

Article Auto-Weighted Structured Graph-Based Regression Method for Heterogeneous Change Detection

Lingjun Zhao *, Yuli Sun ^(D), Lin Lei and Siqian Zhang

College of Electronic Science, National University of Defense Technology, Changsha 410073, China

* Correspondence: nudtzlj@163.com

Abstract: Change detection using heterogeneous remote sensing images is an increasingly interesting and very challenging topic. To make the heterogeneous images comparable, some graphbased methods have been proposed, which first construct a graph for the image to capture the structure information and then use the graph to obtain the structural changes between images. Nonetheless, previous graph-based change detection approaches are insufficient in representing and exploiting the image structure. To address these issues, in this paper, we propose an autoweighted structured graph (AWSG)-based regression method for heterogeneous change detection, which mainly consists of two processes: learning the AWSG to capture the image structure and using the AWSG to perform structure regression to detect changes. In the graph learning process, a self-conducted weighting strategy is employed to make the graph more robust, and the local and global structure information are combined to make the graph more informative. In the structure regression process, we transform one image to the domain of the other image by using the learned AWSG, where the high-order neighbor information hidden in the graph is exploited to obtain a better regression image and change image. Experimental results and comparisons on four real datasets with seven state-of-the-art methods demonstrate the effectiveness of the proposed approach.



1. Introduction

1.1. Background

Change detection (CD) is a very important topic in remote sensing, which can detect changes occurring on the Earth's surface by comparing images collected at different times [1]. CD has a wide range of applications, such as agricultural surveys, urban planning, environmental monitoring, and disaster assessment [2–4].

Previous CD methods often compare multitemporal images from the same sensor to extract changes, which can be referred to as homogeneous CD or monomodal CD, such as the homogeneous CD of SAR images [5], multispectral images [6,7], and hyperspectral images [8]. However, homogeneous CD has the limitation that the compared images from the same sensor may not always be acquired in time. In this case, we need to utilize the heterogeneous CD, i.e., using multitemporal images from different sensors to extract changes.

Heterogeneous CD can be regarded as an extension of homogeneous CD, which has two additional advantages (compared to homogeneous CD) [9,10]: first, it can reduce the response time of change analysis in emergency situations (such as floods, earthquakes, etc.) by using any available image independently of its modality, instead of having to wait for the homogeneous images to be sensed. More importantly, homogeneous images are often unavailable due to the accompanying adverse conditions of weather and lighting. Second, heterogeneous CD can improve the temporal resolution of change analysis by inserting heterogeneous data.



Citation: Zhao, L.; Sun, Y.; Lei, L.; Zhang, S. Auto-Weighted Structured Graph-Based Regression Method for Heterogeneous Change Detection. *Remote Sens.* 2022, *14*, 4570. https:// doi.org/10.3390/rs14184570

Academic Editor: Mohammad Awrangjeb

Received: 10 August 2022 Accepted: 9 September 2022 Published: 13 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).



1.2. Related Work

Compared to homogeneous CD, heterogeneous CD is more challenging because the multitemporal images are not directly comparable owing to the fact that they come from different modalities. In order to make heterogeneous images comparable, the first task addressed by heterogeneous CD is to transform them to the same domain, which is related to the topic of data transformation, feature learning, and domain adaptation [11–13].

As presented in [14–16], a more specific classification of heterogeneous CD methods is challenging. However, the distinction between using supervised data or not in the heterogeneous CD is still obvious. The supervised methods usually use the labeled unchanged data pairs to train the transformation functions [17–19]. However, acquiring supervised data is difficult in practice [20] as it requires high labor labeling costs and strong expert knowledge. To avoid the dependence on labeled data, unsupervised methods usually construct pseudo-training sets or use iterative methods to filter the training data [14,21–23]. At the same time, depending on the method used in the comparison, heterogeneous CD can also be divided into traditional machine learning-based and deep-learning-based. The former include methods using energy model [24], dictionary learning [25], copula theory [26], patch similarity graph [27,28], manifold learning [29], kernel canonical correlation analysis [30], fractal projection [15], graph learning [31], kernel regression [14,17]. The latter consist of methods using image style transferring network [32], cycle-consistent generative adversarial networks [18,33], commonality autoencoder [34], symmetric convolutional coupling network [22], self-supervised learning with Siamese networks [35,36].

Recently, a class of graph-based heterogeneous CD methods has been proposed, which usually first construct a graph for each image to capture the structure information, and then operate on the graphs to obtain changes. Specifically, Wu et al. [37] propose a semi-supervised multiscale graph convolutional network (GCN) to treat the CD problem as a node classification task, which constructs a multiscale graph for the stacked image and then propagates the information from labeled nodes to unlabeled ones by using GCN. The graph fusion-based CD method (GFBCD) [38] first constructs an approximate local graph for each image by using the Nystrom extension, and then fuses the graphs to calculate the changes by minimizing similarity between graphs. Furthermore, the GFBCD is improved in [39] by using a signal smoothness representation in the graph learning stage and a graph Laplacian-based regularization in the change estimating stage, which is referred to as GSMO-CD for short.

Based on the structure consistency between heterogeneous images, two types of graphbased methods are proposed: structure comparison methods and structure regression methods. For the former, they first construct a k-nearest-neighbors (KNN) graph for each image with Euclidean distance metric, and then directly compare the differences between the graphs by graph projection, such as the nonlocal patch similarity graph-based method (NPSG) [40], the improved nonlocal patch graph-based method (INLPG) [28], the iterative robust graph and Markovian co-segmentation-based method (IRG-McS) [23]. For the structure regression-based methods, a graph is first constructed for the pre-event image, and then the pre-event image is translated to the domain of post-event image by a structural regression or projection model, and finally, the translated image and post-event image are compared in the same domain as the homogeneous CD problem, such as the sparse-constrained adaptive structure consistency-based method (SCASC) [10], the fractal projection and Markovian segmentation method (FPMS) [15], and the adaptive graph with structure cycle consistency-based method (AGSCC) [9]. In addition, both types of the structure comparison methods and structure regression methods can be incorporated into the framework of graph signal processing for heterogeneous change detection (HCD-GSP): the structure comparison methods can be regarded as computing the difference of the same graph signal passing through different graphs in the vertex domain filtering of HCD-GSP [41], while the structure regression methods can be regarded as computing the difference of different graph signals passing through the same graph filter with the spectral domain analysis of HCD-GSP [42].

For these graph-based heterogeneous CD methods [9,10,15,23,28,37–40], there are three attractive advantages: first, they focus on structural features rather than illumination features (or pixel-wise features), which are very robust under different imaging conditions, while also being able to mitigate the effects of noise; second, they extract changes by analyzing the graph vertices, whose number is much smaller than the number of pixels, thus significantly reducing the complexity of operations; third, these methods are all very intuitive and interpretable (with clear physical meaning), while being able to be combined with other frameworks and extended to other applications.

1.3. Motivation and Contribution

Motivated by recent advances of the graph-based HCD methods, this paper proposes an auto-weighted structured graph (AWSG)-based method, which falls under the structure regression-based approach described above. Compared with the structure comparison methods that only provide the change results, i.e., difference image (DI) or the change map (CM), the structure regression methods also provide a translated image of another moment, which has a stronger visualization effect.

For the graph-based heterogeneous CD methods, two main aspects need to be solved: first, how to construct the graph to represent the image structure; second, how to detect changes by using the graph. Although the previous methods have achieved relatively promising results, they still have limitations in the above two aspects.

(1) First, the graph constructed by the previous method is not robust enough. The KNN graph is the most widely used to capture the structure information, such as INLPG [28], GSMO-CD [39], IRG-McS [23], SCASC [10], AGSCC [9]. It is well known that the KNN graph faces two challenges: the number (k) of nearest neighbors (NNs) and the distance metric. INLPG and GSMO-CD connect each vertex to its NNs with a fixed number of edges (k) without considering the proportion of objects represented by each vertex, which will under-connect the vertex with many truly similar vertices and over-connect the vertex with few truly similar vertices, thus making the graph unstable. To address this challenge, SCASC, IRG-McS, and AGSCC use an in-degree-based strategy to select an adaptive k for each vertex. However, the previous methods directly use the Euclidean distance metric and ignore the feature difference in the graph construction or graph learning, that is, these methods calculate the distance or similarity between vertices by assigning the same weight to all features while ignoring the distinguishing capability of different features in heterogeneous images.

In this paper, we learn an adaptive KNN graph for the structure regression method, which can connect each vertex to suitable similar neighbors with appropriate weights by using two techniques. First, it uses the *k* selection strategy proposed in IRG-McS [23] to choose the number of NNs for each vertex. Second, but more importantly, it uses an autoweighted distance metric that adaptively assigns different weights for different features by using a self-conducted weight learning, which can make the graph more robust.

(2) Second, the graphs constructed by the previous methods cannot completely characterize the structure of the image, i.e., the representation ability of these graphs is insufficient. For example, the FPMS [15] and PSGM [27] use the self-expression property to learn the graph that can reconstruct each vertex by using other connected vertices, which can capture the global structure of an image. The SCASC [10] and INLPG [28] use the self-similarity property to construct the graph that connects each vertex with its NNs, which can capture the local structure of an image. However, from the perspective of information mining, we certainly hope that the graph can characterize richer structural information, which is more conducive to the subsequent structure comparison or structure regression for heterogeneous CD.

In this paper, we combine the local and global structure information to learn a structured graph, which uses the self-expression property to preserve the global structure of an image and uses the adaptive neighbor approach to capture the local structure of an image. In the graph learning process, we incorporate the local similarity-based regularization and global self-expression-based regularization. Then, the structured graph can find the most informative neighbors for each target vertex, which are not only the NNs of this target node, but can also reconstruct this target node using the corresponding edge weights. Therefore, the proposed graph can contain more structure information than graphs providing only global structure information [15] and graphs providing only local structure information [10].

(3) Third, previous methods have not fully utilized the information contained in the graph to detect changes. Once the graphs are constructed or learned, they are directly fused as in GFBCD [38] and GSMO-CD [39], or compared by projection as in INLPG [28] and IRG-McS [23], or transformed for structure regression as in SCASC [10]. However, these methods only use the first-order neighborhood information of the graphs, that is, they only consider the connections between each vertex and its NNs that are directly adjacent to it, but ignore the high-order neighbor information hidden in the graph. Therefore, this insufficient exploitation of graph information limits the change detection performance.

