*remote sensing*

MDPI

*Article*

# A Mutual Teaching Framework with Momentum Correction for Unsupervised Hyperspectral Image Change Detection

**Jia Sun, Jia Liu, Ling Hu, Zhihui Wei and Liang Xiao ***

School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China; sun_jia@njust.edu.cn (J.S.); omegaliuj@njust.edu.cn (J.L.); njusthuling@njust.edu.cn (L.H.); gswei@njust.edu.cn (Z.W.)

* Correspondence: xiaoliang@mail.njust.edu.cn

**Abstract:** Deep-learning methods rely on massive labeled data, which has become one of the main impediments in hyperspectral image change detection (HSI-CD). To resolve this problem, pseudo-labels generated by traditional methods are widely used to drive model learning. In this paper, we propose a mutual teaching approach with momentum correction for unsupervised HSI-CD to cope with noise in pseudo-labels, which is harmful for model training. First, we adopt two structurally identical models simultaneously, allowing them to select high-confidence samples for each other to suppress self-confidence bias, and continuously update pseudo-labels during iterations to fine-tune the models. Furthermore, a new group confidence-based sample filtering method is designed to obtain reliable training samples for HSI. This method considers both the quality and diversity of the selected samples by determining the confidence of each group instead of single instances. Finally, to better extract the spatial–temporal spectral features of bitemporal HSIs, a 3D convolutional neural network (3DCNN) is designed as an HSI-CD classifier and the basic network of our framework. Due to mutual teaching and dynamic label learning, pseudo-labels can be continuously updated and refined in iterations, and thus, the proposed method can achieve a better performance compared with those with fixed pseudo-labels. Experimental results on several HSI datasets demonstrate the effectiveness of our method.

**Keywords:** change detection; bitemporal hyperspectral image; pseudo-label; mutual teaching



**Citation:** Sun, J.; Liu, J.; Hu, L.; Wei, Z.; Xiao, L. A Mutual Teaching Framework with Momentum Correction for Unsupervised Hyperspectral Image Change Detection. *Remote Sens.* **2022**, *14*, 1000. https://doi.org/10.3390/rs14041000

Academic Editors: Yue Wu, Kai Qin, Maoguo Gong and Qiguang Miao

Received: 7 January 2022
Accepted: 15 February 2022
Published: 18 February 2022



## 1. Introduction

Hyperspectral imaging techniques can obtain continuous spectral information over a wide range of spectral wavelengths. The ability to display the subtle spectral variations of different ground objects has played an important role in many land-cover monitoring applications, such as mineral exploration [1,2], land-use monitoring [3,4], and military defense [5]. Change detection (CD) is the process of identifying differences in the state of an object or phenomenon by observing it at different times [6], which has been an indispensable application in the remote sensing field for a long time. Because of the rapid increase in spectral information, hyperspectral images (HSIs) are able to help detect finer changes than other remote sensing images and observe more change details. However, due to the spectral variability and redundant information, it is still a substantial challenge to effectively mine the spectral–spatial information to complete the HSI-CD task.

For decades, a variety of unsupervised methods have been applied to HSI-CD. Change detection aims to generate an accurate binary change map. In traditional methods, the change map can be obtained by analyzing the difference image (DI), which is usually based on differencing or log-rationing function. The most typical method is the change vector analysis (CVA) method [7], which identifies the changed pixels and the type of change according to the magnitude and direction of the spectral change vector. Some techniques utilize image transformation to extract new features for better performance. Principal component analysis (PCA) [8] retains the main information of original images according

to the statistical characteristics, which reduces data redundancy enormously. Both multivariate alteration detection (MAD) [9] and the improved iteratively reweighted MAD (IRMAD) [10] methods calculate the degree of change by canonical correlation analysis. These measures all assume that the characteristics of unchanged pixels are uniform, but in reality, due to atmospheric conditions, illumination, etc., completely identical features rarely exist [11]. To suppress the difference in unchanged pixels, slow feature analysis (SFA) extracts the most temporally invariant component and then converts images into a new feature space [12]. Furthermore, some methods [13,14] directly determine the changing type of pixels using the postclassification comparison (PCC), but their performances fully depend on the accuracy of the classifier.

Recently, deep-learning (DL) methods have been favored by many researchers because of their strong nonlinear representation ability. Change detection needs to process bitemporal images simultaneously, as feature fusion must be carried out to form a single feature vector, which is usually a similarity measure between those two features [15]. Conventional methods inevitably lose partial information via difference or other processing, while deep-learning methods can avoid this problem. Mou et al. [16] proposed an end-to-end network. A convolutional neural network (CNN) extracts spectral–spatial features and a recurrent neural network (RNN) analyzes the temporal dependence between images. Considering the mixed pixels in HSIs, some methods [17] utilize subpixel-level information obtained by unmixing to improve detection accuracy. Chen et al. [18] proved that the 3D convolution kernel combined with regularization can effectively extract the spectral–spatial features of HSIs for classification tasks. Based on this, a 3D convolutional neural network (3DCNN) for hyperspectral image change detection is designed as the basic model of the proposed framework.

However, the great success of existing deep learning methods in many tasks mainly benefits from a large amount of labeled data. Pixel-wise labels for bitemporal HSIs need to be annotated by experts, which is time-consuming and expensive. Thus, it is difficult to obtain in large quantities. To solve the problem, existing unsupervised HSI-CD methods usually use pseudo-labels generated by traditional algorithms [17,19–21]. One of the main challenges is that the training process of neural networks is susceptible to noise in pseudo-labels. It is difficult to deal with the high-dimensionality of hyperspectral data for traditional CD methods. Additionally, affected by atmospheric conditions, illumination, and topography changes, the spectral variability of ground objects further increases the difficulty of change detection. Due to the limitations of these traditional methods, there certainly exist some discrepancies between the pseudo-labels and the true labels. Zhang et al. [22] proved that advanced neural networks can easily fit training sets with arbitrary labels. Once the network fits inaccurate labels, it will seriously affect the classification results. Wang et al. [17] utilized subpixel information to enhance robustness of the model. Du et al. [19] designed a deep slow feature analysis (DSFA) algorithm based on SFA theory and deep network to extract invariant components. These methods largely ignore handling the noisy labels. Li et al. [20] added a noisy model with zero-mean Gaussian distribution to their loss function, yet the experimental effect was general. The authors in [21] adopted two unsupervised algorithms to jointly generate credible labels. However, the same problem still exists, where it is impossible to filter all noisy labels by only one-time sample selection.

To address the noisy labels, we propose dynamically correcting pseudo-labels instead of safely relying on labels. The momentum correction approach is based on mutual teaching, where two learning models are mutually updated to jointly learn. Dynamic learning approaches by sample selection are popular in robust learning from noisy labels [23–27]. Yao et al. [24] adjusted the number of training samples in each iteration according to the learning curve. Self-paced learning (SPL) [25,26], which reduces the confidence threshold as the number of iterations increases, automatically selects more complex samples. However, the self-training of networks is prone to self-confidence bias and cannot be corrected when errors accumulate. Co-teaching [27] trains two classifiers simultaneously and enables them to select small loss samples for each other in every mini-batch, effectively suppressing the

phenomenon of overfitting. It is generally assumed that small loss samples are more likely to be correctly labeled. Nevertheless, simply using loss to select training data is not suitable for bitemporal HSIs with complex variations. The selected samples are easily concentrated in the two categories of maximum and minimum changes, which is not conducive to the generalization of models. Therefore, we divide all samples into multiple groups according to the similarity of the difference vector in advance and randomly select from the high-confidence groups to ensure that multiclass samples can be selected. In addition, although our approach uses incompletely correct labels during initialization, utilizing the newly derived more reliable results to update the pseudo-labels can further boost the classifier performance [28,29]. The main contributions are summarized as follows:

(1) We introduce to a novel mutual teaching framework with momentum correction for resisting noisy labels generated by traditional methods in unsupervised HSI-CD. Due to mutual teaching and dynamic label learning, pseudo-labels can be continuously updated and refined in iterations, and thus the proposed method can achieve superior results.
(2) A group confidence-based sample selection approach is proposed to avoid selecting the two most extreme types of samples, and it is used alternately with another selection mechanism in iteration to ensure that complex samples can participate in training.
(3) An end-to-end 3DCNN is designed as a classifier for HSI-CD and the basic model of the proposed framework. Experiments on four datasets demonstrate that our framework can effectively improve model performance.

## 2. Related Work

### *2.1. Unsupervised Deep Methods for Change Detection*

Remote sensing image annotation is more difficult than that of natural images, especially for pixel-level change detection. Therefore, unsupervised methods without manual labeling steps have more advantages. Currently, unsupervised deep-learning methods can be divided into two categories. As shown in Figure 1a, the network is treated as a feature extractor to transform original images into a new feature space, and the model parameters are optimized based on the analysis of current output features in each iteration. For example, Liu et al. [30] proposed a symmetric convolutional coupling network (SCCN), which was initialized by a denoising autoencoder, and then minimized the feature difference between those unchanged pixels. Zhang et al. [31] adopted clustering analysis and detected multiple types of changes. Liu et al. [32] established an energy function driven network according to the feature difference. The advantage of these methods is that the newly derived features are used in each iteration to progressively improve the accuracy of the results. However, due to the limitation of optimization, it is difficult to use more complex models without any labels and the high dimension of HSIs is not conducive to model convergence. The other is shown in Figure 1b. The results obtained by the traditional algorithms are assigned to all samples as pseudo-labels to train neural networks, which is more commonly used [17,19–21,25,26,33,34]. These methods are easy to implement and closer to end-to-end patterns, avoiding the intermediate steps of difference image analysis. The only problem is that the pseudo-labels are not completely correct, which may mislead the network training. Inspired by the first category of methods, we utilize new predictive values to update pseudo-labels in multiple iterations to gradually reduce noise labels.

