Received July 30, 2019, accepted August 8, 2019, date of publication August 13, 2019, date of current version September 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2934983

Extreme Self-Paced Learning Machine for On-Orbit SAR Images Change Detection

SHUYUAN YANG[1], ZHI LIU[1], QUANWEI GAO[1], YUTENG GAO[2], AND ZHIXI FENG[1]

[1]School of Artificial Intelligence, Xidian University, Xi'an, China
[2]Electrical Engineering Department, Northwestern Polytechnical University, Xi'an, China

Corresponding author: Shuyuan Yang (syyang2009@gmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61771380, Grant U1730109, Grant 91438103, Grant 61771376, Grant 61703328, Grant 91438201, Grant U1701267, and Grant 61703328, in part by the Equipment Pre-Research Project of the 13th Five-Years Plan under Grant 6140137050206, Grant 414120101026, Grant 6140312010103, Grant 6141A020223, Grant 6141B06160301, and Grant 6141B07090102, in part by the Major Research Plan in Shaanxi Province of China under Grant 2017ZDXM-GY-103 and Grant 017ZDCXL-GY-03-02, in part by the Foundation of the State Key Laboratory of CEMEE under Grant 2017K0202B and Grant 2018K0101B, and in part by the Science Basis Research Program in Shaanxi Province of China under Grant 16JK1823 and Grant 2017JM6086.

ABSTRACT With the rapid development of earth observation satellites, on-orbit data processing is becoming more and more desirable. In this paper, a new on-orbit change detection method for Synthetic Aperture Radar (SAR) images, is proposed via an Extreme Self-paced Learning Machine (ESLM). First, a reflectivity-spatial affinity is defined to measure the similarity between two segmented super-pixels, to identify the initial three groups of pixels: strictly changed, strictly unchanged and fuzzy pixels. Then a new extreme self-paced learning machine is developed, by gradually selecting the most confident changed pixels and predicting the changed pixels in an incremental pattern. Moreover, both the labeled and unlabeled samples are explored to realize semi-supervised classification. Different with the available methods, ESLM works in a self-paced learning pattern and achieves accurate detection, for it can automatically choose the training samples and explore unlabeled samples to enhance the online prediction of changes. Therefore, ESLM has the characteristics of accurate and robust detection, parameter free, low-complexity and rapid implementation, which is very suitable for on-orbit processing. Some experiments are taken on five real benchmark datasets, and the results verify the effectiveness of ESLM.

INDEX TERMS Change detection, synthetic aperture radar, extreme self-paced learning machine, affinity propagation super-pixel clustering, manifold regularizer.

I. INTRODUCTION

Change detection from multi-temporal Synthetic Aperture Radar (SAR) images aims to identify changes in the images of the same scene taken at different times [1], which has extensive applications in many civil and military fields [2]. With the rapid development of earth observation satellites, increasing amount of spaceborne SAR data are collected. Thus, it is desirable to develop on-orbit change detection techniques, which can detect changes on the aircraft and then transmit them to the ground station. Compared with the on-ground change detection, on-orbit change detection can avoid the storage, compression and transmission of a large amount of SAR data, so is potential in coping with the explosion of data.

Making an analysis on the on-orbit change detection technologies, we will find that they should have the following characteristics:

1) *High automation degree*. Different with the on-ground processing, on-orbit change detection algorithms require little involvement of human operation. In order to automatically implement on the spacecraft, on-orbit change detection methods need to have the capability of self-learning and self-evolution, since no manual involvement is allowed in the detection process.
2) *Low complexity*. The spacecraft often has limitations on the power consumption and device volume. For the low power consumption and small size limitation of aerospace platforms, on-orbit change detection algorithms should have the characteristics of simple principle, rapid processing and high efficiency.

The associate editor coordinating the review of this article and approving it for publication was Xin Luo.

3) *Robustness and flexibility*. It is well known that the on-board processing needs robust and adaptive detection algorithms that can work for SAR images with different resolutions, looks and working wavebands. Consequently, developing automated, rapid and robust on-orbit methods, is the future trend of change detection.

Having a glance over the available multi-temporal SAR images change detection approaches, we will find that they can be mainly categorized into unsupervised and supervised ones. Unsupervised methods first calculate a "Difference Map" (DM) and then locate the changes from it using some distribution priors [3]–[6] or clustering algorithms [7]–[9], [31]. For example, the data distributions of "changed" and "unchanged" components in the image are explored by casting various kinds of priors, including generalized Gaussian distribution [3], hidden Bivariate Gamma distribution [4], Gaussian mixture distribution [5], Markov chains [6] and so on [23]–[25]. Then the changes are detected via some statistical estimation methods. However, these methods often cast too rigid assumption on data, and the detection results rely heavily on the validity of priors. Different with these works, some methods utilize clustering algorithms, such as K-means clustering [5], [30], splitting clustering [6], fuzzy clustering [7] and mean-shift clustering [8], graph non-negative matrix factorization [31] algorithms, to distinguish "changed" and "unchanged" components. However, several parameters, such as the clusters number and splitting levels, should be manually tuned. Moreover, due to the presence of speckle noises in images, unsupervised change detection methods always suffer from high detection error ratio and sensitivity to noises.

Supervised methods formulate the change detection as a binary classification task, and utilize some labeled data (or samples) to identify the changes from the DM [10], [11]. For example, a Support Vector Machine (SVM) with a difference-kernel and a ratio kernel is proposed, for accurate change detection [10]. Later an iterative label- information composite kernel based classifier is proposed for change detection with the guidance of anisotropic texture [11]. In a recent work, a relationship learning approach is proposed, which establishes a classifier to learn the relationship between the changed class and unchanged class [12]. Due to the use of labeled samples, supervised methods often achieve accurate detection [28], [29]. However, both training a good classifier and learning the relationship need a lot of labeled data, which will involve much human participation and result in high labeling cost.

A recent trend in machine learning is to integrate naturalistic learning in biological species into learning, such as continuous learning and self-learning. Self-paced learning is inspired by children's learning process, whose basic idea is to establish a simple model first and then gradually learn samples from "simple" to "complex". In order to realize an automatic and efficient on-orbit change detection from multi-temporal SAR images, in this paper we develop a rapid and simple unsupervised change detection method via a new Extreme Self-paced Learning Machine (ESLM). ESLM can gradually select the confident changed pixels and incrementally predict changed pixels in a semi-supervised pattern. First, a DM of two multi-temporal SAR images is calculated by a log-ratio operator. Then the DM is segmented into some connected regions named super-pixels. Second, a reflectivity-spatial affinity is defined for two super-pixels to evaluate their similarity, and an Affinity Propagation Superpixel Clustering (APSC) algorithm is designed to automatically cluster super-pixels. Based on APSC, three groups of pixels: strictly changed pixels, strictly unchanged pixels and fuzzy pixels, are extracted from the DM, to serve as the initial training data. The flowchart of the proposed method is shown in Fig.1.

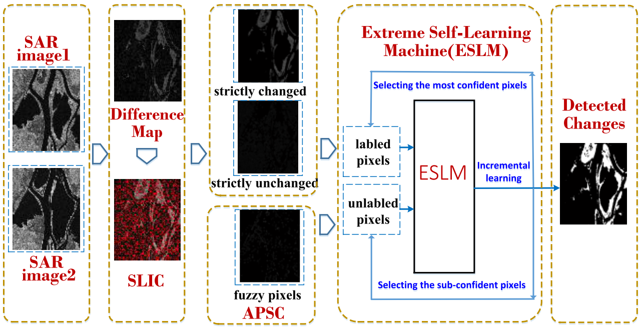


FIGURE 1. The framework of on-orbit change detection via ESLM.

Compared with the available change detection methods, the proposed method can not only work for on-orbit processing, but also have the following characteristics: 1) ESLM works in an incremental self-paced learning pattern, which can automatically choose the training samples itself from "simple" to "complex"; 2) ESLM explores unlabeled samples to enhance the online prediction of changes. So it can not only boost the performance of unsupervised methods, but also avoid high computational complexity of semi-supervised methods; 3) a multistage clustering is used to select initial training samples, which is of high automation degree. Consequently, ESLM has the characteristics of little manual participation, accurate detection and high robustness. Some experiments are taken on the Bern dataset acquired by the ERS-2 SAR, the Ottawa dataset acquired by a RADARSAT SAR, the Yellow River dataset acquired by Radarsat-2 sensor and two single-look SAR images pairs on the Wuhan dataset acquired by PALSAR. The experimental results verify the efficiency of ESLM.

II. MULTISTAGE CLUSTERING

In order to automatically extract the training samples for the subsequent incremental classification, in this section a multistage clustering scheme is proposed. A reflectivity-spatial affinity is first defined, and then a new APSC algorithm is proposed.