Inspired by the GSP framework of the heterogeneous CD [41,42], we exploit the high-order neighbor information of the graph in the structure regression process, that is, the information (or attributes) of vertices that are within *K*-hop ($K \ge 2$) away from the target vertex are also propagate to this vertex. Specifically, in the structure regression model, we decompose the post-event image into the translated pre-event image and the change image, and constrain the translated image to inherit the structure of the pre-event image by incorporating the global self-expression constraint and local similarity constraint, which are derived from the high-order information of the graph learned from the pre-event image. By effectively exploiting the information contained in the graph, the proposed method can obtain a better regression image and change image, thus improving the CD performance.

The main contribution of this paper is to improve the performance of structure regression-based heterogeneous change detection methods from two aspects: structure representation and exploiting the graph information, which can be summarized as follows:

- We learn an auto-weighted structured graph to represent the image structure by using a self-conducted weight learning;
- We combine the local and global structure information in the graph learning process to obtain a more informative graph;
- We exploit the high-order neighbor information of the graph in the structure regression process;
- We conduct comprehensive experiments on different real datasets to demonstrate the
 effectiveness of the proposed method.

1.4. Outline

The remainder of this paper is composed of five sections. In Section 2, we describe the graph learning model. In Section 3, we describe the structure regression model. In Section 4, the experimental results are analyzed. Finally, Section 5 concludes this paper and discusses the future work.

2. Auto-Weighted Structured Graph Learning

We consider a pair of co-registered heterogeneous images of $\tilde{\mathbf{X}} \in \mathbb{R}^{I \times J \times C_x}$ and $\tilde{\mathbf{Y}} \in \mathbb{R}^{I \times J \times C_y}$ obtained by different sensors over the same region before and after a change event, respectively, where C_x and C_y represent the channel numbers of $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$, respectively. The goal of heterogeneous CD is to detect the changed region.

Since the heterogeneous images of $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ are from different domains, we cannot directly compare them to calculate the change image. Therefore, we need to find the connection between them and make them comparable. As illustrated in [23], there is structural consistency between heterogenous images. We first segment $\tilde{\mathbf{X}}$ and $\tilde{\mathbf{Y}}$ into small parts (superpixels) of $\tilde{\mathbf{X}}_i$ and $\tilde{\mathbf{Y}}_i$, $i = 1, \dots, N$, with the same segmentation map, then the structural consistency can be described in the following way: if $\tilde{\mathbf{X}}_i$ and $\tilde{\mathbf{X}}_j$ belong to the same type of object (behaving as they are quite similar), and none of them changed during the event, then $\tilde{\mathbf{Y}}_i$ and $\tilde{\mathbf{Y}}_j$ also belong to the same type of object (behaving as they are also quite similar). If we define the similarity relationships between the superpixels within the pre-event image, i.e., the relationships between \tilde{X}_i and \tilde{X}_j , $i, j = 1, \dots, N$, as the structure of pre-event image, then this structure can be inherited by the post-event image if no change occurs. Therefore, we can employ this structural consistency to perform the image translation, that is, transform the pre-event image to the domain of post-event image and then compare the translated image and post-event image in the same domain to obtain the change map. In this way, the proposed method contains four steps: (1) pre-processing; (2) learning graph to capture the image structure; (3) translating image by using structural consistency; (4) extracting changes by binary segmentation. The framework of the proposed method is shown in Figure 1.



Figure 1. Framework of the proposed AWSG-based HCD method.

2.1. Pre-Processing

As the pairwise relationships is used to represent the image structure, we choose the superpixel as the basic processing unit, which has two advantages: first, it can contain sufficient contextual information and preserve the edges of the object; second, it can greatly reduce the computational complexity of the proposed method.

We choose the fractal net evolution approach (FNEA) [43] to complete the superpixel segmentation because of its good performance on both efficiency and boundary preservation. We first stack the pre-event image and post-event image into a stacked image with $C_x + C_y$ channels, then we use the FNEA to segment this stacked image into *N* superpixels with the segmentation map Λ . Then, we project this segmentation map to the original images to obtain the superpixels of $\mathbf{\tilde{X}}_i$ and $\mathbf{\tilde{Y}}_i$, $i = 1, \dots, N$, so that the pair of superpixels with the same index from different image (such as $\mathbf{\tilde{X}}_i$ and $\mathbf{\tilde{Y}}_i$) represent the same spatial region.

After obtaining the superpixels, different features can be extracted, such as the spectral, spatial, and textural features. For each channel of the *i*-th superpixel, we extract *M* features and define the *m*-th feature extraction operator as $\mathcal{F}^{(m)}$, then we can obtain the feature vectors as $\mathbf{X}_i^m = \mathcal{F}^{(m)}(\mathbf{\tilde{X}}_i) \in \mathbb{R}^{C_x}$ and $\mathbf{Y}_i^m = \mathcal{F}^{(m)}(\mathbf{\tilde{Y}}_i) \in \mathbb{R}^{C_y}$. Then, we can obtain the stacked feature matrices as $\mathbf{X} \in \mathbb{R}^{C_x \times N \times M}$ and $\mathbf{Y} \in \mathbb{R}^{C_y \times N \times M}$, where $\mathbf{X}^m = \mathbf{X}_{:::,m} \in \mathbb{R}^{C_x \times N}$ and $\mathbf{Y}^m = \mathbf{Y}_{:::,m} \in \mathbb{R}^{C_y \times N}$ represents the *m*-th feature matrices.

2.2. Graph Learning

To capture the structure information of the pre-event image, we construct a directed weighted graph $\mathbf{G} = (V, E, S)$, where the vertex set is $V = \{\tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2, \dots, \tilde{\mathbf{X}}_N\}$ that sets each superpixel as a vertex, and the edge set is $E = \{(\tilde{\mathbf{X}}_i, \tilde{\mathbf{X}}_j) | S_{i,j} \neq 0; i, j = 1, \dots, N\}$ that connects $\tilde{\mathbf{X}}_i$ to $\tilde{\mathbf{X}}_j$ with the weight $S_{i,j}$.

2.3. Structured Graph Learning Model

In SCASC [10], a distance-induced graph is proposed, which is based the intuitive constraint: the smaller the distance between \tilde{X}_i and \tilde{X}_j , the larger the weight between them, i.e., $S_{i,j}$, should be. SCASC learns the local structure with the following model:

$$\min_{\mathbf{S}} \sum_{i,j=1}^{N} d_{i,j}^{x} S_{i,j} + \alpha \left(S_{i,j} \right)^{2}, \ s.t. \ \sum_{i=1}^{N} S_{i,j} = 1, \ S_{i,j} \ge 0$$
(1)

where $d_{i,j}^x$ is the distance between superpixels of $\tilde{\mathbf{X}}_i$ and $\tilde{\mathbf{X}}_j$ and $\alpha > 0$ is a penalty parameter. As will be shown latter, the **S** is a sparse column when choosing a suitable α . This type of graph describes the similarity relationships between each vertex and its local nearest neighbors.

Different from SCASC, the self-expression property is applied to learn the global structure in PSGM [27], which assumes that each vertex can be reconstructed by other connected vertices. PSGM learns the graph by using the following model:

$$\min_{\mathbf{s}} f(\mathbf{S}), \ s.t. \ \mathbf{X}^m = \mathbf{X}^m \mathbf{S}, \ S_{i,j} \ge 0$$
⁽²⁾

where $f(\mathbf{S})$ is the regularization of **S**. As (2) uses all the vertices to represent each vertex with the self-weighing matrix as $\mathbf{X}_{i}^{m} = \mathbf{X}^{m}\mathbf{S}_{i}$, this graph is expected to contain global structure information of the pre-event image.

As has been shown in many problems, such as classification [44], clustering [45,46], and feature selection [47], both global and local structural information are essential for graph performance because they can provide complementary information. In this paper, we combine the local structure learning of (1) and global structural learning of (2) into a structured graph learning model as

$$\min_{\mathbf{S},\varepsilon} \sum_{i,j=1}^{N} d_{i,j}^{x} S_{i,j} + \alpha \left(S_{i,j}\right)^{2} + \beta \sum_{m=1}^{M} g(\varepsilon^{m})$$

s.t. $\mathbf{X}^{m} = \mathbf{X}^{m} \mathbf{S} + \varepsilon^{m}, \sum_{i=1}^{N} S_{i,j} = 1, S_{i,j} \ge 0$ (3)

where $\beta > 0$ represents tuning parameters, ε denotes the self-expression error caused by the noise or corruption, $g(\cdot)$ denotes the penalty term. In the model (3), the local structure penalty term of $\sum_{i,j=1}^{N} d_{i,j}^{x} S_{i,j}$ tends to provide a smaller weight $S_{i,j}$ to the edge $(\tilde{\mathbf{X}}_{i}, \tilde{\mathbf{X}}_{j})$ when the distance between $\tilde{\mathbf{X}}_{i}$ and $\tilde{\mathbf{X}}_{j}$ is larger. The global self-expression penalty term of $\sum_{m=1}^{M} g(\varepsilon^{m})$ tends to reconstruct \mathbf{X}_{i} from the whole \mathbf{X} with \mathbf{S}_{i} as $\mathbf{X}_{i}^{m} \approx \sum_{j=1}^{N} \mathbf{X}_{j}^{m} S_{j,i}$. The smooth regularizer of $\sum_{i,j=1}^{N} \alpha \left(S_{i,j}\right)^{2}$ is used to avoid a trivial solution and make \mathbf{S} column-sparse with the constraint of $\sum_{i=1}^{N} S_{i,j} = 1$, $S_{i,j} \ge 0$. Specially, if we choose $\alpha = 0$, we have a trivial solution for \mathbf{S}_{i} as $S_{i,i} = 1$ and $S_{j,i} = 0$, $j \neq i$; if we set $\alpha \to \infty$, we have the trivial solution for \mathbf{S}_{i} as $\mathbf{S}_{i} = \mathbf{1}_{N}/N$, where $\mathbf{1}_{N}$ denotes an $N \times 1$ column vector of ones.

2.4. Weighted Graph Learning Model

In the structured graph learning model of (3), there are two points that directly affect the performance of the graph: the α that determines the number of NNs, and the distance

metric that determines the neighbors position and weight of **S**. That is, α controls the number of non-zero elements in **S**, and $d_{i,j}^x$ and $g(\varepsilon^m)$ controls the index and value of each non-zero element of **S**.

