remote sensing

MDPI

Article

SCAD: A Siamese Cross-Attention Discrimination Network for Bitemporal Building Change Detection

Chuan Xu [1,†], Zhaoyi Ye [1,†], Liye Mei [1,2,†], Sen Shen [3], Qi Zhang [1], Haigang Sui [4], Wei Yang [5,*] and Shaohua Sun [6,‡]

1 School of Computer Science, Hubei University of Technology, Wuhan 430068, China
2 The Institute of Technological Sciences, Wuhan University, Wuhan 430072, China
3 School of Weapon Engineering, Naval Engineering University, Wuhan 430032, China
4 State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China
5 School of Information Science and Engineering, Wuchang Shouyi University, Wuhan 430064, China
6 Air Force Research Academy, Beijing 10085, China
* Correspondence: yangwei403@wsyu.edu.cn
† These authors contributed equally to this work.
‡ Current address: Unit 91049, Qingdao 266001, China.

Abstract: Building change detection (BCD) is crucial for urban construction and planning. The powerful discriminative ability of deep convolutions in deep learning-based BCD methods has considerably increased the accuracy and efficiency. However, dense and continuously distributed buildings contain a wide range of multi-scale features, which render current deep learning methods incapable of discriminating and incorporating multiple features effectively. In this work, we propose a Siamese cross-attention discrimination network (SCADNet) to identify complex information in bitemporal images and improve the change detection accuracy. Specifically, we first use the Siamese cross-attention (SCA) module to learn unchanged and changed feature information, combining multi-head cross-attention to improve the global validity of high-level semantic information. Second, we adapt a multi-scale feature fusion (MFF) module to integrate embedded tokens with context-rich channel transformer outputs. Then, upsampling is performed to fuse the extracted multi-scale information content to recover the original image information to the maximum extent. For information content with a large difference in contextual semantics, we perform filtering using a differential context discrimination (DCD) module, which can help the network to avoid pseudo-change occurrences. The experimental results show that the present SCADNet is able to achieve a significant change detection performance in terms of three public BCD datasets (LEVIR-CD, SYSU-CD, and WHU-CD). For these three datasets, we obtain F1 scores of 90.32%, 81.79%, and 88.62%, as well as OA values of 97.98%, 91.23%, and 98.88%, respectively.

Keywords: building change detection; deep learning; Siamese cross-attention; feature fusion; differential context

Citation: Xu, C.; Ye, Z.; Mei, L.; Shen, S.; Zhang, Q.; Sui, H.; Yang, W.; Sun, S. SCAD: A Siamese Cross-Attention Discrimination Network for Bitemporal Building Change Detection. *Remote Sens.* **2022**, *14*, 6213. https://doi.org/10.3390/rs14246213

Academic Editor: Benoit Vozel

Received: 7 November 2022
Accepted: 6 December 2022
Published: 8 December 2022

1. Introduction

Change detection (CD) methods are used to observe the variety of differences in the same target during different time periods [1,2], separating the image pixel points into label 0 (unchanged) and label 1 (changed) [3]. The change in buildings is a significant indicator of urbanization, as buildings are among the most dynamic structures in a city. In order to obtain reliable information about urban change, it is critical to process building change detection (BCD) accurately and effectively. Nowadays, researchers have performed various studies on the theory and application of CD in remote sensing images, which are of a critical significance for land surveying, land resource management, urban construction and planning, and illegal construction management [4–7]. However, due to the complex texture

of the building, the variations in building forms and changes in vegetation and light during different seasons, BCD still presents considerable challenges [8].

In general, CD methods can be classified as either traditional or deep learning-based. Traditional CD methods can be categorized as pixel-based (PBCD) or object-based (OBCD) [9]. PBCD methods analyze the spectral characteristics of each pixel point by change vector analysis (CVA) [10] and principal component analysis (PCA) [11]. Support vector machines (SVM) [12,13] and random forests [14] are used for coarse matching, followed by setting thresholds to determine the CD results. Additionally, the ease of acquiring high-resolution remote sensing images has been enhanced by the rapid development in aerospace and remote sensing technologies [15]. Hay et al. [16] first introduced the concept of objects in the field from remote sensing images and applied multi-resolution segmentation techniques to extract various objects from images. Then, there have been a variety of approaches proposed for CD using OBCD, mainly based on the spectral, textural, and spatial background information at the object level [17–19]. However, the traditional CD methods still have considerable limitations. The PBCD methods ignore the spatial correlation between adjacent pixels, focusing only on the spectral information. Semantic information is not taken into account by the OBCD algorithms, which makes the model unable to effectively identify pseudo-changes. Furthermore, traditional CD methods cannot adequately characterize the changes in buildings in high-resolution remote sensing images, making them unsuitable for complying with real-world accuracy requirements.

With the advancement in computing power and data, deep learning has produced a large amount of research in the fields of object detection, image classification, and semantic segmentation [20–22]. As a result, deep learning algorithms are currently being applied to CD, a hotspot in remote sensing research [23,24]. Currently, the majority of deep learning-based CD methods involve networks which show efficient results regarding contrastive learning [25] and segmentation tasks [26]. The main purpose of comparative learning is to deduce the differences between similar objects and expand the differences between various kinds of objects. For example, Dong et al. [27] built a network based on a time prediction; specifically, this network can distinguish the different patches in bitemporal images, encode them into more consistent feature information, and finally obtain detection results through a clustering [28] algorithm. However, the operation process leads to the loss of a great deal of semantic information, which leads to missing detection occurrences. Chen et al. [29] presented an unsupervised CD method using self-supervised learning to pretrain a neural network; in this method, contrastive and regression losses are used to calculate varied and similar images. Chen et al. [30] innovatively proposed the pyramid spatial–temporal attention module (PAM), which mitigates the effect of light variations on CD performance; however, this method merely considers the spatial attention weights between bitemporal images. Wang et al. [31] proposed focal contrastive loss to alleviate the imbalance between positive and negative samples in CD, and this method reduces the intra-class variance and increases the inter-class difference so that the binarized output is more easily obtained by setting a threshold.

Even though these methods have achieved effective results as a result of comparative learning, their effectiveness is greatly affected by the sample distribution of the datasets. Using distance metrics in contrastive learning methods for CD tasks remains ineffective. Since buildings are densely distributed, shadows, similar roads, and other factors often affect the change areas of the buildings. Therefore, segmentation methods are able to segment the region of the changed areas, achieving better CD effects. Zhan et al. [32] obtained the CD results by combining a weight-sharing network with a threshold segmentation of the feature graph at the final layer; however, this network structure is relatively simple and unable to extract deeper semantic information. Chen et al. [33] employed spatial and channel-attention mechanisms to extract the feature information from spatial and temporal channels, respectively, more efficiently and comprehensively capturing global dependencies and long-range contextual information. However, large-scale cross-dimensional operations undoubtedly increase the computational time. Mi et al. [34] developed a deep

neural forest based on semantic segmentation, which effectively alleviates the impact of noise on the CD results, but still remains unsatisfactory in terms of missing detection.

Although researchers have proposed various methods, the BCD problems are not yet completely resolved. First, the current attention mechanisms are incapable of efficiently focusing on the unchanged and changed regions when there are large numbers of pseudo-changes in the bitemporal images, which can lead to serious false detection phenomena. Second, there are large numbers of downsampling and upsampling operations in the existing networks, leading to the loss of bitemporal information; furthermore, the direct fusion strategy exacerbates this issue, making the network unable to effectively recover image information during upsampling, and the final detection results will also have issues, such as missed detections and untidy change edges. Finally, the current algorithms only perform CD operations and do not take into account the differential information in the bitemporal images, so they cannot distinguish the pseudo-change information satisfactorily. As a result, detecting building changes in high-resolution remote sensing images remains a significant challenge; improving the detection accuracy and interpreting these images effectively remain an imperative part of BCD research.

Therefore, we propose a novel deep learning-based network (SCADNet) for BCD. In the encoding stage, the shared-weight Siamese network with the Siamese cross-attention (SCA) module is used to extract the features from the bitemporal images, combining them with multi-head cross-attention to enhance feature perception and global effectiveness. To alleviate the network's fusion-stage information loss, we add a multi-scale feature fusion (MFF) module in the decoding stage, which enables it to fuse multi-scale feature information by fusing adjacent scales step-by-step. More importantly, we propose a differential context discrimination (DCD) module, which obtains similar and different features between contexts, increasing the resistance of the model to pseudo-variation by increasing the variation in different contextual features.

The most significant contributions of our work are summarized as follows:

(1) We added a SCA module to the Siamese network, focusing on unchanged and changed regions. The Siamese network is now capable of deploying two-channel targeted attention on the specified feature information, strengthening the network's characterization ability and improving its ability to recognize environmental and building changes, and thus enhancing the network's recognition accuracy.
(2) Our proposed MFF module is able to fuse independent multi-scale information, recover the original feature information of remote sensing images as much as possible, reduce the false detection rate of CD, and make the edge lines of detecting change regions more delicate.
(3) We designed a DCD module by combining differential and concatenation methods, enhancing the feature differences between contexts and focusing on comparing the differences between pseudo and real changes, making the model more responsive to the region where the changes occur, thus reducing the network's missing detection rate.

2. Materials and Methods

The first part of this section describes the overall structure of the SCADNet, followed by a detailed description of the three modules, SCA, MFF, and DCD, respectively. Finally, the loss function is described.

2.1. Network Architecture

An overview of the structure of the SCADNet network is shown in Figure 1. The model first takes the bitemporal remote sensing images from the same area as the input. We receive the feature information from the bitemporal images through a weight-sharing Siamese network. The decoder, composed of SCA and MFF modules, decodes the multi-scale features. Specifically, the SCA module extracts unchanged and changed feature information. The MFF module fuses the extracted multi-scale feature information. Then, our DCD module performs the differential operation between the pre- and post-temporal

images to obtain the differential map and inputs the predicted and difference images together into a discriminator for a contextual difference discrimination. The discriminator will calculate the probability loss, when the probability loss is less than the set threshold, the discriminator will output the final BCD result.