A. REFLECTIVITY-SPATIAL AFFINITY OF SUPERPIXELS

Given two co-registered SAR images $\mathbf{X}_1 = \{X_1(i,j)\}$ and $\mathbf{X}_2 = \{X_2(i,j)\}$ taken at different time of the same scene, where $X_1(i,j)$ and $X_2(i,j)$ are the pixels located at (i,j) in the images, and $1 \leq i \leq I$, $1 \leq j \leq J$. First, we generate a DM by a log-ratio operator [13],

$$\mathbf{DM} = \left| \log \frac{\mathbf{X}_2 + \varepsilon}{\mathbf{X}_1 + \varepsilon} \right| \tag{1}$$

where ε is a small positive constant to avoid the pixel values in $\mathrm{X}_i (i = 1, 2)$ be zero. As soon as **DM** is obtained, we first use the Simple Linear Iterative Cluster (SLIC) [14] to segment **DM** into N super-pixels $\{\mathbf{S}_1, \ldots, \mathbf{S}_N\}$. The number of super-pixels is set as $N = \lfloor I \times J/100 \rfloor$. Then an APSC algorithm is advanced to acquire the initial training samples of ESLM.

It is well known that with the increasing resolution of SAR images, they are not only a set of reflectivities but also a set of data with spatial organization. So in our work we define a reflectivity-spatial affinity $s(i,j)$ between the i^{th} and j^{th} super-pixel, to evaluate their similarity $s(i,j)$,

$$s(i,j) = s_{reflectivity}(i,j) + s_{spatial}(i,j) \tag{2}$$

where $s_{reflectivity}(i,j)$ and $s_{spatial}(i,j)$ are the reflectivity affinity and spatial affinity between the i^{th} and j^{th} super-pixels respectively. The reflectivity affinity is then defined as,

$$s_{reflectivity}(i,j) = -\left\| r_i - r_j \right\|^2 \tag{3}$$

where r_i and r_j represent the average reflectivities of the i^{th} and j^{th} super-pixel. The spatial affinity is defined as,

$$s_{spatial}(i,j) = \sqrt{(i_x - j_x)^2 + (i_y - j_y)^2} \tag{4}$$

where (i_x, j_x) and $\left(i_y, j_y\right)$ represent the coordinates of the two super-pixels in the image.

B. AFFINITY PROPAGATION SUPER-PIXEL CLUSTERING (APSC)

Then we use the calculated $s(i,j)$ and propagation algorithm to locate the potential exemplars (or clustering centers) from N super-pixels [15]. The N super-pixels are considered as nodes in a network and $\left\{e_j\right\}, j = 1, \ldots, N$ is a set of binary hidden variables, where $e_j = 1$ indicates that the j^{th} super-pixel is chosen as a clustering center and $e_j = 0$ indicates that it is not a clustering center. $\left\{h_{ij}\right\} (i,j = 1, \ldots, N)$ is a set of N^2 binary hidden variables, where $h_{ij} = 1$ indicates that the i^{th} super-pixel has chosen the j^{th} pixel as its clustering center. Affinity propagation tries to find the most representative samples that maximize the sum of affinities of nodes to their exemplars. This optimization can be implemented via the update of responsibility $r(i,j)$ and availability $a(i,j)$. The availabilities $a(i,j)$ are initialized as zero, and the responsibility $r(i,j)$ are updated by [15],

$$r(i,j) \leftarrow s(i,j) - \max_{j'=j}\{a(i,j') + s(i,j')\} \tag{5}$$

Then the availabilities $a(i,j)$ are updated by:

$$a(i,j) \leftarrow \min\{0, r(j,j) + \sum_{i' \neq \{i,j\}} \max\{0, r(i',j)\}\}, \quad \mathit{if}\ i \neq j \tag{6}$$

$$a(i,j) \leftarrow \sum_{i' s.t. i' \neq j} \max\{0, r(i',j)\}, \quad \mathit{if}\ i = j \tag{7}$$

This procedure will terminate until the exemplars are stably determined. Compared with traditional clustering methods, the APSC algorithm has some advantages: 1) It is free of parameter and can automatically determine the number of clusters; 2) It can involve many-sides affinity between nodes to find more representative clustering centers; 3) The affinity update is of low complexity.

C. MULTISTAGE CLUSTERING

After APSC, denote the number of clusters as V, and denote the intensity of the centroid of the n^{th} super-pixel in the v^{th} cluster as I_n^v. The average value in each cluster is calculated as $AI^v (v = 1, 2, \ldots, V)$. Then we perform a K-means clustering on AI^v to cluster the super-pixels into three groups: strictly changed class, strictly unchanged class and fuzzy class. According to AI^v, the class that has the highest AI^v value is considered as the strictly changed class. The class that has the lowest AI^v value is considered as the strictly unchanged class. The other class is thus considered as fuzzy class.

In our work, we select the "confident" pixels from the strictly unchanged and strictly changed classes, to serve as the labeled samples. On the other hand, we also select the "confident" pixels from the fuzzy class, as the unlabeled samples. If a pixel and its neighbors (a squared $b \times b$ neighboring window centered around the pixel) belong to the same class, we choose it as the "confident" pixel. The "confident" pixels in the strictly unchanged class, strictly changed class and fuzzy class are denoted as $\{\mathbf{x}_i^u\}$, $\{\mathbf{x}_i^c\}$ and $\{\mathbf{x}_i^f\}$ respectively. In our work we set $b = \sqrt{num}/3$, where num is the number of pixels in the super-pixel. Fig.2 illustrates the procedure of this multistage clustering.

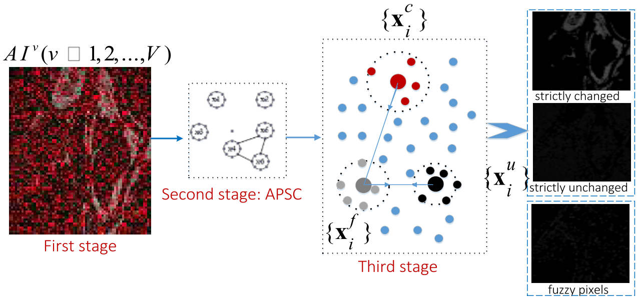


FIGURE 2. Multistage clustering for determining initial training samples.

III. EXTREME SELF-PACED LEARNING MACHINE (ESLM)

Semi-supervised classifier can use a handful of labeled samples and large amount of unlabeled samples to enhance the performance of classifiers [16]–[19], [31]. In this section we first formulate an affinity regularizer to implement the semi-supervised extreme learning. Then an incremental extreme self-paced learning machine is constructed to gradually identify changed pixels.

A. EXTREME LEARNING MACHINE (ELM)

Extreme Learning Machine (ELM) is a single-hidden layer feed-forward neural network [20]. Denote the number of hidden layer as k, and denote the nonlinear activation function of the hidden layer as $g(\cdot)$. Then the output of the hidden layer for the inputs can be written as a matrix,

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g(\mathbf{w}_1 \cdot \mathbf{x}_Q + b_1) \\ g(\mathbf{w}_2 \cdot \mathbf{x}_1 + b_2) & \cdots & g(\mathbf{w}_2 \cdot \mathbf{x}_Q + b_2) \\ \vdots & \ddots & \vdots \\ g(\mathbf{w}_K \cdot \mathbf{x}_1 + b_K) & \cdots & g(\mathbf{w}_K \cdot \mathbf{x}_Q + b_K) \end{bmatrix}_{K\times Q} \tag{8}$$

where $\mathbf{x}_i$ is the i^{th} training samples and $\mathrm{w}_i = [w_{i1}, w_{i2}, \ldots, w_{id}]$ is a random weight vector connecting the i^{th} hidden neuron with input neurons, b_i is the bias of the i^{th} hidden neuron. K and Q are the number of hidden neurons and samples respectively. Here a continuous sigmoid function $g(\cdot)$ is adopted in the hidden layer, according to the universal approximation capability of feed-forward neural networks [32]. The activation function in the output layer is linear, so the network output is $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots \mathbf{y}_m]$, with the output of the j^{th}neuron being,

$$\mathbf{y}_j = \beta_j^T \mathbf{H} \tag{9}$$

where $\beta_j = [\beta_{j1}, \beta_{j2}, \ldots, \beta_{jK}]^T (j = 1, 2, \ldots, m)$ is the weight vector connecting the hidden neurons with the j^{th} output neuron. m is the number of output neurons, which is the length of the label in the classification. Denoting the weight matrix as $\mathbf{B} = [\beta_1, \beta_2, \ldots \beta_m] \in R^{K\times m}$, we solve $\mathbf{B}$ by minimizing the errors of training samples and the norm of weights,

$$\min_{\mathbf{B}} \left\| \mathbf{B}^T \mathbf{H} - \mathbf{T} \right\|_2^2 + \|\mathbf{B}\|_2^2 \tag{10}$$

where $\mathbf{T} = \left[\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_Q\right]^T \in R^{Q\times l}$ is the desired output matrix, and $\mathbf{t}_i$ is the corresponding output of $\mathbf{x}_i$. Therefore $\mathbf{B}$ can be analytically determined by

$$\mathbf{B} = (\mathbf{H}\mathbf{H}^T + \mathbf{I})^{-1}\mathbf{H}\mathbf{T}^T \tag{11}$$

where $\mathbf{I}$ is an identity matrix.

