Adaptive Spatial and Difference Learning for Change Detection

Yangguang Liu, Fang Liu, *Member, IEEE*, Jia Liu, *Member, IEEE*, Xu Tang, *Senior Member, IEEE*, and Liang Xiao, *Member, IEEE*

***Abstract*—Change detection refers to revealing the surface changes from multitemporal images of a given scene, where changed regions of different size and shape may appear anywhere. Convolutional neural network, as one of the most widely used deep learning architectures, has shown good performance in change detection task recently. However, due to multiple convolution and pooling operators in deep architectures, it often happens that small changes are missed and sharp edges are blurred. In this article, an adaptive spatial and difference learning method is proposed to tackle this problem, which mainly contains a feature-adapted difference learning (FADL) module and a difference recalibration (DR) module. In FADL, a kernel-adapted spatial learning part is constructed to capture varying spatial information, which selects proper kernels to adapt different shape and size of changed regions; a joint feature refinement part is employed to learn comprehensive features between bitemporal data, which are further enhanced by difference information. In addition, DR with shallow features is designed to recalibrate spatial and difference details further. From the experiments on three public datasets, our proposed method can relieve the spatial ambiguity problem and obtains the optimal results among the compared change detection methods.**

***Index Terms*—Adaptive spatial, change detection, difference recalibration (DR), feature-adapted difference learning (FADL).**

I. Introduction

CHANGE detection is an increasingly important technique to identify changed and unchanged regions by analyzing multitemporal images acquired at different times. It is widely used in urban planning [1], [2], disaster assessment [3], forest monitoring [4], [5], and environmental monitoring [6].

In recent years, with the development of satellite sensors, the temporal and spatial resolution has gradually improved, providing a rich source of data for the development of change detection. High-resolution images contain more spatial information and more detailed surface information, which is beneficial for the development of change detection. At the same time, it also brings many new challenges due to the increased variability within ground objects caused by the increased resolution. First of all, because the images taken in different periods cause imaging differences due to light changes and seasonal changes, the first difficulty in change detection is how to eliminate the effect of "nonsemantic changes." Second, in the change detection task, since the size variance between the unchanged and changed regions is usually large, it is necessary to eliminate class imbalance problem. Third, the sizes of changed regions in different images are different, and how to adapt to size of changes still needs to be considered more deeply. Therefore, how to reduce "nonsemantic changes," extract useful features from remote sensing images, and accurately detect changes of interest is an important problem to be solved in the task of change detection.

Due to the importance of change detection techniques in many applications, many methods have been proposed in the field of change detection in recent years to meet different needs. The simplest of these is the pixel-based method, which takes a single pixel as the smallest unit of image processing and generates a difference image (DI) by comparing spectral information between pixel pairs of bitemporal images. Finally, the final change map (CM) is generated from the DI via image segmentation. Pixel-based methods include image difference [7], regression analysis [8], change vector analysis (CVA) [9], principal component analysis (PCA) [10], [11], independent component analysis [12], and Kauth–Thomas transformation [13]. The object-based method takes the set of pixels as the smallest unit of image processing, namely the image object, which segments the image into disjoint objects based on their shape, spectral and texture features, and compares the classification results to obtain the CM [14]. Moreover, compared to pixel-based change detection that focuses on pixel-level changes and object-based change detection that focuses on object-level changes, scene change detection [15], [16] takes into account the changes of the whole scene in a comprehensive manner. It compares and analyzes multitemporal remote sensing images to determine the type, scale, and spatial distribution of changes.

Manuscript received 6 April 2023; revised 29 June 2023; accepted 14 July 2023. Date of publication 24 July 2023; date of current version 15 August 2023. This work was supported in part by the National Nature Science Foundations of China under Grant 61802190, Grant 62276133, Grant 62171332, Grant 61871226, Grant 62071233, and Grant 62001226), in part by the Fundamental Research Funds for the Central Universities under Grant JSGP202101 and Grant JSGP202204, in part by the Nature Science Foundations of Jiangsu Province, China under Grant BK20190451, and in part by the Open Research Fund Program of LIESMARS Under Grant 22R04. *(Corresponding author: Fang Liu.)*

Yangguang Liu, Jia Liu, and Liang Xiao are with the Jiangsu Key Laboratory of Spectral Imaging & Intelligent Sense, the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, Jiangsu 210094, China (e-mail: 207099533@qq.com; omegaliuj@gmail.com; xiaoliang@mail.njust.edu.cn).

Fang Liu is with the Nanjing University of Science and Technology, Nanjing 210094, China, and also with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China (e-mail: fayliu77@163.com).

Xu Tang is with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, Xidian University, Xi'an, Shaanxi 710071, China (e-mail: tangxu128@gmail.com).

Digital Object Identifier 10.1109/JSTARS.2023.3298097

As deep learning techniques emerge, the application of convolutional neural network (CNN)-led deep learning techniques in the field of change detection becomes popular. Compared with traditional models, CNN-based models have powerful nonlinear mapping capabilities, enabling them to capture detailed image information and complex texture features. According to the fusion strategy, CNN-based change detection networks are divided into early fusion [17], [18], [19], [20] and late fusion [21], [22], [23], [24] networks. The early fusion networks input multitemporal images as a whole to the deep CNN. The late fusion networks use Siamese networks to extract bitemporal features first, and then, fuse them.

