

Group Self-Paced Learning With a Time-Varying Regularizer for Unsupervised Change Detection

Maoguo Gong[✉], Senior Member, IEEE, Yingying Duan, and Hao Li[✉]

Abstract—Unsupervised change detection based on supervised or semisupervised classifiers has achieved strong adaptability and robustness to obtain satisfactory change detection results. However, these methods suffer from an issue that it is hard to collect reliable training samples in an unsupervised manner. In this article, a group self-paced learning (GSPL) framework is proposed to mine the reliable training samples. In the proposed method, each sample is assigned a weight to indicate its reliability. The proposed scheme is able to learn the weighted samples and update the weights iteratively in a self-paced manner to identify the reliable training samples. In the phase of updating weights, a grouping strategy is designed to avoid selecting training samples from homogeneous regions. Furthermore, a novel time-varying self-paced regularizer is proposed to automatically determine the learning scheme of self-paced learning. Finally, three classifiers, including SoftMax, backpropagation neural network, and support vector machine, are investigated under this proposed framework. Experiments on five change detection data sets demonstrate that the proposed framework can significantly outperform those state-of-art methods for change detection in terms of accuracy and robustness.

Index Terms—Change detection, remote sensing image, self-paced learning (SPL).

I. INTRODUCTION

CHANGE detection aims to detect unchanged and changed areas between images of the same scene at different times, which has attracted widespread interests in recent years [1]–[4]. With the rapid development of various satellite sensors, change detection in remote sensing images has been applied in civil and military fields, such as glacier changes monitoring [5], urbanization construction [6], damage assessment [7], and flood monitoring [8]. The supervised [9], semisupervised [10], and unsupervised [11] change detection methods have been proposed to detect the changes between two remote sensing images. Among them, the unsupervised methods are popular because they do not require labels. The threshold methods and the clustering methods are widely used in unsupervised change detection. The threshold methods, such as Otsu [12], Kittler–Illingworth (KI), [13], and

Manuscript received September 4, 2019; accepted October 31, 2019. Date of publication November 27, 2019; date of current version March 25, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61772393 and Grant 61906146 and in part by the Key Research and Development Program of Shaanxi Province under Grant 2018ZDXM-GY-045. (*Corresponding author: Maoguo Gong*)

The authors are with the Key Laboratory of Intelligent Perception and Image Understanding, Ministry of Education of China, Xidian University, Xi'an 710071, China (e-mail: gong@ieee.org; omegalihao@gmail.com).

This article has supplementary downloadable material available at <http://ieeexplore.ieee.org>, provided by the authors.

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

Digital Object Identifier 10.1109/TGRS.2019.2951441

expectation–maximization (EM) [14], aim to find the optimal threshold value to distinguish the pixels into changed or unchanged categories properly. The main idea of clustering methods is to maximize the similarity between pixels belonging to the same class and minimize the similarity between pixels of different classes. Many local and nonlocal clustering methods have been proposed to deal with the change detection problems, such as the fuzzy local information C-means (FLICM) clustering algorithm [15] and the fuzzy C-means (FCM) clustering algorithm with local information and kernel metric (KWFLICM) [16].

To further improve the change detection performance, some researchers have tried to introduce a supervised classifier into the traditional unsupervised change detection method. Wu *et al.* [17] first extracted speeded up robust features (SURF) key points from both images and then matched them using the random sample consensus algorithm to get training samples of the changed and unchanged categories. An SVM classifier was trained to obtain the change detection results. Li *et al.* [18] employed an object-based Markov random field method to obtain the training samples and then considered logistic regression as the classifier to classify the features extracted by stacked autoencoders into changed and unchanged classes. The above-mentioned change detection methods are generally unsupervised because the training samples are obtained in an unsupervised manner.

There are two main issues to consider in current unsupervised change detection methods based on the supervised classifier. On the one hand, some acquired labels based on unsupervised methods are inaccurate. On the other hand, it is an urgent task to design an efficient classifier to deal with inaccurate labels. In this article, self-paced learning (SPL) is incorporated into traditional change detection methods to address the above-mentioned issues. SPL is able to collect reliable samples by automatically assigning weights to training samples. Furthermore, SPL is able to identify outliers and reduce their impact on the classifier by assigning their weights to zero. Recently, SPL has been proven to be effective in solving many problems, such as human behavior recognition and long-term tracking [19], [20]. Self-paced convolutional neural networks (NNs) [21] incorporated SPL into the convolutional NNs and achieved good classification results in handwritten digit recognition. Shang *et al.* [22] proposed a change detection method based on SPL to detect the changes in synthetic aperture radar (SAR) images.

The multitemporal images may have several heterogeneous regions. For example, the considered image may consist of rivers, mountains, and forests. The unsupervised change detec-

tion methods always tend to choose training samples from homogeneous regions, which would prevent classifiers from being well trained. In this article, a group SPL (GSPL) framework with a time-varying regularizer is proposed for change detection in multitemporal remote sensing images. To avoid selecting training samples from the same region, a grouping strategy based on superpixel segmentation is designed to distinguish heterogeneous regions. In the proposed framework, based on the acquired group information, the sample weights are assigned according to the “easy first” and “more group” principles. Furthermore, a time-varying self-paced regularizer is proposed to automatically provide reasonable learning schemes for SPL.

The rest of this article is organized as follows: Section II gives a brief introduction about the related background and motivation of this article. The GSPL framework proposed for change detection is presented in Section III. Experimental details are given in Section IV. The conclusions are given in Section V.

II. BACKGROUND AND MOTIVATION

A. Self-Paced Learning

Given training set $\mathcal{D} = \{(x_1, y_1), \dots, (x_n, y_n)\}$, let $x_i \in R^m$ represents the i th observed sample, y_i denotes its corresponding label and w denotes the model parameter inside the decision function g . $L(y_i, g(x_i, w))$ represents the loss function which calculates the cost between the ground truth label y_i and the estimated label $g(x_i, w)$. The objective of SPL is to jointly learn the model parameter w and the sample weights $v = [v_1, v_2, \dots, v_n]$ by minimizing

$$\min_{w, v \in [0, 1]^n} E^{SPL}(w, v, \lambda) = \sum_{i=1}^n v_i L(y_i, g(x_i, w)) + f(v; \lambda) \quad (1)$$

where $v_i \in [0, 1]$ is a sample weight that measures how “reliable” or “easy” the training sample is. $f(v; \lambda)$ is the self-paced regularizer that determines how to calculate the sample weights. λ is used to control the learning pace of SPL. Alternative convex search (ACS) is usually adopted to solve (1) [23] during the iteration of SPL. The pseudocode of SPL is shown in Algorithm 1.

Algorithm 1 Algorithm of SPL

```

Input: The training dataset:  $\mathcal{D} = \{(x_1, y_1), \dots, (x_n, y_n)\}$ .
Output: Model parameter  $w$ .
1 Initialize  $v^*$  and pace parameter  $\lambda$ ;
2 while not converged do
3   Update  $w^* = \text{argmin}_w E(w, v^*; \lambda)$ ;
4   Update  $v^* = \text{argmin}_v E(w^*, v; \lambda)$ ;
5   Increase  $\lambda$ ;
6 end
7 return  $w = w^*$ .

```

The SPL model always learns the “reliable” sample first and then gradually learns the “less reliable” sample by increasing λ . When λ is small, only “reliable” samples with small

loss values will be considered into training. With the increase of λ , more and more “less reliable” samples with large losses will be gradually involved to train a more “mature” model.

B. Motivation

1) Motivation of Using GSPL for Unsupervised Change Detection: Supervised classification technique has become a popular and effective tool for change detection because it can significantly improve the performance by using labeled samples. However, there are two main issues to utilize the supervised algorithms for change detection. On the one hand, it is difficult to collect reliable labeled samples. Manual annotation often requires a lot of manpower. The unsupervised approach is both convenient and labor saving, but it will generate many unreliable or noisy labels. On the other hand, it is an urgent task to design a classifier to deal with the acquired labels. Traditional machine learning methods, such as SoftMax (SM) [24], backpropagation NNs (BPNNs) [25], and support vector machine (SVM) [26], may not achieve satisfactory change detection results with samples polluted heavily.

Based on the above-mentioned analysis, the above-mentioned issues can be converted into how to learn a robust classifier using unreliable samples generated by unsupervised methods. Therefore, it is natural to combine SPL with a traditional unsupervised method. The unsupervised method is used to generate the training set with unreliable samples, and then SPL is employed to collect reliable samples from this training set by automatically assigning weights to samples. However, the weights determined by SPL only according to the “easy first” principle may tend to focus on the homogeneous region of the image. The classifier may not be able to learn the characteristics of other regions. The above-mentioned issue can be solved by distinguishing these heterogeneous regions in advance. Therefore, we propose a GSPL framework for change detection, in which the training samples are divided into different groups according to their heterogeneity, and weights are calculated by the “more group” principle together with the “easy first” principle so as to get better change detection performance.

