, Dec. 2017.

[43] Y. Wang, L. Du, and H. Dai, "Unsupervised SAR image change detection based on SIFT keypoints and region information," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 7, pp. 931–935, Jul. 2016.

[44] X. Zhang, H. Su, C. Zhang, X. Gu, X. Tan, and P. M. Atkinson, "Robust unsupervised small area change detection from SAR imagery using deep learning," *ISPRS J. Photogrammetry Remote Sens.*, vol. 173, pp. 79–94, 2021.
[45] Y. Gao, F. Gao, J. Dong, Q. Du, and H.-C. Li, "Synthetic aperture radar image change detection via Siamese adaptive fusion network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 10748–10760, 2021.
[46] F. Gao, J. Dong, B. Li, and Q. Xu, "Automatic change detection in synthetic aperture radar images based on PCANet," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1792–1796, Dec. 2016.
[47] F. Gao, J. Dong, B. Li, Q. Xu, and C. Xie, "Change detection from synthetic aperture radar images based on neighborhood-based ratio and extreme learning machine," *J. Appl. Remote Sens.*, vol. 10, no. 4, 2016, Art. no. 046019.

Qiuze Yu received the Ph.D. degree in electronic engineering from the Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan, China, in 2004.

From 2008 to 2012, he served as a Postdoctoral Fellow and an Associate Professor with Shanghai Jiao Tong University, Shanghai, China. In 2013, he worked as a Visiting Scholar with the California Institute of Technology, Pasadena, CA, USA. He is currently a Principle Investigator with the Laboratory for Multisensor Signal Processing and UAV Navigation, Wuhan University, Wuhan. He has authored over 40 papers in international conference proceedings and journals. His research interests include multisensor (radar-optical) signal and image processing, multisensor automatic target recognition, deep learning, and unmanned aerial vehicle navigation.

Miao Zhang received the bachelor's degree in electronic information engineering from Central South University, Changsha, China, in 2020. She is currently working toward her master's degree in information and communication engineering with the School of Electronic Information, Wuhan University, Wuhan, China.

Her research interests include SAR image change detection.

Lijie Yu received the bachelor's degree in optoelectronic information science and engineering from the Wuhan University of Technology, Wuhan, China, in 2018, and the master's degree in information and communication engineering from the Electronic Information School, Wuhan University, Wuhan, in 2020.

His research interests include SAR image registration and navigation.

Ruikai Wang received the bachelor's degree in communication engineering, in 2021, from Wuhan University, Wuhan, China, where he is currently working toward the master's degree in information and communication engineering with the School of Electronic Information.

His research interests include SAR image registration and navigation.

Jinsheng Xiao received the Ph.D. degree in computational mathematics from the School of mathematics, Wuhan University, Wuhan, China, in 2001.

From 2001 to 2004, he was a Research Associate with the Institution of Multimedia Network Communication and a Lecturer of Communication Engineering with the School of Electronic Information, from 2004 to 2008, with Wuhan University, where he has been an Associate Professor of Information and Communication Engineering, since 2008. From August 2014 to August 2015, he was the Visiting Scholar with the Department of Computer Science, University of California, Santa Barbara, CA, USA. He has authored more than 50 articles. His research interests include video and image processing and computer vision.