B. AFFINITY REGULARIZED SEMI-SUPERVISED EXTREME LEARNING

Recently some semisupervised ELM methods have been used to enhance the prediction accuracy when only very few labeled samples are available [33]–[35]. In our work, we use the confident pixels set $\{\mathbf{x}_i^c\}$ and strictly unchanged pixels set $\{\mathbf{x}_i^u\}$ as the initial labeled samples $L_0 = \{(\mathbf{x}_i^j, \mathbf{t}_i^j) | \mathbf{x}_i^j \in R^d, j \in (c, u), \mathbf{t}_i^j \in R^m\}$, where $\mathbf{t}_i^j$ is a 2-d binary vector indicating the label of $\mathbf{x}_i^j (\mathbf{t}_i^j = [1, 0]^T$ when $j = c$; $\mathbf{t}_i^j = [0, 1]^T$ when $j = u$); $i = 1, 2, \ldots, l (l = |L_0|$, is the sum of the number of strictly unchanged pixels and strictly changed pixels). The fuzzy pixels $\{\mathbf{x}_i^f\}$ are taken as the unlabeled samples $U_0 = \{\mathbf{x}_i^f | \mathbf{x}_i^f \in R^d\}$, where $i = 1, 2, \ldots, u$ ($u = |U_0|$ is the sum of the number of fuzzy pixels).

Based on ELM, we develop an Affinity Regularized Semi-supervised Extreme Learning Machine (AR-SELM), based on a manifold regularizer. A local consistency assumption is cast on both the labeled and unlabeled samples, which indicates that samples with large affinity should have similar labels [21]. Consequently, a new affinity regularizer is defined as follows,

$$R = \sum_{i,j=l+u} s(i,j) \left\| \mathbf{Y}^i - \mathbf{Y}^j \right\|^2 /2 \tag{12}$$

where $\mathbf{Y}^i$, $\mathbf{Y}^j$ are the corresponding outputs of two samples $\mathbf{x}_i$ and $\mathbf{x}_j$ respectively. Because the items in (12) are positive, this regularizer constrains that if $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar to each other, then the predictions $\mathbf{Y}^i$, $\mathbf{Y}^j$ should be similar as well. Moreover, we reformulate (12) as

$$\begin{aligned} R &= \sum_{i,j=l+u} s(i,j) \left(\left\| \mathbf{Y}^i \right\|^2 + \left\| \mathbf{Y}^j \right\|^2 - 2\left\langle \mathbf{Y}^i, \mathbf{Y}^j \right\rangle \right) /2 \\ &= \frac{1}{2}(Tr(\mathbf{Y}_{l+u}\mathbf{Y}_{l+u}^T\mathbf{D}^T) + Tr(\mathbf{Y}_{l+u}\mathbf{Y}_{l+u}^T\mathbf{D}^T)) \\ &\quad - Tr(\mathbf{Y}_{l+u}\mathbf{Y}_{l+u}^T\mathbf{W}^T) \\ &= Tr(\mathbf{Y}_{l+u}^T(\mathbf{D} - \mathbf{W})\mathbf{Y}_{l+u}) \\ &= Tr(\mathbf{Y}_{l+u}^T\mathbf{L}\mathbf{Y}_{l+u}) \end{aligned} \tag{13}$$

where $\mathbf{D} \in R^{(l+u)\times(l+u)}$ is a diagonal matrix with the element $D_{ii} = \sum_j s(i,j)$. Then the graph Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is calculated.

Denote the output of the hidden layer for the inputs L_0 as $\mathbf{H}_0$. By modifying the objection function of ELM in (10), the objective function of AR-SELM can be written as:

$$\min_{\mathbf{B}_0} \left\| \mathbf{B}_0^T \mathbf{H}_0 - \mathbf{Y} \right\|_2^2 + \|\mathbf{B}_0\|_2^2 + \lambda Tr(\mathbf{Y}_{l+u}^T\mathbf{L}\mathbf{Y}_{l+u}) \tag{14}$$

where λ is a parameter to weight the affinity regularizer in the semi-supervised extreme learning machine. The analytical solution to equation (14) can be calculated to determine the weights,

$$\mathbf{B}_0 = (\mathbf{H}_0\mathbf{H}_0^T + \mathbf{I} + \lambda\mathbf{H}_0^T\mathbf{L}\mathbf{H}_0)^{-1}\mathbf{H}_0\mathbf{Y}^T \tag{15}$$

As soon as the weights are calculated, the unknown samples can be predicted as changed or unchanged pixels from $\mathbf{Y}_u = \mathbf{B}_0^T\mathbf{H}_u$.

C. SELF-PACED LEARNING

In our work, an incremental extreme self-paced learning machine is developed to gradually predict the unknown pixels. The affinity regularized semi-supervised classifier can automatically update the decision boundary by involving the confident pixels into the training. The labeled and unlabeled data come chunk by chunk, and the weights are incrementally learned. So the pixels in the image are gradually predicted as the changed or unchanged pixels.

Suppose at time t, the training dataset $S_t = L_t \cup U_t$ has been learned to solve $\mathbf{B}_t$. $\mathbf{H}_t$ is the hidden matrix of S_t, $\mathbf{H}_t^l$ is the hidden matrix of L_t, $\mathbf{H}_t^u$ is the hidden matrix of U_t. The unlabeled samples are predicted by their outputs $\mathbf{Y}_t^u = \mathbf{B}_t^T \mathbf{H}_t^u$, where $\mathbf{H}_t^u$ is the hidden matrix of unlabeled samples in S_t. The samples whose predicted labels are confident are selected to be the new received chunk $S_C = \{\mathbf{x}_j\}_{j=1}^k$ by sequentially max $\max\limits_{j \in u}(\mathbf{Y}_t^u)$. At time $t+1$, the new received chunk S_C can be classified as two groups: $k/2$ labeled samples(denoted as S_C^l) and $k/2$ unlabeled samples (denoted as S_C^u). Denote the hidden matrix respect to the new chunk as $\mathbf{H}_C$. The hidden matrix and target matrix of S_C^l are denoted as $\mathbf{H}_C^l$ and $\mathbf{Y}_C^l$ respectively. The update training dataset is denoted as $S_{t+1} = L_{t+1} \cup U_{t+1}$, where $L_{t+1} = L_{t+1} \cup S_C^l$, $U_{t+1} = U_t \cup S_C^u$. The output matrix of the hidden layer for all the received data and the labeled data has been received up to time $t+1$,

$$\mathbf{H}_{t+1} = \begin{bmatrix} \mathbf{H}_t \\ \mathbf{H}_C \end{bmatrix}, \quad \hat{\mathbf{H}}_{t+1} = \begin{bmatrix} \mathbf{H}_t^l \\ \mathbf{H}_C^l \end{bmatrix} \tag{16}$$

According to (15), the new weights of the extreme self-paced learning machine can be determined by,

$$\mathbf{B}_{t+1} = \left(\mathbf{I} + \hat{\mathbf{H}}_{t+1}^T \hat{\mathbf{H}}_{t+1} + \lambda \mathbf{H}_{t+1}^T \mathbf{L}_{t+1} \mathbf{H}_{t+1}\right)^{-1} \hat{\mathbf{H}}_{t+1}^T \hat{\mathbf{Y}}_{t+1} \tag{17}$$

For simplicity, we denote $\mathbf{A}_t$ as,

$$\mathbf{A}_t = \left(\mathbf{I} + \hat{\mathbf{H}}_{t+1}^T \hat{\mathbf{H}}_{t+1} + \lambda \mathbf{H}_{t+1}^T \mathbf{L}_{t+1} \mathbf{H}_{t+1}\right) \tag{18}$$