2) Motivation of Proposing Time-Varying Self-Paced Regularizer: The key of SPL is the calculation of sample weights, which is determined by the self-paced regularizer. Several efficient self-paced regularizers have already been established, such as hard weight regularizer, linear weight regularizer, logarithmic weight regularizer, and mixture weight regularizer [27]. The hard weight regularizer assigns weights as 0 or 1 and cannot distinguish the differences between the reliable samples. The other three soft weight regularizers assign weights in the range of [0, 1] and can recognize how reliable the considered samples are. According to the learning mechanism of SPL, in the early stage, only a small number of easy samples are assigned to nonzero weights, which would have an impact on the model parameter. However, all these nonzero weights are too small, which will lead to the low speed of model training at the beginning. To address this

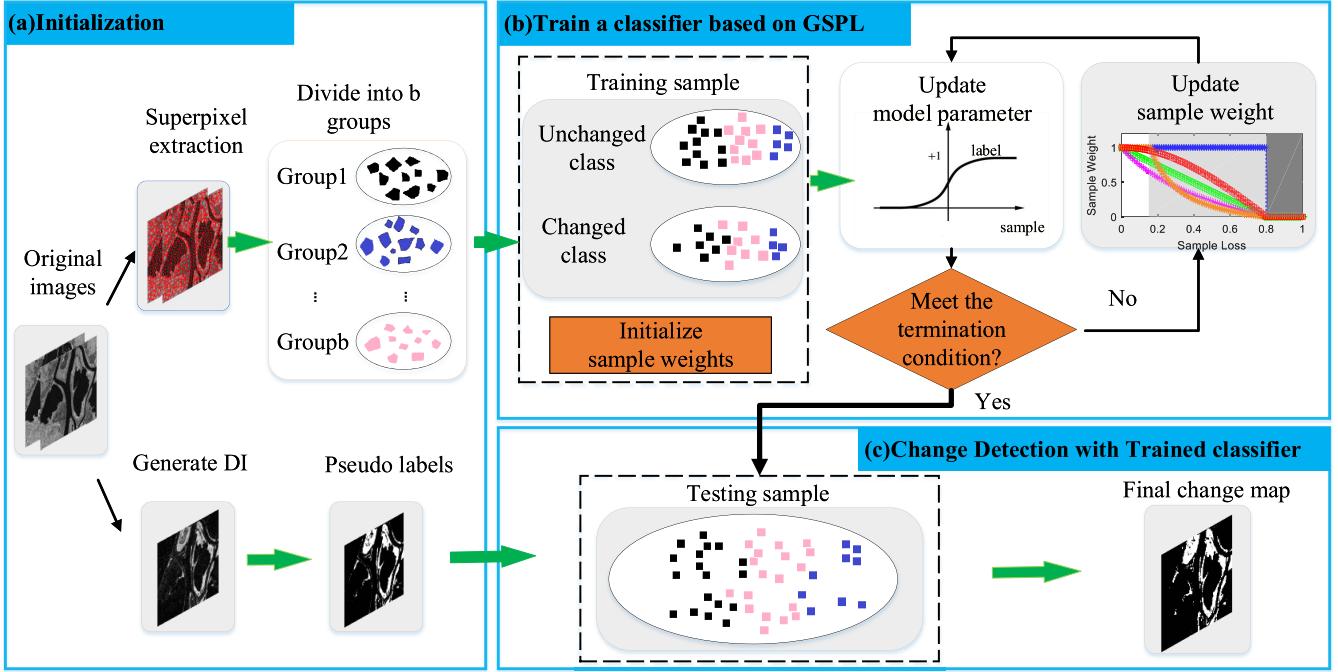


Fig. 1. Workflow of the GSPL framework for change detection.

issue, we propose a time-varying self-paced regularizer for automatically assigning sample weights reasonably.

III. METHODOLOGY

In this section, we first propose the GSPL framework for unsupervised change detection. Then, the generation of pseudolabels and group information of the training set are introduced in detail. Next, we investigate the updating methods of parameters under several popular classifiers. Finally, the time-varying self-paced regularizer is described.

A. GSPL Framework for Unsupervised Change Detection

To further improve the change detection performance, semi-supervised or supervised classifiers are employed to enhance unsupervised change detection performance [28]. In order to achieve satisfactory results, the supervised classifier should be trained with reliable labeled samples, which is not easy to acquire. Unsupervised methods can obtain labeled samples without the need of manpower but may produce many unreliable samples due to the existence of noise in images, such as multiplicative noises in SAR images and additive noises in optical images [29]. Therefore, it is an urgent task to learn a reliable classifier from such a training set with unreliable samples. In this article, SPL is employed to collect reliable samples from the noisy training set. However, the reliable samples selected by the original SPL tend to come from the homogeneous region in images. To address these issues, we propose a GSPL framework for change detection, which introduces group information to help choose reliable samples.

Fig. 1 shows the workflow of the proposed change detection framework. As shown in Fig. 1, the training set, testing set, and group information are obtained in the initialization stage, where the training set is filled with samples with

pseudolabels, and the testing set contains all the samples need to be classified. Then, a robust classifier is established using the training set and its group information. Finally, the change detection result can be obtained by classifying the testing set with the trained classifier.

Assuming that the training set $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\} \in \mathbb{R}^{m \times n}$ is partitioned into b groups: $\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(b)}$, where columns of $\mathbf{X}^{(j)} \in \mathbb{R}^{m \times n_j}$ represents the j th group, $\mathbf{x}_i^{(j)}$ denotes the i th sample in the j th group, n_j is the number of the samples in the j th group and $\sum_{j=1}^b n_j = n$. Accordingly denote the sample weights as $\mathbf{v} = [\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(b)}]$, where $v_i^{(j)}$ denotes the weight of the i th sample in j th group, $\mathbf{v}^{(j)} = (v_1^{(j)}, \dots, v_{n_j}^{(j)})^T \in [0, 1]^{n_j}$. As described earlier, the GSPL model attempts to assign nonzero weights to reliable samples from different groups. This can be realized by optimizing the following objective function:

$$\min_{\mathbf{w}, \mathbf{v}} E^{GSPL}(\mathbf{w}, \mathbf{v}, \lambda) = \sum_{j=1}^b \sum_{i=1}^{n_j} v_i^{(j)} L_i^{(j)}(y_i^{(j)}, g(\mathbf{x}_i^{(j)}, \mathbf{w})) + \sum_{j=1}^b \sum_{i=1}^{n_j} f(v_i^{(j)}; \lambda_i) \quad (2)$$

where $y_i^{(j)}$ denotes the corresponding pseudolabel of the i th sample in j th group, and λ_i is a threshold determining whether the i th sample in a group is “reliable” or not. Similar to SPL, ACS can be applied to optimize (2) for solving the GSPL model, which updates the model parameter and sample weights alternatively. The pseudocode of the GSPL framework for change detection is shown in Algorithm 2.

B. Generation of Pseudotraining Set and Group Information

1) *Generation of Training Set:* In the proposed framework, the training set and testing set have the same data information

Algorithm 2 Algorithm of GSPL for Change Detection

Input: Two images: X_1 and X_2 , the number of groups b and the number of iterations $iterMax$.

Output: A change detection result map.

1 Initialization

2 Generate the DI or the original features;

3 Pre-classify DI or the original features to get pseudo labels;

4 Get the group information of DI or the original features;

5 Generate training sample set $\mathcal{D} = \{\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(b)}\}$ and pseudo label set: $\mathbf{y} = \{\mathbf{y}^{(1)}, \dots, \mathbf{y}^{(b)}\}$;

6 Initialize sample weights $\mathbf{v} = \{\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(b)}\}$, pace parameter: λ^t, γ and the maximum number of iterations: $iterMax$;

7 Cycling

8 if $iter < iterMax$ then

9 Update $\mathbf{w}^* = \arg \min_{\mathbf{w}} E(\mathbf{w}, \mathbf{v}^*; \lambda^t, \gamma)$;

10 Update $\mathbf{v}^* = \arg \min_{\mathbf{v}} E(\mathbf{w}^*, \mathbf{v}; \lambda^t, \gamma)$ using **Algorithm 3**;

11 Calculate λ^t by (18), $iter = iter + 1$;

12 else

13 Stop cycling.

14 end

15 Classify the DI or the original features with the trained classifier.

16 **return** The final change detection result.

obtained from the difference image (DI) or original features of the two original images. In this article, DI is adopted for its simplicity and efficiency in change detection. In general, different types of images should adopt different strategies [30] to generate DI. For example, for SAR images polluted by multiplicative speckle noises, a log-ratio operation is usually adopted [31]. For multispectral images, change vector analysis (CVA) [32] can be adopted to generate a multidimensional DI, containing the magnitude information and direction information. After obtaining DI, a sliding window with size $\rho^2 \times d$ will be used to generate data information for the training set and testing set. Each sample consists of a pixel and its neighborhood of size $\rho^2 \times d$ [28] (d is the dimension of the DI). Then, pseudolabels can be obtained by analyzing the pixels in DI with traditional unsupervised methods.

2) *Generation of Group Information:* An image may have several heterogeneous regions, such as rivers, mountains, wetlands, and grasslands. As shown in Fig. 2(a), unchanged areas contain four types of change information denoted as u_1, u_2, u_3 , and u_4 . The changed areas contain three types of change information denoted as c_1, c_2 , and c_3 . The reliable samples determined by SPL are likely to come from homogeneous regions. For example, Fig. 2(b) (1) indicates that the reliable positive samples are all from c_3 , and the reliable negative samples are all from u_4 . However, the pixel information varies greatly between different regions. To get a more robust classifier, those reliable samples should come from different areas, as shown in Fig. 2(b) (2).

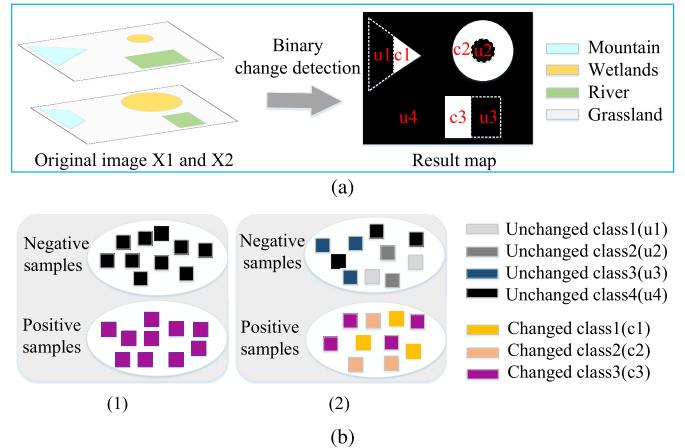


Fig. 2. Examples of binary change detection and sample selection using SPL and GSPL. (a) Example of binary change detection (white areas represent the changed areas and the black areas represent the unchanged areas). (b) Comparison of samples selected by SPL and GSPL. (1) Select samples with SPL. (2) Select samples with GSPL.