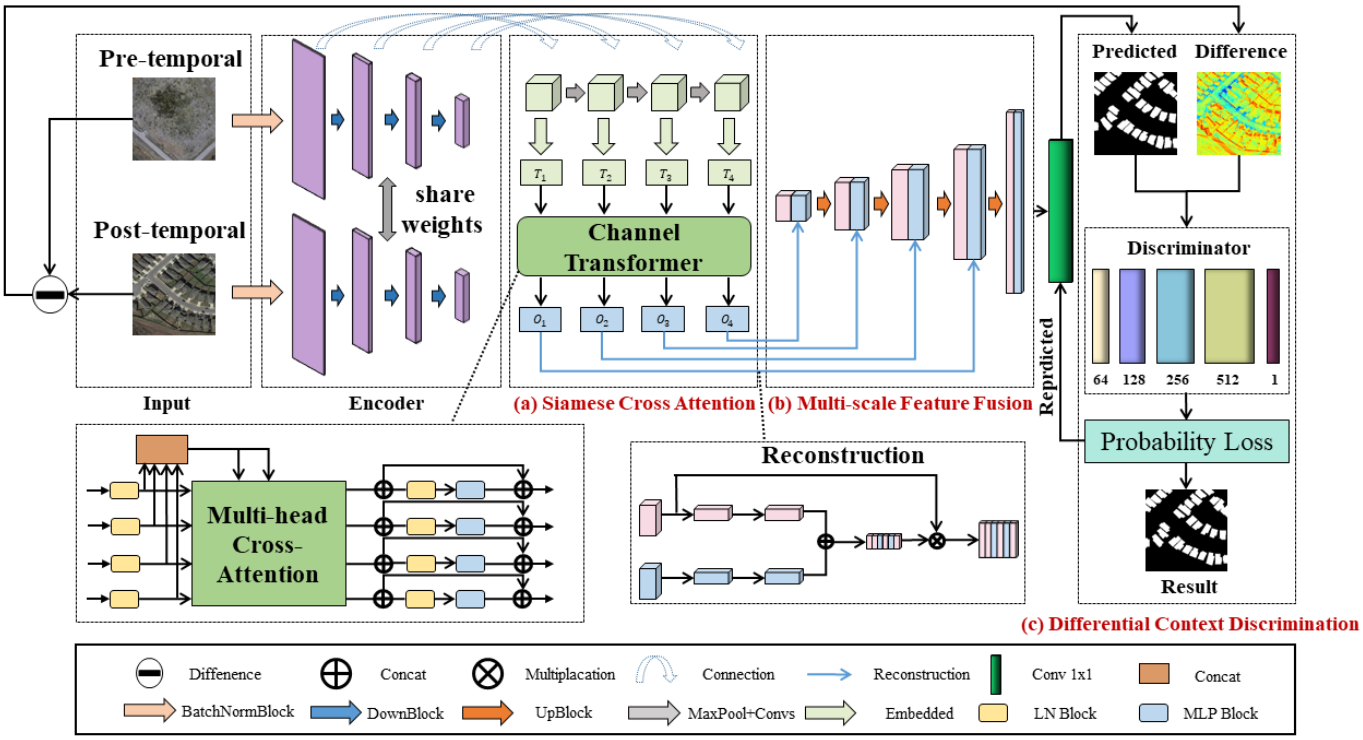

Figure 1. Overview of the proposed SCADNet. Pre- and post-temporal remote sensing images were entered into the encoder to obtain multi-scale features. Additionally, the multi-head cross-attention feature was combined with the SCA module to strengthen attention to the specified change information. MFF module integrated multi-scale feature information twice by Reconstruction and UpBlock. Finally, the DCD module received predicted and difference images. Discriminators was used to obtain the probability loss and iterated continuously until the probability loss reached the minimum value to obtain the CD result.

As part of the encoding process, we collect the characteristic features of the bitemporal images using a weight-sharing Siamese network; subsequently, we perform a preprocessing operation on the two-channel image input to the BatchNormBlock, including a convolution kernel of 3, a 2D convolution with a step size of 1, a 2D BatchNorm, and an ReLU activation function with an output channel number of 64. The numbers of channels for extracting the feature information are 128, 256, and 512, respectively.

Our decoding stage consists of two parts: SCA and MFF modules. The SCA module uses the Siamese cross-attention mechanism and fuses the transformer multi-head cross-attention mechanism to obtain the changed and unchanged features of the bitemporal images. The MFF module first reconstructs the acquired multi-scale feature information; then, the image feature information is recovered through four upsampling operations for the predictive results of the changed buildings.

Finally, our DCD module feeds the predicted and difference pictures into the discriminator to determine the current probability loss of CD and it repeats this process until the probability loss is minimized.

2.2. Siamese Cross-Attention Module

Two-stream Siamese networks can achieve relatively effective results in BCD tasks. The principle is to use two Siamese channels to pre- and post-temporal images, and then extract the features for the BCD in parallel.

However, the traditional fully convolutional Siamese neural network does not improve the extraction of the image's features and rich contextual semantic information, and it focuses excessively on low-level feature information, which is irrelevant for CD. Therefore, the traditional fully convolutional Siamese neural network suffers from a number of difficulties, such as inaccurate region boundaries, and missed or false detections.

In light of the need to extract fine-grained and abundant image features as well as the combination of contextual semantic information for BCD, our SCA module enhances the traditional Siamese network by adding a Siamese cross-attention mechanism to and adds a multi-head cross-attention mechanism in order to obtain more comprehensive spatiotemporal semantic information.

As illustrated in Figure 1, we also use the Siamese channel with shared weights for the SCA module. Four outputs from the encoder stage are received as the inputs, and we perform the embedded operation on these four inputs, starting with a 2D convolution, followed by flattening the features into two-dimensional sequences with patch sizes of 32, 16, 8, and 4. Therefore, we obtain the four scales of feature information tokens $T_i(i = 1, 2, 3, 4)$, then concatenate them as $T_\sum$, including the key and value.

Figure 2 shows the multi-head cross-attention mechanism of the channel transformer. We input all four of the above tokens and $T_\sum$ into the multi-head cross-attention mechanism and enable each token to learn more abundant multi-scale features:

$$Q_i = T_i W_{Q_i}, K = T_\sum W_K, V = T_\sum W_V \tag{1}$$

where W_{Q_i}, W_K, and W_V are the weights of different inputs and C_i is the channel dimension of the four input tokens [35]. In our network, C_1 = 64, C_2 = 128, C_3 = 256, C_4 = 512.

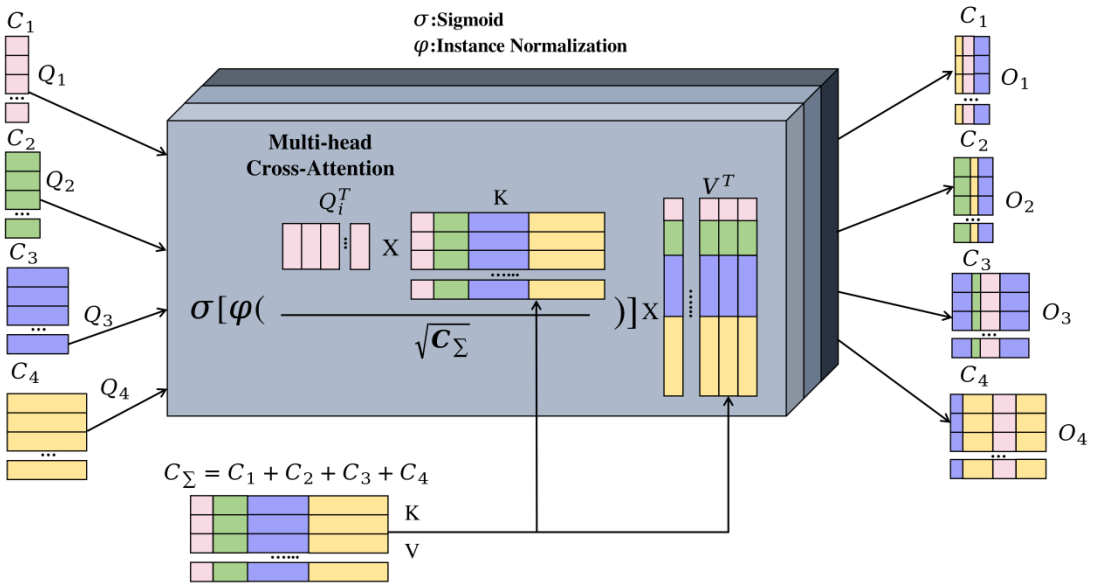

Figure 2. Overview of multi-head cross-attention mechanism.

Equation (2) shows that we generate similarity matrix M_i by Q_i, K, and V by weighting M_i and value V^T to obtain the cross-attention:

$$CA_i = M_i V^T = \sigma[\varphi(\frac{Q_i^T K}{\sqrt{C_\sum}})]V^T = \sigma[\varphi(\frac{W_{Q_i}^T T_i^T T_\sum W_K}{\sqrt{C_\sum}})]W_V^T Q_\sum^T \tag{2}$$

where $\sigma(\cdot)$ and $\varphi(\cdot)$ denote the instance normalization [36] and softmax function, respectively. Using instance normalization, each similarity matrix can be normalized so that the gradients are propagated more smoothly. In our implementation, after multi-head cross-attention, we calculate the N-head output as follows:

$$MCA_i = (CA_i^1 + CA_i^2 +, \ldots, +CA_i^N)/N \tag{3}$$

where N is the number of heads. We then execute the *MLP* and residual operator to obtain the final output:

$$O_i = MCA_i + MLP(Q_i + MCA_i) \tag{4}$$

We perform the operation of Equation (4) four times to obtain four outputs: O_1, O_2, O_3, and O_4. In addition to high-level semantic information, these outputs include information about the region of interest for change.

2.3. Multi-Scale Feature Fusion Module

Since the main purpose of CD is to detect changes in each pixel point, if only the individual pixel points themselves are considered, the extracted feature information is completely independent and cannot represent the entire image information properly. Thus, insufficient feature information will lead to false and missing detections. Moreover, bitemporal feature fusion is a critical part of Siamese network CD. Using the feature information directly will result in information loss and redundancy, which will negatively impact the accuracy of the detection. Therefore, we propose an MFF module to fuse multi-scale feature information.

Figure 1 shows that our MFF module is divided into two parts for the execution: Reconstruction and UpBlock. The Reconstruction operation receives two inputs, the Token output after the embedded operation, $T_i(i = 1, 2, 3, 4)$, and the output from the Channel Transformer, $O_i(i = 1, 2, 3, 4)$. These two inputs are spatially squeezed by the global pooling layer to obtain vector $V(x)$ and its kth channel. We start with generating an attention mask:

$$M_i = W_1 \cdot V(T_i) + W_2 \cdot V(O_i) \tag{5}$$

where W_1 and W_2 are the weights of the two linear layers; then, the individual channels are connected. We repeat the above operation four times to obtain the four outputs. We also perform upsampling operations to integrate the above four outputs in order to better fuse the multi-scale feature information. Additionally, the output channels of the four UpBlocks are 256, 128, 64, and 64. Finally, we convolve the output results of the fourth UpBlock once with a convolution kernel of 1 and a step size of 1 to obtain the fusion results of the BCD, which are then input to the DCD module for discrimination.