The previous methods usually connect each vertex to its NNs with a fixed number of edges (*k*) without considering the proportion of objects, such as INLPG [28], NPSG [40], GSMO-CD [39], which will cause under-connection or over-connection of some vertices. Meanwhile, the previous methods directly use the Euclidean distance metric and ignore the distinguishing capability of different features in heterogeneous images in the graph learning, such as SCASC [10], INLPG [28], IRG-McS [23], which is not robust and limits the structural representational capacity of the graph. To address these two problems, a weighted structured graph learning model can be used

$$\min_{\mathbf{S},\varepsilon,w} \sum_{m=1}^{M} w_m \left\{ \sum_{i,j=1}^{N} d_{i,j,m}^x S_{i,j} + \beta g(\varepsilon^m) \right\} + \sum_{i=1}^{N} \alpha_i \|\mathbf{S}_i\|_2^2
s.t. \mathbf{X}^m = \mathbf{X}^m \mathbf{S} + \varepsilon^m, \sum_{i=1}^{N} S_{i,j} = 1, \ S_{i,j} \ge 0, \sum_{m=1}^{M} w_m^\eta = 1, w_m \ge 0$$
(4)

where $d_{i,j,m}^x = \left\| \mathbf{X}_i^m - \mathbf{X}_j^m \right\|_2^2$ denotes the *m*-th feature distance between $\mathbf{\tilde{X}}_i$ and $\mathbf{\tilde{X}}_j$, w_m denotes the *m*-th feature weight, and $\alpha_i > 0$ is used to determine the NNs number of the *i*-th vertex, and $\eta > 0$ is used to control the weights distribution. By comparing the models (3) and (4), we find that the weighted structured graph learning model (4) employs the parameter α_i for each vertex to control the number of NNs, and introduces the w_m to assign the weights for different features.

2.5. Auto-Weighted Structured Graph Learning

As we know, the fewer the parameters to be adjusted, the more robust the graph learning model. We can find that η can be adjusted in a larger range and η is crucial to the performance of graph learning in (4). Additionally, we want to eliminate the need for manual adjustment of this parameter for different images, that is, we expect to remove such a parameter while pursuing good learning performance.

Motivated by recently proposed iteratively re-weighted technique [48], we propose an auto-weighted structured graph learning model by using a self-conducted weight learning

$$\min_{\mathbf{S},\varepsilon} \sum_{m=1}^{M} \sqrt{\sum_{i,j=1}^{N} d_{i,j,m}^{x} S_{i,j}^{} + \beta g(\varepsilon^{m}) + \sum_{i=1}^{N} \alpha_{i} \|\mathbf{S}_{i}\|_{2}^{2}}$$

$$s.t. \, \mathbf{X}^{m} = \mathbf{X}^{m} \mathbf{S} + \varepsilon^{m}, \sum_{i=1}^{N} S_{i,j}^{} = 1, \, S_{i,j}^{} \ge 0$$
(5)

We can find that no weight factors are explicitly defined in (5). The Lagrange function of the minimization problem (5) can be formulated as

$$\mathcal{L}(\mathbf{S},\varepsilon,\Psi) = \sum_{m=1}^{M} \sqrt{\sum_{i,j=1}^{N} d_{i,j,m}^{x} S_{i,j}} + \beta g(\varepsilon^{m}) + \sum_{i=1}^{N} \alpha_{i} \|\mathbf{S}_{i}\|_{2}^{2} + \mathcal{G}(\Psi, S, \varepsilon)$$
(6)

where $\mathcal{G}(\Psi, S, \varepsilon)$ is the formalized term derived from the constraints of (5) with Ψ representing the Lagrange multipliers. Taking the derivative of (6) with respect to **S** and ε , and setting them to zero, we can obtain

$$\sum_{m=1}^{M} w_m \frac{\partial \sum_{i,j=1}^{N} d_{i,j,m}^x S_{i,j} + \beta g(\varepsilon^m)}{\partial \mathbf{S}} + \frac{\partial \mathcal{G}(\Psi, S, \varepsilon)}{\partial \mathbf{S}} = 0;$$

$$\sum_{m=1}^{M} w_m \frac{\partial \sum_{i,j=1}^{N} d_{i,j,m}^x S_{i,j} + \beta g(\varepsilon^m)}{\partial \varepsilon^m} + \frac{\partial \mathcal{G}(\Psi, S, \varepsilon)}{\partial \varepsilon^m} = 0$$
(7)

where

$$w_m = \frac{1}{2\sqrt{\sum_{i,j=1}^N d_{i,j,m}^x S_{i,j} + \beta g(\varepsilon^m)}}$$
(8)

Note that w_m is determined by the target variables of **S** and ε , so (5) cannot be directly solved. However, if w_m is set to be satisfactory, (5) can be regarded as the solution of the problem

$$\min_{\mathbf{S},\varepsilon} \sum_{m=1}^{M} w_m \left\{ \sum_{i,j=1}^{N} d_{i,j,m}^x S_{i,j} + \beta g(\varepsilon^m) \right\} + \sum_{i=1}^{N} \alpha_i \|\mathbf{S}_i\|_2^2
s.t. \mathbf{X}^m = \mathbf{X}^m \mathbf{S} + \varepsilon^m, \sum_{i=1}^{N} S_{i,j} = 1, \ S_{i,j} \ge 0,$$
(9)

which can be solved. This inspires us to use an iterative method to optimize problem (5): updating **S** by solving problem (9) with fixed w_m , and then updating w_m by using (8) with fixed **S** and ε .

2.6. Optimization

Define the *m*-th feature distance matrix as \mathbf{D}_{x}^{m} , whose *i*, *j*-th element is $d_{i,j,m}^{x}$, i.e., the *m*-th feature distance between superpixels $\mathbf{\tilde{X}}_{i}$ and $\mathbf{\tilde{X}}_{j}$, and define $\alpha^{\frac{1}{2}}$ as a diagonal matrix with the *i*-th diagonal element being $\sqrt{\alpha_{i}}$. Then, we can rewrite the problem (9) as

$$\min_{\mathbf{S},\varepsilon} \sum_{m=1}^{M} w_m \left\{ Tr \left(\mathbf{S}^T \mathbf{D}_x^m \right) + \beta g(\varepsilon^m) \right\} + \left\| \mathbf{S} \alpha^{\frac{1}{2}} \right\|_F^2$$

$$s.t. \mathbf{X}^m = \mathbf{X}^m \mathbf{S} + \varepsilon^m, \mathbf{S}^T \mathbf{1}_N = \mathbf{1}_N, \mathbf{S} \ge 0$$
(10)

where $Tr(\cdot)$ represents the matrix trace.

Next, we use the alternating direction method of multipliers (ADMM) to solve the minimization problem of (10). First, we introduce an auxiliary variable $\mathbf{P} \in \mathbb{R}^{N \times N}$, and rewrite the model (9) as the minimization of

$$\mathcal{L}(\mathbf{S}, \varepsilon, \mathbf{P}) = \sum_{m=1}^{M} w_m \Big\{ Tr \Big(\mathbf{S}^T \mathbf{D}_x^m \Big) + \beta g(\varepsilon^m) \Big\} + \Big\| \mathbf{S} \alpha^{\frac{1}{2}} \Big\|_F^2 + Tr \Big((\mathbf{S} - \mathbf{P})^T \mathbf{R}_1 \Big) + \sum_{m=1}^{M} Tr \Big((\mathbf{X}^m - \mathbf{X}^m \mathbf{P} - \varepsilon^m)^T \mathbf{R}_2^m \Big) + \frac{\mu_1}{2} \| \mathbf{S} - \mathbf{P} \|_F^2 + \frac{\mu_2}{2} \sum_{m=1}^{M} \| \mathbf{X}^m - \mathbf{X}^m \mathbf{P} - \varepsilon^m \|_F^2 s.t. \, \mathbf{S}^T \mathbf{1}_N = \mathbf{1}_N, \mathbf{S} \ge 0$$
(11)

where $\mathbf{R}_1 \in \mathbb{R}^{N \times N}$ and $\mathbf{R}_2^m \in \mathbb{R}^{C_x \times N}$, $m = 1, \dots, M$ are the Lagrangian multipliers, and $\mu_1, \mu_2 > 0$ are two penalty parameters. Then, the alternating direction method (ADM) can be used to solve the minimization of (11) by iteratively updating one variable at a time and fixing the others.

 ε -subproblem. The minimization of (11) with respect to ε^m can be formulated as

$$\min_{\varepsilon^m} \beta w_m g(\varepsilon^m) - Tr\left((\varepsilon^m)^T \mathbf{R}_2^m\right) + \frac{\mu_2}{2} \|\mathbf{X}^m - \mathbf{X}^m \mathbf{P} - \varepsilon^m\|_F^2$$
(12)

We define the proximal operator as

$$prox_{\lambda g}(\mathbf{b}) := \arg\min_{\mathbf{a}} g(\mathbf{a}) + \frac{1}{2\lambda} \|\mathbf{a} - \mathbf{b}\|_{F}^{2}$$
(13)

We have that ε^m can be obtained by

$$\varepsilon^{m} = prox_{\frac{\beta w_{m}}{\mu_{2}}g}(\mathbf{Q}^{m})$$
(14)

where $\mathbf{Q}^m = \mathbf{X}^m - \mathbf{X}^m \mathbf{P} + \frac{\mathbf{R}_2^m}{\mu_2}$. According to the different regularization functions of $g(\cdot)$, ε^m can be updated with different closed-form solutions of (14). When we select the squared Frobenius norm $g(\varepsilon^m) = \|\varepsilon^m\|_F^2$, which is suitable for the case that the corruption of ε^m follows a Gaussian distribution, we have

$$\varepsilon^m = \frac{\mathbf{Q}^m}{1 + 2\beta w_m/\mu_2} \tag{15}$$

When we select the ℓ_1 -norm of $g(\varepsilon^m) = \sum_{i,j} |\varepsilon^m_{i,j}|$, which is usually adopted in the case that the corruption ε^m is the random impulse noise, we have an element-wise updating as

$$\varepsilon_{i,j}^{m} = sign\left(Q_{i,j}^{m}\right) \max\left(\left|Q_{i,j}^{m}\right| - \beta w_{m}/\mu_{2}, 0\right)$$
(16)

where $sign(\cdot)$ denotes the signum function. When we select the $\ell_{2,1}$ -norm $g(\varepsilon^m) = \|\varepsilon^m\|_{2,1}$, which is more suitable to characterize sample-specific corruption and outliers, we can update ε by using the Lemma 3.3 of [49]

$$\varepsilon_i^m = \max\left\{ \|\mathbf{Q}_i^m\|_2 - \frac{\beta w_m}{\mu_2}, 0 \right\} \frac{\mathbf{Q}_i^m}{\|\mathbf{Q}_i^m\|_2} \tag{17}$$

where the convention $0 \cdot (0/0) = 0$ is followed. For the ε -subproblem, we provide three closed forms of ε^m updates (15)–(17) based on different commonly used regularization functions of $g(\cdot)$. However, it should be noted that there may be other more appropriate g() to match the specific noise conditions in practice, e.g., impulsive noise with the Middleton noise model [50] or alpha distribution [51].