Different types of images may be classified into different numbers of groups, which is determined by the imaging mechanism of the image. For example, an SAR image is the reflection of the ground object to the radar beam; therefore, it can only reflect the strength information. Therefore, the number of groups in the SAR image is two, indicating that the reflection coefficient is strong or weak. Therefore, the number of groups in the corresponding change detection map is three, reflecting the information of becoming stronger, weaker, or unchanged. Multispectral images are obtained by capturing multiple bands on the same target and usually contain a large amount of information. According to different spectral features, ground objects can be identified. Therefore, the number of groups in a multispectral image is roughly the number of objects in the image. The change detection map can contain more than three categories, and different categories indicate different change types. Unsupervised methods can be adapted to generate group information. However, group information generated by unsupervised methods directly is susceptible to noise. Therefore, the superpixel segmentation technique, simple linear iterative clustering (SLIC) [33], is used in this framework.

C. Update Model Parameters Under Different Classifiers

Given sample weights \mathbf{v} , the updating of model parameters is only related to the selected samples from the training set and their sample weights. Therefore, in the process of updating model parameters, the training set can be expressed as $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$, and the model parameters can be updated as follows:

$$\begin{aligned} \mathbf{w}^* &= \arg \min_{\mathbf{w}} \sum_{j=1}^b \sum_{i=1}^{n_j} v_i L(y_i, g(\mathbf{x}_i, \mathbf{w})) \\ &= \arg \min_{\mathbf{w}} \sum_{i=1}^n v_i L(y_i, g(\mathbf{x}_i, \mathbf{w})) \end{aligned} \quad (3)$$

where n is the total number of training samples in all groups. A variety of classifiers can be embedded into the

GSPL framework. In this article, SM, BPNN, and SVM are investigated under the GSPL framework for their efficiency in the classification task. Their model parameter updating process will be described in Sections III-C.1–III-C.3.

1) *Group Self-Paced Softmax*: Given training set $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$ and their corresponding sample weights. The model parameters $\boldsymbol{\theta}$ can be obtained as

$$\boldsymbol{\theta} = \arg \min_{\boldsymbol{\theta}} \left(\sum_{i=1}^n v_i L(y_i, g(\mathbf{x}_i; \boldsymbol{\theta})) + \frac{\lambda}{2} \sum_{k=1}^K \sum_{j=0}^m \theta_{kj}^2 \right) \quad (4)$$

where \mathbf{x}_i is the i th sample with $m+1$ features, $\mathbf{x}_i = (1, x_{i1}, x_{i2}, \dots, x_{im})^T$, y_i is the corresponding one-hot encoded label, n is the total number of training samples, $L(y_i, g(\mathbf{x}_i; \boldsymbol{\theta}))$ is the loss value of the i th sample, $\boldsymbol{\theta} = (\theta_0, \theta_1, \dots, \theta_m)$, $\lambda/2 \sum_{k=1}^K \sum_{j=0}^m \theta_{kj}^2$ is a weight penalty item, and K is the number of categories.

With fixed sample weights, the parameter $\boldsymbol{\theta}$ is updated based on the gradient descent method as follows:

$$\boldsymbol{\theta}^{t+1} = \boldsymbol{\theta}^t - \eta \frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \quad (5)$$

where

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_j} = -\sum_{i=1}^n v_i \mathbf{x}_i (\mathbf{I}\{y_i = j\} - p(y_i = j | \mathbf{x}_i; \boldsymbol{\theta})) + \lambda \theta_j \quad (6)$$

$$p(y_i = j | \mathbf{x}_i; \boldsymbol{\theta}) = \frac{e^{\theta_j^T \mathbf{x}_i}}{\sum_{j=1}^K e^{\theta_j^T \mathbf{x}_i}} \quad (7)$$

where $\mathbf{I}\{\cdot\}$ is an indicator function, whose value is 1 or 0 when the expression inside is true or false. The derivation of (6) is described in Section II-A of the Supplementary Materials. We can see from (5) and (6) that the sample weight v together with the step parameter η is the real control parameter of the learning rate. When the samples are reliable, the corresponding weights are high, and the model parameters can be updated in a large step, and vice versa. Samples with 0 weights will have no influence on the model parameters.

2) *Group Self-Paced Neural Networks*: Let $h_0 = \mathcal{D}_i$ denote the input of the networks, $\mathbf{h}_l = f(\mathbf{W}_l \mathbf{h}_{l-1} + b_l)$, $l = 1, 2, \dots, L$ denote the output of the l th layer, and $\mathbf{O}_l = \mathbf{h}_L = f(\mathbf{W}_L \mathbf{h}_{L-1} + b_L)$ denote the final output of the networks. The objective of GSPNN is defined as follows:

$$\min_{\boldsymbol{\theta}, \mathbf{v}} E(\boldsymbol{\theta}, \mathbf{v}; \lambda) = \sum_{i=1}^n (v_i L(\mathbf{O}_i, y_i) + f(v_i; \lambda)) \quad (8)$$

where $\boldsymbol{\theta} = \{\mathbf{W}_1, b_1, \dots, \mathbf{W}_L, b_L\}$, $L(\mathbf{O}_i, y_i)$ is the loss value.

With fixed sample weights, the model parameter is updated based on gradient descent as follows:

$$\mathbf{W}_l = \mathbf{W}_l - \eta v_i (\delta_{l+1} \odot \mathbf{h}_l) \quad (9)$$

$$\begin{aligned} b_l &= b_l - \eta \nabla b_l \\ &= b_l - \eta \frac{\partial v_i \frac{1}{2} \|\mathbf{O}_i - y_i\|^2}{\partial \mathbf{W}_l} \\ &= b_l - \eta v_i \delta_{l+1} \end{aligned} \quad (10)$$

where \odot represents the dot product operation and δ_l is the residual error of the l th layer. The derivation process is explained in Section II-B of the Supplementary Materials.

Different from BPNN, it can be observed from (9) and (10) that the true learning rate of GSPNN is ηv_i , which means that the real learning rate of the GSPNN model is controlled by the sample weights together with η . Furthermore, only samples with nonzero weights can have impact on the model parameter.

3) *Group Self-Paced Support Vector Machine*: Given training set: $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)\}$ and their corresponding weights, the objective of GSPSVM is defined as follows:

$$\begin{aligned} \min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^n (C v_i l_i + f(v_i; \lambda)) \\ \text{s.t. } y_i (\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - l_i \\ l_i \geq 0, \quad i = 1, 2, \dots, n \end{aligned} \quad (11)$$

where l_i is the standard hinge loss, which is calculated as

$$l_i = \max(0, 1 - y_i (\mathbf{w}^T \phi(\mathbf{x}_i) + b)) \quad (12)$$

where $\phi(\cdot)$ is a feature mapping function to obtain nonlinear decision boundaries. C ($C > 0$) is the standard regularization parameter to trade off the hinge loss and the margin. $v_i l_i$ is the weighted loss value of the i th sample.

With fixed sample weights, the model parameter in (11) can be solved by the following dual problem:

$$\begin{aligned} \max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{s.t. } \sum_{i=1}^n \alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq C v_i, \quad i = 1, 2, \dots, n \end{aligned} \quad (13)$$

where $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ is the kernel function. The proof process of (13) is described in Section II-C of the Supplementary Materials.

Compared with the dual form of the original SVM, (13) imposes a sample-specific upper bound on the support vector coefficient, which is the key to computing decision boundary [27]. A sample upper bound is proportional to its weight, and therefore, a sample with a small weight v_i will have less influence on the decision boundary as its support vector coefficient is limited by a small value $C v_i$. Only samples with nonzero weights can be selected to train the classifier in the change detection task. When samples' weights are 0, the corresponding samples will not be selected into the training set and will have no influence on the decision boundary.

D. Update Sample Weights With a Time-Varying Self-Paced Regularizer

With a fixed model parameter \mathbf{w} in (2), the sample weights can be updated by optimizing

$$\min_{\mathbf{v}} E(\mathbf{v}) = \sum_{j=1}^b \sum_{i=1}^{n_j} (v_i^{(j)} L_i^{(j)} + f(v_i^{(j)}; \lambda_i^t)). \quad (14)$$

To calculate the sample weights, the pace parameter λ_i^t and the self-paced regularizer $f(v; \lambda_i^t)$ should be specified in advance.

For the self-paced regularizer, several soft self-paced regularizers have been proposed, such as linear soft weight, logarithmic soft weight, and mixture weight [27]. However, all the soft weight methods assign small weights to the reliable samples in the early stage of SPL. To deal with this situation, we propose a novel self-paced regularizer in this framework. The proof of it is explained in Section I-B of the Supplementary Materials. The proposed regularizer $f(v_i; \lambda^t)$ can be described as follows:

$$f(v_i; \lambda^t) = -\frac{2\lambda^t}{\pi} (v_i \arccos(v_i) - \sqrt{1 - v_i^2}). \quad (15)$$

Supplementary materials can prove that (15) is convex with respect to v in $[0, 1]$. Therefore, the global optimal solution of (14) can be obtained at $\nabla_v E(\mathbf{v}) = 0$. Therefore, we have

$$\frac{\partial E}{\partial v_i} = L_i - \frac{2\lambda^t}{\pi} \arccos(v_i) = 0. \quad (16)$$

From (16), the closed-form optimal solution for v_i ($i = 1, 2, \dots, n$) can be obtained as

$$v_i = \begin{cases} \cos\left(\frac{\pi}{2\lambda^t} L_i\right), & L_i < \lambda^t \\ 0, & \text{else.} \end{cases} \quad (17)$$

A time-varying pace parameter λ_i^t is designed in this article so that reliable samples distributed in the heterogeneous regions of the image with low-loss values will be assigned a large weight in the early stage. The calculation of λ_i^t is defined as follows:

$$\lambda_i^t = \lambda + \gamma q^t(i) \quad (18)$$

$$q^t(i) = \frac{1}{C^t \sqrt{i}}, \quad i = 1, 2, \dots \quad (19)$$

$q^t(i)$ in (18) is a decreasing function indicating that samples in the same group have a monotonically decreasing weights with respect to their corresponding rank i in the group. When $\gamma = 0$, λ_i^t is a constant value; only easy samples with low loss values will be selected into the training set, which may also tend to come from a homogeneous region in images. When $\gamma \neq 0$, the pace parameter λ_i^t will decrease, and the samples distributed in heterogeneous regions will be selected.