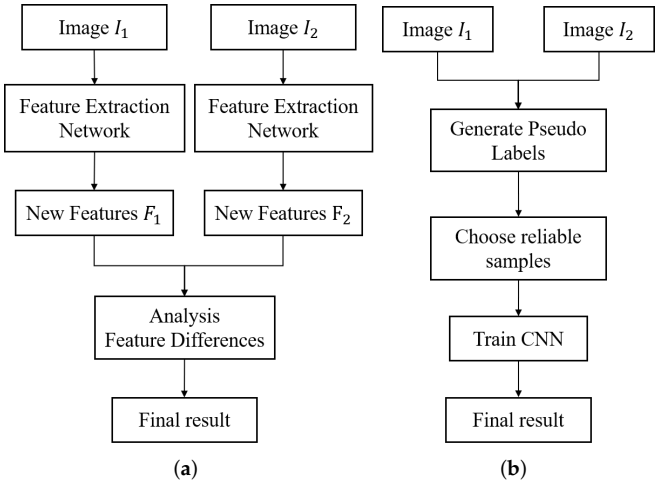



**Figure 1.** Architectures of two unsupervised deep methods for change detection. (**a**) Feature analysis-driven model training. (**b**) Pseudo-labels-driven model training.

### *2.2. Deep Learning with Noisy Labels*

Noisy labels are ubiquitous in deep-learning applications, such as in large-scale low-quality datasets collected from the internet or crowdsourcing platforms in supervised learning, predictive pseudo-labels in semi-supervised learning and in domain adaptation learning. The overfitting of noisy labels will directly weaken the generalization of the models. Thus, learning with noisy labels still attracts researchers' attention. The noise transition matrix and robust loss function are commonly used for antinoise training. Goldberger et al. [35] added another softmax layer to capture the transitional relationship between the noisy and true labels. Ghosh et al. [36] confirmed that the loss function based on the mean-absolute error is inherently robust to noise. CleanNet [37] determined whether the sample label is correct by comparing it with a representative "class prototype". However, these methods generally require prior knowledge or rely on certain constraints. To avoid consuming additional resources or more complex networks, it is a good way to select clean parts from noisy instances to update models. The memorization effects of deep neural networks show that the samples with smaller values collected from the loss function are more likely to be correctly annotated. Therefore, some studies [24–27] allow the model to select reliable samples for itself in each iteration to improve classification accuracy, which is similar to active learning and reinforcement learning.

### *2.3. Mutual Teaching Paradigm*

Although sample selection can effectively prevent noisy labels from participating in training, it is difficult to ensure that the selected labels are absolutely clean. The self-training process is sensitive to noise and outliers, and multiple iterations will accumulate the model bias caused by a few wrongly selected instances or unbalanced samples. For this purpose, MentorNet [38] is learned to compute time-varying weights for each training sample based on a predefined course, which provides meaningful supervision to help StudentNet overcome corrupted labels. However, the problem of error accumulation still exists. Inspired by co-training [39], co-teaching [27] trains two identical deep networks and lets them select small loss samples for each other in every minibatch. The difference between them is that co-training needs to establish two viewpoints to generate reliable pseudo-labels, which are generally used for semisupervised learning. Co-teaching only needs one viewpoint, which utilizes the randomness of the network training process to resist self-confidence bias, similar to finding their potential shortcomings by "peer-review". Likewise, deep mutual learning [40] enables multiple student networks to learn from each other to produce a more robust and generalized network in model distillation. This simple and effective learning paradigm is easily extended to other applications [41–44].

Unfortunately, few studies have focused on pseudo-label noise in HSI-CD. Therefore, we develop a dynamic change detection framework using the novel mutual teaching approach and an improved sample selection method.

## 3. Methodology

In this section, we detail the proposed method from three aspects: the training process of the mutual teaching framework, sample selection and class balancing, and the classifier for HSI-CD.

### *3.1. The Mutual Teaching Framework*

An overview of the proposed mutual teaching framework for bitemporal HSI-CD is shown in Figure 2. First, the original pseudo-labels are obtained from a traditional method, and after screening, they are used to initialize two DL models with the same structure. After each iteration, the two DL models update pseudo-labels for each other with new predictions and alternately use two different sample selection procedures to ensure the accuracy and diversity of training instances. In the end, the final result is generated by combining the predictions of the two models.

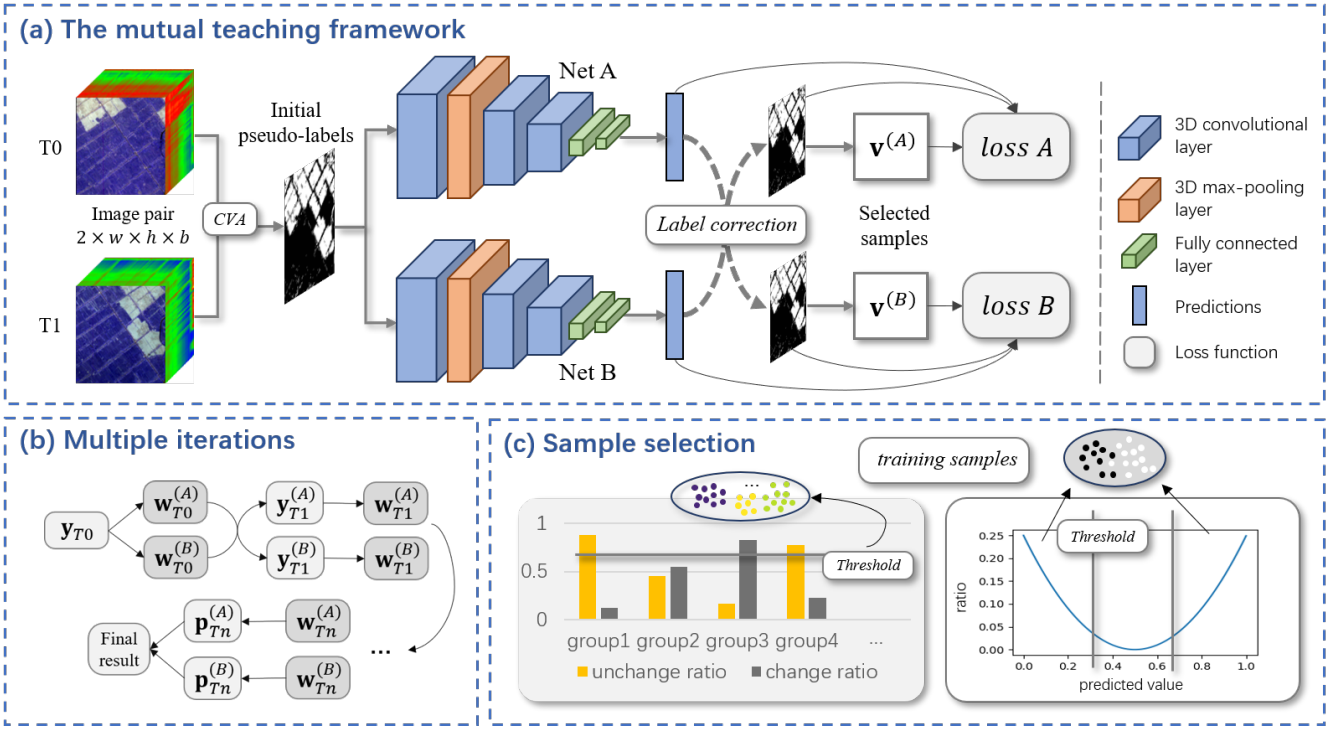



**Figure 2.** Graphical illustration of the proposed method. (**a**) Overall framework of the proposed mutual teaching based on collaborative training and label correction. Both models update their pseudo-labels with each other's predictions and select clean samples to optimize parameters. (**b**) The model parameters and pseudo-labels are alternately updated and the final result is generated by the predictions of the two models. (**c**) Two sample selection methods jointly ensure the accuracy and diversity of training samples.

In this work, the HSI-CD task is regarded as a classification problem, that is, to determine whether the sample corresponding to each pixel belongs to the changed or unchanged class. Taking a pair of pixels in the same position of bitemporal images as a training instance. With a total of $m$ samples, $x_i$ is the $i$th sample, and $m$ equals $w \times h$, where $w$ and $h$ are the width and height of the original image, respectively. We adopt the CVA algorithm to obtain initial pseudo-labels $\mathbf{y} \in \mathbb{R}^m$, $y_i$ equals 1 to represent the change sample, and 0 is unchanged. In contrast to the existing deep-learning-based change detection methods, the pseudo-labels generated by CVA are only used to initialize the

parameters of the two networks $\mathbf{w}^{(A)}$ and $\mathbf{w}^{(B)}$ and serve as initial values of $\mathbf{y}^{(A)}$ and $\mathbf{y}^{(B)}$. They will be updated dynamically by mutually training the two networks.

To further improve the accuracy of classifiers, it is necessary to select samples with the label as correct as possible. After feeding all pseudo-labels into the sample selection program, two sets of training data $\mathbf{v}^{(A)}$ and $\mathbf{v}^{(B)}$ can be obtained. Here $\mathbf{v} \in \mathbb{R}^m$ indicates whether the sample $x_i$ is selected, where $v_i$ equals 1 indicating selected and 0 unselected. The parameter updating procedures of both models are described below:

$$\begin{aligned} \widehat{\mathbf{w}}^{(A)} &= \underset{\mathbf{w}^{(A)}}{\operatorname{argmin}} \sum_{i=0}^{m-1} v_i^{(A)} L\left(y_i^{(A)}, f\left(x_i, \mathbf{w}^{(A)}\right)\right) \\ \widehat{\mathbf{w}}^{(B)} &= \underset{\mathbf{w}^{(B)}}{\operatorname{argmin}} \sum_{i=0}^{m-1} v_i^{(B)} L\left(y_i^{(B)}, f\left(x_i, \mathbf{w}^{(B)}\right)\right) \end{aligned} \tag{1}$$

where $L(y_i, f(x_i, \mathbf{w}))$ is the loss between the classifier's predicted value $f(x_i, \mathbf{w})$ and the pseudo-label $y_i$. Then, we can update the pseudo-labels with a new prediction:

$$\begin{aligned} \hat{y}_i^{(A)} &= \alpha y_i^{(A)} + (1-\alpha) f\left(x_i, \mathbf{w}^{(B)}\right) \\ \hat{y}_i^{(B)} &= \alpha y_i^{(B)} + (1-\alpha) f\left(x_i, \mathbf{w}^{(A)}\right) \end{aligned} \tag{2}$$

where $\alpha$ is the momentum parameter. Note that both models use each other's predicted values to update their own pseudo-labels for mutual teaching purposes.