The graph Laplacian matrix at time $t+1$ can be expressed as

$$\mathbf{L}_{t+1} = \begin{bmatrix} \mathbf{L}_t + \mathbf{D}_{t,C} & -S_{t,C} \\ -S_{C,t} & \mathbf{D}_{C,t} + \mathbf{L}_C \end{bmatrix} \tag{19}$$

where $\mathbf{L}_t$ and $\mathbf{L}_C$ are the graph Laplacian matrices with respected to the dataset S_t and S_C respectively. $S_{t,C}$ is affinity matrix from the dataset S_t to S_C. $S_{C,t}$is affinity matrix from the dataset S_C to S_t. $\mathbf{D}_{t,C}$ and $\mathbf{D}_{C,t}$ are the diagonal matrices whose main diagonal elements are row sums of $S_{t,C}$ and $S_{C,t}$ respectively. According to the relationship between time t and $t+1$, $\mathbf{A}_{t+1}$ can be written as

$$\mathbf{A}_{t+1} \approx \mathbf{A}_t + \hat{\mathbf{H}}_C^T \hat{\mathbf{H}}_C + \lambda \mathbf{H}_C^T \mathbf{L}_C \mathbf{H}_C \tag{20}$$

Then we calculate,

$$\begin{aligned} \mathbf{A}_{t+1}^{-1} &\approx \left(\mathbf{A}_t + \hat{\mathbf{H}}_C^T \hat{\mathbf{H}}_C + \lambda \mathbf{H}_C^T \mathbf{L}_C \mathbf{H}_C\right)^{-1} \\ &= \mathbf{F}_t^{-1} - \lambda \mathbf{F}_t^{-1} \mathbf{H}_C^T \left(\mathbf{I}_k + \lambda \mathbf{L}_C \mathbf{H}_C \mathbf{F}_t^{-1} \mathbf{H}_C^T\right)^{-1} \\ &\quad \times \mathbf{L}_C \mathbf{H}_C \mathbf{F}_t^{-1} \end{aligned} \tag{21}$$

where $\mathbf{F}_t^{-1} = \left(\mathbf{A}_t + \hat{\mathbf{H}}_C^T \hat{\mathbf{H}}_C\right)^{-1} = \mathbf{A}_t^{-1} - \mathbf{A}_t^{-1} \hat{\mathbf{H}}_C^T \left(\mathbf{I} + \hat{\mathbf{H}}_C \mathbf{A}_t^{-1} \hat{\mathbf{H}}_C^T\right)^{-1} \hat{\mathbf{H}}_C \mathbf{A}_t^{-1}$. If we denote $\mathbf{K}_t$ as

$$\mathbf{K}_t = \mathbf{I} - \mathbf{A}_t^{-1} \hat{\mathbf{H}}_C^T \left(\mathbf{I} + \hat{\mathbf{H}}_C \mathbf{A}_t^{-1} \hat{\mathbf{H}}_C^T\right)^{-1} \hat{\mathbf{H}}_C \tag{22}$$

Substituting $\mathbf{K}_t$ into $\mathbf{F}_t^{-1}$, we can get $\mathbf{F}_t^{-1} = \mathbf{K}_t \mathbf{A}_t^{-1}$ and $\mathbf{A}_{t+1}^{-1} = \mathbf{J}_t \mathbf{A}_t^{-1}$, where $\mathbf{J}_t = \mathbf{K}_t - \lambda \mathbf{K}_t \mathbf{A}_t^{-1} \mathbf{H}_C^T \left(\mathbf{I} + \lambda \mathbf{L}_C \mathbf{H}_C \mathbf{K}_t \mathbf{A}_t^{-1} \mathbf{H}_C^T\right)^{-1} \mathbf{L}_C \mathbf{H}_C \mathbf{K}_t$. Thus we can obtain the weights of the AR-SELM,

$$\mathbf{B}_{t+1} = \mathbf{A}_{t+1}^{-1} \hat{\mathbf{H}}_{t+1}^T \hat{\mathbf{Y}}_{t+1} = \mathbf{J}_t \mathbf{B}_t + \mathbf{J}_t \mathbf{A}_t^{-1} \hat{\mathbf{H}}_C^T \hat{\mathbf{Y}}_C \tag{23}$$

D. ESLM BASED ON-ORBIT CHANGE DETECTION

In the on-orbit change detection, the working process can be divided into a self-paced learning phase and then a prediction phase. The procedure of ESLM based change detection algorithm is described in Algorithm 1.

```
Algorithm 1 ESLM Based Change Detection Algorithm
Input: Two multi-temporal SAR images
Output: Changes of the two images
Initialization phase: The samples after multistage clustering
are used as the initial training samples: $S_0 = L_0 \cup U_0$
    1) Generate the hidden matrix $\mathbf{H}_0^l$, $\mathbf{H}_0$ for $S_0$;
    2) Calculate the graph Laplacian matrix $\mathbf{L}_0$ for $S_0$;
    3) Randomly initialize the parameter of ELM;
    4) Calculate $\mathbf{B}_0$from (15);
Self-paced learning phase: Set $t = 0$ and repeat the following steps,
    1) Formulate the data chunk $S_C$ from $\mathbf{B}_t$;
    2) Calculate the mapping matrix $\mathbf{H}_C$ for the newest chunk;
    3) Calculate the Laplacian matrix $\mathbf{L}_C$ with respected to the
       input of dataset $S_C$;
    4) Calculate the mapping matrix $\hat{\mathbf{H}}_C$ for the newest
       labeled samples, calculate $\mathbf{K}_t$, $\mathbf{J}_t$, $\mathbf{B}_{t+1}$ and $\mathbf{A}_{t+1}^{-1}$;
    5) Set $t = t + 1$, go to step 1);
    Until $||B_{t+1} - \mathbf{B}_t|| < \varepsilon$;
Predicting Phase: Use the learned B to evaluate the output
of unknown pixels to locate the changed pixels.
```

IV. EXPERIMENTAL RESULTS

To investigate the performance of the proposed method, in this section some experiments are taken on five real multi-temporal SAR images, including the Bern dataset acquired by the ERS-2 SAR, the Ottawa dataset acquired by a RADARSAT SAR, the Yellow River dataset acquired by Radarsat-2, and the Wuhan dataset acquired by PALSAR. The first three datasets have the ground-truth and the last two datasets have not got the ground-truth.

A. DATASETS AND COMPARATIVE METHODS

1) *Dataset 1*: The first dataset covers a region near the city of Bern, Switzerland, in April and May 1999. The size of the two SAR images is 301 × 301, with the resolution being 30m, which are shown in Fig.3(a) and Fig.3(b) [3]. The images are collected by the SAR on the European Remote Sensing 2 satellite. The radar works in C-band and has VV polarization. The ground-truth is shown in Fig.3(c).

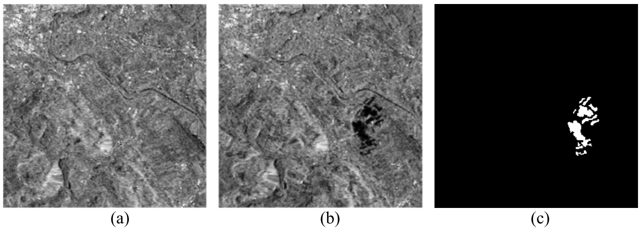


FIGURE 3. SAR images of bern. (a) Image in 04, 1999. (b) Image in 05, 1999. (c) Ground-truth.

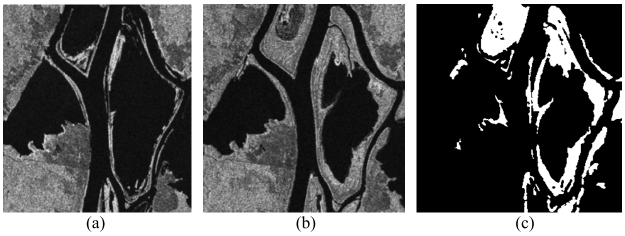


FIGURE 4. SAR images of Ottawa. (a) Image in 05, 1997. (b) Image in 08, 1997. (c) Ground-truth.

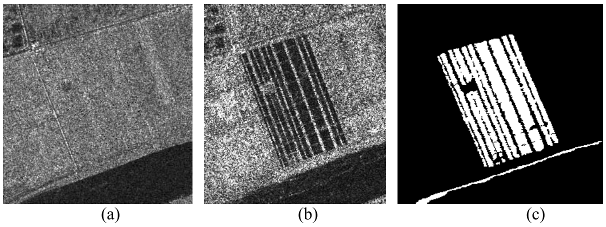


FIGURE 5. SAR images of the Yellow river estuary. (a) Image in 06, 2008. (b) Image in 06, 2009. (c) Ground-truth.