2.4. Differential Context Discrimination Module

Remote sensing images contain complex image contents, as well as a variety of building shapes. The same building can have large variations in different scenes and time sequences. The effective discrimination of differential context information can help the network to extract valuable information more efficiently, thereby improving the recognition accuracy and robustness to pseudo-variation features.

The current mainstream context discrimination methods are differential and concatenation methods [37]. The differential method can obtain bitemporal change information; however, it is affected by changes in the shooting angle and light. Although the concatenation method can extract continuous features from images, it does not adequately capture bitemporal changes. Therefore, our proposed DCD module combines the advantages of the above two methods. It focuses not only on regions with small semantic changes but also on regions of change where there are large differences between the contexts. Specifically, we use the difference operation to bitemporal images as a difference image, and it is worth noting that the input of the discriminator is multichannel images created by concatenating

the difference image and the generator's predicted change map in the channel dimension, aiming to provide prior information for better-discriminating features.

Figure 1 shows that there are two inputs to the DCD module. One is the predicted image obtained after the Siamese network processing and the other is the different image obtained by the difference operation between the pre- and post-temporal images. We input these two images into the discriminator for differential context discrimination. We define the real one as GT, whereas the fake one is the generator's predicted change map. Probability loss is the result of the discriminator calculating the difference between the fake and the real. Our discriminator consists of a fully connected convolutional neural network; specifically, there are five convolutional layers in the discriminator, each with a convolution kernel size of 4, and the numbers of channels in each layer are 64, 128, 256, 512, and 1. Each convolutional layer has a convolutional padding of 2, while the first 4 layers have a stride size of 2. The last layer has a stride size of 1. Additionally, a Leaky-ReLU operation is performed after each convolutional layer. The discriminator will finally output the probability loss of this CD result, and the loop executes the DCD several times to guide the probability loss to the minimum. Therefore, the DCD module makes the outputs of the network more closely resemble GT, which ultimately produces a higher accuracy map of the BCD results.

2.5. Loss Function

By optimizing the correct loss during training, the Jaccard index [38] can substantially improve the accuracy of CD. The loss of our network can be simplified by Equation (6), which is based on the Jaccard index:

$$L_{SCAD} = \rho \frac{1}{C} \sum_{c \in C} J_C(v(c)) \tag{6}$$

where $v(c)$ is a vector of the pixel errors for class $c \in C$ aiming to construct the loss surrogate to J_C. It is defined by:

$$v(c) = \begin{cases} 1 - s_i(c), & if\ c = y_i \\ s_i(c), & if\ c \neq y_i \end{cases} \tag{7}$$

where y_i represents the ground truth and $s_i(c)$ is the network's prediction result. A set function J_C encodes a submodular Jaccard loss for class c and indicates a set of generated error results.

As a result of choosing a suitable loss function, we improved the accuracy of BCD, bringing the edge and detail information closer to the target image. If we had used a regular GAN [39], there would be problems, such as difficulty in convergence and model explosion. Based on previous experimental experience [38], we used the least-squares generative adversarial network (*LSGAN*) [40] as the loss function in our work. The *LSGAN* is more stable and can detect changes more accurately. It is defined by Equation (8):

$$L_{LSGAN}(D) = E_{i,y\sim} P_{data(i,y)}[(D(i,y) - 1)^2] + E_{i\sim} P_{data(i)}[(D(i, G(i)))^2] \tag{8}$$

LSGAN is also used to optimize adversarial learning, and is formulated as follows:

$$L_{LSGAN}(G) = E_{i\sim} P_{data(i)}[(D(i, G(i)) - 1)^2] \tag{9}$$

In addition, we employ a supervised training, which enhances the accuracy of CD. SCADNet's objective function, therefore, can be defined as follows:

$$\begin{array}{c} \min L(D) = L_{LSGAN}(D) \\ \min L(G) = L_{LSGAN}(G) + \alpha L_{SCAD} \end{array} \tag{10}$$

the relative weights of the two objective functions are controlled by α. Our task is best suited by setting α to 5.

3. Experiments and Results

The following is a description of this section. The datasets we used are LEVIR-CD, SYSU-CD, and WHU-CD. Our evaluation metrics and the parameters for the experiments are then described. Finally, we describe an ablation experiment on the LEVIR-CD dataset and compare the various methods comprehensively. As a result of our experimental results, our method outperformed the alternatives.

3.1. Datasets

We used three public BCD datasets: LEVIR-CD, SYSU-CD, and WHU-CD to evaluate the superiority of SCADNet. Those datasets contain pre-temporal and post-temporal images as well as the labels of the changed building areas. The experimental datasets are briefly described in Table 1.

Table 1. An overview of experimental datasets.

Name	Bands	Image Pairs	Resolution (m)	Image Size	Training Set	Validation Set	Testing Set
LEVIR-CD	3	637	0.5	1024 × 1024	3096	432	921
SYSU-CD	3	20,000	0.5	256 × 256	12,000	4000	4000
WHU-CD	3	1	0.5	32,207 × 15,354	5201	744	1487

1. LEVIR-CD [30], created by Bei-hang University, contains a variety of architectural images, including original Google Earth images collected between 2002 and 2018. There are 1024 × 1024 pixels in each image with a resolution of 0.5 m. Due to GPU memory limitations, we divided each image into 16 patches of 256 × 256 pixels without an overlap. As a result, we obtained 3096, 432, and 921 pairs of patches for training, validation, and testing, respectively. As shown in Figure 3, a few scenes are taken from the LEVIR-CD dataset.

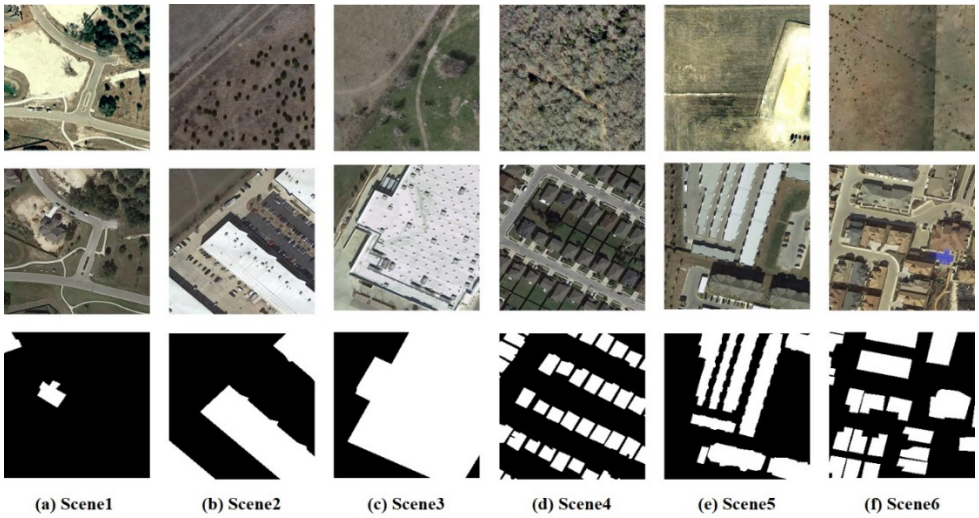

Figure 3. Display of sample images from the LEVIR-CD dataset (first row indicates pre-temporal images, second row indicates post-temporal images, and third row indicates labelled images).

2. Sun Yat-Sen University created the challenging SYSU-CD dataset [41] for CD. This dataset contains changes in vegetation and buildings in a forest, buildings along a coastline, and the appearance and disappearance of ships in an ocean. The image size is 256 × 256 pixels with a resolution of 0.5 m. In our work, the training, validation, and test sample proportion was 6:2:2. Therefore, we obtained 12,000, 4000, 4000 pairs of patches for training, validation, and testing, respectively. Figure 4 illustrates some of the variety of the scenarios included in the SYSU-CD dataset.

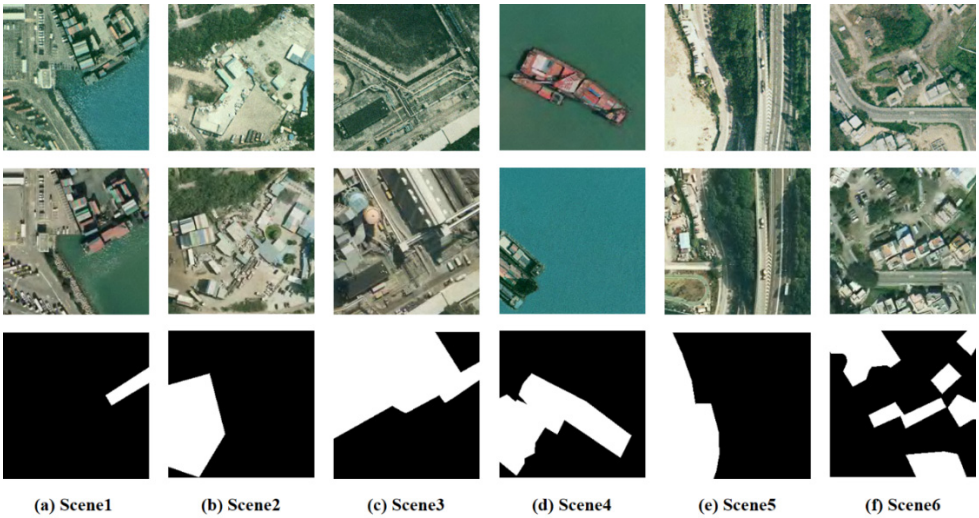

Figure 4. Display of sample images from the SYSU-CD dataset (first row indicates pre-temporal images, second row indicates post-temporal images, and third row indicates the labelled images).

3. The WHU-CD dataset [42] was released by the University of Wuhan as a public CD dataset. Only one image is included in the original dataset, which is 15,354 × 32,507 pixels. In order to be consistent with the two datasets mentioned above, 7432 patches were generated by cropping the image into 256 × 256 pixels. During the splitting, no overlap was used. In the end, we obtained 5201, 744, and 1487 pairs of patches for the training, validation, and testing, respectively. A few scenes from the WHU-CD building dataset are shown in Figure 5.