Sample selection can effectively reduce noisy labels, but it is impossible to completely screen them. Due to various factors such as unbalanced samples and noisy labels, it is inevitable for the classifier to generate confidence bias. The error will be transferred back to itself in the next iteration, and it should be increasingly accumulated in the self-training process. Benefiting from the respective training of the two models, they can filter out different types of errors by mutual teaching and effectively reduce the accumulation of these errors. Meanwhile, with the improvement of model prediction accuracy, the influence of noisy labels can also be mitigated by gradually modifying pseudo-labels. After multiple iterations, the final results are derived from the predicted values of two classifiers. When their predictions are different, we choose one with less loss.

### *3.2. Sample Selection*

To reduce the impact of noisy labels, sample selection is utilized in large studies. The most common approach is to judge the credibility according to the sample loss, which can be formulated as:

$$v_i = \begin{cases} 1, & \text{if } |y_i - f(x_i, w)| < \lambda \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

The threshold $\lambda$ is a critical parameter. When $\lambda$ is too large, noisy labels will increase, and when $\lambda$ is too small, the lack of complex samples is not conducive to the generalization of the classifier.

In previous studies [25,26], the above selection method was used, and it is designed for synthetic aperture radar (SAR) image data, which is comparatively simple. However, it performs poorly on HSIs. The selected samples tend to focus on the simplest regions and ignore other types. Thus, we need to design a more appropriate sample selection algorithm for such HSIs. Considering the distribution of data, we use a clustering-based method to select data in blocks. The data in the same cluster have high similarity. When most samples in the cluster have consistent prediction, it is relatively reliable. To select samples of different types evenly and ensure correct labels, a group confidence-based sample selection approach is designed, as shown in Figure 3. First, the PCA is used to reduce the feature dimension of the difference image, and k-means algorithm is applied on the results to obtain the grouping information. Then, we can obtain a grouping label

vector $\mathbf{c} \in \mathbb{R}^m$, which indicates the grouping information of all samples and takes the value in $\{0, 1, \ldots, n-1\}$, while $n$ is the total number of groups. We consider the label with the highest proportion in each group as the group label:

$$g_j = \max_{l \in \{0,1\}} \left\{ \sum_{i=0}^{m-1} (c_i == j) \times (y_i == l) \right\}, \; j = 0, 1, \cdots, n-1 \tag{4}$$

where $y_i$ is the pseudo-label; $\mathbf{g} = [g_0, \ldots, g_{n-1}] \in \mathbb{R}^n$, $g_j$ is the group label. If there are more changed samples than unchanged samples within the group, $g_j$ takes 1; otherwise, it takes 0. The group confidence is determined by the proportion of group labels:

$$r_j = \frac{\sum_{i=0}^{m-1} (c_i == j) \times (y_i == g_j)}{\sum_{i=0}^{m-1} (c_i == j)}, \; j = 0, 1, \cdots, n-1 \tag{5}$$

where $\mathbf{r} \in \mathbb{R}^n$ represents a group confidence vector. When the value exceeds a certain threshold, the sample in the group is considered reliable:

$$v_i = \begin{cases} y_i == g_j, & \text{if } r_j \geq \sigma \\ 0, & \text{otherwise} \end{cases} \quad \text{s.t. } j = c_i \tag{6}$$

where $\sigma$ is the group confidence threshold. Note that we only select samples with the same label as the group. In this way, the selected training dataset contains samples of varying degrees of change and has a low proportion of noisy labels.

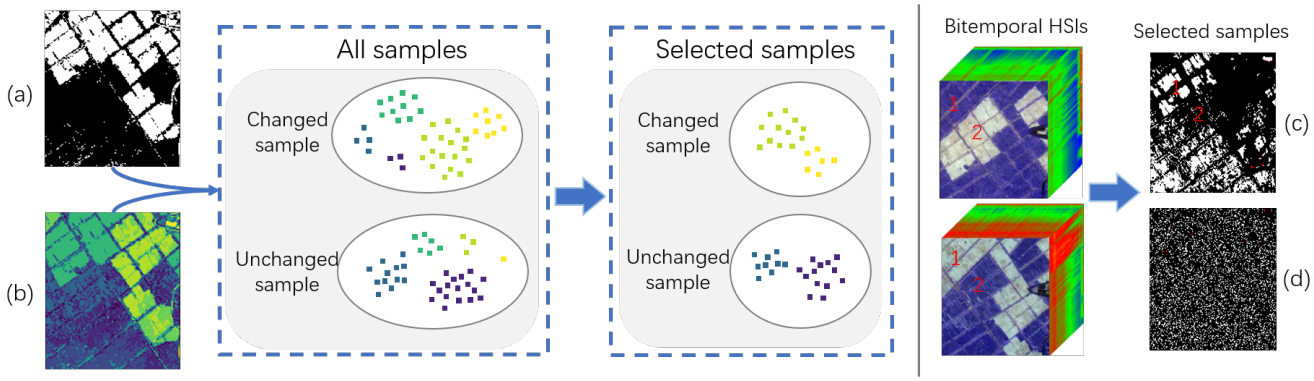



**Figure 3.** Sample selection based on group confidence. (**a**) Pseudo-labels. (**b**) Multiclass map. (**c**) Sample loss-based method. (**d**) Group confidence-based method. In (**c**,**d**), white is the selected samples with correct label, red is the selected noisy samples, and black is the discarded samples.

Figure 3c,d shows the samples selected in two ways respectively. It is obvious that the dataset contains two main changes, and the changed samples selected by a single sample confidence focus on one class while ignoring the other. Moreover, the sampling of our method is more even, and the noisy labels contained in both methods are negligible. Another advantage of our method is that it can be used on pseudo-labels of discrete values, such as 0 or 1. The sample loss-based algorithm can only be used on continuous values, which are usually between 0 and 1.

The group confidence-based sample selection approach can consider screening noisy labels and the diversity of the training data. There are mainly two kinds of samples to be discarded: one with low group confidence and the other has a different label from most of the samples in the group. Thus, some complex samples may never participate in training, and the two models cannot adequately exchange information. Therefore, we alternately use two selection strategies to jointly guarantee the accuracy and the stability of the final results, as shown in Figure 2c. The group confidence-based sample selection method is used to select samples which are as clean possible, to improve the accuracy of models, and

the parameter $\sigma$ is set to 0.8. The loss-based method is used to select as many samples as possible to encourage these easily overlooked complex samples to participate in training, and the parameter $\lambda$ is set to 0.4.

In addition, we apply different weights for sample loss to balance class. General binary classification uses cross-entropy loss, which can be defined as follows:

$$l_{ce}(y_i, p_i) = -y_i \log p_i - (1 - y_i) \log(1 - p_i) \tag{7}$$

where $p_i$ is the predicted value. Then, the final weighted loss function is:

$$L(y_i, f(x_i, \mathbf{w})) = |y_i - f(x_i, \mathbf{w})|^{\gamma} \cdot l_{ce}(y_i, f(x_i, \mathbf{w})) \tag{8}$$

where the first item enhances the weight of large loss samples, and $\gamma$ is set to 2 according to article [45]. The weighted loss can balance the multiclass samples to avoid a large deviation of the model. The entire procedure for the proposed method is summarized in Algorithm 1.

**Algorithm 1:** Procedure of the proposed method.

**Input:** Two images $\mathbf{I}_1$ and $\mathbf{I}_2$; thresholds $\sigma$ and $\lambda$; the number of iterations $n_t$; the momentum parameter $\alpha$.
**Output:** The final result $\mathbf{p}$.

*// Initialization*
Get pseudo-labels $\mathbf{y}$ and multiclass map $\mathbf{c}$; initialize $\mathbf{y}^{(A)}$ and $\mathbf{y}^{(B)}$;
Randomly initialize $\mathbf{w}^{(A)}$ and $\mathbf{w}^{(B)}$;
**for** $i \leftarrow 1$ **to** $n_t$ **do**
  **if** $i$ % 2 == 1 **then**
    Update selected sample $\mathbf{v}^{(A)}$ and $\mathbf{v}^{(B)}$ by (6);
  **else**
    Update selected sample $\mathbf{v}^{(A)}$ and $\mathbf{v}^{(B)}$ by (3);
  **end**
  Update model parameters $\mathbf{w}^{(A)}$ and $\mathbf{w}^{(B)}$ by (1) and (8);
  Update pseudo-labels $\mathbf{y}^{(A)}$ and $\mathbf{y}^{(B)}$ by (2);
**end**
$\mathbf{p} \Leftarrow \{\mathbf{p}^{(A)} = f\left(\mathbf{x}, \mathbf{w}^{(A)}\right), \mathbf{p}^{(B)} = f\left(\mathbf{x}, \mathbf{w}^{(B)}\right)\}$

### *3.3. A 3D Convolutional Neural Network Establishment*

The bitemporal hyperspectral data have four dimensions, two spatial axes, a spectral axis and a temporal axis. To extract features using general 2D convolution kernels, most change detection methods reduce one dimension of data by stacking or with a difference operation. However, direct stacking increases the number of convolution kernel channels and network parameters, especially for HSIs with hundreds of channels, and the difference operation leads to the loss of original information. In HSI classification, the authors in [18] verified that 3D convolution can better extract spectral spatial features of HSI than 2D convolution. In some video processing applications, 3D convolution kernel has been used to extract temporal and spatial features simultaneously. Similar to change detection, these kinds of data have an additional temporal dimension relative to a single image. Therefore, 3D convolution is an appropriate feature extractor without additional operations in HSI-CD.

In convolutional layers, the calculation of new features uses convolution kernels to multiply local domain features of the previous layer, then adds a bias and passes through an activation function. For 2D convolution, the value of the feature map extracted by the $i$th convolution kernel of the $l$th layer at position $(x, y)$ is calculated as:

$$X_{l,i}^{xy} = f\left(\sum_{m} \sum_{p=0}^{P_i - 1} \sum_{q=0}^{Q_i - 1} W_{l,i}^{pq} X_{l-1,m}^{(x+p)(x+q)} + b_{l,i}\right) \tag{9}$$

where $f(\cdot)$ is the activation function, $P_i$ and $Q_i$ are the height and width of the kernel, $W_{l,i}^{pq}$ is the value of the kernel connected to the feature map at position $(p, q)$, $m$ represents the $l-1$th layer feature map connected to the current feature, and $b_{l,i}$ is the bias. For 3D convolution, the value of the feature map extracted by the $i$th convolution kernel of the $l$th layer at position $(x, y, z)$ is calculated as:

$$X_{l,i}^{xyz} = f\left(\sum_{m}\sum_{p=0}^{P_i-1}\sum_{q=0}^{Q_i-1}\sum_{r=0}^{R_i-1} W_{l,i}^{pqr} X_{l-1,m}^{(x+p)(y+q)(z+r)} + b_{l,i}\right) \tag{10}$$

where $R_i$ is the size of the 3D kernal along the spectral dimension. Because the adjacent spectral channels of HSIs have a strong correlation, it is reasonable to extract the spatial and spectral neighborhood information of two images simultaneously with 3D convolution.