C^t in (19) is used to automatically provide reasonable learning schemes at different stages of change detection. In the early stage, only a small number of reliable samples will be given nonzero weights to train a classifier, these nonzero weights need to be large numbers so that the parameters of the classifier can be updated in a large step. In the later stages, more reliable samples will be given nonzero weights, but only a part of these reliable samples with lowest loss values should be given large weights, and others should be given small weights. Considering the above-mentioned situations, we design C^t by the following formula:

$$C^t = \tan\left(\frac{\pi}{2}\left(1 - \frac{\text{iter}^t}{\text{iterMax} + 1}\right)\right), \quad \text{iter}^t = 1, 2, \dots \quad (20)$$

where iter^t is the current iteration number and iterMax is the maximum number of iterations.

The algorithm of calculating sample weights is presented in Algorithm 3. In Fig. 3, we use a simple example to show

Algorithm 3 Algorithm of Calculating Sample Weights

Input: The training sample data set:
 $\mathcal{D} = \{\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(b)}\}$ and pseudo label set:
 $\mathbf{y} = \{y^{(1)}, \dots, y^{(b)}\}$.

Output: The global solution $\mathbf{v} = \{\mathbf{v}^{(1)}, \dots, \mathbf{v}^{(b)}\}$.

```

1 for  $j = 1$  to  $b$  do
2   Sort the samples in  $\mathbf{X}^{(j)}$  in ascending order of their
      loss values;
3   Accordingly, denote the labels and weights of  $\mathbf{X}^{(j)}$  as
       $(y_1^{(j)}, \dots, y_{n_j}^{(j)})$  and  $(v_1^{(j)}, \dots, v_{n_j}^{(j)})$ ;
4   for  $i = 1$  to  $n_j$  do
5     if  $L_i^{(j)} < (\lambda^t + \gamma \frac{1}{C^t \sqrt{i}})$  then
6        $v_i^{(j)} = \cos(\frac{\pi}{2(\lambda^t + \gamma \frac{1}{C^t \sqrt{i}})} L_i^{(j)})$ ;
7     else
8        $v_i^{(j)} = 0$  ;
9     end
10   end
11 end
12 return  $\mathbf{v}$ .

```

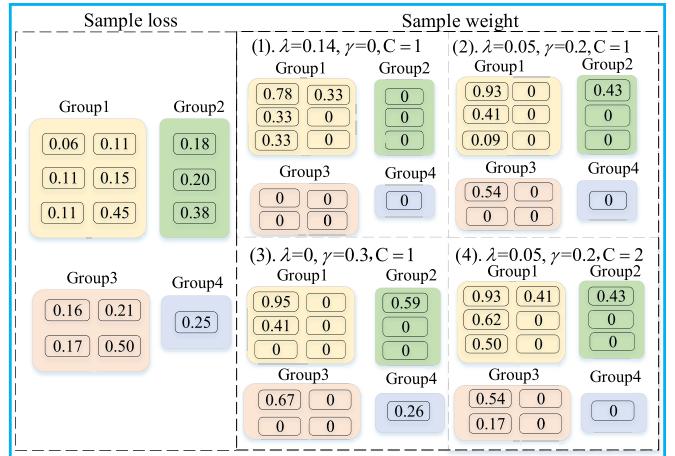


Fig. 3. Example of sample weights calculated by Algorithm 3.

the calculation of sample weight under different parameters. In Fig. 3, squares with different colors represent different areas in the image, and boxes in colored blocks represent samples collected from the DI or the original features. When $\gamma = 0$ in Fig. 3(1), GSPL considers only the “easy first” principle and assigns nonzero weights only according to the sample loss, which is the same as the original SPL. Meanwhile, samples with the same loss value in a group share the same sample weight. When $\gamma \neq 0$ and $\lambda^t \neq 0$ in Fig. 3(2) and 3(4), GSPL considers both “easy first” principle and “more group” principle so that samples scattered in more different groups are assigned to nonzero weights, thus producing more reasonable weight distributions. In addition, samples sharing the same loss value in a group are assigned to different weights because of the decrease of threshold in step 6 in Algorithm 3. Moreover, GSPL can control the distribution of weights in a group by

TABLE I
PROPERTIES OF DATA SET

Name	Size	Location	Time t_1	Time t_2
Ottawa dataset	290×350	Ottawa, Canada	May, 1997	August, 1997
Stone-Gate dataset	349×252	Taiwan, China	August, 2004	September, 2004
San Francisco dataset	390×370	San Francisco, America	August, 2003	May, 2004
Yellow River dataset	350×400	Yellow River Estuary, China	June, 2008	June, 2009
Bahe dataset	350×350	Xi'an, China	August, 2013	August, 2015

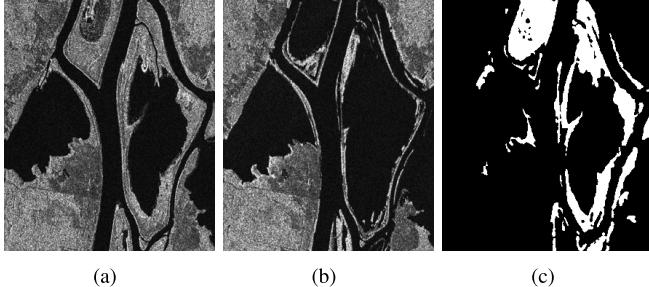


Fig. 4. Ottawa data set. (a) Image acquired in May 1997. (b) Image acquired in August 1997. (c) Reference image.

the value of C —weights in a group calculated with $C = 1$ is not greater than the corresponding weights calculated with $C = 2$, as shown in Fig. 3(2) and Fig. 3(4). When $\gamma = 0$ and $\lambda^t \neq 0$ in Fig. 3(3), GSPL considers only “more group” principle, ignoring the “easy first” principle, which would lead to outliers being assigned to nonzero weights, as seen that the sample in Group 4, whose loss value is 0.25, has a nonzero weight 0.26. Therefore, considering both “easy first” principle and “more group” principle seems to be more reasonable for change detection than considering anyone alone.

IV. EXPERIMENTAL STUDY

In the proposed framework, SLIC [33] is applied to extract superpixel blocks. The FCM algorithm is employed as the unsupervised method to generate group information. In this section, the data sets and evaluation criteria will be presented first. Then, four parts of the experiments are carried out for evaluating the proposed framework. At last, contrast experiments with other methods will be shown.

A. Data Sets and Evaluation Criteria

Five remote sensing image data sets are investigated to validate the effectiveness of the proposed framework for change detection, including an SAR image data set for binary change detection, an optical image data set of panchromatic band for binary change detection, two SAR image data sets for ternary change detection, and a multispectral image data set for multiclass change detection, as shown in Fig. 4–Fig. 8. The size, the imaging location, and imaging time of each data set are listed in Table I.

To evaluate the change detection results quantitatively, it is necessary to adopt some criteria. Four evaluation criteria are adopted in our experiments, including the overall error (OE),

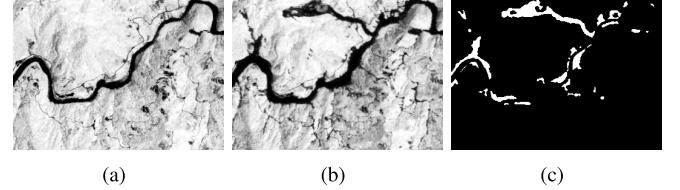


Fig. 5. Stone-Gate data set. (a) Image acquired in August 2004. (b) Image acquired in September 2004. (c) Reference image.

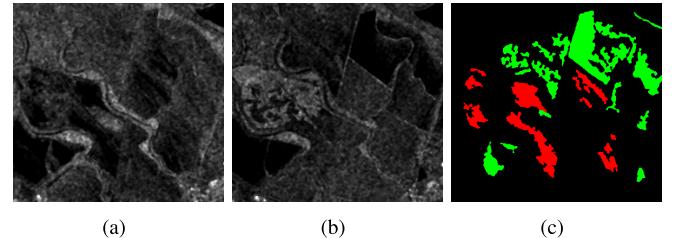


Fig. 6. San Francisco data set. (a) Image acquired in August 2003. (b) Image acquired in May 2004. (c) Reference image.

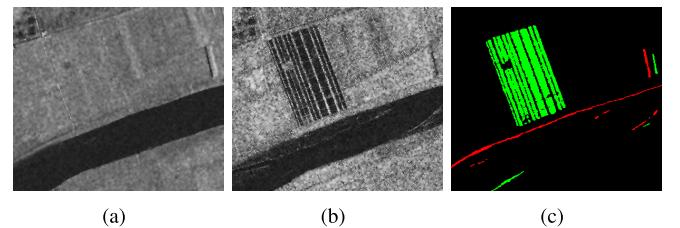


Fig. 7. Yellow River data set. (a) Image acquired in June 2008. (b) Image acquired in June 2009. (c) Reference image.