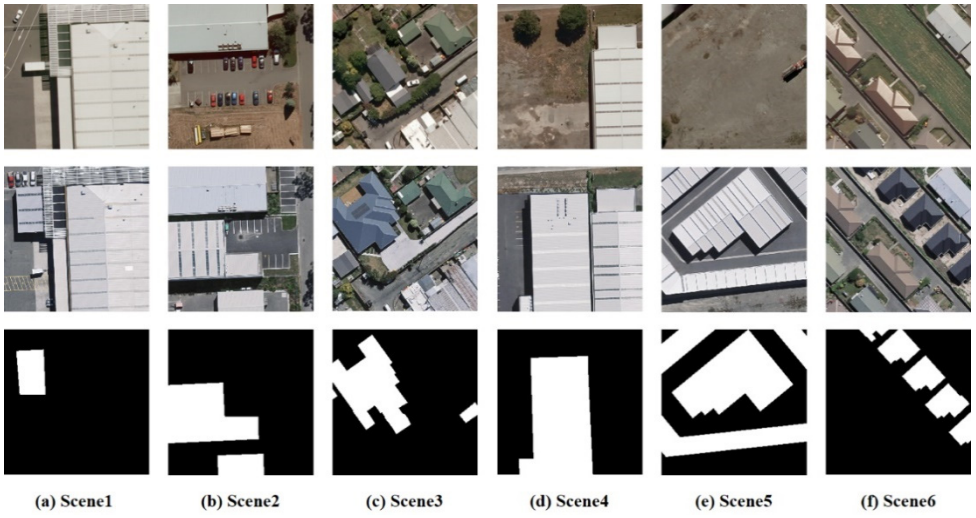

Figure 5. Display of sample images from the WHU-CD dataset (first row indicates pre-temporal images, second row indicates post-temporal images, and third row indicates labelled images).

3.2. Experimental Details

3.2.1. Evaluation Metrics

Remote sensing CD presents a problem regarding the binary classification of the pixels. CD algorithms are therefore evaluated using the following quantitative evaluation metrics that are commonly used in binary classification problems: precision (P), recall (R), F1 score, mean intersection over union (mIOU), overall accuracy (OA), and kappa coefficient. Precision indicates fewer false detections, while recall indicates fewer missed detections. The higher the mIOU and F1 scores, the better the performance. We also added two additional values, where IOU_0 and IOU_1 indicate that an unchanged or changed

region is detected, respectively. The OA provides an overall assessment of the model's performance, with higher values representing a better performance. The consistency is checked using the kappa coefficient. Specifically, we defined the evaluation metrics as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{11}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{12}$$

$$\text{F1} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \tag{13}$$

$$\text{IOU\_0} = \frac{\text{TN}}{\text{TN} + \text{FP} + \text{FN}} \tag{14}$$

$$\text{IOU\_1} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{15}$$

$$\text{mIOU} = (\text{IOU\_0} + \text{IOU\_1})/2 \tag{16}$$

$$\text{OA} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{17}$$

$$\text{Pe} = \frac{(\text{TP} + \text{FP})(\text{TP} + \text{FN}) + (\text{FP} + \text{TN})(\text{FN} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})^2} \tag{18}$$

$$\text{Kappa} = \frac{\text{OA} - \text{Pe}}{1 - \text{Pe}} \tag{19}$$

In the above formulas, true positive is abbreviated as TP. False positive is referred to as FP. True negative is abbreviated as TN. False negative is referred to as FN.

3.2.2. Parameter Settings

Throughout the experiments, we employed the PyTorch framework to build all the models. We used NVIDIA GeForce RTX 3090 GPU in our experiments. Based on the limitations of the GPU memory, we set the batch training size to eight when configuring the parameters of the network model training. The maximum number of epochs we used for training the model was 200. Training was stopped early during the process in order to prevent overfitting. Our initial learning rate was 0.0002. In the overall training process, the model with the highest performance on the validation set will be applied to the test set for testing.

3.3. Ablation Experiment

In order to confirm the effectiveness of our proposed SCA and DCD modules, we conducted ablation experiments using the LEVIR-CD dataset, and the results are shown in Table 2.

Table 2. Results of ablation experiments using LEVIR-CD dataset.

Method	Precision	Recall	F1 Score	mIOU	IOU_0	IOU_1	OA	Kappa
Baseline	87.73	91.31	89.48	89.16	97.36	80.96	97.63	88.15
Baseline + SCA	88.37	91.29	89.81	89.48	97.45	81.50	97.71	88.52
Baseline + SCA + DCD	**90.14**	**91.74**	**90.32**	**90.56**	**97.75**	**83.37**	**97.98**	**89.79**

Note that the best results are in bold.

The F1 score increased from 89.48% in the baseline to 89.81% when the SCA module was added separately. As a result of the simultaneous addition of the SCA and DCD modules, the model's F1 score increased by 0.84% over the baseline, reaching 90.32%. With just the SCA module, the precision improved by only 0.64%, but with the additional use of the DCD module, the precision improved by 2.41%, demonstrating that the DCD module is

effective at reducing missed detections. Finally, when all the modules were applied to the baseline, the precision and IOU_1 for the CD reached their highest values of 90.14% and 83.37%, respectively. These values are significantly higher than the baseline.

3.4. Comparative Experiments

We selected several classical CD models as well as the existing SOTA models for comparison experiments to demonstrate the accuracy and effectiveness of SCADNet. The selected algorithms are described in detail as follows:

1. FC-EF [43]: A method of image-level fusion in which bitemporal images are concatenated to shift the single input to a fully convolutional network, and feature mapping is performed through skip connections.
2. FC-Siam-conc [43]: This method fuses the multiscale information in the decoder. A Siamese FCN is employed, which uses the same structure and shared weights to extract multilevel features.
3. FC-Siam-diff [43]: Only the skip connection is different between this method and FC-Siam-conc. Instead of concatenating the absolute values, FC-Siam-diff uses absolute value differences.
4. CDNet [44]: CDNet is initially used to detect street changes. The core part of the network is four compression blocks and four extension blocks. The compression blocks acquire feature information about the images and the extension blocks refine the change regions. Softmax is used to classify each pixel point for the prediction, balancing performance, and model size.
5. IFNet [45]: An image fusion network for CD that is deeply supervised. The bitemporal images are first extracted using a two-stream network. The feature information is then transferred to the deep supervised difference discrimination network (DDN) for analysis. Finally, channel attention and spatial attention are applied to fuse the two-stream feature information to ensure the integrity of the change region boundaries.
6. SNUNet [46]: SNUNet reduces the loss of deep information in neural networks by combining the Siamese network and NestedUNet. In addition, an ensemble channel attention module (ECAM) is used to achieve an accurate feature extraction.
7. BITNet [47]: A transformer module is added to the Siamese network to transform each image into a set of semantic tokens. This is followed by building a contextual model of this set of semantic tokens using an encoder. Finally, a decoder restores the original information of the image through a decoder, thus enhancing the feature representation in the pixel space.
8. LUNet [48]: LUNet is implemented by incorporating an LSTM neural network based on UNet. It adds an integrated LSTM before each encoding process, which makes the network operation more lightweight by adjusting the weight of each LSTM, the bias, and the switch of the forgetting gate, thus achieving an end-to-end network structure.

We employed a total of twelve methods to conduct comparative experiments; we do not have an open source well reproducible code for the four methods (DeepLabV3, UNet++, STANet, and HDANet) [49]. Therefore, to respect the existing work, we directly referenced the available accuracy evaluation metrics on the BCD datasets.

Table 3 shows the comparison results of the various methods on the LEVIR-CD dataset. SCADNet outperformed other networks in all the metrics except precision. The F1 score and recall metrics also showed an excellent performance (90.32% and 91.74%, respectively). HDANet achieved the highest precision of 92.26%, which was 2.12% better than our method. Since SCADNet is focused not only on correctly detecting the areas that are really changing, but also on all areas where a change occurs. The recall metric of our method was higher than HDANet's of 4.13%. In light of the F1 score metric, we can conclude that our method had the most robust overall performance.

Table 3. Comparative results for different methods used on the LEVIR-CD dataset.

Method	Precision	Recall	F1 Score	mIOU	IOU_0	IOU_1	OA	Kappa
FC-EF	79.91	82.84	81.35	81.97	95.38	68.56	95.80	78.99
FC-Siam-conc	81.84	83.55	82.68	83.11	95.74	70.48	96.13	80.51
FC-Siam-diff	78.60	89.30	83.61	83.77	95.71	71.84	96.13	81.43
CDNet	84.21	87.10	85.63	85.65	96.43	74.87	96.77	83.81
IFNet	85.37	90.24	87.74	87.53	96.91	78.16	97.21	86.17
SNUNet	91.05	88.87	89.94	89.65	97.57	81.73	97.81	88.71
BITNet	87.32	91.41	89.32	89.00	97.31	80.70	97.59	87.96
LUNet	85.69	90.99	88.73	88.44	97.13	79.75	97.42	87.28
DeepLabV3	90.03	82.51	86.11	-	-	-	-	85.39
UNet++	91.44	85.24	88.23	-	-	-	-	87.62
STANet	92.01	83.33	87.46	-	-	-	-	86.82
HDANet	**92.26**	87.61	89.87	-	-	-	-	89.34
SCADNet(ours)	90.14	**91.74**	**90.32**	**90.56**	**97.75**	**83.37**	**97.98**	**89.79**

Note that the best results are in bold.

We selected two scenarios from each of the three BCD datasets for visualization. As can be seen in Figure 6, there is only one variation in the building. Due to the changes in lighting, all the methods except SNUNet and SCADNet detected the house in the lower-right corner as a building change. In addition, we achieved fewer false detections compared with SNUNet in the upper-left corner of the building.