For HSI-CD tasks, we design a 3D convolutional neural network as the basis classifier, as shown in Figure 4. A sample consists of two data blocks of a neighborhood size of 3 extracted from bitemporal images at the same location, and is filled to size 5 with 0 when input into the network. After three 3D convolution layers and a pooling layer, the fused feature vector is extracted, and finally, the change information is output through two fully connected layers. The last layer is activated by the softmax function, and the other uses the ReLU function. After all of the samples are fed into the network, a result map with the same size as the original image representing the degree of change can be obtained. To verify the effectiveness of 3D convolution, we design a similar 2D convolutional neural network for comparison in subsequent experiments.

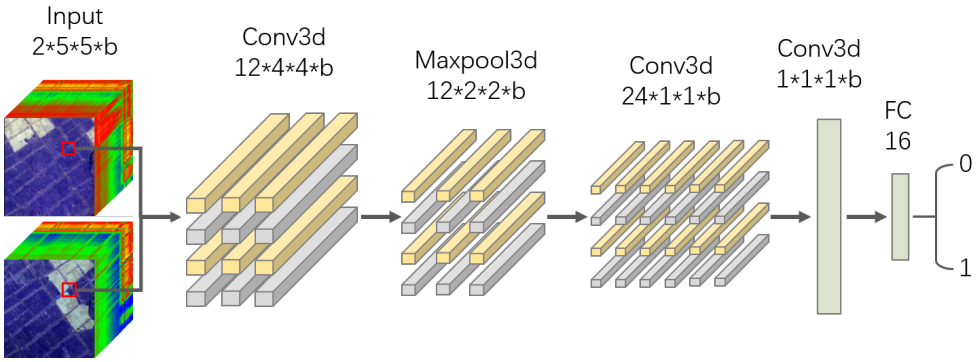



**Figure 4.** Architecture of the proposed 3DCNN.

## 4. Result

To verity the effectiveness of the proposed method, several experiments were conducted using a multispectral dataset and three popular hyperspectral datasets. This section first introduces the datasets used in the experiment. Then the evaluation measures of change detection and experimental setup are described. Finally, comparative experiments with other methods are analyzed in detail.

### *4.1. Introduction to Datasets*

The first dataset "Bastrop" is shown in Figure 5a,b, which consists of two multispectral images (MSI) taken before and after a forest fire in Bastrop County, Texas, in September 2011 and October 2011 [46]. This multispectral dataset was selected from Landsat 5 Thematic Mapper (TM) multispectral images consisting of six spectral bands with a spatial resolution of 30 m for bands 1–5 and 7 and one thermal band (band 6). Their spatial size is 1534 × 808 pixels with 7 bands.

The other three HSI datasets were collected from Earth Observing-1 (EO-1) Hyperion data. EO-1 has a spectral resolution of 10 nm and a spatial resolution of approximately 30 m, with a total of 242 different bands. The second dataset "Umatilla" is irrigated farmland in Umatilla County, OR, USA, as shown in Figure 5d,e. The images contain

390 × 200 pixels and 242 bands. The third dataset, "Yancheng", was acquired on 3 May 2006, and 23 April 2007, in Yancheng, Jiangsu Province, China, as shown in Figure 5g,h. The two images both consist of 450 × 140 pixels with 155 bands after eliminating the noise. The fourth dataset, "river", was obtained on 3 May 2013, and 31 December 2013, in Jiangsu Province, China, as shown in Figure 5j,k. This dataset contains two HSIs with 463 × 241 pixels and 198 channels [17].

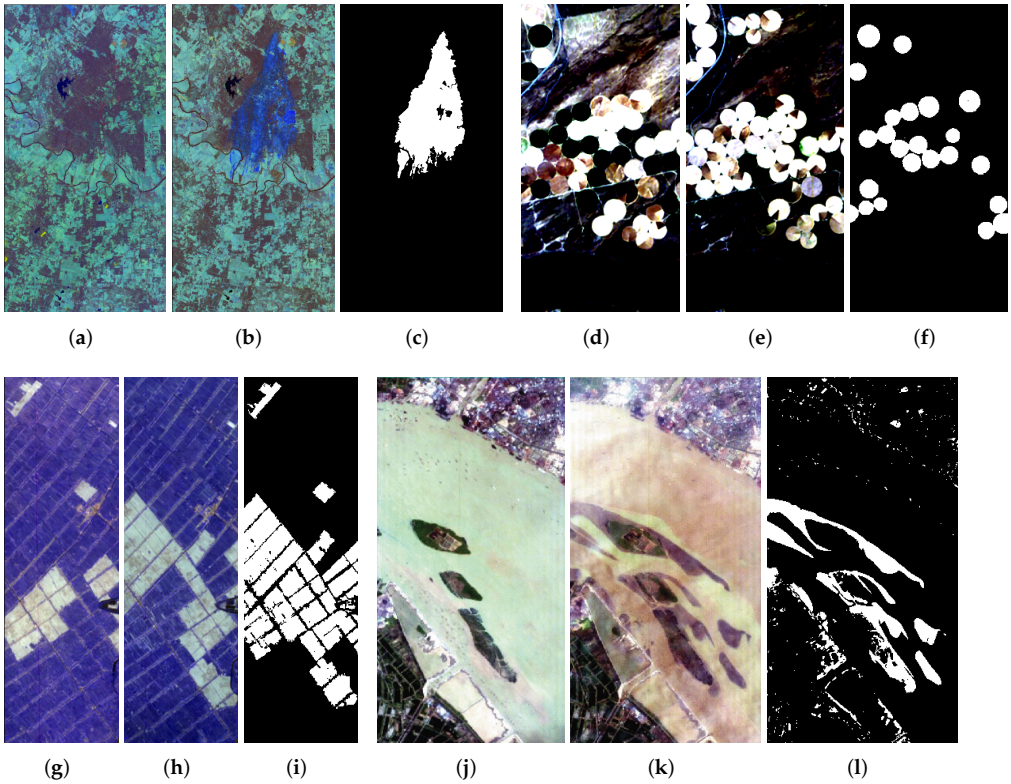



**Figure 5.** Experimental datasets. (**a**) Bastrop dataset in September 2011. (**b**) Bastrop dataset in October 2011. (**d**) Umatilla dataset on 1 May 2004. (**e**) Umatilla dataset on 8 May 2007. (**g**) Yancheng dataset on 3 May 2006. (**h**) Yancheng dataset on 23 April 2007. (**j**) River dataset on 3 May 2013. (**k**) River dataset on 31 December 2013. (**c**,**f**,**i**,**l**) groundtruth change map for Bastrop, Umatilla, Yancheng and River dataset, respectively.

### *4.2. Evaluation Measures and Experimental Configurations*

In this paper, specific evaluation metrics are used to evaluate the change detection results of all methods on the datasets. Generally, the results of change detection use pixel-level indicators, which mainly include the following four metrics: true positives (*TP*), true negatives (*TN*), false positives (*FP*), and false negatives (*FN*). The positive sample refers to the changed samples, displayed in white in the result image, and the negative sample refers to unchanged samples, displayed in black. The correct rate of classification is represented by the overall accuracy (*OA*), and the formula is

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

Compared with *OA*, the kappa coefficient and *F*1 score can better reflect the consistency between the predicted results and the actual results. It is calculated as

$$PRE = \frac{(TP + FP) \times (TP + FN) + (FN + TN) \times (FP + TN)}{(TP + TN + FP + FN)^2} \tag{12}$$

$$Kappa = \frac{OA - PRE}{1 - PRE} \tag{13}$$

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{16}$$

For the experimental setting, the proposed 3DCNN network structure is shown in Figure 4. For hyperspectral images, the size of the first two 3D convolution kernels is $2 \times 2 \times 5$, while on the Bastrop dataset, the size of the 3D convolution kernels is $2 \times 2 \times 1$ because it has only seven bands. The parameter $n$ is the total number of groups. For the Bastrop datasets, $n$ is set to 10, and for the other three datasets, $n$ is set to 20. The momentum parameter $\alpha$ is set to 0.4.

### *4.3. Comparison with Other Methods*

To verity the effectiveness of the proposed method, we tested our method on three hyperspectral datasets and a multispectral dataset, then compared it with other classical methods, including the change vector analysis (CVA) [7], iteratively reweighted multivariate alteration detection (IRMAD) [10], iterative slow feature analysis (ISFA) [12], support vector machines (SVM), GETNET [17], 2DCNN, and 3DCNN. The Otsu threshold algorithm is used in CVA to generate the final change detection result, which is used as pseudo-labels for other methods that require labeled data. Among the above methods, only CVA, IRMAD and ISFA do not require labeled samples. Other classification-based methods use the same pseudo-labels for supervised training and select training samples through our proposed sample selection method.

#### 4.3.1. Experiments on the Bastrop Dataset

Figure 6 shows the final binary result images of the eight methods, and Table 1 lists the results of the numerical evaluation. It can be clearly seen from the figure that there is a large amount of misclassification noise in CVA, mainly false negative samples, and the kappa coefficient is only 0.7241. IRMAD is the worst and ISFA is relatively better among the three unsupervised traditional algorithms, but they have the same problems. Other methods use the results of CVA as pseudo-labels to train their models. Although SVM significantly reduces FP values, it also leads to a huge deviation that causes the changed samples to be mistaken for unchanged. Then, the kappa coefficient was reduced by 22%. Deep-learning methods have wonderful advantages. They all increase the OA and the kappa coefficients and outperform traditional algorithms visually. Remarkably, the performance of our method is considerably better than other methods on this dataset, with kappa rising to 0.9406, which is 9% higher than the second-highest value. In particular, a large number of false negative samples have been well-corrected.