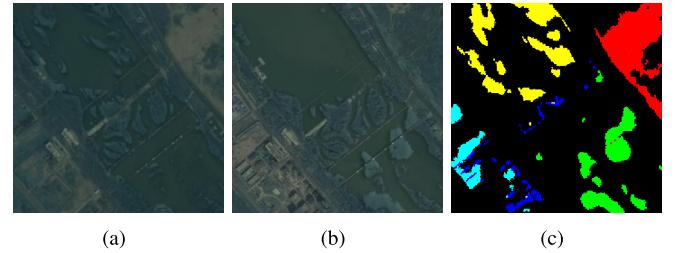


Fig. 8. Bahe data set. (a) Image acquired in August 2013. (b) Image acquired in August 2015. (c) Reference image.

percentage of correct classification (PCC), kappa coefficient (KC) [34], and normalized mutual information (NMI). OE is the number of pixels misclassified. PCC represents the proportion of pixels classified correctly. KC and NMI are two different indices that can measure the agreement of experimental results and the reference map.

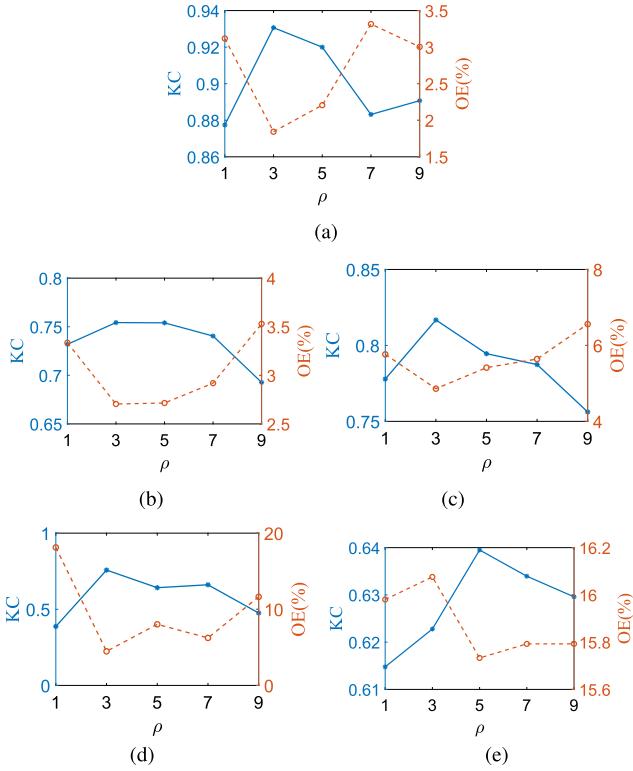


Fig. 9. Change detection results with different neighborhood size ρ . (a) Ottawa data set. (b) Stone-Gate data set. (c) San Francisco data set. (d) Yellow River data set. (e) Bahe data set.

B. Experiments on the Proposed GSPL Framework

This part of the experiment aims to demonstrate the effectiveness of adopting SPL, group information, and time-varying self-paced regularizer in the proposed framework. Specifically, we first analyzed the influence of neighborhood size on change detection performance, and then, tested the detection results based on the original classifier, SPL-based classifier, and GSPL-based classifier. Finally, we compared the proposed time-varying self-paced regularizer with other regularizers to verify the validity of the proposed regularizer. In addition, for the sake of handwriting convenience, we abbreviate the SM, NN, and SVM based on SPL as SPSM, SPNN, and SPSVM, respectively.

1) *Test of the Neighborhood Size ρ* : In the proposed GSPL framework, a sample is generated by extracting the local neighborhood of the current pixel in the DI with the size of $\rho \times \rho$. In the experiments, the parameter ρ is set to 1, 3, 5, 7, and 9, respectively.

The values of OE and KC on the five data sets are displayed in Fig. 9 in the form of a broken line graph. According to Fig. 9, the best change detection results are acquired by the proposed method on the first four data sets when the parameter ρ is set to 3. For the Bahe data set, the best results are obtained when $\rho = 5$. When ρ is small, the generated samples may lose some vital neighborhood information. With the increase of ρ , more neighborhood information is considered to learn the difference between the multitemporal images. However, when ρ is too large, it is difficult to distinguish the difference

between two adjacent samples since they almost share the same neighborhood information. In general, satisfactory results can be obtained by the proposed method when the parameter ρ is set to 3 or 5.

2) *Test of the Proposed Time-Varying Self-Paced Regularizer*: In order to demonstrate the effectiveness of the proposed time-varying self-paced regularizer, we investigate it in the proposed GSPL framework in comparison with the linear soft weight, the logarithmic soft weight, and the mixture weight regularizers. The results are drawn in Fig. 10, where the ordinate in the bar charts of Fig. 10 represents the KC values.

As shown in Fig. 10, except for the results of GSPSVM on the Ottawa data set, GSPSM, GSPNN, and GSPSVM always acquire the highest KC values with the time-varying regularizer because the proposed regularizer is able to provide reasonable learning schemes during iterations. In the early stage of learning with a small pace parameter λ , the involved samples tend to be easy and should be given large weights. The proposed self-paced regularizer can automatically assign reasonable weight schemes at different stages of SPL.

C. Ablation Studies

In this experiment, we aim at investigating the proposed grouping strategy based on the five change detection data sets. Fig. 11 shows the curves of the KC values obtained by SPNN and GSPNN with respect to the pace parameter. All the SPL-based methods gradually increase the pace parameter to involve more samples into training. Due to the increasing pace parameter, we cannot guarantee the convergence of the whole SPL algorithm. However, for a fixed pace parameter, according to the theoretical studies [20], the procedure of SPL is able to converge to a stationary solution. Therefore, it consists of an inner loop for each pace, and the resulting stationary results are reported in Fig. 11. As shown in Fig. 11, the best KC values obtained by GSPNN are larger than those of SPNN. Note that the algorithm can be terminated when the best results have been acquired. From Fig. 11, GSPNN is always the first to acquire the best KC values on the five change detection data sets in comparison with SPNN. It is reasonable to assign the weights under the “more group” principle together with the “easy first” principle.

Fig. 12 shows the change detection results of NN, SPNN, and GSPNN on the five change detection data sets. It can be seen from Fig. 12 that the results obtained by SPNN and GSPNN are much better than those of NN. Therefore the proposed SPL framework for change detection is efficient to detect the changes between the multitemporal images. For the Yellow River data set and the Bahe data set, the results of GSPNN are much better than those obtained by SPNN. It is difficult to distinguish the difference between the results obtained by SPNN and GSPNN for the other data sets.

In order to further investigate the proposed grouping strategy, we employ two tests of significance for comparison of multiple algorithms over multiple data sets [35], [36]. First, we take the Friedman test followed by the Nemenyi test into consideration [35]. This comparison involves the SPL and GSPL methods based on SM, NN, and SVM with five change

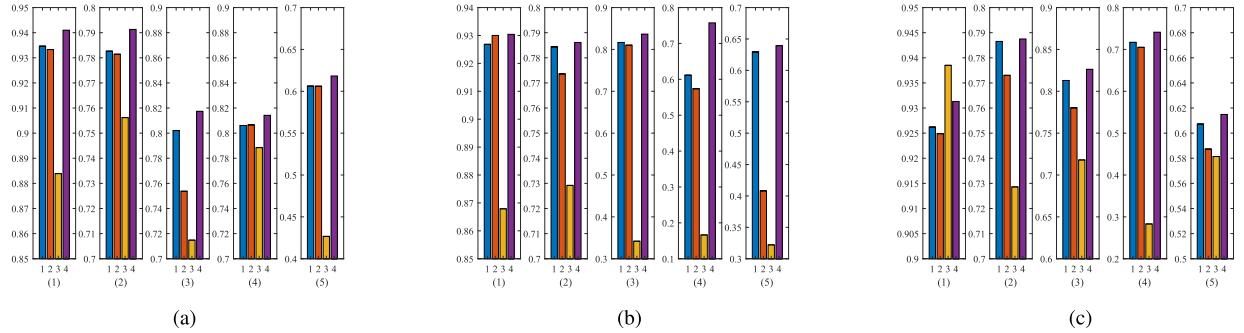


Fig. 10. KC values of GSPSM, GSPNN and GSPSVM under 1: linear soft weight, 2: logarithmic soft weight, 3: mixture weight, and 4: the proposed time-varying self-paced regularizer on five real data sets. (a) GSPSM. (b) GSPNN. (c) GSPSVM.

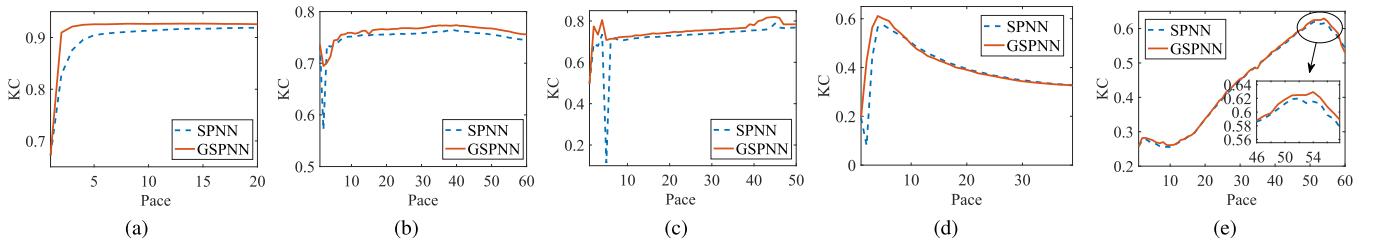


Fig. 11. Curves of KC value with respect to iteration number in SPNN and GSPNN on five real data sets. (a) Ottawa data set. (b) Stone-Gate data set. (c) San Francisco data set. (d) Yellow River data set. (e) Bahe data set.