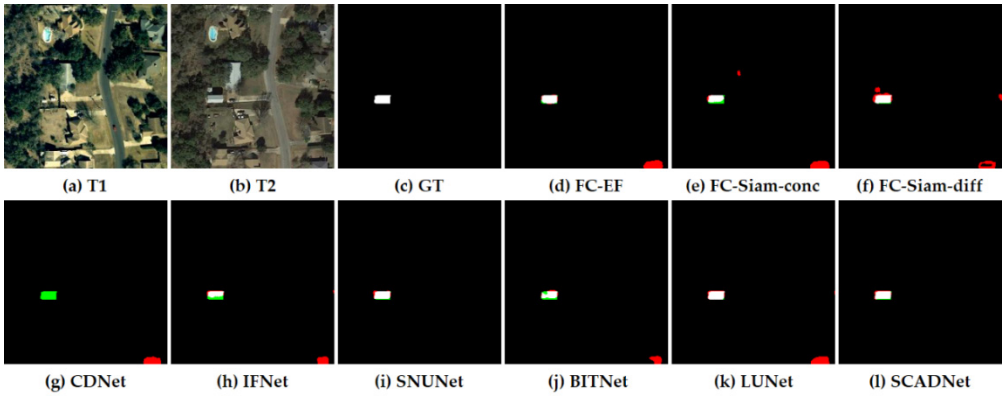

Figure 6. LEVIR-CD dataset comparison results visualization for the first image. TP is visualized in white, TN is visualized in black, FP is visualized in red, and FN is visualized in green.

As shown in Figure 7, our method was able to detect a dense multi-objective building change scene with a total of three rows of buildings changing. In contrast, all the other methods failed to detect the relatively small changes in the second row of the buildings. While FC-EF, FC-Siam-conc, and FC-Siam-diff almost completely missed these small targets, our method was successful in identifying them. In addition, SCADNet was also able to achieve a low false detection rate for slightly larger buildings.

Table 4 shows the comparison results for the SYSU-CD dataset. Our method achieved optimal results in terms of the F1 score, mIOU, IOU_0, IOU_1, OA, and kappa metrics. One of the most notable results was our F1 score of 81.79%. Among the other methods, the F1 score of the IFNet also reached 80.98%, which is only 0.81% lower than that of our method, while the SCADNet was second only to UNet++, DeepLabV3, BITNet, and STANet in terms of the precision. With the multi-scale feature fusion strategy, IFNet exhibited the highest recall rate of all the methods, reaching 87.60%. DeepLabV3 and the three FC-based methods had a relatively insufficient detection accuracy. For example, the F1 score and precision for FC-EF were 75.13% and 64.58%, respectively, which were 6.66% and 15.54% lower than those for our method.

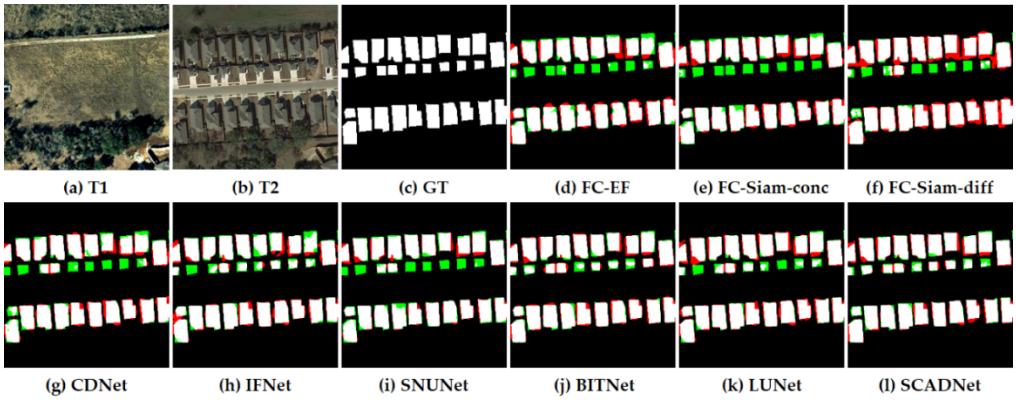

Figure 7. LEVIR-CD dataset comparison results visualization for the second image. TP is visualized in white, TN is visualized in black, FP is visualized in red, and FN is visualized in green.

Table 4. Comparative results for different methods used on the SYSU-CD dataset.

Method	Precision	Recall	F1 Score	mIOU	IOU_0	IOU_1	OA	Kappa
FC-EF	64.58	89.79	75.13	71.19	82.21	60.17	85.98	65.73
FC-Siam-conc	65.98	89.39	75.99	72.18	83.08	61.28	86.65	67.04
FC-Siam-diff	70.84	84.87	77.22	74.07	85.24	62.90	88.19	69.34
CDNet	74.61	84.10	79.08	76.15	86.90	65.39	89.50	72.10
IFNet	75.29	**87.60**	80.98	77.91	87.77	68.04	90.30	74.52
SNUNet	76.90	79.59	78.22	75.68	87.13	64.23	89.55	71.35
BITNet	80.61	79.29	79.95	77.53	88.46	66.59	90.62	73.83
LUNet	76.14	81.74	78.84	76.13	87.18	65.08	89.65	72.01
DeepLabV3	80.99	70.65	75.47	-	-	-	-	68.56
UNet++	**81.44**	74.66	77.90	-	-	-	-	71.76
STANet	80.38	74.75	77.46	-	-	-	-	70.84
HDANet	78.53	79.88	79.20	-	-	-	-	72.71
SCADNet(ours)	80.12	83.53	**81.79**	**79.13**	**89.08**	**69.19**	**91.23**	**76.02**

Note that the best results are in bold.

Figure 8 shows a scene from the SYSU-CD dataset, where only a small portion of the upper part of the image has building changes and the site environment is complex. Aside from the SCADNet, FC-EF, CDNet, and IFNet, all the other methods misidentified vegetation changes as building changes. Further, the CDNet performed second only to the SCADNet and caused only a few false detections around the change area. Even though the SCADNet did not completely detect the actual change area, it was able to avoid the occurrence of false detections.

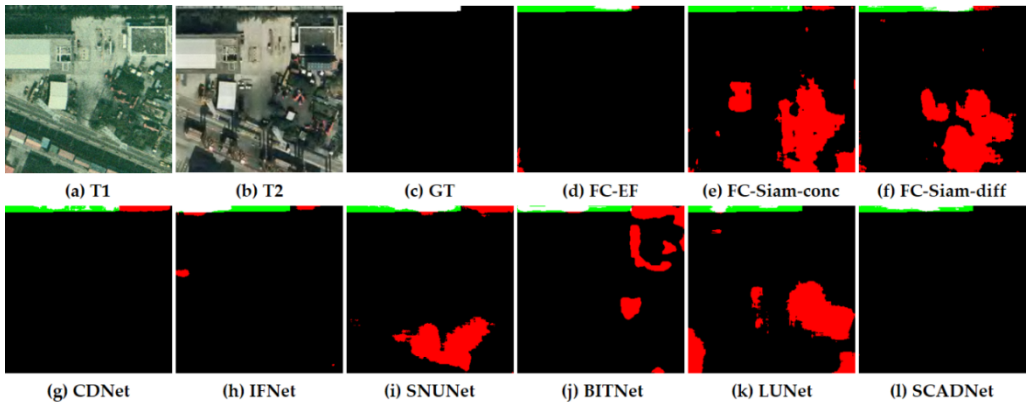

Figure 8. SYSU-CD dataset comparison results visualization for the first image. TP is visualized in white, TN is visualized in black, FP is visualized in red, and FN is visualized in green.

There are large-scale building changes in Figure 9. In spite of the fact that all the methods were able to detect the main part of the changed building, they did not perform the BCD effectively. Among them, the BITNet missed a significant part of the changed area, while other methods except the LUNet and SCADNet had a large area of false detection. Furthermore, the SCADNet detected edges more accurately when it came to identifying the changed regions.

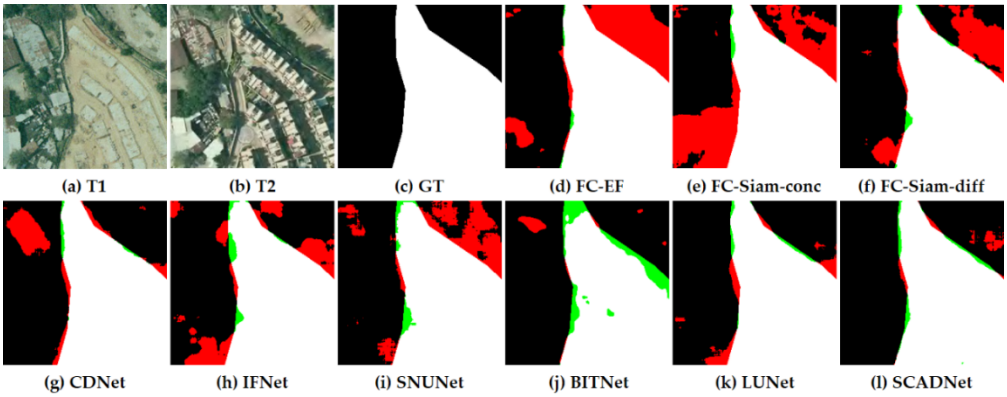

Figure 9. SYSU-CD dataset comparison results visualization for the second image. TP is visualized in white, TN is visualized in black, FP is visualized in red, and FN is visualized in green.

The metrics for each method are compared in Table 5 on the WHU-CD dataset. Our method was still able to outperform the other methods in terms of the F1 score, mIOU, IOU_0, IOU_1, OA, and kappa metrics. Although FC-Siam-diff achieved 94.30% in recall, outperforming our method by 1.8%, its precision only reached 65.98%, which is 19.08% lower than ours. Similarly, the HDANet achieved an optimum precision of 89.87%, which was 4.81% higher than our method, but its recall was 9.55% lower than that of the SCADNet. Moreover, our method achieved an F1 score value of 88.62%, indicating that it has a better comprehensive BCD performance.

Table 5. Comparative results for different methods used on the WHU-CD dataset.

Method	Precision	Recall	F1 Score	mIOU	IOU_0	IOU_1	OA	Kappa
FC-EF	70.43	92.31	79.90	82.12	97.72	66.53	97.82	78.77
FC-Siam-conc	63.80	91.81	75.28	78.70	97.04	60.36	97.16	73.83
FC-Siam-diff	65.98	**94.30**	77.63	80.38	97.33	63.44	97.44	76.32
CDNet	81.75	88.69	85.08	86.25	98.47	74.03	98.54	84.31
IFNet	86.51	87.69	87.09	87.93	98.72	77.14	98.78	86.45
SNUNet	81.92	85.33	83.59	85.08	98.36	71.80	98.42	82.76
BITNet	82.35	92.59	87.17	87.96	98.66	77.26	98.72	86.50
LUNet	66.32	93.06	77.45	80.26	97.33	63.19	97.45	76.13
DeepLabV3	82.56	81.97	82.26	-	-	-	-	81.58
UNet++	89.06	78.98	83.72	-	-	-	-	83.13
STANet	86.01	83.40	84.68	-	-	-	-	84.10
HDANet	**89.87**	82.55	86.05	-	-	-	-	85.54
SCADNet(ours)	85.06	92.50	**88.62**	**89.20**	**98.83**	**79.57**	**98.88**	**88.04**

Note that the best results are in bold.