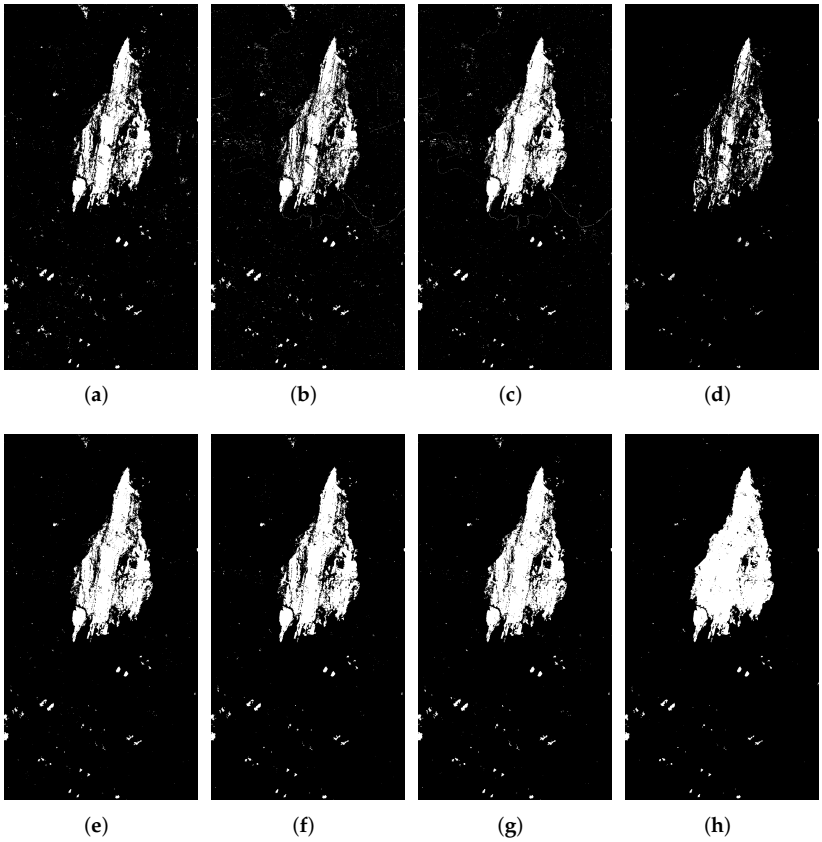


**Figure 6.** The change map on Bastrop dataset. (**a**) CVA. (**b**) IRMAD. (**c**) ISFA. (**d**) SVM. (**e**) GETNET. (**f**) 2DCNN. (**g**) 3DCNN. (**h**) ours.

**Table 1.** Quantitative evaluation of CD results by different methods for Bastrop dataset.

| Methods | FP | FN | OA | Kappa | F1 |
|---|---|---|---|---|---|
| CVA | 10,272 | 46,813 | 0.9539 | 0.7241 | 0.7487 |
| IRMAD | 13,490 | 54,000 | 0.9455 | 0.6688 | 0.6977 |
| ISFA | 10,082 | 32,010 | 0.9660 | 0.8073 | 0.8259 |
| SVM | **2799** | 83,435 | 0.9304 | 0.4992 | 0.5290 |
| GETNET | 6744 | 31,319 | 0.9693 | 0.8241 | 0.8408 |
| 2DCNN | 6923 | 34,276 | 0.9668 | 0.8076 | 0.8257 |
| 3DCNN | 7420 | 26,299 | 0.9728 | 0.8473 | 0.8623 |
| ours | 7212 | **6811** | **0.9887** | **0.9406** | **0.9469** |

#### 4.3.2. Experiments on the Umatilla Dataset

These dataset results are shown in Figure 7 and listed in Table 2. Among the three unsupervised traditional algorithms, CVA has the most serious noise and the lowest accuracy. From the visual effect, the results of IRMAD are closest to the real labels, but there is no substantial advantage compared with ISFA in quantitative analysis. Deep-learning methods can basically filter out the background noise, which also confirms the effectiveness of our sample selection method. Although the gap is small, our method has achieved the best performance in quantitative analysis.

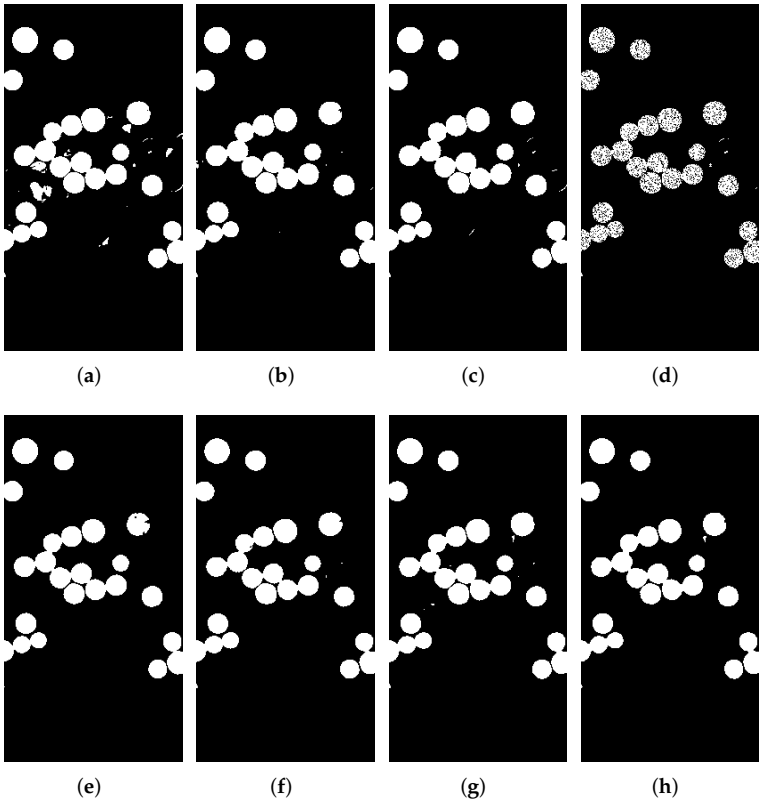


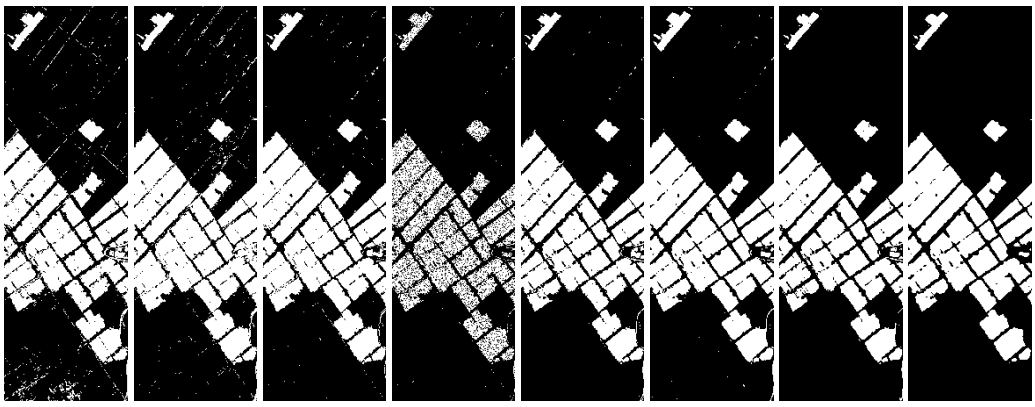

**Figure 7.** The change map on Umatilla dataset. (**a**) CVA. (**b**) IRMAD. (**c**) ISFA. (**d**) SVM. (**e**) GETNET. (**f**) 2DCNN. (**g**) 3DCNN. (**h**) ours.

**Table 2.** Quantitative evaluation of CD results by different methods for Umatilla dataset.

| Methods | FP | FN | OA | Kappa | F1 |
|---|---|---|---|---|---|
| CVA | 1092 | 198 | 0.9835 | 0.9258 | 0.9352 |
| IRMAD | 452 | 246 | 0.9911 | 0.9586 | 0.9637 |
| ISFA | 506 | **191** | 0.9911 | 0.9588 | 0.9639 |
| SVM | 256 | 2125 | 0.9695 | 0.8442 | 0.8612 |
| GETNET | 216 | 337 | 0.9929 | 0.9667 | 0.9707 |
| 2DCNN | 210 | 445 | 0.9916 | 0.9604 | 0.9651 |
| 3DCNN | 277 | 291 | 0.9927 | 0.9660 | 0.9701 |
| ours | **151** | 309 | **0.9941** | **0.9723** | **0.9756** |

#### 4.3.3. Experiments on the Yancheng Dataset

The changes in this dataset are mainly related to farmland. The results are shown in Figure 8 and listed in Table 3. The Yancheng dataset is a relatively simple, traditional method that can also achieve a good performance, especially the performance of ISFA and deep-learning methods that are very similar. Additionally, their OAs are all over 97%. The performance of SVM is the worst and there is too much noise in the changed area. In addition, these four deep-learning methods have all performed very well, but GETNET and 2DCNN still have obvious noise in the unchanged regions, and 3DCNN performs poorly in the changed regions. Only our method eliminates the background noise and also ensures the accuracy of the changed region with multiple iterations.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**) (**g**) (**h**)

**Figure 8.** The change map on Yancheng dataset. (**a**) CVA. (**b**) IRMAD. (**c**) ISFA. (**d**) SVM. (**e**) GETNET. (**f**) 2DCNN. (**g**) 3DCNN. (**h**) ours.

**Table 3.** Quantitative evaluation of CD results by different methods for Yancheng dataset.

| Methods | FP | FN | OA | Kappa | F1 |
|---|---|---|---|---|---|
| CVA | 1833 | 1158 | 0.9525 | 0.8860 | 0.9197 |
| IRMAD | 2268 | 356 | 0.9583 | 0.9019 | 0.9318 |
| ISFA | 1303 | **296** | 0.9746 | 0.9394 | 0.9574 |
| SVM | **512** | 4619 | 0.9186 | 0.7882 | 0.8419 |
| GETNET | 810 | 792 | 0.9746 | 0.9383 | 0.9562 |
| 2DCNN | 1162 | 611 | 0.9719 | 0.9323 | 0.9522 |
| 3DCNN | 554 | 1059 | 0.9744 | 0.9373 | 0.9553 |
| ours | 548 | 817 | **0.9783** | **0.9472** | **0.9624** |

#### 4.3.4. Experiments on the River Dataset

The River dataset is more complex than the other datasets and contains a variety of changes, mainly the disappearance of substances in rivers. Figure 9 shows the maps obtained by eight methods and the quantitative comparison is shown in Table 4. It is obvious from the numerical indicators that the results of CVA are extremely unbalanced, and the number of false-positive samples is approximately 6 times that of the false-negative samples. In addition, ISFA, which performs relatively well in the other datasets, has the worst accuracy here. There is no significant difference among the results of the three networks. The OA can grow to more than 95%, which once again proves that the deep neural network has a strong learning ability and that sample selection can effectively suppress noisy labels. Through multiple iterations and sample selection, the proposed method eliminates the huge deviation of the initial pseudo-labels and obtains the best performance.