P-subproblem. By taking the derivative of $\mathcal{L}(\mathbf{S}, \varepsilon, \mathbf{P})$ of (11) with respect to **P** and setting it to zero, we can obtain

$$\sum_{m=1}^{M} \mu_2 (\mathbf{X}^m)^T (\mathbf{X}^m \mathbf{P} + \varepsilon^m - \mathbf{X}^m) - (\mathbf{X}^m)^T \mathbf{R}_2^m = \mathbf{R}_1 - \mu_1 (\mathbf{P} - \mathbf{S})$$
(18)

Then, we can update **P** by using

$$\mathbf{P} = \left(\mathbf{I}_{N} + \frac{\mu_{2}}{\mu_{1}} \sum_{m=1}^{M} \left(\mathbf{X}^{m}\right)^{T} \mathbf{X}^{m}\right)^{-1} \times \left(\mathbf{S} + \frac{\mathbf{R}_{1} + \sum_{m=1}^{M} \left(\mathbf{X}^{m}\right)^{T} \mathbf{R}_{2}^{m}}{\mu_{1}} + \frac{\mu_{2} \sum_{m=1}^{M} \left(\mathbf{X}^{m}\right)^{T} \left(\mathbf{X}^{m} - \varepsilon^{m}\right)}{\mu_{1}}\right)$$
(19)

where I_N denotes the $N \times N$ identity matrix.

S-subproblem. The minimization of (9) with respect to **S** can be formulated as a column independent form as

$$\min_{\mathbf{S}_{i}} \left(\sum_{m=1}^{M} w_{m} \mathbf{D}_{x}^{m} + \mathbf{R}_{1}\right)_{i}^{T} \mathbf{S}_{i} + \alpha_{i} \|\mathbf{S}_{i}\|_{2}^{2} + \frac{\mu_{1}}{2} \|\mathbf{S}_{i} - \mathbf{P}_{i}\|_{2}^{2}$$

$$s.t. \ \mathbf{S}_{i}^{T} \mathbf{1}_{N} = 1, \mathbf{S}_{i} \ge 0$$
(20)

By defining $\mathbf{E} = \sum_{m=1}^{M} w_m \mathbf{D}_x^m + \mathbf{R}_1 - \mu_1 \mathbf{P}$ and $\gamma_i = \frac{\mu_1}{2} + \alpha_i$, the minimization problem of (20) can be rewritten as

$$\min_{\mathbf{S}_i^T \mathbf{1}_N = 1, \mathbf{S}_i \ge 0} \left\| \mathbf{S}_i + \frac{1}{2\gamma_i} \mathbf{E}_i \right\|_2^2$$
(21)

According to the solving method proposed in [52], the closed form solution of S_i can be obtained by

$$S_{(j),i} = \begin{cases} \frac{E_{(k_i+1),i}-E_{(j),i}}{k_i E_{(k_i+1),i}-\sum_{h=1}^{k_i} E_{(h),i}}, & j \le k_i \\ 0, & j > k_i \end{cases}$$
(22)

when we set γ_i as

$$\gamma_i = \frac{k_i}{2} E_{(k_i+1),i} - \frac{1}{2} \sum_{h=1}^{k_i} E_{(h),i}$$
(23)

where the (*j*) of $E_{(h),i}$ represents the position of the *j*-th smallest value in **E**_{*i*}.

k-selection: from (22) and (23), we can find that the learned graph S is a KNN-type graph, and the parameter α_i controls the number of NNs, which is very important in the graph. Obviously, a very small k_i will under-connect the vertices and make the graph not robust enough, while a very large k_i will over-connect the vertices and make the graph confusing. Therefore, we use an in-degree-based adaptive *k*-selection strategy proposed in the IRG-McS [23], which aims to connect each superpixel with as many truly similar superpixels as possible.

w-subproblem. According to the self-weighting strategy, the w_m can be updated by using (8), which can be rewritten as

$$w_m = \frac{1}{2\sqrt{Tr(\mathbf{S}^T \mathbf{D}_x^m) + \beta g(\varepsilon^m)}}$$
(24)

Multipliers: finally, the two Lagrangian multipliers can be updated as

$$\mathbf{R}_{1} \leftarrow \mathbf{R}_{1} + \mu_{1}(\mathbf{S} - \mathbf{P}) \mathbf{R}_{2}^{m} \leftarrow \mathbf{R}_{2}^{m} + \mu_{2}(\mathbf{X}^{m} - \mathbf{X}^{m}\mathbf{P} - \varepsilon^{m})$$
(25)

The procedure of the proposed auto-weighted structured graph learning (5) is summarized in Algorithm 1. The algorithm terminates when the iteration reaches the maximum number of N_{iter} or the relative difference between two adjacent iterations is less than the given threshold.

Algorithm 1: AWSG learning.
Input: The feature matrix X , parameter $\beta > 0$.
Initialize: Set P , R ₁ , and R ₂ ^{<i>m</i>} = 0, and adaptively select k_i .
Repeat:
1: Update ε^m through (14) according to different $g(\cdot)$.
2: Update P through (19).
3: Update S through (22).
4: Update w_m through (24).
5: Update \mathbf{R}_1 and \mathbf{R}_2^m through (25).
Until stopping criterion is met.
Output: The learned graph S.

3. Structure Regression

Once we obtain the structured graph **S** that represents the structure information of the pre-event image, we can use it to transform the pre-event image to the domain of post-event image based on structure consistency between heterogeneous images. We define the translation function as \mathcal{M} and the translated image as $\tilde{\mathbf{Z}} \in \mathbb{R}^{I \times J \times C_y}$, that is, $\tilde{\mathbf{Z}} = \mathcal{M}(\tilde{\mathbf{X}})$, and define the feature matrix of translated image as $\tilde{\mathbf{Z}}$ with $\mathbf{Z}^m = \mathcal{F}^{(m)}(\tilde{\mathbf{Z}})$.

3.1. Structure Consistency-Based Regularization

The structure consistency between a pre-event image $\tilde{\mathbf{X}}$ and its translated image $\tilde{\mathbf{Z}}$ constrains that they should share the same similarity relationships. This consists of two aspects: first, if $\tilde{\mathbf{X}}_i$ and $\tilde{\mathbf{X}}_j$ represent the same type of objects (they are very similar), then $\tilde{\mathbf{Z}}_i$ and $\tilde{\mathbf{Z}}_j$ also represent the same type of objects (they are also very similar); second, if $\tilde{\mathbf{X}}_i$ can be reconstructed by its NNs with \mathbf{S}_i , i.e., $\mathbf{X}_i^m = \mathbf{X}^m \mathbf{S}_i$, then $\tilde{\mathbf{X}}_i$ should also be represented by its NNs with $\mathbf{Z}_i^m = \mathbf{Z}^m \mathbf{S}_i$. Combining these two aspects, we have the following local similarity-based regularization and global self-expression-based regularization as

$$\sum_{m=1}^{M} \omega_m \left(\sum_{i,j=1'}^{N} \left\| \mathbf{Z}_i^m - \mathbf{Z}_j^m \right\|_2^2 S_{i,j} + \beta g(\mathbf{Z}^m - \mathbf{Z}^m \mathbf{S}) \right)$$
(26)

where ω_m represents the *m*-th feature weights similar to those in (9) that can be autodetermined by using the self-conducted weight learning. We define the degree matrix $\mathbf{D}_s \in \mathbb{R}^{N \times N}$ as a diagonal matrix with the *i*-th diagonal element being $\sum_{j=1}^{N} (S_{i,j} + S_{j,i})/2$, and the Laplacian matrix as $\mathbf{L} = \mathbf{D}_s - \frac{\mathbf{S} + \mathbf{S}^T}{2}$, then the structure consistency-based regularization can be rewritten as

$$\sum_{m=1}^{M} \omega_m \left(2Tr \left(\mathbf{Z}^m \mathbf{L} (\mathbf{Z}^m)^T \right) + \beta g (\mathbf{Z}^m - \mathbf{Z}^m \mathbf{S}) \right)$$
(27)

3.2. Structure Regression Model

We can decompose the post-event image $\tilde{\mathbf{Y}}$ into a translated image $\tilde{\mathbf{Z}}$ and a change image $\tilde{\mathbf{\Delta}}$. Then, we have $\mathbf{Y} = \mathbf{Z} - \mathbf{\Delta}$, where $\mathbf{\Delta}$ is the changed feature matrix (note that $\mathbf{\Delta}$ is not the feature matrix of $\tilde{\mathbf{\Delta}}$ because the feature extraction operator may be not a linear operator). Based on the fact that changed region is always only a minority in the practical CD problem, while most of the region is unchanged, we have a change sparsity-based penalty as $\sum_{m=1}^{M} \|\mathbf{\Delta}^{m}\|_{2,1}$. By combining the structure consistency-based regularization (27) and the change sparsity-based penalty, we obtain the structure regression model as

$$\min_{\mathbf{Z}, \boldsymbol{\Delta}, \boldsymbol{\omega}} \sum_{m=1}^{M} \omega_m \left(2Tr \left(\mathbf{Z}^m \mathbf{L} (\mathbf{Z}^m)^T \right) + \beta g(\varepsilon^m) \right) + \lambda \| \boldsymbol{\Delta}^m \|_{2,1}$$

$$s.t. \, \mathbf{Y}^m = \mathbf{Z}^m - \boldsymbol{\Delta}^m, \, \mathbf{Z}^m = \mathbf{Z}^m \mathbf{S} + \varepsilon^m$$
(28)

where $\lambda > 0$ is a penalty parameter.

In the regression model (28), we only use the first-order neighborhood information of the graph **S**. However, as demonstrated in HCD-GSP [41,42], the high-order neighbor information is also very important for the heterogeneous CD. In order to incorporate the hidden high-order information of the graph **S** into the structure regression model (28) and use the self-weighting strategy, we propose a new structure regression model as

$$\min_{\mathbf{Z}, \Delta} \sum_{m=1}^{M} \sqrt{2Tr(\mathbf{Z}^{m} \mathcal{H}(\mathbf{L})(\mathbf{Z}^{m})^{T}) + \beta g(\varepsilon^{m})} + \lambda \|\Delta^{m}\|_{2,1}$$

$$s.t. \mathbf{Y}^{m} = \mathbf{Z}^{m} - \Delta^{m}, \ K\mathbf{Z}^{m} = \mathbf{Z}^{m} \mathcal{H}(\mathbf{S}) + \varepsilon^{m}$$
(29)

where $\mathcal{H}(\mathbf{L}) = \sum_{k=1}^{K} \mathbf{L}^{(k)}$ and $\mathcal{H}(\mathbf{S}) = \sum_{k=1}^{K} \mathbf{S}^{(k)}$ denoting the sum of 1 to *K*-th power polynomial of **L** and **S**, respectively.