Figure 10 shows that there is no change in the building, and the change in the roof color of the building caused all the other methods to produce false detections. The FC-EF, CDNet, and SNUNet all identified the area where the building already exists as the change area, while the BITNet and LUNet classified the road change as a building change. Our method did not cause any false detections or omissions, and the detection results were consistent with GT, achieving optimal results.

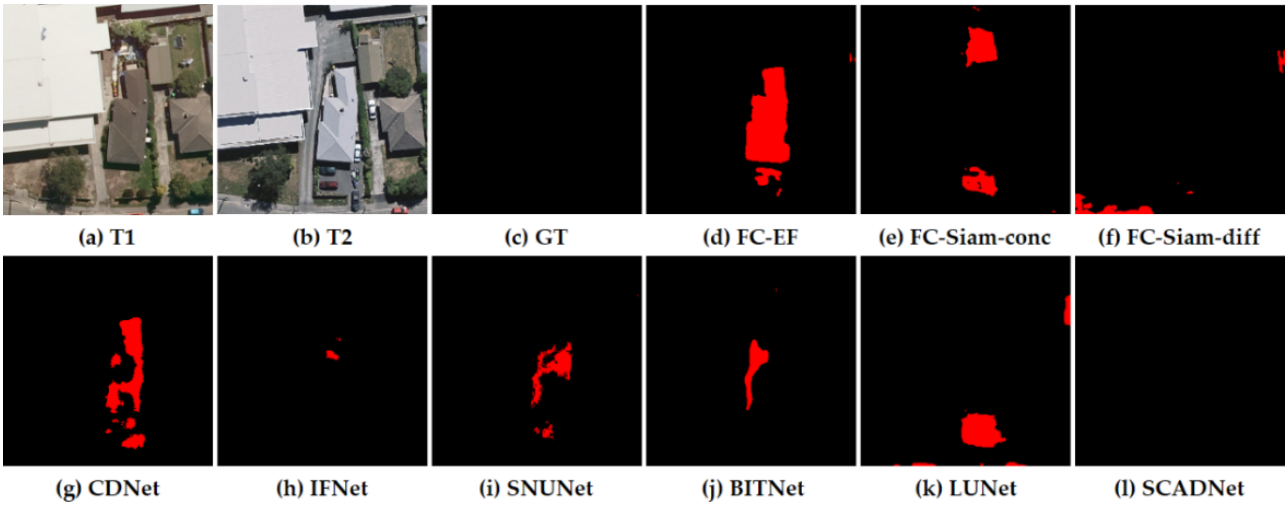

Figure 10. WHU-CD dataset comparison results visualization for the first image. TP is visualized in white, TN is visualized in black, FP is visualized in red, and FN is visualized in green.

The T1 and T2 images in Figure 11 have changed drastically, but there are only two building changes, one of which is a large building change. All the methods were able to detect these two building change areas; however, the FC-EF, FC-Siam-conc, FC-Siam-diff, BITNet, and LUNet all incorrectly identified the parking space as a changed building. Moreover, the IFNet missed part of the change area, while our SCADNet completed the BCD to near perfection.

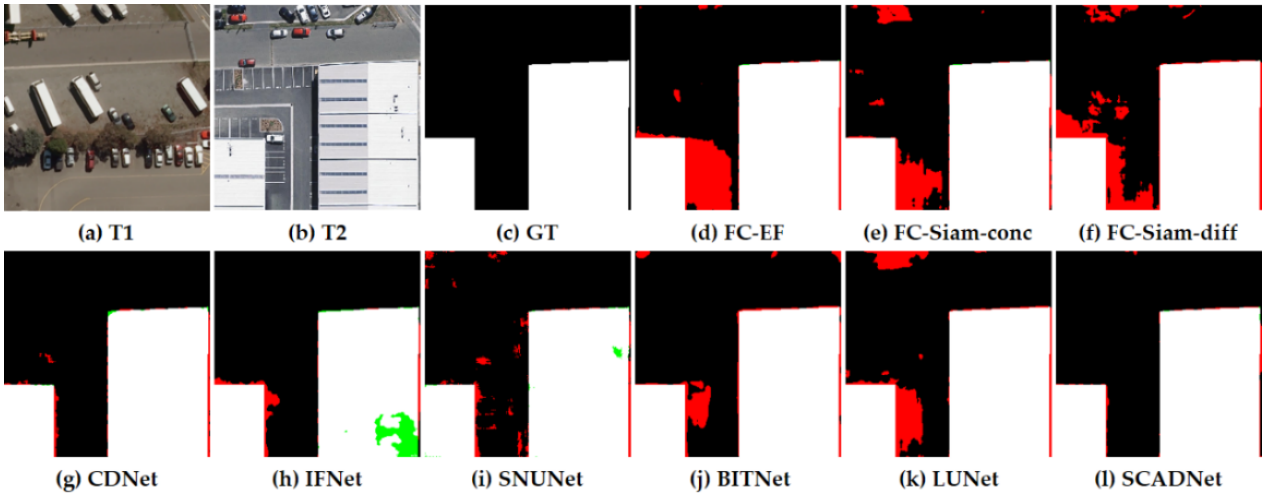

Figure 11. WHU-CD dataset comparison results visualization for the second image. TP is visualized in white, TN is visualized in black, FP is visualized in red, and FN is visualized in green.

3.5. Quantitative Results

We selected 100 images from the LEVIR-CD dataset for a quantitative evaluation and comparison. Figure 12 shows the comparison of various methods across eight metrics.

The SCADNet presented a remarkable superiority in terms of the F1 score, mIOU, IOU_0, IOU_1, OA, and kappa metrics. As far as the F1 score is concerned, our method was significantly better than the others. The SCADNet also achieved relatively positive results for the precision metric (on par with SNUNet). Finally, the LUNet outperformed the SCADNet by only a slim margin in the recall metric.

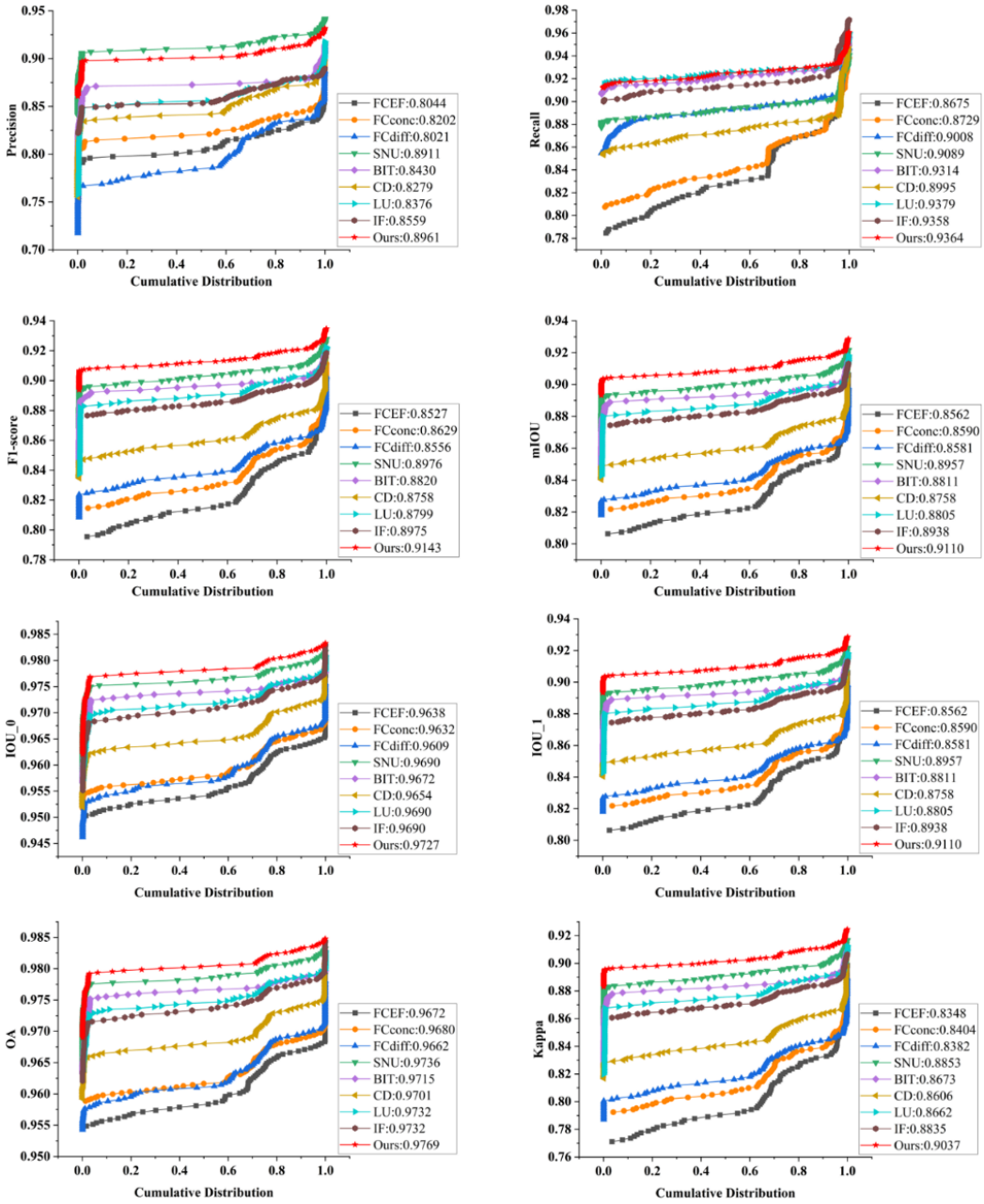

Figure 12. Quantitative comparisons of eight metrics, including Precision, Recall, F1 score, mIOU, IOU_0, IOU_1, OA, and kappa, on 100 images from the LEVIR-CD dataset. A point (x, y) on the curve indicates there are 100 * x percent of images with metric values not exceeding y.