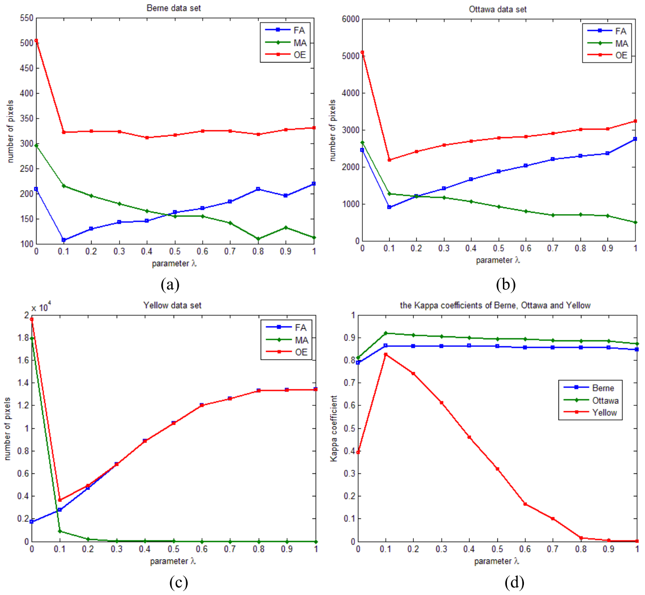


FIGURE 6. Performances of ESLM with the variation of parameter λ. (a) Bern dataset. (b) Ottawa dataset. (c) Yellow dataset. (d) Kappa coefficient.

2) *Dataset 2*: The second dataset covers a region over the city of Ottawa, in May and August 1997. The size of the two SAR images is 290×350, with the resolution being 10m, which are shown in Fig.4(a) and Fig.4(b). The images are collected by the SAR on the RADARSAT. The radar works in C-band and has HH polarization. The ground-truth is shown in Fig.4(c).

3) *Dataset 3*: The third dataset covers a region over the Yellow River estuary in China in June 2008 and June 2009. The size of the two SAR images is 257×289, with the resolution being 3m, which are shown in Fig.5(a) and Fig.5(b). The images are collected by the SAR on the Radarsat-2. The radar works in C-band and has HH polarization. The ground-truth is shown in Fig.5(c). In addition, speckle noise on the image Fig.5(a) is much more than that of Fig.5(b), because Fig.5(a) is single-look image and Fig.5(a) is four-look.

4) *Dataset 4 and 5*: The last two datasets cover the region of Wuhan Province in China. The original two SAR images are obtained by PALSAR in June 2006 and in March 2009. The radar works in L-band and has HH polarization. The resolution of the two images is 10m, and the images are shown in Fig.5(a) and Fig.5(b) respectively. The range and size of the two SAR images are 40km × 70km and 500 × 500 respectively. The Wuhan dataset is single-look, so the two images have more speckle noises than the first four datasets. Two obviously changed areas are selected in original SAR images to verify the performance of the proposed algorithm.

In order to verify the effectiveness of ESLM, five related and some state-of-the-art algorithms are used for a comparison, including the Mathematical Morphology and K-means clustering (MMK) [30], Deformable Dictionary Learning (DDL) [26], Neighborhood-based Ratio (NR) [27], Local Restricted Convolutional Network (LRCN) [28] and Contractive Autoencoder (CA) [29]. Both the visual results and numerical results are demonstrated, along with the consumed time of these methods. All the experiments are taken on Intel ® Core$^{\text{TM}}$ i3-3210, CPU @2.10GHz 4.0GB Windows 10 systems, Matlab 2014a.

B. EVALUATION CRITERION AND PARAMETER SELECTION

In order to evaluate the performance of various kinds of algorithms, some evaluation criterions are used, including Missed Alarms (MA: the number of undetected pixels in the changed region), False Alarms (FA: the number of changed pixels that are wrongly detected as unchanged pixels), Overall Error (OE) and Kappa Coefficient (KC) [22]. Moreover, in this section we make an analysis of the parameter on the first three datasets.

N_c is the number of changed pixels in the ground-truth and N_u is the number of unchanged pixels in the ground-truth. Then some evaluation metrics are calculated [23]:

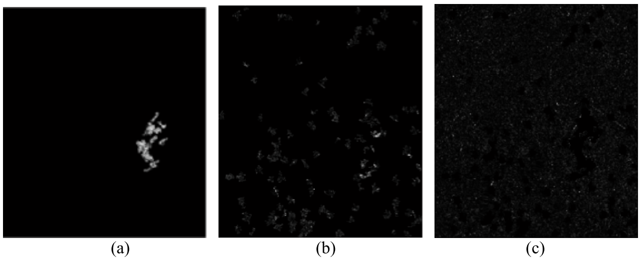


FIGURE 7. Extracted pixels by APSC on the Bern dataset. (a) Strictly changed pixels. (b) Fuzzy pixels. (c) Strictly unchanged pixels.

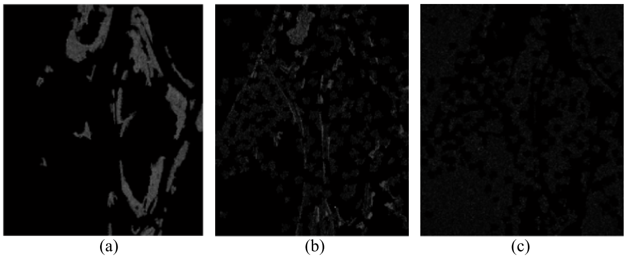


FIGURE 8. Extracted pixels by APSC on the Ottawa dataset. (a) Strictly changed pixels. (b) Fuzzy pixels. (c) Strictly unchanged pixels.

(1) false alarm rate: $P_{PA} = \text{FA}/N_u$. (2) missed alarm rate: $P_{MA} = \text{MA}/N_c$. (3) total errors rate: $P_{OE} = OE/(N_c + N_u)$. In our method, there is a single parameter λ. In order to choose an appropriate λ, we vary its value from 0 to 1, and the results of the first three datasets are shown in Fig.6. From the variation of MA, FA and OE with the parameter λ in Fig.6(a)(b)(c), we can observe that for the datasets, these values first increase and then decrease. Whenλtakes the value in the range of [0.08, 0.15], the three metrics can achieve relatively lower values, which indicates more accurate change detection. Fig.6(d) shows the variation of KC with the parameter λ for the first three datasets, and form it we can observe that KC also takes a relatively larger value in this range for the three datasets. So in the following experiments, we set λ as 0.1.

C. EXPERIMENTAL RESULTS ON THE FIRST THREE DATASETS

In this section we first investigate the performance of the proposed APSC algorithm. For the first three datasets, the labeled pixels, fuzzy pixels and unlabeled pixels extracted by APSC are shown in Fig.7-9. Fig.7(a), Fig.8(a) and Fig.9(a) show the strictly changed pixels on the Bern, Ottawa and Yellow datasets respectively. When compared them with the ground-truth, we can observe that most of the detected labeled pixels are correct, which validate the effectiveness of APSC.

Moreover, from Fig.7(a), Fig.8(a) and Fig.9(a) we can observe that the detected strictly changed pixels are locally connected. That is, APSC is robust to the speckle noises existed in the SAR images. Fig.7(b), Fig.8(b) and Fig.9(b) show the fuzzy pixels detected by APSC on the Bern, Ottawa and Yellow datasets respectively. From them we can observe that the fuzzy pixels prone to be uniformly distributed in the whole image, which contain both changed and unchanged pixels. This is consistent with the uncertainty of fuzzy pixels. Fig.7(c), Fig.8(c) and Fig.9(c) show the strictly unchanged pixels by APSC on the Bern, Ottawa and Yellow datasets respectively. From Fig.7(c), Fig.8(c) and Fig.9(c) we can find that most of them are unchanged pixels, and this is especially obvious in Fig.9(c). However, the detected strictly unchanged pixels not only contain pixels with low reflectivity but also include some speckle noises, which will bring some misclassification to ESLM.

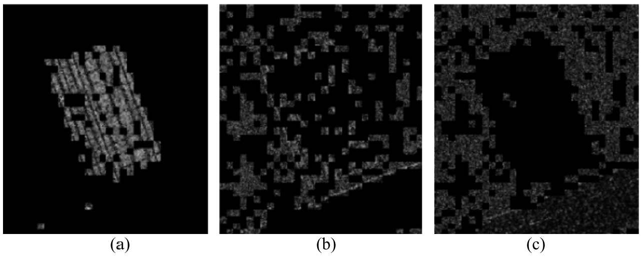


FIGURE 9. Extracted pixels by APSC on the Yellow dataset. (a) Strictly changed pixels. (b) Fuzzy pixels. (c) Strictly unchanged pixels.