3.3. Optimization

By using the ADMM and using the iteratively re-weighted technique similar as (5), the model (29) can be rewritten as

$$\mathcal{L}(\varepsilon, \mathbf{Z}, \mathbf{\Delta}) = \sum_{m=1}^{M} \omega_m \Big(2Tr \Big(\mathbf{Z}^m \mathcal{H}(\mathbf{L}) (\mathbf{Z}^m)^T \Big) + \beta g(\varepsilon^m) \Big) \\ + \lambda \|\mathbf{\Delta}^m\|_{2,1} + Tr \Big((\mathbf{Z}^m - \mathbf{Y}^m - \mathbf{\Delta}^m)^T \mathbf{R}_1^m \Big) + \frac{\mu_1}{2} \|\mathbf{Z}^m - \mathbf{Y}^m - \mathbf{\Delta}^m\|_F^2$$
(30)
$$+ Tr \Big((K\mathbf{Z}^m - \mathbf{Z}^m \mathcal{H}(\mathbf{S}) - \varepsilon^m)^T \mathbf{R}_2^m \Big) + \frac{\mu_2}{2} \|K\mathbf{Z}^m - \mathbf{Z}^m \mathcal{H}(\mathbf{S}) - \varepsilon^m\|_F^2$$

where $\mathbf{R}_1^m, \mathbf{R}_2^m \in \mathbb{R}^{C_y \times N}$, $m = 1, \dots, M$ are Lagrangian multipliers, and $\mu_1, \mu_2 > 0$ are two penalty parameters. Then, we use the ADM to separate the minimization of (30) into ε -subproblem, **Z**-subproblem, **\Delta**-subproblem and ω -subproblem, which is similar to the procedure of solving problem (5) in Algorithm 1. The updating scheme for solving the auto-weighted structure regression model (29) with a high-order structured graph is given by

$$\varepsilon^{m} = prox_{\frac{\beta\omega_{m}}{\mu_{2}}g} \left(K \mathbf{Z}^{m} - \mathbf{Z}^{m} \mathcal{H}(\mathbf{S}) + \frac{\mathbf{R}_{2}^{m}}{\mu_{2}} \right)$$
(31a)

$$\boldsymbol{\Delta}^{m} = prox_{\frac{\lambda}{\mu_{1}} \parallel \cdot \parallel_{2,1}} \left(\mathbf{Z}^{m} - \mathbf{Y} + \frac{\mathbf{R}_{1}^{m}}{\mu_{1}} \right)$$
(31b)

$$\mathbf{Z}^m = \mathbf{\Theta}^m \mathbf{\Psi}^{-1} \tag{31c}$$

$$\omega_m = \frac{1}{2\sqrt{2Tr\left(\mathbf{Z}^m \mathcal{H}(\mathbf{L})(\mathbf{Z}^m)^T\right) + \beta g(\varepsilon^m)}}$$
(31d)

$$\mathbf{R}_{1}^{m} = \mathbf{R}_{1}^{m} + \mu_{1}(\mathbf{Z}^{m} - \mathbf{Y} - \boldsymbol{\Delta}^{m})$$
(31e)

$$\mathbf{R}_{2}^{m} = \mathbf{R}_{2}^{m} + \mu_{2}(K\mathbf{Z}^{m} - \mathbf{Z}^{m}\mathcal{H}(\mathbf{S}) - \varepsilon^{m})$$
(31f)

where the proximal operator in the updating of (31a) and (31b) with different closed-form solutions are given in (15)–(17). The matrices of Θ^m and Ψ in (31c) are defined as

$$\boldsymbol{\Theta}^{m} = (\mu_{2}\varepsilon^{m} - \mathbf{R}_{2}^{m})(K\mathbf{I}_{N} - \mathcal{H}(\mathbf{S}))^{T} + \mu_{1}(\mathbf{Y} + \boldsymbol{\Delta}^{m}) - \mathbf{R}_{1}^{m}$$
(32a)

$$\Psi = \mu_1 \mathbf{I}_N + \mu_2 (K \mathbf{I}_N - \mathcal{H}(\mathbf{S})) (K \mathbf{I}_N - \mathcal{H}(\mathbf{S}))^T + 4\omega_m \mathcal{H}(\mathbf{L})$$
(32b)

The procedure of the proposed structure regression model (29) is summarized in Algorithm 2. The algorithm terminates when the iteration reaches the maximum number of N_{iter} or the relative difference between two adjacent iterations is less than the given threshold.

Algorithm 2: Structure regression.
Input: The matrices of Y , S and L , parameters β , $\lambda > 0$.
Initialize: Set $\mathbf{Z} = \mathbf{Y}$, and $\mathbf{R}_1^m = \mathbf{R}_2^m = 0$.
Repeat:
1: Update ε^m through (31a) according to different $g(\cdot)$.
2: Update Δ^m through (31b).
3: Update \mathbb{Z}^m through (31c).
4: Update ω_m through (31d).
5: Update \mathbf{R}_1^m and \mathbf{R}_2^m through (31e) and (31f).
Until stopping criterion is met.
Output: The feature matrices of Z and Δ .

3.4. Change Extraction

After obtaining the regression feature matrix \mathbf{Z} and the changed feature matrix Δ by solving the regression model with Algorithm 2, we can compute the translated image $\tilde{\mathbf{Z}}$ by using $\tilde{\mathbf{Z}} = \mathcal{F}^{-1}(\mathbf{Z})$ and the difference image as

$$DI(i,j) = \sum_{m=1}^{M} \|\mathbf{\Delta}_{n}^{m}\|_{2}^{2}, \quad (i,j) \in \Lambda_{n}, \ n = 1, \cdots, N$$
(33)

Then, we can divided this DI into changed and unchanged classes to object the binary CM, which can be regarded as a binary segmentation problem. Therefore, the thresholding methods such as Otsu threshold [53], or clustering methods such as the K-means [54], or the random field-based methods such as MRF [10,23] can be selected to complete the image segmentation. In this paper, we directly choose the MRF-based segmentation method proposed in SCASC [10], which can incorporate the change information and spatial information in the energy. There is only one hyperparameter to be adjusted in the segmentation algorithm that is the balancing parameter, which we fix to 0.5 according to the SCASC [10].

4. Experimental Results and Discussions

In this section, we experimentally test the performance of the proposed AWSG-based structure regression method for HCD (named AWSG for short), which is achieved by comparing it with some existing state-of-the-art (SOTA) algorithms on four different datasets.

4.1. Datasets and Evaluation Metrics

Four real heterogeneous datasets are used to evaluate the proposed AWSG as listed in Table 1, all these heterogeneous image pairs have already been registered. These datasets contain different kinds of heterogeneity: (1) The cross-sensor cases, e.g., the image pair consisting of the near-infrared (NIR) band image from Landsat-5 and the RGB bands optical image from Google Earth in Dataset #1, and the image pair consisting of the optical images from Pleiades and WorldView2 in Dataset #2; (2) The multisource cases, e.g., the image pair consisting of the SAR image from Radarsat-2 and the optical image from Google Earth in Dataset #3, and image pair consisting of the multispectral image from Landsat-8 and the SAR image from Sentinel-1A. These datasets also cover different spatial resolutions, different image sizes, and different change events, which can evaluate the robustness of the propose algorithm, as shown in Figure 2.

Two types of metrics are utilized to evaluate change detection results. First, the quality of DI is evaluated by the precision-recall (PR) curve and empirical receiver operating characteristics (ROC) curve, which are plotted with the precision rate vs. recall rate and the true positive rate (TPR) vs. false positive rate (FPR), respectively. The precision rate, recall rate (also is the TPR), and FPR are calculated by Precision = TP/(TP + FP), Recall = TP/(TP + FN), and FPR = FP/(TN + FP) respectively, with the true positives, false positives, true negatives, and false negatives abbreviated as TP, FP, TN, and FN, respectively. Second, the quality of binary CM is evaluated by the overall accuracy (OA), the Kappa coefficient (κ) and F1-score (F1), which are computed as

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

$$\kappa = \frac{2 \times (TP \times TN - FN \times FP)}{(TP + FP) \times (FP + TN) + (TP + FN) \times (FN + TN)}$$

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN}$$
(34)

For all the experiments of AWSG, we choose the global self-expression penalty term as $g(\varepsilon^m) = \|\varepsilon^m\|_F^2$ for Algorithms 1 and 2; we adjust the scale parameter of FNEA to make the number of segmented superpixels around 5000, i.e., $N \approx 5000$, and extract the features of mean, median, and variance values for each band of the superpixel; we fix $\beta = 1$,



 $\mu_1 = \mu_2 = 0.4$ for Algorithms 1 and 2; we set $\lambda = 0.1$ and fix K = 3 (according to the HCD-GSP [42]) for Algorithm 2.

Figure 2. Heterogeneous datasets, regression images and DIs. From top to bottom, they correspond to Datasets #1 to #4, respectively. From left to right: (**a**) pre-event image \tilde{X} ; (**b**) post-event image \tilde{Y} ; (**c**) the ground truth; (**d**) the regression images \tilde{Z} ; (**e**) the DI calculated by using (33) with changed feature matrix Δ .

Dataset	Location	Event (& Spatial Resolution)	Sensing Date	Collection Sensor	Image Size (Pixels)
#1	Sardinia, Italy	Lake expansion (30 m)	Sept. 1995 July 1996	Landsat-5 Google Earth	$\begin{array}{c} 300\times412\times1\\ 300\times412\times3 \end{array}$
#2	Toulouse, France	Construction (0.52 m)	May 2012 July 2013	Pleiades WorldView2	$\begin{array}{c} 2000\times2000\times3\\ 2000\times2000\times3 \end{array}$
#3	Shuguang Village, China	Building construction (8 m)	June 2008 Sept. 2012	Radarsat-2 Google Earth	$\begin{array}{c} 593 \times 921 \times 1 \\ 593 \times 921 \times 3 \end{array}$
#4	Sutter County, CA, USA	Flooding ($\approx 15 \text{ m}$)	Jan. 2017 Feb. 2017	Landsat-8 Sentinel-1A	$\begin{array}{c} 875\times500\times11\\ 875\times500\times3 \end{array}$

Table 1. Description of the four heterogeneous data sets.