3.6. Computational Efficiency Experiment

We employed two metrics, the number of parameters (Params) and floating points of the operations (FLOPs) to further compare the efficiency of the model of the eight comparative methods; note that the smaller the number of model's Params and FLOPs, the lower the model's complexity and computational cost. For each method, we gave two images of size $1 \times 3 \times 256 \times 256$ as the inputs, and the computational efficiency results are shown in Table 6. Because of the simple network architecture, the Params and FLOPS values for the FC-based method and CDNet were relatively low. Due to the deep layer of the networks and the multi-scale feature fusion strategy, both the IFNet and SCADNet generated a large number of Params, but our FLOPs were still lower than IFNet.

Table 6. Analysis of the computational efficiency of various methods.

Method	Params(M)	FLOPs(G)
FC-EF	1.35	3.56
FC-Siam-conc	1.54	5.31
FC-Siam-diff	1.35	4.71
CDNet	1.43	23.45
IFNet	35.99	82.26
SNUNet	27.06	123.11
BITNet	3.01	8.48
LUNet	8.45	17.33
SCADNet(ours)	66.94	70.72

4. Discussion

Based on three public BCD datasets, the SCADNet was comprehensively evaluated. The SCADNet was further evaluated quantitatively and qualitatively against several popular methods to demonstrate its superiority. This is mainly due to the following three aspects: first, the conventional methods focus primarily on the changed regions of the bitemporal images. As a result, if the background of the bitemporal images changes significantly, it will cause a large area of false detection. Our SCA module is able to differentiate the changed and the unchanged regions in the bitemporal images and extracts the actual changed feature information accurately for an improved BCD performance. Second, due to the different number of channels and the degree of representation, the multi-scale feature information extracted by direct fusion will result in the redundancy of unimportant information and the loss of key information. To reduce the error detection rates, the MFF module gradually integrates the multi-scale feature information with rich semantic information and recovers the original feature information of the image to the maximum extent possible. Third, in contrast to the traditional method of directly generating BCD results, our DCD module can jointly consider the predicted image and the difference image to calculate the probability loss. This allows us to provide a more detailed analysis of the building changes, thus reducing the missed detection rates.

Furthermore, we will analyze our SCA module by visualizing the attention maps. For the BCD, it is imperative to identify not only the changed buildings but also the unchanged environments. As a Siamese neural network consists of multiple layers, the downsampling results in the loss of local details. The abundant detail information in the image can be perceived by adding an attention mechanism, thus alleviating the above problem. In addition, the cross-attention mechanism can also effectively focus on the similarities and differences between the pixel points in the bitemporal images, enhancing the neural network's attention to tiny buildings in the images, reducing the noise by correlating the bitemporal images, clarifying the edges of the changes, and completing the CD by connecting the two streams of cross-attention information. As shown in Figure 13, we provide several examples of how our method visualizes attention maps at different stages of the SCADNet. Blue indicates lower attention values, while red indicates higher attention values.

We performed the attention map visualization operations on all three of our datasets and took two images from each dataset as a display. There are two parts to our cross-attention mechanism, invariant attention, and changing attention channels.

There are a number of changes in dense, tiny buildings in the first image of the LEVIR-CD dataset, and our attention channels focus on the environmental information that has not changed in the before and after time series, as well as the regions where there are buildings changes. In Stage 1, the change attention channel only detects a small portion of the changed buildings, and the color is not particularly red; however, as the network layers deepen, our change attention channel gradually becomes more focused on the changed areas. Until Stage 4, our changing attention channel peaks for the changed region and shows a dark blue color for the content of the unchanged region. At the same time, the

invariant attention channel also reaches its maximum attention in the unchanged regions. In the second scene of LEVIR-CD, the attention module does not only focus on changes in large buildings in the middle region, but also on changes in buildings on the left all the time.

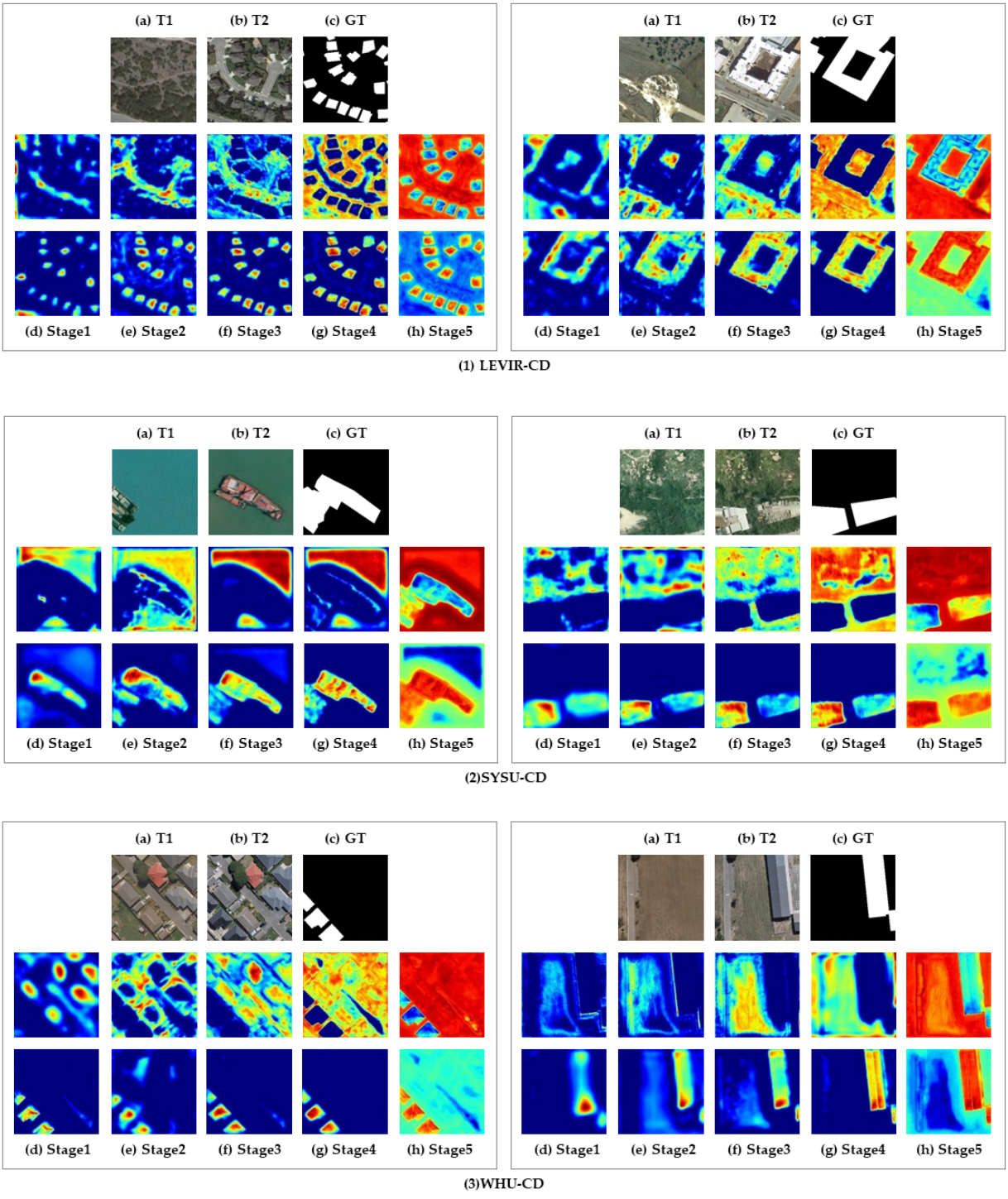

Figure 13. Attention map visualization in LEVIR-CD, SYSU-CD, and WHU-CD datasets. Blue indicates lower attention values and red indicates higher attention values.

For the SYSU-CD dataset, with an increasing number of network layers, the invariant attention channel focuses more on non-building change information, such as the ocean and

vegetation, etc. The changing attention channel focuses on ship information in the first picture and the emerging buildings on both sides of the mountain in the second picture.

As part of the WHU-CD dataset, we also collected two scenarios, which show regular building changes. Our two-channel attention is still able to better distinguish between unchanged and changed regions, especially in the second scenario, where the invariant attention channel concentrates most of its attention on the unchanged regions, whereas the changing attention channel focuses only on regions where building changes occur.

According to the visualization results, our network's attention mechanism module efficiently captures the high-level semantic information needed for BCD, and this high-level semantic information serves as strong data support. Using two-channel attention, the SCADNet is capable of focusing on its own region of interest, resulting in a superior accuracy.

5. Conclusions

A novel BCD method called the SCADNet was proposed in our study. An SCA module was used to identify the changed and unchanged regions in the bitemporal images. An MFF module was proposed in order to fuse the multi-scale feature information and reduce the key information loss during the feature map fusion process. To distinguish whether the change information in the extracted feature maps is a pseudo-change, we applied the DCD module to filter out the regions where real changes occur. The experimental results demonstrate that the SCADNet is superior to other methods using the LEVIR-CD, SYSU-CD, and WHU-CD datasets. The F1 score on the three datasets above can reach 90.32%, 81.79%, and 88.62%, respectively.

In this study, we conducted experiments on only three datasets, which do not effectively represent the generalization of the SCADNet, and subsequent experiments can be conducted on more remote sensing image CD datasets.

The combination of CD and semantic segmentation will be further investigated in the future. Additionally, the proposed method is based on an extensive collection of annotated samples, which is essential for supervised learning, in order to reduce the dependence of the CD methods on high-quality datasets. Therefore, we intend to conduct future research using a combination of the unsupervised methods and CD.

Author Contributions: Conceptualization, C.X., Z.Y., L.M. and W.Y.; methodology, Z.Y.; software, S.S. (Shaohua Sun); validation, L.M., S.S. (Sen Shen), and Q.Z.; formal analysis, W.Y.; investigation, L.M.; data curation, S.S. (Sen Shen); writing—original draft preparation, Z.Y.; writing—review and editing, C.X.; visualization, H.S.; supervision, C.X., W.Y. and H.S.; project administration, C.X.; funding acquisition, C.X. and W.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China (Nos. 41601443); Scientific Research Foundation for Doctoral Program of Hubei University of Technology (BSQD2020056); Guangxi Science and Technology Major Project (AA22068072); Science and Technology Research Project of Education Department of Hubei Province (B2021351); Natural Science Foundation of Hubei Province (2022CFB501).

Data Availability Statement: Not applicable.