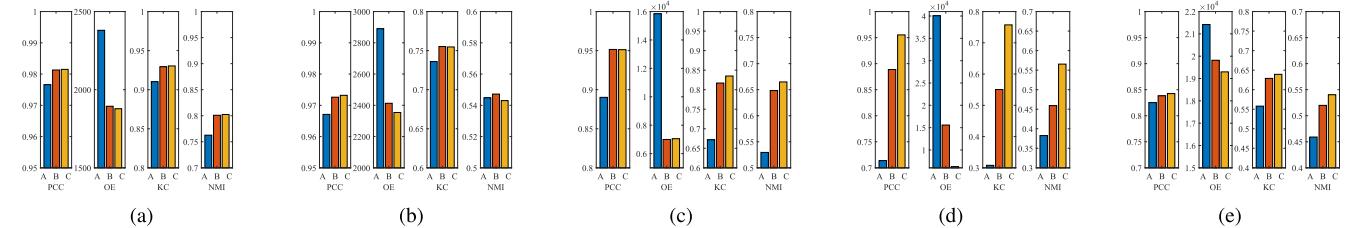


Fig. 12. Change detection results of A: NN (blue bar), B: SPNN (red bar), and C: GSPNN (yellow bar) on the five data sets. (a) Ottawa data set. (b) Stone-Gate data set. (c) San Francisco data set. (d) Yellow River data set. (e) Bahe data set.

detection data sets. Therefore, we can acquire 15 sets of change detection results for each method. The Friedman statistics for PCC, OE, KC, and NMI are 7.8750, 7.8750, 16.2885, and 5.5652, respectively. Then, the p -values for PCC, OE, KC, and NMI are 0.0049, 0.0049, 2.9758E-04, and 0.0135, respectively. All the above-mentioned values are less than 0.05. Therefore, we conclude that the tested algorithms are significantly different in the four criteria, with a confidence level of 95%. Next, the Nemenyi test is employed to compare the SPL and GSPL frameworks. The differences between SPL and GSPL for PCC, OE, KC, and NMI are 0.6000, 0.6000, 0.7333, and 0.5333, respectively. All the above-mentioned values are larger than the critical difference 0.5061. It can be observed that the SPL and GSPL frameworks are significantly different in terms of the four criteria based on the Nemenyi test.

D. Comparison of the Proposed Method With Other Methods

The contrast experiments are conducted on the five data sets to substantiate the superiority of the proposed GSPL change detection framework. For the binary change detection task,

the FCM algorithm [37], the FLICM algorithm [15], the fuzzy clustering algorithm with a modified MRF energy function (MRFFCM) [38], the EM algorithm, and the generalized KI algorithm (GKI) [39] are considered as the competing schemes. For the ternary change detection task, CM, FLICM, KWFLICM, EM, and Otsu [12] are selected. For multiclass change detection in multispectral images, CVA [32], principal components analysis (PCA) [40], multivariate alteration detection (MAD) [41], the iteratively reweighted multivariate alteration detection (IRMAD) [42], and CVA_T [43] are adopted as the contrast methods.

1) Results on Ottawa Data Set: The Ottawa data set is a section of two SAR images over the city of Ottawa, ON, Canada, acquired by the RADARSAT SAR sensor. The two images were taken in May and August in 1997, respectively. The areas were once afflicted with floods.

The change detection maps obtained by different methods on the Ottawa data set are shown in Fig. 13. As shown in Fig. 13, the change maps obtained by FCM, EM, and GKI consist of a lot of noise. Many unchanged areas are wrongly detected as the changed ones. FCM clusters samples according

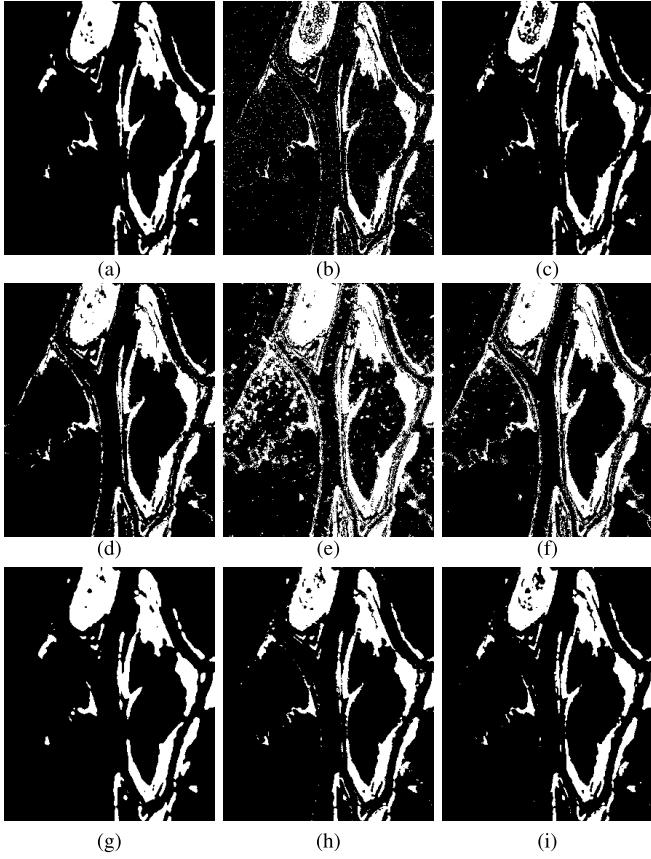


Fig. 13. Reference image and change detection results on Ottawa data set. (a) Reference. (b) FCM. (c) FLICM. (d) MRFFCM. (e) EM. (f) GKI. (g) GSPSM. (h) GSPNN. (i) GSPSV.

TABLE II

QUANTITATIVE COMPARISON AMONG DIFFERENT METHODS ON OTTAWA DATA SET. THE BEST RESULTS ARE REPRESENTED IN BOLD AND THE SECOND BEST RESULTS ARE IN ITALICS

Methods	PCC	OE	KC	NMI
FCM	0.9524	4829	0.8185	0.5956
FLICM	0.9727	2776	0.8913	0.7461
MRFFCM	0.9604	4024	0.8480	0.6455
EM	0.8696	13234	0.6295	0.4400
GKI	0.9387	6220	0.7971	0.6041
GSPSM	0.9789	2144	0.9217	0.7844
GSPNN	<i>0.9815</i>	<i>1878</i>	<i>0.9304</i>	<i>0.8024</i>
GSPSV	0.9822	1800	0.9314	0.8149

to the membership between the samples and the centers, which cannot mitigate the impacts of noise pixels. FLICM takes the neighborhood information into consideration to classify the DI, and MRFFCM utilizes a modified MRF energy function to modify the membership of each pixel. Therefore, both FLICM and MRFFCM are robust to noise. It is obvious that the change maps obtained by the proposed GSPSM, GSPNN, and GSPSV are close to the reference map. Table II lists the evaluation criteria values. EM obtains the worst results with quite a number of pixels misclassified. The KC values of the proposed three methods are larger than 0.9.

2) *Results on Stone-Gate Data Set:* This data set records the changes in the Stone-Gate Reservoir, Taiwan. The two images

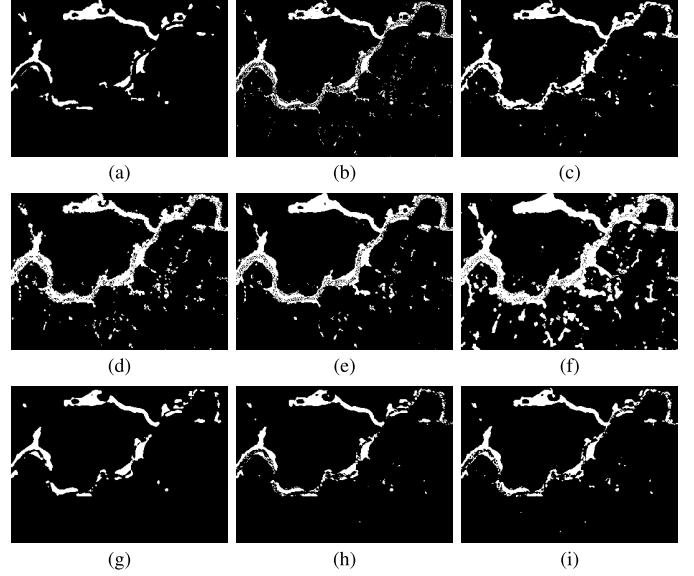


Fig. 14. Reference image and change detection results on Stone-Gate data set. (a) Reference. (b) FCM. (c) FLICM. (d) MRFFCM. (e) EM. (f) GKI. (g) GSPSM. (h) GSPNN. (i) GSPSV.

TABLE III
QUANTITATIVE COMPARISON AMONG DIFFERENT METHODS ON STONE-GATE DATA SET. THE BEST RESULTS ARE REPRESENTED IN BOLD AND THE SECOND BEST RESULTS ARE IN ITALICS

Methods	PCC	OE	KC	NMI
FCM	0.9673	2875	0.7286	0.5241
FLICM	0.9663	2965	0.7342	0.5492
MRFFCM	0.9376	5489	0.6050	0.4580
EM	0.9450	4835	0.6375	0.4891
GKI	0.8733	11141	0.4145	0.3162
GSPSM	0.9773	1995	0.7842	0.5833
GSPNN	0.9772	2008	<i>0.7861</i>	<i>0.5868</i>
GSPSV	0.9760	2112	0.7875	0.5946

were acquired by the Formosat-2 sensor in August and September in 2004, respectively. The typhoon led to the collapse of the ground with bare land emerging.

The change detection maps obtained by different methods on the Stone-Gate data set are shown in Fig. 14. It can be observed that MRFFCM, EM, and GKI get the worst results with numerous noise points in the final maps. The change maps obtained by the proposed three methods have a relatively clear background. Table III lists the change detection results based on the four criteria. GKI receives the highest OEs because many unchanged regions are wrongly classified. The PCC values of the proposed three schemes are larger than 0.97. The proposed methods obtain the highest PCC, KC, and NMI values and the lowest OE values.

3) *Results on San Francisco Data Set:* San Francisco data set was acquired by the ESAERS-2 satellite over the city of San Francisco, CA, USA, in August 2003, and May 2004, respectively. In Fig. 15, the black areas indicate the unchanged areas. The red and green areas indicate two different kinds of change information.