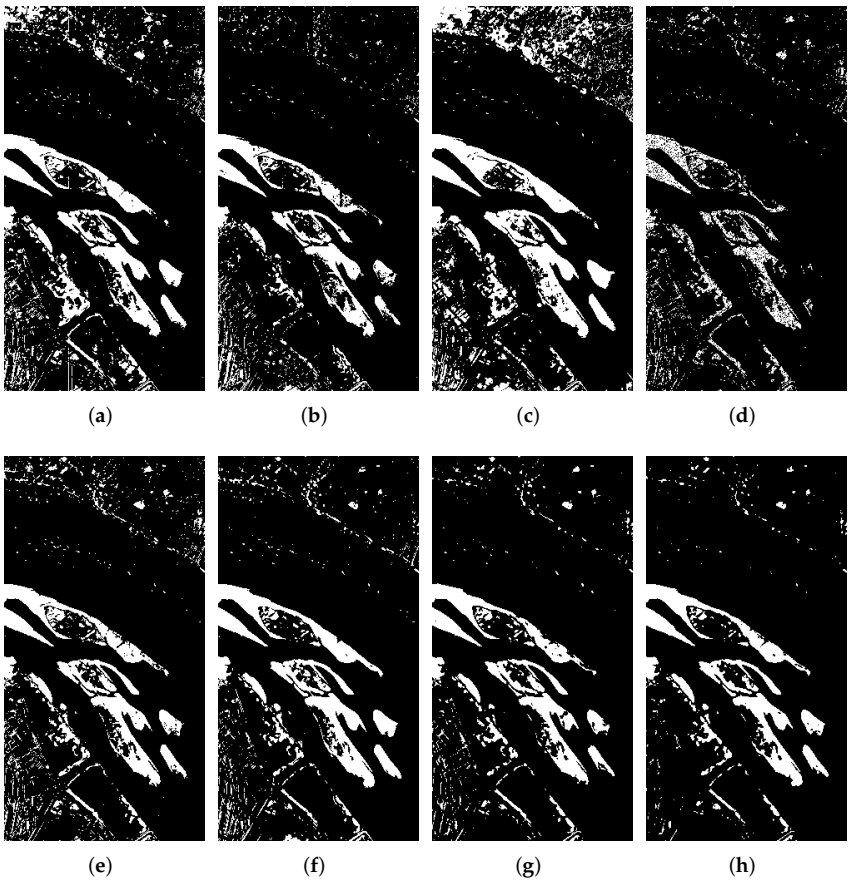


**Figure 9.** The change map on River dataset. (**a**) CVA. (**b**) IRMAD. (**c**) ISFA. (**d**) SVM. (**e**) GETNET. (**f**) 2DCNN. (**g**) 3DCNN. (**h**) ours.

**Table 4.** Quantitative evaluation of CD results by different methods for River dataset.

| Methods | FP | FN | OA | Kappa | F1 |
|---|---|---|---|---|---|
| CVA | 6196 | **1123** | 0.9344 | 0.7103 | 0.7467 |
| IRMAD | 3343 | 3089 | 0.9424 | 0.7005 | 0.7328 |
| ISFA | 10,244 | 1355 | 0.8961 | 0.5897 | 0.6453 |
| SVM | 2595 | 6007 | 0.9229 | 0.5373 | 0.5784 |
| GETNET | 4185 | 1369 | 0.9502 | 0.7636 | 0.7915 |
| 2DCNN | 3618 | 1127 | 0.9575 | 0.7958 | 0.8196 |
| 3DCNN | 2447 | 2215 | 0.9582 | 0.7827 | 0.8061 |
| ours | **1595** | 1809 | **0.9695** | **0.8387** | **0.8558** |

## 5. Discussion

### *5.1. Ablation Study*

To argue the effectiveness of the mutual teaching paradigm, on the above four datasets we make the two networks perform mutual teaching and separate training under the same conditions. Figure 10 shows the OA of the results in 10 consecutive iterations. Classifiers A and B refer to each other's predicted values, while A′ and B′ only use their own results. The two sets of experiments have the same initialization. In the mutual teaching framework, high-precision classifiers are often dragged down by low-precision classifiers, undergoing raising and lowering changes. However, overall, the performances of the two models basically show an upward trend. Although this process has some fluctuations, it does not affect the overall performance. The self-training performance is relatively poor, and the Umatilla and Yancheng datasets are almost not improved. The improvement of the Bastrop

and River datasets is mainly due to sample selection and label correction, but it is also inferior to the mutual teaching models.

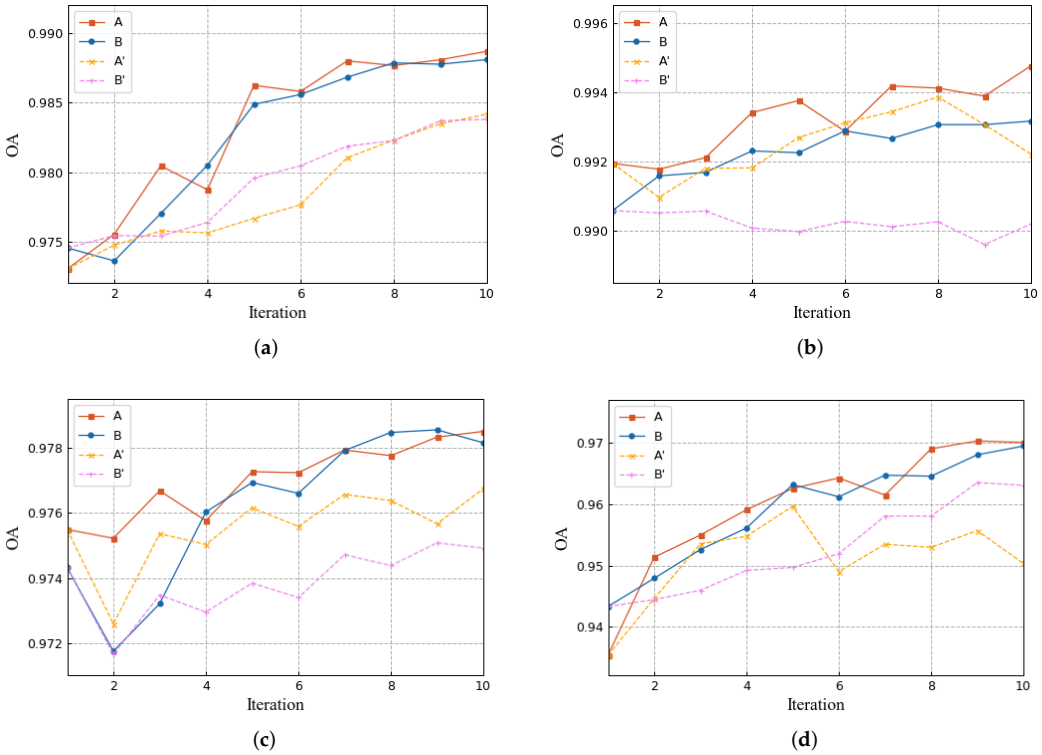



**Figure 10.** The iterative performance of the mutual teaching framework, where A and B use mutual teaching and A′ and B′ are self-training. (**a**) Bastrop dataset. (**b**) Umatilla dataset. (**c**) Yancheng dataset. (**d**) River dataset.

As shown in Figure 11, we compare the alternating training with only one sample selection method. If a dataset itself is relatively simple, the difference between these results is not large. To show the difference in performance, we only use the most complicated River dataset. Figure 11 shows the overall accuracy of the final results and the variance between two models under three settings in each iteration on the River dataset. Although the model accuracy increases faster when only the group confidence-based sample selection method is used, the accuracy no longer increases and remains stable from the sixth iteration. The overall accuracy is further improved by alternating training and significantly exceeds other settings, which proves that the participation of complex samples in training is beneficial to improving the model performance and preserving the details of the change map.

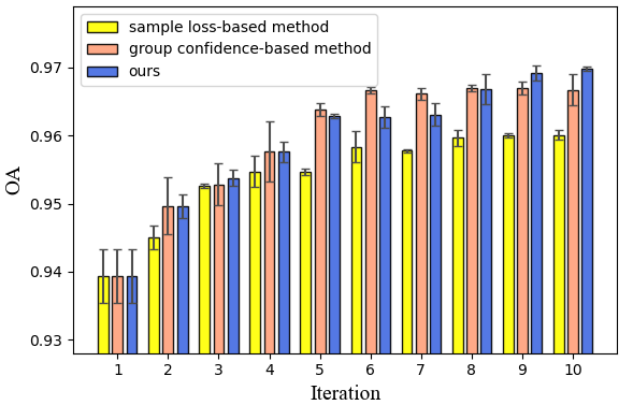



**Figure 11.** The result accuracy and the variance of two models under three settings on River dataset.

### *5.2. Compatibility of the Proposed Framework with Other Models*

Figure 12 displays the change detection results of three networks during initialization and after 10 iterations in our framework. GETNET and 3DCNN have considerably more false-positive samples during initialization, which is mainly misled by the pseudo-labels generated by CVA. However, after multiple corrections, these noises have been improved to a certain extent, especially for the 3DCNN (as shown in the red box). The main error of the 2DCNN result is that some changed regions were not detected, and it had also been recovered after iteration. In other words, both false-positive and false-negative noise labels have the opportunity to be corrected under the proposed framework. The results demonstrate that the mutual teaching framework can also benefit other deep-learning methods based on pseudo-labels.