4.2. Regression Images and Difference Images

In the first experiment, we demonstrate the effectiveness of the AWSG-based regression model by displaying the regression image \tilde{Z} (using the mean feature of Z as the value of each pixel within the superpixel) and the difference image computed by (33). In Figure 2d, we show the regression images generated by the AWSG on different heterogeneous datasets. We can find that the structures of the regression images \tilde{Z} are consistent with the pre-event images \tilde{X} in both the changed and unchanged areas. At the same time, it can also be found that the regression images \tilde{Z} have the same image style as the post-event images \tilde{Y} . These two aspects demonstrate that the AWSG-based regression model can well translate the pre-event image to the domain of post-event image.

In Figure 2e, we show the DI generated by AWSG on Datasets #1 to #4. There are two observations that can be visualized: first, the DIs are able to distinguish the unchanged and changed areas by giving a larger value to the changed superpixels, which shows the effectiveness of the AWSG-based structure representation of Algorithm 1 and the highorder-based structure regression with the learned AWSG of Algorithm 2; second, the DIs are sparse with the change sparsity-based $\sum_{m=1}^{M} \| \mathbf{\Delta}^{m} \|_{2,1}$ in the regression model (29), which means that satisfactory segmentation maps can be obtained even by the simple thresholding methods or clustering methods. Figure 3 plots the ROC and PR curves of the difference images, the areas under the ROC curves (AUR) of Datasets #1 to #4 are 0.938, 0.828, 0.954, and 0.878, respectively, and the the areas under the PR curves (AUP) of Datasets #1 to #4 are 0.656, 0.586, 0.735, and 0.456, respectively. From the curves in Figure 3 and the values of AUR and AUP, we can find that the DIs are able to distinguish well between changed and unchanged areas, by simply using the commonly used classification methods, such as thresholding methods (e.g., the Otsu threshold [53]), and clustering methods (e.g., the Kmeans [54] and fuzzy c-means [55]). For example, if we plot a line with Precision = Recall on the PR curves of Figure 3b and calculate the intersection of this diagonal line with the PR curve of each dataset, we can obtain the horizontal coordinates of the points with Precision = Recall on the PR curves: 0.689, 0.563, 0.661, 0.519 corresponding to Datasets #1 to #4, respectively. These are also the F1 values (F1 = Precision when Precision = Recall) that can be obtained by a simple threshold segmentation on the DIs. From the DIs of Figure 2 and the ROC, PR curves of Figure 3, we can find that the proposed method is able to effectively identify changes from heterogenous image pairs.



Figure 3. ROC (a) and PR (b) curves of AWSG generated DIs.

4.3. Change Maps

In order to evaluate the effectiveness of the AWSG based HCD algorithm, we select seven SOTA methods (including two deep-learning-based methods) for comparison, where we use the default parameters in the codes that are also consistent with their original papers.

M3CD (M3CD is available at http://www-labs.iro.umontreal.ca/~mignotte (accessed on 9 August 2022)) [16]. An MRF-based method that builds up the observation field from a pixel pairwise modeling and solves the energy minimization problem by using iterative conditional estimation.

FPMS (FPMS is available at http://www-labs.iro.umontreal.ca/~mignotte (accessed on 9 August 2022)) [15]. A fractal projection and Markovian segmentation-based method that translates the image by using a spatial fractal decomposition and a contractive projection.

CICM (CICM is available at http://www-labs.iro.umontreal.ca/~mignotte (accessed on 9 August 2022)) [56]. It is a circular invariant convolution model-based method that aims at transforming images by using a concentric circular invariant convolution representation and a multiscale Markov segmentation model.

SCASC (SCASC is available at https://github.com/yulisun/SCASC (accessed on 9 August 2022)) [10]. It is a sparse-constrained adaptive structure consistency-based method that translates the images by using the structure consistency with a local similarity based graph.

AGSCC (AGSCC is available at https://github.com/yulisun/AGSCC (accessed on 9 August 2022)) [9]. It is an adaptive graph with a structure cycle consistency-based method that translates the images by combining the forward transformation and cycle transformation.

SCCN (SCCN is implemented at https://github.com/llu025 (accessed on 9 August 2022)) [22]. It is a deep-learning-based method that uses a symmetric convolutional coupling network (SCCN) to transform the heterogeneous images into a common feature space.

ACE-Net (ACE-Net is available at https://github.com/llu025 (accessed on 9 August 2022)) [21]. It is a deep image translation method that uses an adversarial cyclic encoder network consisting of two auto-encoders and an affinity-based change prior for network training.

Figure 4 shows the CMs of all the methods on the Datasets #1 to #4, and Table 2 lists the quantitative measures of these CMs. We find that most methods can roughly reflect the change information, whereas some methods do not perform robustly enough on some datasets. To be specific, in Dataset #1, most methods can well identify the changes, but the false negatives in CICM and SCASC are more than for other methods, and the false positives in the SCCN and FPMS are more than others, which leads to a lower F1 and κ of CICM, SCCN, FPMS, and SCASC in Table 2. In Dataset #2, the resolution of the

multitemporal images is very high (0.52 m) and the proportion of the changed region is also relatively larger than in other datasets, which bring difficulties in accurately detecting changes. From Figure 4 and Table 2, we find that most methods can only detect a small part of the changes, such as FPMS ($\kappa = 0.215$) and CICM ($\kappa = 0.270$), however, the proposed AWSG can obtain a more accurate CM with better visual effects and highest evaluation metrics ($\kappa = 0.524$). In Dataset #3, there are a large number of false alarms in the SCCN, resulting in a poorer detection performance (F1 = 0.386) than other methods. Additionally, the proposed AWSG is able to obtain good detection results on this dataset, scoring second only to the ACE-Net. In the last Dataset #4, there are more types of objects in the pre- and post-event images than in the other datasets (such as buildings, farmland, rivers, forests, roads, mountains) and these objects are very unevenly proportioned, as shown in Figure 2, which poses difficulties in capturing image structure and completing image regression. Nevertheless, the proposed method is still able to achieve optimal detection results and gain the highest evaluation metrics (F1 = 0.547), which further demonstrates the structural expressiveness of AWSG and the importance of higher-order structure information in the image regression model. The average OA, κ and F1 of the proposed AWSG on these four datasets are about 0.951, 0.628, and 0.656, respectively, which shows that the proposed AWSG is very competitive, even compared with the deep-learning-based methods.



Figure 4. Binary CMs of different methods on heterogeneous datasets. From top to bottom, they correspond to Datasets #1 to #4, respectively. From left to right are binary CMs generated by: (a) M3CD; (b) FPMS; (c) CICM; (d) SCASC; (e) AGSCC; (f) SCCN; (g) ACE-Net; (h) AWSG. In the binary CM, White: true positives (TP); Red: false positives (FP); Black: true negatives (TN); Green: false negatives (FN).

Dataset #1		Dataset #2		Dataset #3		Dataset #4			Average						
Methous	OA	κ	F1	OA	κ	F1	OA	κ	F1	OA	κ	F1	OA	κ	F1
M3CD	0.963	0.669	0.689	0.863	0.405	0.481	0.962	0.602	0.622	0.575	0.021	0.077	0.841	0.424	0.467
FPMS	0.925	0.552	0.588	0.838	0.215	0.296	0.938	0.569	0.597	0.947	0.329	0.356	0.912	0.416	0.459
CICM	0.943	0.451	0.481	0.867	0.270	0.321	0.974	0.745	0.759	0.899	0.081	0.131	0.921	0.387	0.423
SCASC	0.947	0.593	0.621	0.892	0.464	0.516	0.979	0.741	0.751	0.961	0.479	0.500	0.945	0.569	0.597
AGSCC	0.959	0.658	0.680	0.897	0.490	0.540	0.983	0.773	0.782	0.960	0.486	0.506	0.950	0.602	0.627
SCCN	0.900	0.478	0.524	0.848	0.417	0.507	0.908	0.344	0.386	0.906	0.392	0.431	0.891	0.408	0.462
ACE-Net	0.964	0.718	0.737	0.886	0.480	0.541	0.981	0.788	0.798	0.940	0.468	0.497	0.943	0.614	0.643
AWSG	0.963	0.690	0.711	0.896	0.524	0.581	0.984	0.775	0.783	0.962	0.524	0.547	0.951	0.628	0.656

Table 2. Quantitative measures of binary CMs on the heterogeneous datasets. Note that SCCN and ACE-Net are deep-learning-based methods.

4.4. Discussion

4.4.1. Ablation Study

The proposed AWSG mainly contains two processes: the structure representation by learning an auto-weighted structured graph and the structure regression by using highorder neighbor information of the graph. To further investigate the contributions of the proposed method in the heterogeneous CD, that is, the roles of the auto-weighted strategy, the structured graph combining local and global information, the adaptive k-selection strategy, and the high-order structure regression, an ablation experiment is carried out, i.e., eliminating the corresponding parts on the basis of the complete AWSG for comparison. The detection performances (F1 measure of CM) of the AWSG without (w/o) auto-weighted strategy (AW), local structure information (LS), global structure information (GS), adaptive k-selection (AK), high-order neighbor information (HN) are reported in Table 3. In addition, we also show the DI and CM generated by AWSG without these parts on Dataset #1 in Figure 5.



(c) w/o GS

(d) w/o AK

(f) AWSG

Figure 5. Ablation study of AWSG. From top to bottom, they correspond to the DI and CM generated by different methods on the Dataset #1. From left to right are the results generated by: (a) AWSG without auto-weighted strategy; (b) AWSG without local structure information; (c) AWSG without global structure information; (d) AWSG without adaptive k-selection; (e) AWSG without high-order neighbor information; (f) the original AWSG.

According to Table 3, the detection performance degrades without (w/o) AW, LS, GS, AK, and HN on all the evaluated datasets. Specifically, compared to the other parts, the effect of adaptive k-selection structure is slightly weaker, which has a more pronounced effect on datasets with uneven feature distribution, e.g., the F1 score is improved by 2.6% in Dataset #4. When applied with the auto-weighted strategy, the constructed graph can adaptively assign different weights for different features by using a self-conducted weight learning, which makes the graph more robust. Therefore, the score of F1 increases by about 4.6% on Dataset #1 and 3.2% on Dataset #2 compared to that without AW. By using the

structured graph that combines the local similarity structure and global self-expression property, the average score of the F1 is about 2.0% higher than that of using only local structural information and 2.3% higher than that of using only global structural information as reported by Table 3. It can also be seen in Figure 5 that the CM of AWSG combining global and local structure information has fewer false alarms and missed detections than the AWSG w/o LS or GS, which demonstrates the effectiveness of structured graph that can improve the ability of characterizing structures. By exploiting the high-order neighbor information ($\mathcal{H}(L)$ and $\mathcal{H}(S)$) in the structure regression model (29), the final detection results is more accurate, as shown in Figure 5e,f, and the average F1 can be improved by about 2.1% compared to that which only uses the first-order neighborhood information, as reported in Table 3.