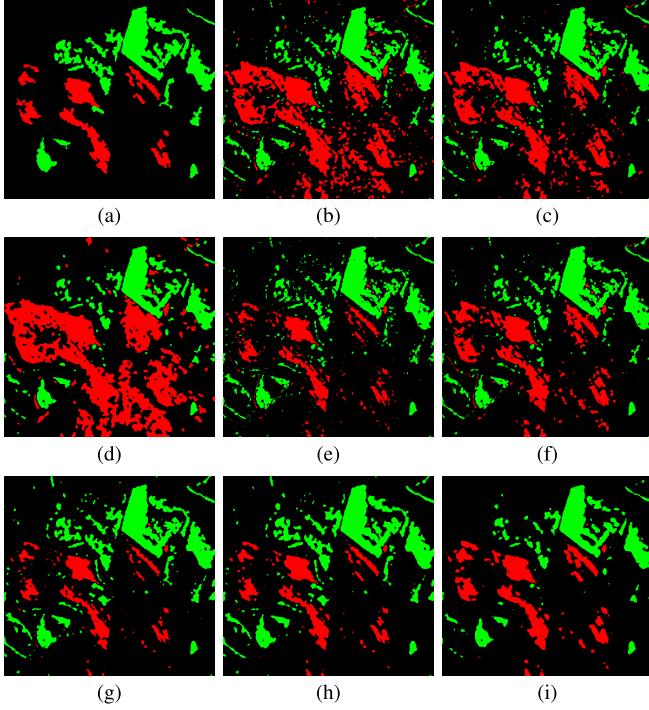


Fig. 15. Reference image and change detection results on San Francisco data set. (a) Reference. (b) FCM. (c) FLICM. (d) KWFLICM. (e) EM. (f) OTSU. (g) GSPSM. (h) GSPNN. (i) GSPSVM.

TABLE IV

QUANTITATIVE COMPARISON AMONG DIFFERENT METHODS ON SAN FRANCISCO DATA SET. THE BEST RESULTS ARE REPRESENTED IN BOLD AND THE SECOND BEST RESULTS ARE IN ITALICS

Methods	PCC	OE	KC	NMI
FCM	0.8672	19161	0.6249	0.4823
FLICM	0.9089	13144	0.7176	0.5530
KWFLICM	0.8052	28111	0.5117	0.4417
EM	0.9380	8940	0.7812	0.5812
OTSU	0.9277	10436	0.7628	0.5764
GSPSM	0.9471	<i>7636</i>	0.8174	0.6370
GSPNN	0.9512	7049	0.8350	0.6649
GSPSVM	0.9512	7049	0.8263	0.6621

The change detection maps obtained by different methods on the San Francisco data set are shown in Fig. 15. It can be observed that many unchanged regions are detected as the changed ones in the change maps obtained by FCM, FLICM, and KWFLICM. The FCM-based algorithms may not be good at classifying the unbalanced data sets since the number of red pixels is smaller than that of green pixels. The change detection results obtained by the eight algorithms are shown in Table IV. The OE values of the proposed three methods are smaller than 8000, which indicates the superiority of the proposed GSPL framework.

4) *Results on Yellow River Data Set:* The Yellow River data set was acquired by Radarsat-2 over the Yellow River Estuary, China, in June 2008 and June 2009, respectively. It is worth noting that the two images are the single-look image and four-look image, respectively. It means that the

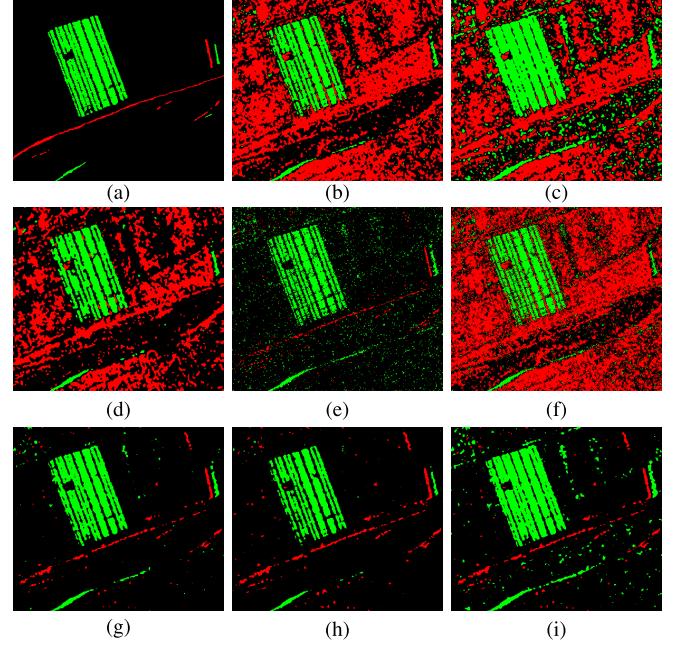


Fig. 16. Reference image and change detection results on Yellow River data set. (a) Reference. (b) FCM. (c) FLICM. (d) KWFLICM. (e) EM. (f) OTSU. (g) GSPSM. (h) GSPNN. (i) GSPSVM.

TABLE V
QUANTITATIVE COMPARISON AMONG DIFFERENT METHODS ON YELLOW RIVER DATA SET. THE BEST RESULTS ARE REPRESENTED IN BOLD AND THE SECOND BEST RESULTS ARE IN ITALICS

Methods	PCC	OE	KC	NMI
FCM	0.5399	64413	0.1975	0.3746
FLICM	0.5553	62253	0.2220	0.2971
KWFLICM	0.7138	40062	0.3080	0.3825
EM	0.9290	9941	0.6529	0.4067
OTSU	0.4862	71932	0.1557	0.3090
GSPSM	0.9636	5090	0.8142	0.6360
GSPNN	<i>0.9552</i>	<i>6273</i>	<i>0.7574</i>	<i>0.5655</i>
GSPSVM	0.9423	8078	0.7412	0.5498

influence of speckle noise on the image acquired in 2008 is much greater than that acquired in 2009. The huge difference in the speckle noise level between the two images makes the change detection task fairly complicated.

The change detection maps obtained by different methods on the Yellow River data set are shown in Fig. 16. These traditional unsupervised methods, except the EM algorithm, obtain poor results with a great number of noise points in the results. On the contrary, the maps obtained by the methods under the proposed framework have a relatively clear background. The two types of changed regions are detected precisely. Table V lists the evaluation criteria values obtained by the eight algorithms. The values of PCC obtained by the proposed GSPL framework are larger than 0.94. The results obtained by GSPSM rank first in comparison with other methods.

5) *Results on Bahe Data Set:* The Bahe data set was acquired by the GF-1 sensor over the city of Xi'an, China,

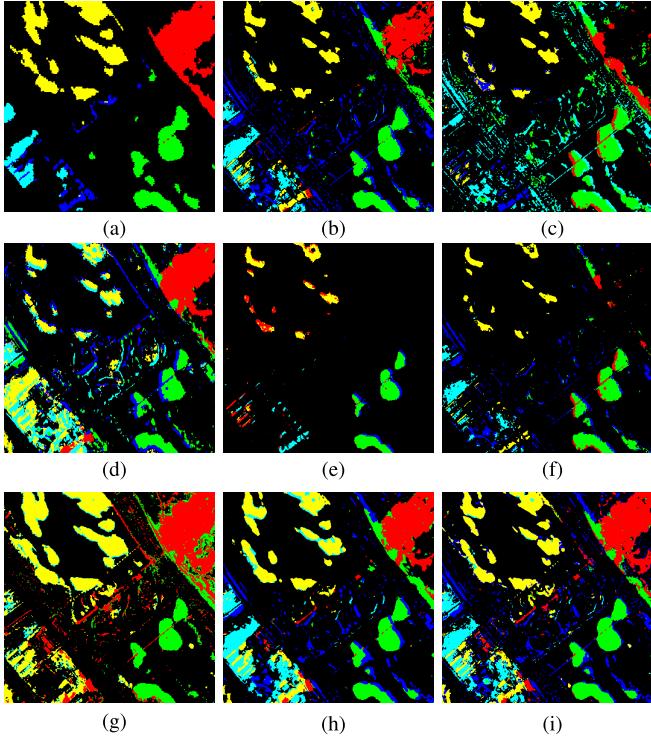


Fig. 17. Reference image and change detection results on Bahe data set. (a) Reference. (b) CVA. (c) PCA. (d) MAD. (e) IRMAD. (f) CVA_T. (g) GSPSM. (h) GSPNN. (i) GSPSVM.

TABLE VI

QUANTITATIVE COMPARISON AMONG DIFFERENT METHODS ON BAHE DATA SET. THE BEST RESULTS ARE REPRESENTED IN BOLD AND THE SECOND BEST RESULTS ARE IN ITALICS

Methods	PCC	OE	KC	NMI
CVA	0.7945	25172	0.4125	0.3428
PCA	0.7703	28138	0.3812	0.3249
MAD	0.7874	26039	0.5165	0.3875
IRMAD	0.8105	23220	0.2859	0.3274
CVA_T	0.7995	24563	0.3355	0.3063
GSPSM	0.8278	21096	0.6183	0.4639
GSPNN	0.8425	19298	0.6393	0.5404
GSPSVM	0.8263	21279	0.6149	0.4958

in August 2013 and August 2015, respectively. The two images have the same spatial resolution of 2 m. In the reference, the black areas indicate the unchanged areas, and the other colored areas indicate changed areas with different change information.

The change detection maps obtained by different methods on the Bahe data set are shown in Fig. 17. Among these competing methods, CVA achieves the best result visually. On the contrary, IRMAD obtains the worst result with many undetected regions. Table VI lists the evaluation criteria values obtained by the eight algorithms. PCA achieves the worst result with a lot of misclassified pixels because the dimensionality reduction of PCA leads to the loss of useful information. The KC values of GSPSM, GSPNN, and GSPSVM are larger than 0.6, and the NMI values of GSPSM, GSPNN, and GSPSVM are larger than 0.4. The results have demonstrated the effectiveness of the proposed GSPL framework.