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

References

1. Qin, R.; Tian, J.; Reinartz, P. 3D change detection—Approaches and applications. *ISPRS J. Photogramm. Remote Sens.* **2016**, *122*, 41–56. [CrossRef]
2. Xue, J.; Xu, H.; Yang, H.; Wang, B.; Wu, P.; Choi, J.; Cai, L.; Wu, Y. Multi-Feature Enhanced Building Change Detection Based on Semantic Information Guidance. *Remote Sens.* **2021**, *13*, 4171. [CrossRef]
3. Song, L.; Xia, M.; Jin, J.; Qian, M.; Zhang, Y. SUACDNet: Attentional change detection network based on siamese U-shaped structure. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *105*, 102597. [CrossRef]

4. Chen, Z.; Zhou, Y.; Wang, B.; Xu, X.; He, N.; Jin, S.; Jin, S. EGDE-Net: A building change detection method for high-resolution remote sensing imagery based on edge guidance and differential enhancement. *ISPRS J. Photogramm. Remote Sens.* **2022**, *191*, 203–222. [CrossRef]
5. Islam, M.A.; Jia, S.; Bruce, N.D. How much position information do convolutional neural networks encode? *arXiv* **2020**, arXiv:2001.08248.
6. Sefrin, O.; Riese, F.M.; Keller, S. Deep learning for land cover change detection. *Remote Sens.* **2020**, *13*, 78. [CrossRef]
7. Vivekananda, G.; Swathi, R.; Sujith, A. Multi-temporal image analysis for LULC classification and change detection. *Eur. J. Remote Sens.* **2021**, *54*, 189–199. [CrossRef]
8. Wang, H.; Lv, X.; Zhang, K.; Guo, B. Building Change Detection Based on 3D Co-Segmentation Using Satellite Stereo Imagery. *Remote Sens.* **2022**, *14*, 628. [CrossRef]
9. Zhang, Z.; Li, Z.; Tian, X. Vegetation change detection research of Dunhuang city based on GF-1 data. *Int. J. Coal Sci. Technol.* **2018**, *5*, 105–111. [CrossRef]
10. Bruzzone, L.; Prieto, D.F. Automatic analysis of the difference image for unsupervised change detection. *IEEE Trans. Geosci. Remote Sens.* **2000**, *38*, 1171–1182. [CrossRef]
11. 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]
12. Nemmour, H.; Chibani, Y. Multiple support vector machines for land cover change detection: An application for mapping urban extensions. *ISPRS J. Photogramm. Remote Sens.* **2006**, *61*, 125–133. [CrossRef]
13. Li, P.; Xu, H. Land-cover change detection using one-class support vector machine. *Photogramm. Engineer. Remote Sens.* **2010**, *76*, 255–263. [CrossRef]
14. Seo, D.K.; Kim, Y.H.; Eo, Y.D.; Park, W.Y.; Park, H.C. Generation of radiometric, phenological normalized image based on random forest regression for change detection. *Remote Sens.* **2017**, *9*, 1163. [CrossRef]
15. Ke, L.; Lin, Y.; Zeng, Z.; Zhang, L.; Meng, L. Adaptive change detection with significance test. *IEEE Access.* **2018**, *6*, 27442–27450. [CrossRef]
16. Hay, G.J.; Niemann, K.O. Visualizing 3-D texture: A three-dimensional structural approach to model forest texture. *Can. J. Remote Sens.* **1994**, *20*, 90–101.
17. Jabari, S.; Rezaee, M.; Fathollahi, F.; Zhang, Y. Multispectral change detection using multivariate Kullback-Leibler distance. *ISPRS J. Photogramm. Remote Sens.* **2019**, *147*, 163–177. [CrossRef]
18. Huang, X.; Cao, Y.; Li, J. An automatic change detection method for monitoring newly constructed building areas using time-series multi-view high-resolution optical satellite images. *Remote Sens. Environ.* **2020**, *244*, 111802. [CrossRef]
19. Javed, A.; Jung, S.; Lee, W.H.; Han, Y. Object-based building change detection by fusing pixel-level change detection results generated from morphological building index. *Remote Sens.* **2020**, *12*, 2952. [CrossRef]
20. Guo, X.; Meng, L.; Mei, L.; Weng, Y.; Tong, H. Multi-focus image fusion with Siamese self-attention network. *IET Image Process.* **2020**, *14*, 1339–1346. [CrossRef]
21. Zhu, Q.; Guo, X.; Deng, W.; Guan, Q.; Zhong, Y.; Zhang, L.; Li, D. Land-use/land-cover change detection based on a Siamese global learning framework for high spatial resolution remote sensing imagery. *ISPRS J. Photogramm. Remote Sens.* **2022**, *184*, 63–78. [CrossRef]
22. Bai, B.; Fu, W.; Lu, T.; Li, S. Edge-guided recurrent convolutional neural network for multitemporal remote sensing image building change detection. *IEEE Trans. Geosci. Remote Sen.* **2021**, *60*, 1–13. [CrossRef]
23. Gao, Y.; Gao, F.; Dong, J.; Wang, S. Change detection from synthetic aperture radar images based on channel weighting-based deep cascade network. *IEEE J. Sel. Top. Appl. Earth Observ.* **2019**, *12*, 4517–4529. [CrossRef]
24. Kang, M.; Baek, J. Sar image change detection via multiple-window processing with structural similarity. *Sensors* **2021**, *21*, 6645. [CrossRef]
25. Dong, H.; Ma, W.; Jiao, L.; Liu, F.; Shang, R.; Li, Y.; Bai, J. *A Contrastive Learning Transformer for Change Detection in High-Resolution Sar Images*; SSRN 4169439; SSRN: Rochester, NY, USA, 2022.
26. Lei, Y.; Liu, X.; Shi, J.; Lei, C.; Wang, J. Multiscale superpixel segmentation with deep features for change detection. *IEEE Access.* **2019**, *7*, 36600–36616. [CrossRef]
27. Dong, H.; Ma, W.; Wu, Y.; Zhang, J.; Jiao, L. Self-supervised representation learning for remote sensing image change detection based on temporal prediction. *Remote Sens.* **2020**, *12*, 1868. [CrossRef]
28. Lv, Z.; Liu, T.; Shi, C.; Benediktsson, J.A.; Du, H. Novel land cover change detection method based on K-means clustering and adaptive majority voting using bitemporal remote sensing images. *IEEE Access.* **2019**, *7*, 34425–34437. [CrossRef]
29. Chen, Y.; Bruzzone, L. Self-supervised Remote Sensing Images Change Detection at Pixel-level. *arXiv* **2021**, arXiv:2105.08501.
30. Chen, H.; Shi, Z. A spatial-temporal attention-based method and a new dataset for remote sensing image change detection. *Remote Sens.* **2020**, *12*, 1662. [CrossRef]
31. Wang, Z.; Peng, C.; Zhang, Y.; Wang, N.; Luo, L. Fully convolutional siamese networks based change detection for optical aerial images with focal contrastive loss. *Neurocomputing* **2021**, *457*, 155–167. [CrossRef]
32. Zhan, Y.; Fu, K.; Yan, M.; Sun, X.; Wang, H.; Qiu, X. Change detection based on deep siamese convolutional network for optical aerial images. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1845–1849. [CrossRef]

33. Chen, J.; Yuan, Z.; Peng, J.; Chen, L.; Huang, H.; Zhu, J.; Liu, Y.; Li, H. DASNet: Dual attentive fully convolutional Siamese networks for change detection in high-resolution satellite images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *14*, 1194–1206. [CrossRef]
34. Mi, L.; Chen, Z. Superpixel-enhanced deep neural forest for remote sensing image semantic segmentation. *ISPRS J. Photogramm. Remote Sens.* **2020**, *159*, 140–152. [CrossRef]
35. Wang, H.; Cao, P.; Wang, J.; Zaiane, O.R. Uctransnet: Rethinking the skip connections in u-net from a channel-wise perspective with transformer. In Proceedings of the 36th AAAI Conference on Artificial Intelligence, virtual, 22 February–1 March 2022; pp. 2441–2449.
36. Ulyanov, D.; Vedaldi, A.; Lempitsky, V. Instance normalization: The missing ingredient for fast stylization. *arXiv* **2016**, arXiv:1607.08022.
37. West, R.D.; Riley, R.M. Polarmetric interferomtric SAR change detection discrimination. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 3091–3104. [CrossRef]
38. Mei, L.; Yu, Y.; Shen, H.; Weng, Y.; Liu, Y.; Wang, D.; Liu, S.; Zhou, F.; Lei, C. Adversarial Multiscale Feature Learning Framework for Overlapping Chromosome Segmentation. *Entropy* **2022**, *24*, 522. [CrossRef] [PubMed]
39. Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv* **2015**, arXiv:1511.06434.
40. Mao, X.; Li, Q.; Xie, H.; Lau, R.Y.K.; Wang, Z.; Smolley, S.P. The Least squares generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2813–2821.
41. Shi, Q.; Liu, M.; Li, S.; Liu, X.; Wang, F.; Zhang, L. A deeply supervised attention metric-based network and an open aerial image dataset for remote sensing change detection. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–16. [CrossRef]
42. Ji, S.; Wei, S.; Lu, M. Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set. *IEEE Trans. Geosci. Remote Sens.* **2018**, *57*, 574–586. [CrossRef]
43. Daudt, R.C.; Le Saux, B.; Boulch, A. Fully convolutional Siamese networks for change detection. In Proceedings of the 25th IEEE International Conference on Image Processing, Athens, Greece, 7–10 October 2018; pp. 4063–4067.
44. Alcantarilla, P.F.; Stent, S.; Ros, G.; Arroyo, R.; Gherardi, R. Street-view change detection with deconvolutional networks. *Auton. Robots* **2018**, *42*, 1301–1322. [CrossRef]
45. Zhang, C.; Yue, P.; Tapete, D.; Jiang, L.; Shangguan, B.; Huang, L.; Liu, G. A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images. *ISPRS J. Photogramm. Remote Sens.* **2020**, *166*, 183–200. [CrossRef]
46. Fang, S.; Li, K.; Shao, J.; Li, Z. SNUNet-CD: A densely connected Siamese network for change detection of VHR images. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [CrossRef]
47. Chen, H.; Qi, Z.; Shi, Z. Remote sensing image change detection with transformers. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–14. [CrossRef]
48. Papadomanolaki, M.; Vakalopoulou, M.; Karantzalos, K. A deep multitask learning framework coupling semantic segmentation and fully convolutional LSTM networks for urban change detection. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 7651–7668. [CrossRef]
49. Wang, X.; Du, J.; Tan, K.; Ding, J.; Liu, Z.; Pan, C.; Han, B. A high-resolution feature difference attention network for the application of building change detection. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *112*, 102950. [CrossRef]