Mathada		A			
Methods	#1	#2	#3	#4	Average
AWSG w/o AW	0.665	0.549	0.751	0.513	0.620
AWSG w/o LS	0.682	0.564	0.761	0.525	0.633
AWSG w/o GS	0.695	0.558	0.752	0.540	0.636
AWSG w/o AK	0.703	0.570	0.772	0.521	0.642
AWSG w/o HN	0.677	0.565	0.760	0.539	0.635
AWSG	0.711	0.581	0.783	0.547	0.656

Table 3. Ablation study of AWSG measured by the F1 score of CM.

4.4.2. Parameter Analysis

The main parameters of AWSG are the superpixel number *N*, maximum iterations N_{iter} , the balancing parameter β , and the sparse penalty parameter λ .

Generally, the number N of superpixels should be determined based on the resolution of the dataset and should also take into account the CD task's requirements on computational efficiency. A large N increases the computational burden while improving the detection granularity.

To investigate the convergence performance of the proposed AWSG with parameter N_{iter} , we plot the convergence curves of Algorithms 1 and 2, which are measured by the relative difference (Rel.diff) ξ^k versus iteration number k, as shown in Figure 6, where $\xi^{[k+1]} = \sum_{m=1}^{M} \frac{\|\mathbf{S}^{[k+1]} - \mathbf{S}^{[k]}\|_{F}}{\|\mathbf{S}^{[k+1]}\|_{F}}$ for the Algorithm 1 and $\xi^{[k+1]} = \sum_{m=1}^{M} \frac{\|\mathbf{\Delta}^{m[k+1]} - \mathbf{\Delta}^{m[k]}\|_{F}}{\|\mathbf{\Delta}^{m[k+1]}\|_{F}}$ for the Algorithm 2. It is clear that the ADMM in the solving of Algorithms 1 and 2 converges quickly, and the exit criterion of $N_{iter} = 15$ or $\xi^{0} = 0.01$ is enough for the algorithms when taking into account the computational time.

The parameter β balances the weight of local structure penalty term and global selfexpression penalty term in the graph learning model (5) and structure regression model (29).

Given the penalty function of $g(\varepsilon^m) = \|\varepsilon^m\|_F^2$, we have $\sum_{i=1}^N \sum_{j=1}^N \|\mathbf{X}_i^m - \mathbf{X}_j^m\|_2^2 S_{i,j} \ge \|\mathbf{X}_i^m - \mathbf{X}_j^m \mathbf{S}\|_F^2$ by using the condition of $\sum_{i=1}^N S_{i,j} = 1$. Therefore, it is recommended that $\beta \ge 1$.

For the sparse penalty parameter λ , it is used to control the sparsity level of DI, which should be determined based on the percentage of real change in the ground truth. However, we experimentally find that the proposed algorithm is not sensitive to this parameter, as shown in Figure 7, where we show the AUR at different values of λ (from 10^{-5} to 1 with the ratio of 10). It can be found that the AWSG performs well in a broad range of λ .



Figure 6. Convergence performance of AWSG.



Figure 7. Influence of parameter λ on the proposed AWSG.

4.4.3. Complexity Analysis

The main complexity of the proposed method is concentrated in the AWSG learning (Algorithm 1) and the structure regression with AWSG (Algorithm 2).

Algorithm 1: Step 1, update ε^m , requires $\mathcal{O}(C_x N^2)$ for calculating \mathbf{Q}^m and $\mathcal{O}(C_x N)$ for the proximal operator of (14); Step 2, update **P**, requires $\mathcal{O}(N^3)$ for the matrix inversion of $(\mathbf{I}_N + \frac{\mu_1}{\mu_2} \sum_{m=1}^M (\mathbf{X}^m)^T \mathbf{X}^m)^{-1}$ and $\mathcal{O}(C_x N^2)$ for the matrix multiplication; Step 3, update **S**, requires $\mathcal{O}(C_x N^2)$ for calculating **E**, $\mathcal{O}(N^2 \log N)$ for sorting all columns of **E**, and $\mathcal{O}(N^2)$ for the closed-form solution of (22); Step 4, update w_m , requires $\mathcal{O}(N^2)$ by using (8) or (24); Step 5, update the Lagrangian multipliers of \mathbf{R}_1 and \mathbf{R}_2^m , requires $\mathcal{O}(C_x N^2)$ for matrix multiplication.

Algorithm 2: Step 1, update ε^m , requires $\mathcal{O}(C_y N^2)$ for the matrix multiplication and $\mathcal{O}(C_y N)$ for the proximal operator of (31a); Step 2, update Δ^m , requires $\mathcal{O}(C_y N)$ by using (31b); Step 3, update \mathbb{Z}^m , requires for $\mathcal{O}(C_y N^2)$ calculating Θ^m , $\mathcal{O}(N^3)$ for the matrix inversion of Ψ^{-1} , and $\mathcal{O}(C_y N^2)$ for the matrix multiplication; Step 4, update w_m , requires $\mathcal{O}(N^2)$ by using (31d); Step 5, update the Lagrangian multipliers of \mathbb{R}_1^m and \mathbb{R}_2^m , requires $\mathcal{O}(C_y N^2)$ for the matrix multiplication.

Although the complexity of AWSG is very high in the above theoretical analysis, which requires $O(N^3)$ for the matrix inversion of each iteration in Algorithms 1 and 2, it can be accelerated by using some iterative solvers for updating **P** with (19) in Algorithm 1 and **Z**^{*m*} with (31c) in Algorithm 2. The linear systems of updating **P** and **Z**^{*m*} can be solved efficiently by using the conjugate gradient (CG) method. In addition, some pre-conditioners can also

be used to accelerate CG method [57]. In our experiments, we use the CG with incomplete Cholesky pre-conditioner for updating **P** and \mathbf{Z}^m . The computational time of each process of AWSG with different *N* on Datasets #2 and #3 is listed in Table 4, where the AWSG is performed in MATLAB 2016a running with Intel Core i9-10980HK CPU.

Table 4. Computational time (s) of AWSG, where t_{A1} , t_{A2} , and t_{total} denote the computational time of Algorithms 1 and 2 and the whole AWSG, respectively.

Datasets	N	t_{A1}	t_{A2}	t _{total}
Dataset #2 2000 \times 2000 \times 3(3)	2500 5000 10,000	3.8 12.1 69.8	2.0 5.9 28.7	9.1 29.3 127.5
Dataset #3 593 \times 921 \times 1(3)	2500 5000 10,000	3.5 10.4 65.9	1.8 5.1 26.6	8.4 24.2 120.4

5. Conclusions

In this paper, we proposed an unsupervised image regression method for heterogeneous change detection. Specifically, the proposed method first segments the multitemporal images into superpixels, and learns an adaptive KNN graph for the pre-event image to capture the structure information, which uses a self-conducted weighting strategy and combines the local and global structure information to make the graph more capable of structural characterization. Then, based on the structure consistency between heterogeneous images, the proposed method transforms the pre-event image to the domain of post-event image by using the learned graph. By exploiting the high-order neighbor information hidden in the graph, it can obtain a better translated image and change image and thus improve the accuracy of change detection. The experimental results confirm the effectiveness of the proposed method by comparing with other related methods. In future work, we will focus on the distributional properties of the image, especially ways to improve the structure representation of the graph in the case of uneven object distribution.

Author Contributions: Methodology, L.Z.; software, L.Z. and Y.S.; validation, L.Z., L.L. and S.Z. Original draft preparation, L.Z.; writing—review and editing, Y.S.; supervision, L.L. and S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Natural Science Foundation of Hunan Province, China of grant number 2021JJ30780.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

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

References

- Singh, A. Review Article Digital change detection techniques using remotely-sensed data. *Int. J. Remote Sens.* 1989, 10, 989–1003. [CrossRef]
- Liu, S.; Marinelli, D.; Bruzzone, L.; Bovolo, F. A Review of Change Detection in Multitemporal Hyperspectral Images: Current Techniques, Applications, and Challenges. *IEEE Geosci. Remote Sens. Mag.* 2019, 7, 140–158. [CrossRef]
- Lv, Z.; Liu, T.; Benediktsson, J.A.; Falco, N. Land Cover Change Detection Techniques: Very-High-Resolution Optical Images: A Review. IEEE Geosci. Remote Sens. Mag. 2021, 10, 2–21. [CrossRef]
- Wen, D.; Huang, X.; Bovolo, F.; Li, J.; Ke, X.; Zhang, A.; Benediktsson, J.A. Change Detection From Very-High-Spatial-Resolution Optical Remote Sensing Images: Methods, applications, and future directions. *IEEE Geosci. Remote Sens. Mag.* 2021, 9, 68–101. [CrossRef]
- 5. Moser, G.; Serpico, S. Generalized minimum-error thresholding for unsupervised change detection from SAR amplitude imagery. *IEEE Trans. Geosci. Remote Sens.* **2006**, *44*, 2972–2982. [CrossRef]
- Lv, Z.; Liu, T.; Shi, C.; Benediktsson, J.A. Local Histogram-Based Analysis for Detecting Land Cover Change Using VHR Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* 2021, 18, 1284–1287. [CrossRef]