V. CONCLUSION

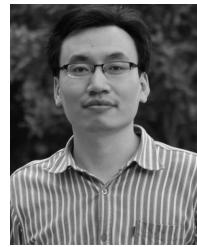
In the image change detection task, unsupervised change detection methods based on a supervised classifier have achieved strong adaptability and robustness and have achieved good change detection performance. However, it is difficult to obtain high-quality labeled samples for training. In this article, we put forward a GSPL framework to address the above issue. Different from the existing change detection methods using a supervised classifier, in the proposed framework, each sample is assigned to weight for indicating its reliability, and only samples with nonzero sample weights can have an influence on the model parameter. Specifically, in the phase of updating weights, group information is integrated to avoid the training samples coming from the homogeneous region. Besides, a time-varying self-paced regularizer is proposed to automatically determine the learning schemes for SPL.

Our theoretical analysis shows that the proposed framework can mine reliable samples from heterogeneous regions in images, and the proposed time-varying self-paced regularizer can provide a reasonable learning scheme for SPL. Furthermore, experiments on five real data sets have demonstrated the feasibility and effectiveness of the proposed framework, which can acquire satisfactory change detection results with high PCC, KC, and NMI values.

REFERENCES

- [1] S. Liu, L. Bruzzone, F. Bovolo, M. Zanetti, and P. Du, "Sequential spectral change vector analysis for iteratively discovering and detecting multiple changes in hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 8, pp. 4363–4378, Aug. 2015.
- [2] G. Liu, Y. Gousseau, and F. Tupin, "A contrario comparison of local descriptors for change detection in very high spatial resolution satellite images of urban areas," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 3904–3918, Jun. 2019.
- [3] S. Saha, F. Bovolo, and L. Bruzzone, "Unsupervised deep change vector analysis for multiple-change detection in VHR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 6, pp. 3677–3693, Jun. 2019.
- [4] H. Li, M. Gong, C. Wang, and Q. Miao, "Self-paced stacked denoising autoencoders based on differential evolution for change detection," *Appl. Soft Comput.*, vol. 71, pp. 698–714, Oct. 2018.
- [5] V. Akbari, A. P. Doulgeris, and T. Eltoft, "Monitoring glacier changes using multitemporal multipolarization SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 6, pp. 3729–3741, Jun. 2014.
- [6] O. Yousif and Y. Ban, "Improving SAR-based urban change detection by combining MAP-MRF classifier and nonlocal means similarity weights," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 10, pp. 4288–4300, Oct. 2014.
- [7] L. Gueguen and R. Hamid, "Toward a generalizable image representation for large-scale change detection: Application to generic damage analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 6, pp. 3378–3387, Jun. 2016.
- [8] J. Lu, J. Li, G. Chen, L. Zhao, B. Xiong, and G. Kuang, "Improving pixel-based change detection accuracy using an object-based approach in multitemporal SAR flood images," *IEEE Trans. Geosci. Remote Sens.*, vol. 8, no. 7, pp. 3486–3496, Jul. 2015.
- [9] F. D. Frate, F. Pacifici, and D. Solimini, "Monitoring urban land cover in Rome, Italy, and its changes by single-polarization multitemporal SAR images," *IEEE J. Sel. Topics Appl. Earth Observat. Remote Sens.*, vol. 1, no. 2, pp. 87–97, Jun. 2008.
- [10] 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.
- [11] Y. Zheng, L. Jiao, H. Liu, X. Zhang, B. Hou, and S. Wang, "Unsupervised saliency-guided SAR image change detection," *Pattern Recognit.*, vol. 61, pp. 309–326, Jan. 2017.
- [12] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

- [13] J. Kittler and J. Illingworth, "Minimum error thresholding," *Pattern Recognit.*, vol. 19, no. 1, pp. 41–47, 1986.
- [14] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc., B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.
- [15] N. S. Mishra, S. Ghosh, and A. Ghosh, "Fuzzy clustering algorithms incorporating local information for change detection in remotely sensed images," *Appl. Soft Comput.*, vol. 12, no. 8, pp. 2683–2692, 2012.
- [16] M. Gong, Y. Liang, J. Shi, W. Ma, and J. Ma, "Fuzzy C-means clustering with local information and kernel metric for image segmentation," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 573–584, Feb. 2013.
- [17] L. Wu, B. Liu, and B. Zhao, "Unsupervised change detection of remote sensing images based on SURF and SVM," in *Proc. Int. Conf. Comput. Intell. Inf. Syst.*, Apr. 2017, pp. 214–218.
- [18] Y. Li, L. Xu, and T. Liu, "Unsupervised change detection for remote sensing images based on object-based MRF and stacked autoencoders," in *Proc. Int. Conf. Orange Technol.*, Dec. 2016, pp. 64–67.
- [19] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2010, pp. 1189–1197.
- [20] M. Gong, H. Li, D. Meng, Q. Miao, and J. Liu, "Decomposition-based evolutionary multiobjective optimization to self-paced learning," *IEEE Trans. Evol. Comput.*, vol. 23, no. 2, pp. 288–302, Apr. 2019.
- [21] H. Li and M. Gong, "Self-paced convolutional neural networks," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 2110–2116.
- [22] R. Shang, Y. Yuan, L. Jiao, Y. Meng, and A. M. Ghalamzan, "A self-paced learning algorithm for change detection in synthetic aperture radar images," *Signal Process.*, vol. 142, pp. 375–387, Jan. 2018.
- [23] K. Tang, V. Ramanathan, L. Fei-Fei, and D. Koller, "Shifting weights: Adapting object detectors from image to video," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 638–646.
- [24] J. Ma, C. Lu, W. Zhang, and Y. Tang, "Health assessment and fault diagnosis for centrifugal pumps using softmax regression," *J. Vibroeng.*, vol. 16, no. 3, pp. 1464–1474, 2014.
- [25] J. Li, J.-H. Cheng, J.-Y. Shi, and F. Huang, "Brief introduction of back propagation (BP) neural network algorithm and its improvement," in *Advances in Computer Science and Information Engineering*. Berlin, Germany: Springer, 2012, pp. 553–558.
- [26] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [27] J. Lu, D. Meng, T. Mitamura, and A. G. Hauptmann, "Easy samples first: Self-paced reranking for zero-example multimedia search," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 547–556.
- [28] 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.
- [29] 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.
- [30] P. R. Coppin and M. E. Bauer, "Processing of multitemporal Landsat TM imagery to optimize extraction of forest cover change features," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 4, pp. 918–927, Jul. 1994.
- [31] F. Bovolo and L. Bruzzone, "A detail-preserving scale-driven approach to change detection in multitemporal SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 12, pp. 2963–2972, Dec. 2005.
- [32] L. Bruzzone and F. Bovolo, "A novel framework for the design of change-detection systems for very-high-resolution remote sensing images," *Proc. IEEE*, vol. 101, no. 3, pp. 609–630, Mar. 2013.
- [33] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.
- [34] G. H. Rosenfield and K. Fitzpatrick-Lins, "A coefficient of agreement as a measure of thematic classification accuracy," *Photogram. Eng. Remote Sens.*, vol. 52, no. 2, pp. 223–227, 1986.
- [35] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.
- [36] A. Benavoli, G. Corani, and F. Mangili, "Should we really use post-hoc tests based on mean-ranks?" *J. Mach. Learn. Res.*, vol. 17, no. 5, pp. 152–161, 2016.
- [37] Y. Wang, L. Chen, and J.-P. Mei, "Incremental fuzzy clustering with multiple medoids for large data," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 6, pp. 1557–1568, Dec. 2014.
- [38] 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.
- [39] 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.
- [40] J. S. Deng, K. Wang, Y. H. Deng, and G. Qi, "PCA-based land-use change detection and analysis using multitemporal and multisensor satellite data," *Int. J. Remote Sens.*, vol. 29, no. 16, pp. 4823–4838, 2008.
- [41] A. A. Nielsen, "Multi-channel remote sensing data and orthogonal transformations for change detection," in *Machine Vision and Advanced Image Processing in Remote Sensing*. Berlin, Germany: Springer, 2007.
- [42] Q. Xu, Z. Liu, F. Li, H. Ren, and M. Yang, "The regularized iteratively reweighted object-based MAD method for change detection in bi-temporal, multispectral data," *Proc. SPIE*, vol. 10156, Oct. 2016, Art. no. 101560P.
- [43] F. Bovolo, S. Marchesi, and L. Bruzzone, "A framework for automatic and unsupervised detection of multiple changes in multitemporal images," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 6, pp. 2196–2212, May 2012.



Maoguo Gong (M'07–SM'14) received the B.S. degree (Hons.) in electronic engineering and the Ph.D. degree in electronic science and technology from Xidian University, Xi'an, China, in 2003 and 2009, respectively.

Since 2006, he has been a Teacher with Xidian University. In 2008 and 2010, he was promoted as an Associate Professor and as a Full Professor, respectively, both with exception admission. His research interests include computational intelligence with applications to optimization, learning, data mining, and image understanding.

Dr. Gong received the support of Top-Notch Young Professionals for the prestigious National Program from the Central Organization Department of China, the Excellent Young Scientist Foundation Program from the National Natural Science Foundation of China, and the New Century Excellent Talent in University from the Ministry of Education of China. He is an Associate Editor of the *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION* and the *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*.



Yingying Duan received the M.S. degree in circuits and systems from Xidian University, Xi'an, China, in 2019.

Her research interests include remote sensing image analysis and machine learning.



Hao Li received the B.S. degree in electronic engineering and the Ph.D. degree in pattern recognition and intelligent systems from Xidian University, Xi'an, China, in 2013 and 2018, respectively.

He is currently an Associate Professor with the School of Electronic Engineering, Xidian University. His research interests include computational intelligence and machine learning.