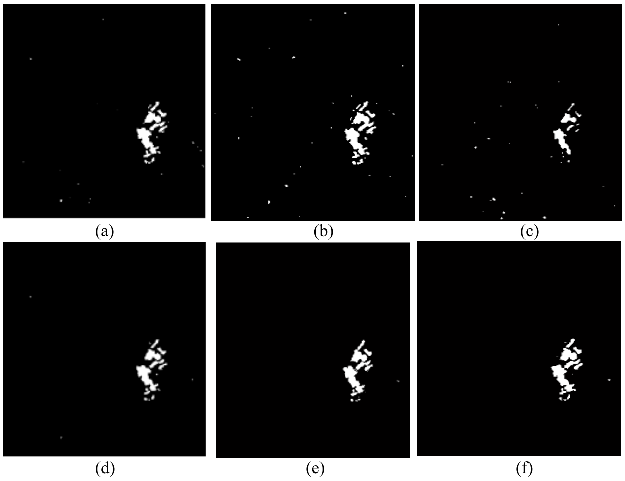


FIGURE 10. Change maps of different methods on the Bern data set. (a) MMK [30]. (b) DDL [26]. (c) NR [27]. (d) LRCN [28]. (e) CA [29]. (f) ESLM.

When the three groups of pixels are identified by our proposed clustering algorithm, then ESLM is incrementally trained to detect changes. Consequently we compare the detection results of the five comparative methods and ESLM. The change detection results of the first three datasets are shown in Fig.10∼Fig.12 respectively. Fig.10(a)-(f) show the detected change maps of the first dataset, by MMK [30], DDL [26], NR [27], LRCN [28], CA [29] and ESLM respectively. From them we can see that there are little difference among the detection results of different methods, since the original images are relatively clean. MMK [30], LRCN [28], CA [29] and ESLM present similar results, and there are some wrongly detected pixels in the result of DDL [26] and NR [27]. Compared with the unsupervised methods, ESLM not only provides more refined edges but also less noisy pixels when compared with other methods. Moreover, ESLM present comparable results with the supervised methods that explore a lot of labeled pixels for training, including LRCN [28] and CA [29]. However, ESLM need not manually label large number of changed and unchanged pixels. By using the self-paced learning, ESLM can automatically label change pixels in an incremental manner. Besides it, ESLM also explore fuzzy pixels to enhance the detection accuracy, and the parameters of ESLM are analytically determined, which can also achieve a rapid detection.

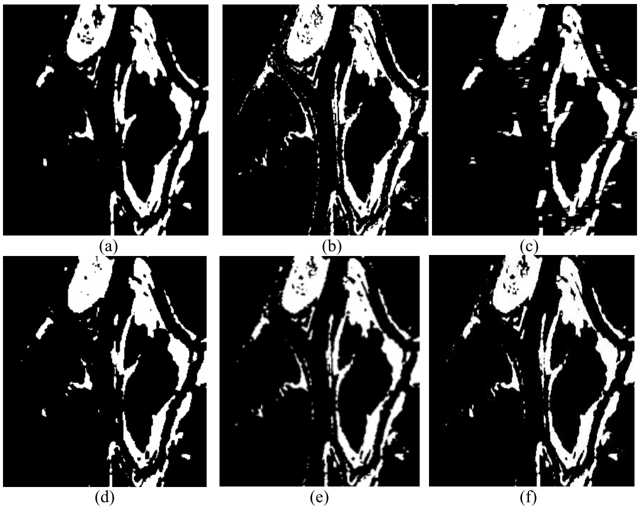


FIGURE 11. Change maps of different methods on the Ottawa dataset. (a) MMK [30]. (b)DDL [26]. (c) NR [27]. (d) LRCN [28]. (e) CA [29]. (f) ESLM.

Fig.11(a)-(f) show the detected changes of the second dataset by MMK [30], DDL [26], NR [27], LRCN [28], CA [29] and ESLM respectively. From the results we can observe that NR [27] detects too smooth regions and its OE is the highest among these methods. DDL [26] has the lowest missed alarms and ESLM has the lowest false alarms. LRCN [28], CA [29] present better results than MMK [30], DDL [26] and NR [27]. According to OE, ESLM present better results when compared with other methods. However, the missed alarm rate is slightly high when compared with LRCN [28] and DDL [26].

Fig12(a)-(f) show the detected changes maps of the third dataset by MMK [30], DDL [26], NR [27], LRCN [28], CA [29] and ESLM respectively. From Fig.12(f) we can see that our method can present comparable results with other supervised methods. However, ESLM consumes less time than the supervised methods, such as LRCN [28] and CA [29]. Among the comparative methods, MMK [30] present very homogeneous detection results. Because the ground-truth of the third dataset is also homogeneous, the change map generated by MMK [30] are more accord with the groud-truth when compared with other methods.

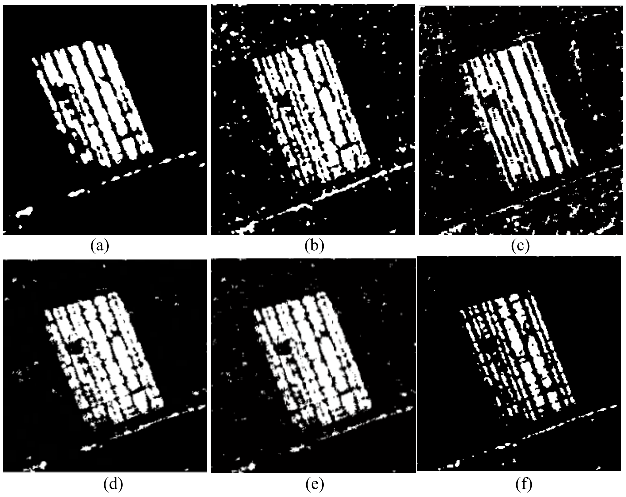


FIGURE 12. Change maps of different methods on the Yellow data set. (a) MMK [30]. (b) DDL [26]. (c) NR [27]. (d) LRCN [28]. (e) CA [29]. (f) ESLM.

TABLE 1. Numerical results of bern dataset.

Methods	FA	P_{FA}	MA	P_{MA}	OE	P_{OE}	KC	T(s)
MMK[30]	116	0.13	190	16.50	306	0.33	0.874	104.5
DDL[26]	141	0.16	199	17.20	340	0.37	0.743	64.3
NR[27]	127	0.14	402	34.80	529	0.58	0.760	**20.3**
LRCN[28]	110	0.12	194	16.80	304	0.33	0.895	248.1
CA[29]	109	0.12	**185**	**16.02**	294	0.32	0.901	132.7
ESLM	**107**	**0.12**	187	16.19	**294**	**0.32**	**0.908**	21.6

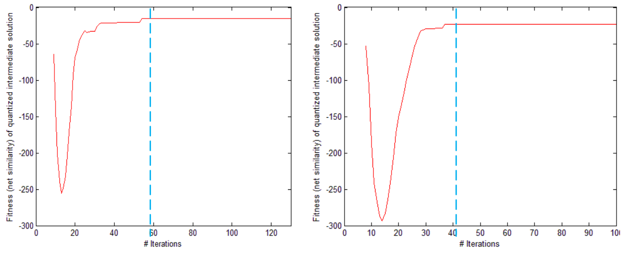
TABLE 2. Numerical results of ottawa dataset.

Methods	FA	P_{FA}	MA	P_{MA}	OE	P_{OE}	KC	T(s)
MMK[30]	1349	1.58	1330	8.29	2679	2.63	0.903	115.2
DDL[26]	1366	1.60	**760**	**4.73**	2126	2.09	0.922	67.1
NR[27]	2148	2.51	2191	13.65	4339	4.27	0.839	29.4
LRCN[28]	1107	1.10	1258	7.84	2365	2.33	0.926	253.8
CA[29]	832	0.83	1301	8.10	2133	2.10	0.911	203.3
ESLM	**787**	**0.78**	1200	7.48	**1987**	**1.96**	**0.927**	**27.1**

TABLE 3. Numerical results of the Yellow dataset.

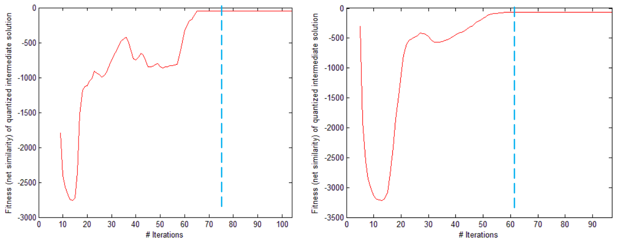
Methods	FA	P_{FA}	MA	P_{MA}	OE	P_{OE}	KC	T(s)
MMK[30]	**2691**	**3.62**	**1864**	**13.9**	**4555**	**6.13**	0.807	139.3
DDL[26]	3907	5.26	2180	16.2	6087	8.19	0.647	49.1
NR[27]	3858	5.19	2002	14.9	5860	7.89	0.663	**33.4**
LRCN[28]	2979	4.01	2117	15.8	5096	6.86	0.725	217.6
CA[29]	3023	4.07	2292	17.1	5315	7.15	0.754	182.4
ESLM	2732	3.68	1994	14.85	4726	6.36	**0.781**	35.2

The numerical metric (FA, MA,OE, P_{FA}, P_{MA}, P_{OE}, KC) and the consumed time of the six methods are shown in Table 1, Table 2 and Table 3 respectively. As to these evaluations, the best results of the six methods are denoted as bold. From the results we can see that in most cases, ESLM has better performance in detecting change details. As can be seen from the three tables, ESLM can totally achieve



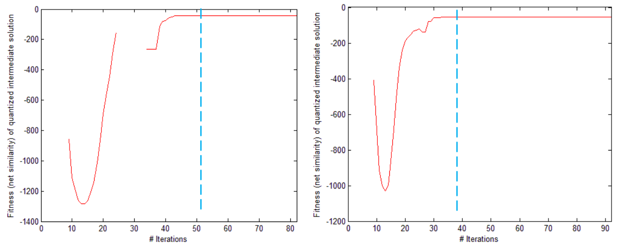
(a) AP algorithm. (b) APSC algorithm.