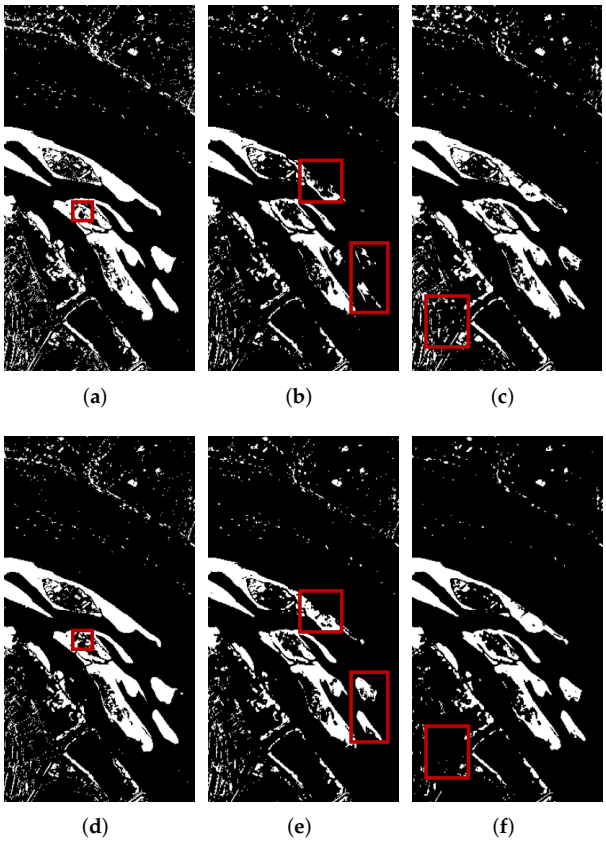
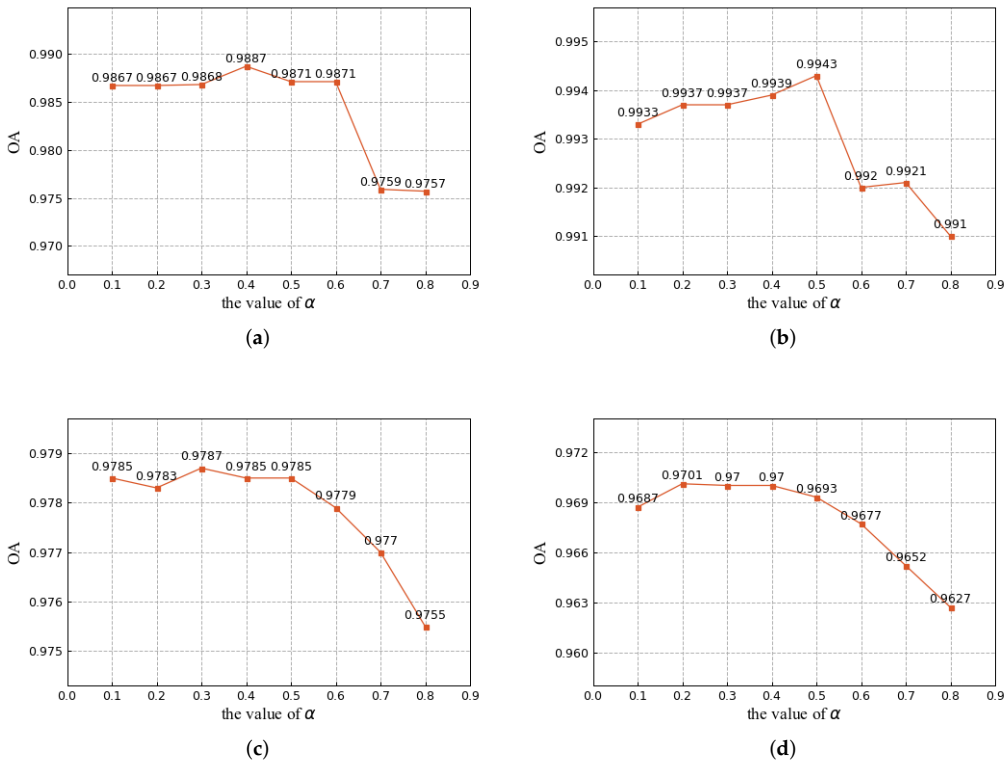



**Figure 12.** Results of different networks in the first and last iterations on River dataset. (**a**) GETNET in the first iteration. (**b**) 2DCNN in the first iteration. (**c**) 3DCNN in the first iteration. (**d**) GETNET in the last iteration. (**e**) 2DCNN in the last iteration. (**f**) 3DCNN in the last iteration.

### *5.3. Hyperparametric Analysis*

#### 5.3.1. Analysis of the Pseudo-Label Update Rate

In the process of pseudo-label correction, the momentum parameter $\alpha$ (in Equation (2)) selection is worth discussing. When $\alpha = 0$, pseudo-labels in each iteration are determined only by new predicted values; when $\alpha = 1$, our method depends entirely on the initial pseudo-label without any updates. We measure the results of different parameters on four datasets, as shown in Figure 13. Experimental results show that the update of pseudo-labels can bring a better performance. If the false labels are not corrected, they will inevitably limit the final result. With the increase in $\alpha$, the overall accuracy shows a downward trend. A value of 0.2∼0.5 is a suitable range for all datasets. Therefore, we choose $\alpha = 0.4$ for our experiments.

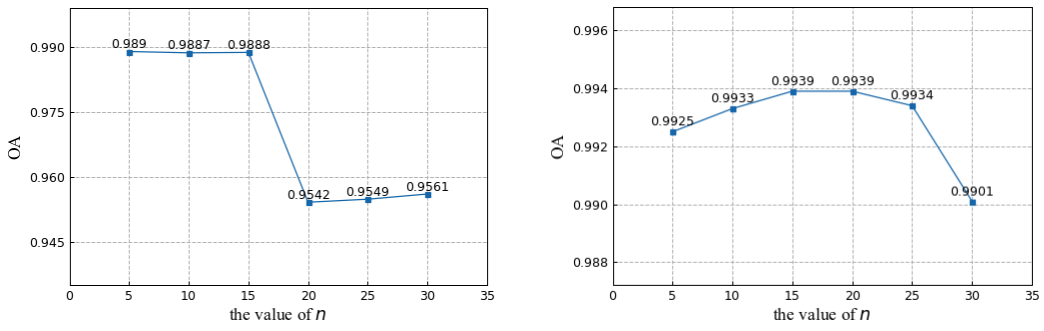



(**a**) (**b**) (**c**) (**d**)

**Figure 13.** Analysis of the momentum parameter $\alpha$. (**a**) Bastrop dataset. (**b**) Umatilla dataset. (**c**) Yancheng dataset. (**d**) River dataset.

### 5.3.2. Analysis of the Number of Groups

Sample selection by clustering methods can ensure the diversity of training samples. However, too detailed a classification makes it difficult to remove noise for datasets with simple ground objects types (the sample loss-based method is more effective here). For complicated datasets, especially HSI that are sensitive to ground changes, classified sampling is crucial to class balance and model generalization. The results for different numbers of groups are shown in Figure 14. Since the Bastrop dataset has only one change type, and the spectral information of MSI is much less than that of HSI, the value of $n$ needs to be small. Based on the experiment, we choose $n = 10$ on the Bastrop dataset. For the other three HSI datasets, we choose $n = 20$. Moreover, other clustering methods that automatically determine the number of groups can be considered to avoid parameter selection.




(**a**) (**b**)

**Figure 14.** *Cont.*

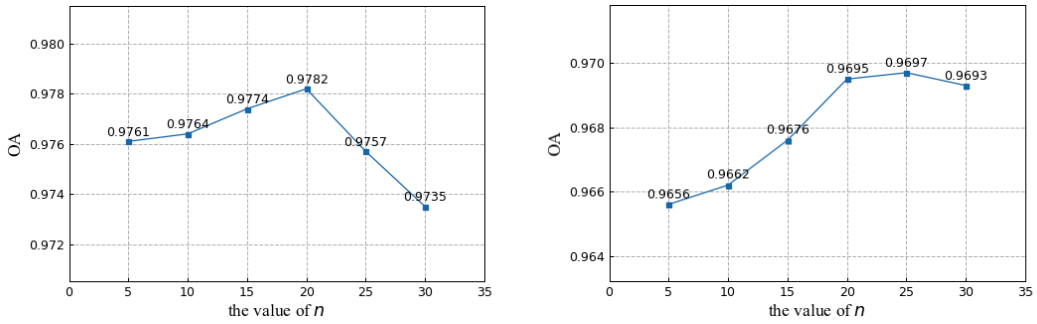



**Figure 14.** Analysis of the number of groups $n$. (**a**) Bastrop dataset. (**b**) Umatilla dataset. (**c**) Yancheng dataset. (**d**) River dataset.

### *5.4. Computing Time*

The computing device is equipped with Intel i7-9700K CPU (3.6 GHz) and NVIDIA GeForce RTX2080Ti GPU. The program is written in Python via the code library of PyTorch. Here, we list the computing time for each dataset in Table 5. With the multiple models and numerous iterations for optimization, the proposed method suffers from high computational complexity. Theoretically, the optimization process of the two models is independent. Therefore, the method can be accelerated by parallel computing to reduce the computing time by 50%, which is the same as the self-training time of a single model.

**Table 5.** Time cost (seconds) of each dataset.

| | **Bastrop** | **Umatilla** | **Yancheng** | **River** |
|---|---|---|---|---|
| 3DCNN | 343.52 | 51.18 | 46.94 | 67.99 |
| ours | 2919.93 | 414.29 | 339.44 | 496.96 |

## 6. Conclusions

In this article, a general mutual teaching framework with momentum correction is proposed for the HSI-CD task by dual-3DCNN. It aims to perform robust training for deep-learning methods using pseudo-labels generated by traditional approaches. Adopting the idea of collaborative training, the proposed framework encourages the two models to teach each other to mitigate self-confidence bias and boosts label correction in the iterative process to further improve performance. Then, focusing on the complexity of HSI change types, a new sample selection method based on group confidence is designed to extract better quality and diverse training data. Furthermore, the 3DCNN can effectively extract spatiotemporal spectral features of bitemporal HSIs, and thus, it is developed as the basic classifier of the above framework. Our approach uses pseudo-labels obtained by unsupervised algorithms, which means it can also be compatible with other networks that require labeled data.

We implemented our approach and performed experiments on a multispectral dataset, as well as on three public hyperspectral datasets. The visual and quantitative results show that our method can effectively improve the robustness and generalization of the deep neural network for the HSI-CD task.