- Lv, Z.; Wang, F.; Cui, G.; Benediktsson, J.A.; Lei, T.; Sun, W. Spatial-Spectral Attention Network Guided with Change Magnitude Image for Land Cover Change Detection Using Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* 2022, 60, 1. [CrossRef]
- 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.* 2019, 57, 3–13. [CrossRef]
- Sun, Y.; Lei, L.; Guan, D.; Wu, J.; Kuang, G.; Liu, L. Image Regression With Structure Cycle Consistency for Heterogeneous Change Detection. *IEEE Trans. Neural Netw. Learn. Syst.* 2022, 1–15. [CrossRef]
- Sun, Y.; Lei, L.; Guan, D.; Li, M.; Kuang, G. Sparse-Constrained Adaptive Structure Consistency-Based Unsupervised Image Regression for Heterogeneous Remote Sensing Change Detection. *IEEE Trans. Geosci. Remote Sens.* 2021, 60, 1–14. [CrossRef]
- 11. Deng, W.; Liao, Q.; Zhao, L.; Guo, D.; Kuang, G.; Hu, D.; Liu, L. Joint Clustering and Discriminative Feature Alignment for Unsupervised Domain Adaptation. *IEEE Trans. Image Process.* 2021, *30*, 7842–7855. [CrossRef] [PubMed]
- 12. Deng, W.; Zhao, L.; Kuang, G.; Hu, D.; Pietikäinen, M.; Liu, L. Deep Ladder-Suppression Network for Unsupervised Domain Adaptation. *IEEE Trans. Cybern.* 2021, 1–15. [CrossRef] [PubMed]
- Deng, W.; Cui, Y.; Liu, Z.; Kuang, G.; Hu, D.; Pietikäinen, M.; Liu, L. Informative Class-Conditioned Feature Alignment for Unsupervised Domain Adaptation. In Proceedings of the 29th ACM International Conference on Multimedia, Nice, France, 21–25 October 2021; pp. 1303–1312.
- 14. Luppino, L.T.; Bianchi, F.M.; Moser, G.; Anfinsen, S.N. Unsupervised Image Regression for Heterogeneous Change Detection. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 9960–9975. [CrossRef]
- Mignotte, M. A Fractal Projection and Markovian Segmentation-Based Approach for Multimodal Change Detection. *IEEE Trans. Geosci. Remote Sens.* 2020, 58, 8046–8058. [CrossRef]
- 16. Touati, R.; Mignotte, M.; Dahmane, M. Multimodal Change Detection in Remote Sensing Images Using an Unsupervised Pixel Pairwise-Based Markov Random Field Model. *IEEE Trans. Image Process.* **2020**, *29*, 757–767. [CrossRef]
- 17. Liu, Z.; Li, G.; Mercier, G.; He, Y.; Pan, Q. Change Detection in Heterogenous Remote Sensing Images via Homogeneous Pixel Transformation. *IEEE Trans. Image Process.* **2018**, *27*, 1822–1834. [CrossRef]
- Li, X.; Du, Z.; Huang, Y.; Tan, Z. A deep translation (GAN) based change detection network for optical and SAR remote sensing images. *ISPRS J. Photogramm. Remote Sens.* 2021, 179, 14–34. [CrossRef]
- Lv, Z.; Huang, H.; Gao, L.; Benediktsson, J.A.; Zhao, M.; Shi, C. Simple Multiscale UNet for Change Detection With Heterogeneous Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* 2022, 19, 1–5. [CrossRef]
- Lv, Z.; Li, G.; Jin, Z.; Benediktsson, J.A.; Foody, G.M. Iterative Training Sample Expansion to Increase and Balance the Accuracy of Land Classification From VHR Imagery. *IEEE Trans. Geosci. Remote Sens.* 2021, 59, 139–150. [CrossRef]
- Luppino, L.T.; Kampffmeyer, M.; Bianchi, F.M.; Moser, G.; Serpico, S.B.; Jenssen, R.; Anfinsen, S.N. Deep Image Translation With an Affinity-Based Change Prior for Unsupervised Multimodal Change Detection. *IEEE Trans. Geosci. Remote Sens.* 2022, 60, 1–22. [CrossRef]
- 22. 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.* **2018**, *29*, 545–559. [CrossRef] [PubMed]
- 23. Sun, Y.; Lei, L.; Guan, D.; Kuang, G. Iterative Robust Graph for Unsupervised Change Detection of Heterogeneous Remote Sensing Images. *IEEE Trans. Image Process.* 2021, *30*, 6277–6291. [CrossRef] [PubMed]
- 24. Touati, R.; Mignotte, M. An Energy-Based Model Encoding Nonlocal Pairwise Pixel Interactions for Multisensor Change Detection. *IEEE Trans. Geosci. Remote Sens.* 2018, 56, 1046–1058. [CrossRef]
- Gong, M.; Zhang, P.; Su, L.; Liu, J. Coupled Dictionary Learning for Change Detection From Multisource Data. *IEEE Trans. Geosci. Remote Sens.* 2016, 54, 7077–7091. [CrossRef]
- Mercier, G.; Moser, G.; Serpico, S.B. Conditional Copulas for Change Detection in Heterogeneous Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* 2008, 46, 1428–1441. [CrossRef]
- 27. Sun, Y.; Lei, L.; Li, X.; Tan, X.; Kuang, G. Patch Similarity Graph Matrix-Based Unsupervised Remote Sensing Change Detection With Homogeneous and Heterogeneous Sensors. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 4841–4861. [CrossRef]
- 28. Sun, Y.; Lei, L.; Li, X.; Tan, X.; Kuang, G. Structure Consistency-Based Graph for Unsupervised Change Detection With Homogeneous and Heterogeneous Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–21. [CrossRef]
- Prendes, J.; Chabert, M.; Pascal, F.; Giros, A.; Tourneret, J.Y. A New Multivariate Statistical Model for Change Detection in Images Acquired by Homogeneous and Heterogeneous Sensors. *IEEE Trans. Image Process.* 2015, 24, 799–812. [CrossRef]
- 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]
- 31. Sun, Y.; Lei, L.; Tan, X.; Guan, D.; Wu, J.; Kuang, G. Structured graph based image regression for unsupervised multimodal change detection. *ISPRS J. Photogramm. Remote Sens.* **2022**, *185*, 16–31. [CrossRef]
- 32. Jiang, X.; Li, G.; Liu, Y.; Zhang, X.P.; He, Y. Change Detection in Heterogeneous Optical and SAR Remote Sensing Images Via Deep Homogeneous Feature Fusion. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 1551–1566. [CrossRef]
- 33. Liu, Z.; Zhang, Z.; Pan, Q.; Ning, L. Unsupervised Change Detection From Heterogeneous Data Based on Image Translation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–13. [CrossRef]
- Wu, Y.; Li, J.; Yuan, Y.; Qin, A.K.; Miao, Q.G.; Gong, M.G. Commonality Autoencoder: Learning Common Features for Change Detection From Heterogeneous Images. *IEEE Trans. Neural Netw. Learn. Syst.* 2021, 33, 1–14. [CrossRef]

- 35. Saha, S.; Ebel, P.; Zhu, X.X. Self-Supervised Multisensor Change Detection. *IEEE Trans. Geosci. Remote Sens.* 2022, 60, 1–10. [CrossRef]
- Chen, Y.; Bruzzone, L. Self-Supervised Change Detection in Multiview Remote Sensing Images. *IEEE Trans. Geosci. Remote Sens.* 2022, 60, 1–12. [CrossRef]
- 37. Wu, J.; Li, B.; Qin, Y.; Ni, W.; Zhang, H.; Fu, R.; Sun, Y. A multiscale graph convolutional network for change detection in homogeneous and heterogeneous remote sensing images. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, 105, 102615. [CrossRef]
- Jimenez-Sierra, D.A.; Benítez-Restrepo, H.D.; Vargas-Cardona, H.D.; Chanussot, J. Graph-Based Data Fusion Applied to: Change Detection and Biomass Estimation in Rice Crops. *Remote Sens.* 2020, 12, 2683. [CrossRef]
- Jimenez-Sierra, D.A.; Quintero-Olaya, D.A.; Alvear-Muñoz, J.C.; Benítez-Restrepo, H.D.; Florez-Ospina, J.F.; Chanussot, J. Graph Learning Based on Signal Smoothness Representation for Homogeneous and Heterogeneous Change Detection. *IEEE Trans. Geosci. Remote Sens.* 2022, 60, 1–16. [CrossRef]
- 40. Sun, Y.; Lei, L.; Li, X.; Sun, H.; Kuang, G. Nonlocal patch similarity based heterogeneous remote sensing change detection. *Pattern Recognit*. **2021**, *109*, 107598. [CrossRef]
- Sun, Y.; Lei, L.; Guan, D.; Kuang, G.; Liu, L. Graph Signal Processing for Heterogeneous Change Detection–Part I: Vertex Domain Filtering. arXiv 2022, arXiv:2208.01881. [CrossRef]
- 42. Sun, Y.; Lei, L.; Guan, D.; Kuang, G.; Liu, L. Graph Signal Processing for Heterogeneous Change Detection–Part II: Spectral Domain Analysis. *arXiv* 2022, arXiv:2208.01905. [CrossRef]
- Baatz, M. Multi resolution segmentation: An optimum approach for high quality multi scale image segmentation. In *Proceedings* of the Beutrage zum AGIT-Symposium; Springer: Salzburg, Germany, 2000; pp. 12–23.
- 44. Zhou, D.; Bousquet, O.; Lal, T.N.; Weston, J.; Schölkopf, B. Learning with local and global consistency. *Adv. Neural Inf. Process. Syst.* **2004**, *16*, 321–328.
- 45. Wang, F.; Zhang, C.; Li, T. Clustering with local and global regularization. *IEEE Trans. Knowl. Data Eng.* **2009**, *21*, 1665–1678. [CrossRef]
- Kang, Z.; Peng, C.; Cheng, Q.; Liu, X.; Peng, X.; Xu, Z.; Tian, L. Structured graph learning for clustering and semi-supervised classification. *Pattern Recognit.* 2021, 110, 107627. [CrossRef]
- Zhu, X.; Li, X.; Zhang, S.; Ju, C.; Wu, X. Robust joint graph sparse coding for unsupervised spectral feature selection. *IEEE Trans. Neural Netw. Learn. Syst.* 2016, 28, 1263–1275. [CrossRef]
- Nie, F.; Li, J.; Li, X. Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semisupervised classification. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 1881–1887.
- 49. Yang, J.; Yin, W.; Zhang, Y.; Wang, Y. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM J. Imaging Sci.* **2009**, *2*, 569–592. [CrossRef]
- 50. Zhang, X.; Ying, W.; Yang, P.; Sun, M. Parameter estimation of underwater impulsive noise with the Class B model. *IET Radar* Sonar Navig. 2020, 14, 1055–1060. [CrossRef]
- Mahmood, A.; Chitre, M. Modeling colored impulsive noise by Markov chains and alpha-stable processes. In Proceedings of the OCEANS 2015, Genova, Italy, 18–21 May 2015; pp. 1–7.
- Nie, F.; Wang, X.; Huang, H. Clustering and projected clustering with adaptive neighbors. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 977–986.
- 53. Otsu, N. A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. 1979, 9, 62–66. [CrossRef]
- 54. Hartigan, J.A.; Wong, M.A. Algorithm AS 136: A k-means clustering algorithm. J. R. Stat. Soc. Ser. C (Appl. Stat.) 1979, 28, 100–108. [CrossRef]
- 55. Bezdek, J.C.; Ehrlich, R.; Full, W. FCM: The fuzzy c-means clustering algorithm. Comput. Geosci. 1984, 10, 191–203. [CrossRef]
- 56. Touati, R. Détection de Changement en Imagerie Satellitaire Multimodale. Ph.D. Thesis, Université de Montréal, Montréal, QC, Canada, 2019.
- Nar, F.; Özgür, A.; Saran, A.N. Sparsity-Driven Change Detection in Multitemporal SAR Images. *IEEE Geosci. Remote Sens. Lett.* 2016, 13, 1032–1036. [CrossRef]