FIGURE 13. Comparison of AP with APSC on the Bern dataset.



(a) AP algorithm. (b) APSC algorithm.

FIGURE 14. Comparison of AP with APSC on the Ottawa dataset.



(a) AP algorithm. (b) APSC algorithm.

FIGURE 15. Comparison of AP with APSC on the Yellow dataset.

small detection error. Moreover, ESLM not only has the highest kappa coefficients, but also has acceptable running time.

D. EVALUATION CRITERION AND PARAMETER SELECTION

In order to separately investigate the role of the clustering algorithm and the classifier in our method, in this test we first compare APSC algorithm with AP algorithm, and then compare AR-SELM with ELM. The performance of the AP and APSC algorithms, are compared on the three datasets. Fig.13-15 show the convergence curves of AP and APSC for the three datasets respectively. From Fig.13-15, we can observe that for all the three datasets, APSC required less number of iterations before convergence than that of AP. In other words, APSC has faster clustering speed than AP. Moreover, we can also find that APSC is superior to AP as to the convergence speed, which can be observed in Fig.13(b), Fig.14(b) and Fig.15(b).

Moreover, two methods are used to compare with our proposed method, including AP-ELM (which uses AP for clustering and ELM without affinity regularizer), AP-ARSELM (which uses AP for clustering and AR-SELM with affinity regularizer). The comparison results of their performance on the first three datasets are shown in Table 4∼6 respectively. It can be clearly seen from them that APSC has an improvement over AP about the performance, and our

TABLE 4. Comparison of AP-ELM, AP-ARSELM and ESLM on bern dataset.

Methods	FA	P_{FA}	MA	P_{MA}	OE	KC
AP-ELM	370	0.41	250	21.65	620	0.713
AP-ARSELM	189	0.21	130	11.26	319	0.857
ESLM	**107**	**0.12**	**187**	**16.19**	**294**	**0.875**

TABLE 5. Comparison of AP-ELM, AP-ARSELM and ESLM on ottawa dataset.

Methods	FA	P_{FA}	MA	P_{MA}	OE	KC
AP-ELM	3325	3.28	1094	6.82	4419	0.827
AP-ARSELM	1253	1.23	958	5.97	2211	0.918
ESLM	**787**	**0.78**	**1200**	**4.66**	**1987**	**0.927**

TABLE 6. Comparison of AP-ELM, AP-ARSELM and ESLM on Yellow dataset.

Methods	FA	P_{FA}	MA	P_{MA}	OE	KC
AP-ELM	4723	6.36	5492	40.89	10215	0.546
AP-ARSELM	3309	4.46	1549	11.53	4858	0.767
ESLM	**2732**	**3.68**	**1994**	**14.85**	**4726**	**0.781**

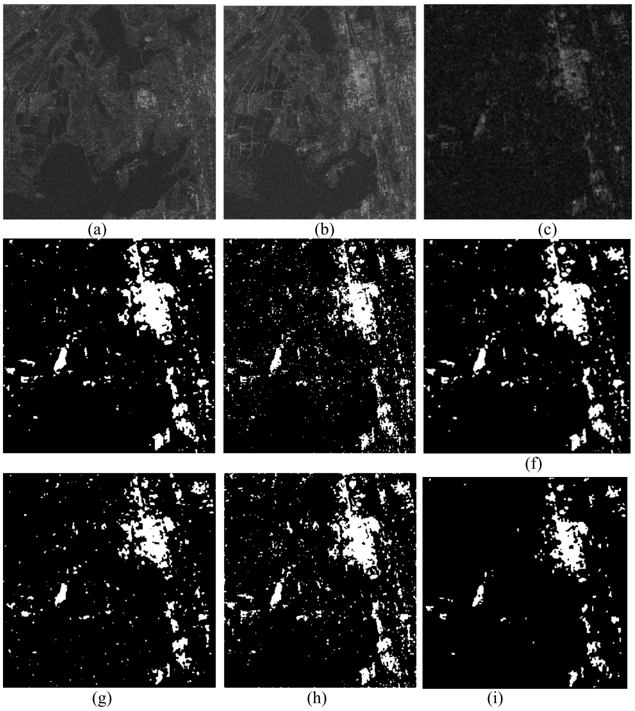


FIGURE 16. Experimental results of the east lack region, Wuhan, China. (a) Image acquired in 06, 2006. (b) Image acquired in 03, 2007. (c) Log-ratio DM. (d) MMK [30]. (e) DDL [26]. (f) NR [27]. (g) LRCN [28]. (h) CA [29]. (i)ESLM.

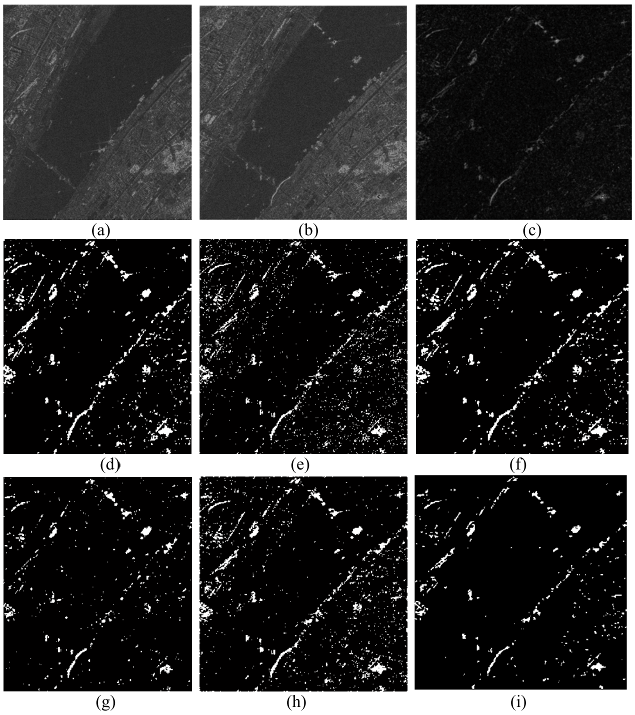


FIGURE 17. Experimental results of Erqi Changjiang River Bridge region, Wuhan, China. (a) Image acquired in 06, 2006. (b) Image acquired in 03, 2007. (c) Log-ratio DM. (d) MMK [30]. (e) DDL [26]. (f) NR [27]. (g) LRCN [28]. (h) CA [29]. (i) ESLM.

proposed ESLM is superior to the original ELM by adding the spatial and reflectivity affinity between pixels in the classification.

E. EXPERIMENTAL RESULTS ON THE LAST TWO DATASETS

In this test, we test our proposed method on the last two datasets. The detection results of six methods are shown in Fig.16 and Fig.17 for visual comparison. Fig.16(a)-(b) and Fig.17(a)-(b) show the two source SAR images, and the log-ratio DMs of the forth and fifth dataset are shown in Fig.16(c) and Fig.17(c) respectively. The main changes between Fig.16(a) and Fig.16(b) are some building and filled lakes. The main changes between Fig.17(a) and Fig.17(b) include the Erqi Changjiang River Bridge, some ships and buildings.

The detection results of the six methods on the two datasets are shown in Fig.16(c)-(i) and Fig.17(c)-(i), respectively. As shown in Fig.16 (e) and (h), we could observe that DDL [26] and CA [29] produce many isolate pixels, which denotes that they are not robust enough to heavy speckle noises. However, from Fig.16(i) we can observe that our method produce less isolate pixels and detect more homogeneous changes when compared with other methods. Similarly, the detected changes by ESLM in Fig.17 (i) have less isolated pixels.