**Author Contributions:** Conceptualization, J.S. and L.H.; methodology, L.X.; software, J.S.; validation, J.S., L.X. and J.L.; formal analysis, L.X. and J.L.; writing—original draft preparation, J.S.; writing—review and editing, L.X. and J.L.; visualization, J.S., L.H. and Z.W.; supervision, L.X., J.L. and Z.W.; project administration, L.X.; funding acquisition, L.X. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China under Grant 61571230, 61871226, and 61906093, in part by the Jiangsu Provincial Social Developing Project

under Grant BE2018727, in part by the Jiangsu Provincial Nature Science Foundations of China under Grant BK20190451, in part by the Fundamental Research Funds for the Central Universities under Grant 30918011104 and 30920021134; and in part by the National Major Research Plan of China under Grant 2016YFF0103604.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Scafutto, R.D.M.; de Souza Filho, C.R.; de Oliveira, W.J. Hyperspectral remote sensing detection of petroleum hydrocarbons in mixtures with mineral substrates: Implications for onshore exploration and monitoring. *ISPRS J. Photogramm. Remote Sens.* **2017**, *128*, 146–157. [CrossRef]
2. Carrino, T.A.; Crósta, A.P.; Toledo, C.L.B.; Silva, A.M. Hyperspectral remote sensing applied to mineral exploration in southern Peru: A multiple data integration approach in the Chapi Chiara gold prospect. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *64*, 287–300. [CrossRef]
3. Zhang, X.; Sun, Y.; Shang, K.; Zhang, L.; Wang, S. Crop classification based on feature band set construction and object-oriented approach using hyperspectral images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 4117–4128. [CrossRef]
4. Vali, A.; Comai, S.; Matteucci, M. Deep learning for land use and land cover classification based on hyperspectral and multispectral earth observation data: A review. *Remote Sens.* **2020**, *12*, 2495. [CrossRef]
5. Shimoni, M.; Haelterman, R.; Perneel, C. Hypersectral imaging for military and security applications: Combining myriad processing and sensing techniques. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 101–117. [CrossRef]
6. Singh, A. Review article digital change detection techniques using remotely-sensed data. *Int. J. Remote Sens.* **1989**, *10*, 989–1003. [CrossRef]
7. Bovolo, F.; Bruzzone, L. A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain. *IEEE Trans. Geosci. Remote Sens.* **2006**, *45*, 218–236. [CrossRef]
8. Celik, T. Unsupervised change detection in satellite images using principal component analysis and *k*-means clustering. *IEEE Geosci. Remote Sens. Lett.* **2009**, *6*, 772–776. [CrossRef]
9. Nielsen, A.A.; Conradsen, K.; Simpson, J.J. Multivariate alteration detection (MAD) and MAF postprocessing in multispectral, bitemporal image data: New approaches to change detection studies. *Remote Sens. Environ.* **1998**, *64*, 1–19. [CrossRef]
10. Nielsen, A.A. The regularized iteratively reweighted MAD method for change detection in multi-and hyperspectral data. *IEEE Trans. Image Process.* **2007**, *16*, 463–478. [CrossRef]
11. Coppin, P.; Jonckheere, I.; Nackaerts, K.; Muys, B.; Lambin, E. Review ArticleDigital change detection methods in ecosystem monitoring: A review. *Int. J. Remote Sens.* **2004**, *25*, 1565–1596. [CrossRef]
12. Wu, C.; Du, B.; Zhang, L. Slow feature analysis for change detection in multispectral imagery. *IEEE Trans. Geosci. Remote Sens.* **2013**, *52*, 2858–2874. [CrossRef]
13. Ahlqvist, O. Extending post-classification change detection using semantic similarity metrics to overcome class heterogeneity: A study of 1992 and 2001 US National Land Cover Database changes. *Remote Sens. Environ.* **2008**, *112*, 1226–1241. [CrossRef]
14. Wan, L.; Xiang, Y.; You, H. A post-classification comparison method for SAR and optical images change detection. *IEEE Geosci. Remote Sens. Lett.* **2019**, *16*, 1026–1030. [CrossRef]
15. Liu, T.; Yang, L.; Lunga, D. Change detection using deep learning approach with object-based image analysis. *Remote Sens. Environ.* **2021**, *256*, 112308. [CrossRef]
16. Mou, L.; Bruzzone, L.; Zhu, X.X. Learning spectral-spatial-temporal features via a recurrent convolutional neural network for change detection in multispectral imagery. *IEEE Trans. Geosci. Remote Sens.* **2018**, *57*, 924–935. [CrossRef]
17. Wang, Q.; Yuan, Z.; Du, Q.; Li, X. GETNET: A general end-to-end 2-D CNN framework for hyperspectral image change detection. *IEEE Trans. Geosci. Remote Sens.* **2018**, *57*, 3–13. [CrossRef]
18. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6232–6251. [CrossRef]
19. Du, B.; Ru, L.; Wu, C.; Zhang, L. Unsupervised deep slow feature analysis for change detection in multi-temporal remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 9976–9992. [CrossRef]
20. Li, X.; Yuan, Z.; Wang, Q. Unsupervised deep noise modeling for hyperspectral image change detection. *Remote Sens.* **2019**, *11*, 258. [CrossRef]
21. Li, Q.; Gong, H.; Dai, H.; Li, C.; He, Z.; Wang, W.; Feng, Y.; Han, F.; Tuniyazi, A.; Li, H.; et al. Unsupervised Hyperspectral Image Change Detection via Deep Learning Self-generated Credible Labels. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 9012–9024. [CrossRef]
22. Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; Vinyals, O. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM* **2021**, *64*, 107–115. [CrossRef]

23. Yu, X.; Han, B.; Yao, J.; Niu, G.; Tsang, I.; Sugiyama, M. How does disagreement help generalization against label corruption? In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019; pp. 7164–7173.
24. Yao, Q.; Yang, H.; Han, B.; Niu, G.; Kwok, J.T.Y. Searching to exploit memorization effect in learning with noisy labels. In Proceedings of the International Conference on Machine Learning, Virtual Event, 13–18 July 2020; pp. 10789–10798.
25. Shang, R.; Yuan, Y.; Jiao, L.; Meng, Y.; Ghalamzan, A.M. A self-paced learning algorithm for change detection in synthetic aperture radar images. *Signal Proc.* **2018**, *142*, 375–387. [CrossRef]
26. Gong, M.; Duan, Y.; Li, H. Group self-paced learning with a time-varying regularizer for unsupervised change detection. *IEEE Trans. Geosci. Remote Sens.* **2019**, *58*, 2481–2493. [CrossRef]
27. Han, B.; Yao, Q.; Yu, X.; Niu, G.; Xu, M.; Hu, W.; Tsang, I.; Sugiyama, M. Co-teaching: Robust Training of Deep Neural Networks with Extremely Noisy Labels. In Proceedings of the International Conference on Neural Information Processing Systems, Stockholm, Sweden, 10–15 July 2018; pp. 8536–8546.
28. Li, P.; Xu, Y.; Wei, Y.; Yang, Y. Self-correction for human parsing. *arXiv* **2020**, arXiv:1910.09777.
29. Zheng, G.; Awadallah, A.H.; Dumais, S. Meta label correction for noisy label learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual Event, 2–9 February 2021.
30. Liu, J.; Gong, M.; Qin, K.; Zhang, P. A deep convolutional coupling network for change detection based on heterogeneous optical and radar images. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *29*, 545–559. [CrossRef] [PubMed]
31. Zhang, P.; Gong, M.; Zhang, H.; Liu, J.; Ban, Y. Unsupervised difference representation learning for detecting multiple types of changes in multitemporal remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2018**, *57*, 2277–2289. [CrossRef]
32. Liu, J.; Zhang, W.; Liu, F.; Xiao, L. A Probabilistic Model Based on Bipartite Convolutional Neural Network for Unsupervised Change Detection. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 4701514. [CrossRef]
33. Gong, M.; Zhao, J.; Liu, J.; Miao, Q.; Jiao, L. Change detection in synthetic aperture radar images based on deep neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *27*, 125–138. [CrossRef]
34. Li, Y.; Peng, C.; Chen, Y.; Jiao, L.; Zhou, L.; Shang, R. A deep learning method for change detection in synthetic aperture radar images. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 5751–5763. [CrossRef]
35. Goldberger, J.; Ben-Reuven, E. Training deep neural-networks using a noise adaptation layer. In Proceedings of the International Conference of Learning Representations (ICLR), San Juan, Puerto Rico, 2–4 May 2016.
36. Ghosh, A.; Kumar, H.; Sastry, P. Robust loss functions under label noise for deep neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31.
37. Lee, K.H.; He, X.; Zhang, L.; Yang, L. Cleannet: Transfer learning for scalable image classifier training with label noise. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 5447–5456.
38. Jiang, L.; Zhou, Z.; Leung, T.; Li, L.J.; Li, F.-F. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In Proceedings of the International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 2304–2313.
39. Blum, A.; Mitchell, T. Combining labeled and unlabeled data with co-training. In Proceedings of the Eleventh Annual Conference on Computational Learning Theory, Madison, WI, USA, 24–26 July 1998; pp. 92–100.
40. Zhang, Y.; Xiang, T.; Hospedales, T.M.; Lu, H. Deep mutual learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4320–4328.
41. Ge, Y.; Chen, D.; Li, H. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
42. Yang, F.; Li, K.; Zhong, Z.; Luo, Z.; Sun, X.; Cheng, H.; Guo, X.; Huang, F.; Ji, R.; Li, S. Asymmetric co-teaching for unsupervised cross-domain person re-identification. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 12597–12604.
43. Liu, J.; Li, R.; Sun, C. Co-Correcting: Noise-tolerant Medical Image Classification via mutual Label Correction. *IEEE Trans. Med. Imag.* **2021**, *40*, 3580–3592 [CrossRef] [PubMed]
44. Tai, X.; Li, M.; Xiang, M.; Ren, P. A Mutual Guide Framework for Training Hyperspectral Image Classifiers with Small Data. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 5510417. [CrossRef]
45. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
46. Volpi, M.; Camps-Valls, G.; Tuia, D. Spectral alignment of multi-temporal cross-sensor images with automated kernel canonical correlation analysis. *ISPRS J. Photogramm. Remote Sens.* **2015**, *107*, 50–63. [CrossRef]