V. CONCLUSION

Human intelligence indicates a learning agent interacts with a dynamic environment and updates its action policies to maximize its long-term rewards. Inspired by it, an automatic and robust unsupervised method via multistage clustering and new extreme self-paced learning machine, is proposed for on-orbit change detection of SAR images. The advantage of ESLM is two folds: 1) it can gradually make avail of the most confident samples, so it has low memory cost and low computational complexity, which is desirable for on-orbit processing; 2) it adopts a semi-supervised model to realize a gradual classification, so achieving accurate detection results. The performance of the proposed method is investigated on several real SAR images, and the results are promising for on-orbit detection. However, as we can see in the results of the Yellow River dataset, our algorithm is also influenced by heavy speckle noises. Therefore, improving the performance of the algorithm to deal with heavy speckle noises will be considered in our future work.

REFERENCES

[1] A. Singh, "Review article digital change detection techniques using remotely-sensed data," *Int. J. Remote Sens.*, vol. 10, no. 6, pp. 989–1003, 1988.

[2] E. J. M. Rignot and J. J. van Zyl, "Change detection techniques for ERS-1 SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 31, no. 4, pp. 896–906, Jul. 1993.

[3] Y. Bazi, L. Bruzzone, and F. Melgani, "An unsupervised approach based on the generalized Gaussian model to automatic change detection in multitemporal SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 874–887, Apr. 2005.

[4] F. Chatelain, J.-Y. Tourneret, and J. Inglada, "Change detection in multisensor SAR images using bivariate Gamma distributions," *IEEE Trans. Image Process.*, vol. 17, no. 3, pp. 249–258, Mar. 2008.

[5] Z. Yetgin, "Unsupervised change detection of satellite images using local gradual descent," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 5, pp. 1919–1929, May 2012.

[6] C. Carincotte, S. Derrode, and S. Bourennane, "Unsupervised change detection on SAR images using fuzzy hidden Markov chains," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 2, pp. 432–441, Feb. 2006.

[7] F. Bovolo and L. Bruzzone, "A split-based approach to unsupervised change detection in large-size multitemporal images: Application to tsunami-damage assessment," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1658–1670, Jun. 2007.

[8] 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.

[9] B. Aiazzi, L. Alparone, S. Baronti, A. Garzelli, and C. Zoppetti, "Nonparametric change detection in multitemporal SAR images based on mean-shift clustering," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 4, pp. 2022–2031, Apr. 2013.

[10] G. Camps-Valls, L. Gómez-Chova, J. Muñoz-Marí, J. L. Rojo-Álvarez, and M. Martínez-Ramón, "Kernel-based framework for multitemporal and multisource remote sensing data classification and change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 6, pp. 1822–1835, Jun. 2008.

[11] L. Jia, M. Li, Y. Wu, P. Zhang, G. Liu, H. Chen, and L. An, "SAR image change detection based on iterative label-information composite kernel supervised by anisotropic texture," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3960–3973, Jul. 2015.

[12] C. Huo, K. Chen, K. Ding, Z. Zhou, and C. Pan, "Learning relationship for very high resolution image change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 8, pp. 3384–3394, Aug. 2016.

[13] 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.

[14] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels," EPFL, Vaud, Switzerland, Tech. Rep., 2010.

[15] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.

[16] L. Jia, M. Li, Y. Wu, P. Zhang, H. Chen, and L. An, "Semisupervised SAR image change detection using a cluster-neighborhood kernel," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 8, pp. 1443–1447, Aug. 2014.

[17] L. An, M. Li, P. Zhang, Y. Wu, L. Jia, and W. Song, "Discriminative random fields based on maximum entropy principle for semisupervised SAR image change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 8, pp. 3395–3404, Aug. 2016.

[18] K. Chen, Z. Zhou, C. Huo, X. Sun, and K. Fu, "A semisupervised context-sensitive change detection technique via Gaussian process," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 2, pp. 236–240, Mar. 2013.

[19] F. Bovolo, L. Bruzzone, and M. Marconcini, "A novel approach to unsupervised change detection based on a semisupervised SVM and a similarity measure," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 7, pp. 2070–2082, Jul. 2008.

[20] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," in *Proc. Int. Joint Conf. Neural Netw.*, vol. 2, 2004, pp. 985–990.

[21] Y. Peng, S. Wang, X. Long, and B.-L. Lu, "Discriminative graph regularized extreme learning machine and its application to face recognition," *Neurocomputing*, vol. 149, pp. 340–353, Feb. 2015.

[22] G. H. Rosenfield and K. Fitzpatrick-Lins, "A coefficient of agreement as a measure of thematic classification accuracy," *Photogramm. Eng. Remote Sens.*, vol. 52, no. 2, pp. 223–227, 1986.

[23] L. Su, M. Gong, B. Sun, and L. Jiao, "Unsupervised change detection in SAR images based on locally fitting model and semi-EM algorithm," *Int. J. Remote Sens.*, vol. 35, no. 2, pp. 621–650, 2014.

[24] T. Celik, "Unsupervised change detection in satellite images using principal component analysis and k-means clustering," *IEEE Geosci. Remote Sens. Lett.*, vol. 6, no. 4, pp. 772–776, Oct. 2009.

[25] 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.

[26] L. Li, Y. Zhao, J. Sun, R. Stolkin, Q. Pan, J. C.-W. Chan, S. G. Kong, and Z. Liu, "Deformable dictionary learning for SAR image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4605–4617, Aug. 2018.

[27] M. Gong, Y. Cao, and Q. Wu, "A neighborhood-based ratio approach for change detection in SAR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 2, pp. 307–311, Mar. 2012.

[28] F. Liu, L. Jiao, X. Tang, S. Yang, W. Ma, and B. Hou, "Local restricted convolutional neural network for change detection in polarimetric SAR images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 818–833, Mar. 2019.

[29] N. Lv, C. Chen, T. Qiu, and A. K. Sangaiah, "Deep learning and superpixel feature extraction based on contractive autoencoder for change detection in SAR images," *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5530–5538, Dec. 2018.

[30] L. Liu, Z. Jia, J. Yang, and N. K. Kasabov, "SAR image change detection based on mathematical morphology and the K-means clustering algorithm," *IEEE Access*, vol. 7, pp. 43970–43978, 2019.

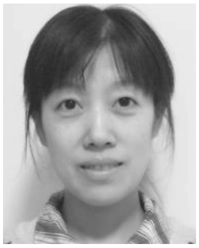
[31] X. He, Q. Wang, and X. Li, "Robust adaptive graph regularized non-negative matrix factorization," *IEEE Access*, vol. 7, pp. 83101–83110, 2019.

[32] X. Chen and H. White, "Improved rates and asymptotic normality for non-parametric neural network estimators," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 682–691, Mar. 1999.

[33] H. Al-Behadili, A. Grumpe, C. Dopp, and C. Wöhler, "Extreme learning machine based novelty detection for incremental semi-supervised learning," in *Proc. 3rd Int. Conf. Image Inf. Process. (ICIIP)*, Waknaghat, India, 2015, pp. 230–235.

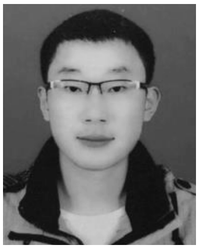
[34] G. Huang, S. Song, J. N. D. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2405–2417, Dec. 2014.

[35] L. Yao and Z. Ge, "Deep learning of semisupervised process data with hierarchical extreme learning machine and soft sensor application," *IEEE Trans. Ind. Electron.*, vol. 65, no. 2, pp. 1490–1498, Feb. 2018.



SHUYUAN YANG received the B.S. degree in electrical engineering and the M.S. and Ph.D. degrees in circuit and system from Xidian University, Xi'an, China, in 2000, 2003, and 2005, respectively. Her research interests include intelligent signal processing, machine learning, and image processing. She is a Senior Member of IET.

ZHI LIU received the B.S. degree in intelligence science and technology from Xidian University, Xi'an, China, in 2014, where he is currently pursuing the Ph.D. degree in intelligent information processing with the School of Artificial Intelligence. His research interests include deep learning, synthetic aperture radar imaging, and image understanding.



QUANWEI GAO received the B.S. degree in electronic information science and technology from the Shandong University of Technology, Zibo, China, in 2017. He is currently pursuing the master's degree in electronics and communication engineering with the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China, Xidian University, Xi'an, China. His research interests include machine learning and image processing.

YUTENG GAO received the B.S. and M.S. degrees from Northwestern Polytechnical University, in 2010 and 2013, respectively, where he is currently pursuing the Ph.D. degree with the School of Electronics and Information. His research interests include antenna design and radar imaging.

ZHIXI FENG received the Ph.D. degree in intelligence science and technology from Xidian University, Xi'an, China, in 2018, where he is currently a Lecturer with the School of Artificial Intelligence. His research interests include deep learning and image understanding.