While deep learning techniques are driving the rapid development of change detection, they are also facing some new problems. Since the receptive fields corresponding to different sizes of convolutional kernels are different, and the sizes of change regions in images usually vary widely, how to capture features of varying sizes is especially important in the change detection task. To address the aforementioned problems, many scholars have proposed different solutions. InceptionNets [25], [26], [27], [28] use multiple groups to capture multiscale features, each with a different receptive field. UNet [29] captures multiscale features by building a top-down architecture. Res2Net [30] extracts multiscale features by constructing hierarchical residual-style connections using rich-scale blocks within a single residual block. Furthermore, spatial and channel attention mechanisms [17], [31], [32] are proposed for feature refinement. Although these change detection methods have some effectiveness in extracting multiscale features, a common problem they face is that they cannot adaptively select the proper size convolution kernel according to the different size and shape of changed regions, which leads to small changes being missed and sharp edges being blurred. Since objects of different size and shape have different spatial information, how to choose proper convolution kernels according to the changed regions of different size and shape is still a problem to be considered.

To this end, we propose an adaptive spatial and difference learning method. It follows an encoder–decoder structure, where the encoder is used to capture multilevel features from bitemporal images. The feature-adapted difference learning (FADL) module is constructed to acquire adaptive spatial knowledge and perform difference learning. In detail, it selects proper convolutional kernels according to the size and shape of changed regions to capture the varying spatial information, which allows some small changed regions of spatial information to be retained. Then, difference learning is performed to make the changed regions of different size and shape to be enhanced in the comprehensive feature. At the same time, the distinguishability between the changed and unchanged regions is further enhanced by difference learning, which facilitates the formation of sharp edges. In addition, to better highlight the changed features, a difference recalibration (DR) module with shallow features is introduced in the decoder to further recalibrate the spatial and difference details. To summarize, the contributions of this article are described as follows.

1) Aiming at the problems of missing of small changes and blurring of sharp edges caused by traditional convolution kernels when dealing with changed regions of different size and shape, an adaptive spatial and difference learning method is proposed, which mainly contains FADL module and DR module.
2) In FADL, a kernel-adapted spatial learning (KASL) part is constructed to capture varying spatial information, which selects proper convolution kernels to adapt different size and shape of changed regions; a joint feature refinement (JFR) part is employed to learn comprehensive features between bitemporal data, which are further enhanced by difference information.
3) The proposed DR uses the precise location information of the shallow features to weight the deep difference features to further recalibrate the spatial and difference details.

The rest of this article is organized as follows. Section II reviews related work. Section III describes our proposed adaptive spatial and difference learning method in detail. Section IV conducts comparison experiments and ablation experiments on three public datasets. Section V discusses the model efficiency and the comparison between different loss functions. Finally, Section VI concludes this article.

II. Related Works

Remote sensing image change detection is usually divided into two processes. The first process is feature extraction of multitemporal images, and the other process is to identify changed regions in the images. The target of the first process is to extract meaningful features in multitemporal images, such as texture features and contextual information. The target of the second process is to detect changes of interest by analyzing the extracted multitemporal image features using change detection techniques. Depending on the change detection techniques used, they are divided into traditional change detection and deep learning-based change detection.

A. *Traditional Change Detection*

Traditional change detection methods have been extensively studied since the early stages of change detection. Depending on the size of the processing unit, traditional methods are usually divided into two categories: pixel based and object based. The pixel-based methods are mainly classified as arithmetic based, transformation based, and classification based. The simplest of these pixel-based methods are arithmetic-based methods, such as image difference [7]. Image transformation-based methods enhance spectral differences by mapping image spectral information into other spectral spaces. CVA [33] obtains the changes of the corresponding pixel pairs in each waveband by spectral measurement and calculation of the bitemporal images, forming change vectors and using threshold segmentation to obtain the final CM. PCA [34] first extracts the effective information of the bitemporal image frequency band to generate DI, then compares it to the first principal component to get the difference, and finally, obtains the final CM by image segmentation. Classification-based methods use existing image classification algorithms that first classify images, and then, perform change detection, like decision tree (DT) [35] and support vector machine (SVM) [36]. There are also some other algorithms. Multivariate change detection (MAD) [37] highlights the features of change by performing variance on the DI. Slow feature snalysis (SFA) [38] extracts time-invariant features from bitemporal images and improves the separability of changed pixels by suppressing the radiometric difference of unchanged pixels. Pixel-based methods are simple in principle and easy to implement, but they only focus on the spectral variation of single pixels and ignore the spatial context information, resulting in noisy outputs, isolated changing pixels, and jagged boundaries. The advantage of object-based methods over these pixel-based methods is that they exploit the spatial context information of bitemporal images. Chen et al. [39] propose a spatial contrast-enhanced change detection method based on image objects to identify change regions by shape differences between bitemporal images. In [40], a statistical object-based method for forest monitoring is proposed, which identifies changed objects by a chi-square test during an iterative trimming process. Although object-based methods have some advantages in delineating object boundaries, their accuracy is very sensitive to the segmentation algorithm [41] employed, which leads to instability in their detection results. Moreover, these traditional change detection methods rely on handcrafted features of remote sensing images and are susceptible to sensor and lighting conditions, which limits their application in the field of change detection.

B. *Deep Learning-Based Change Detection*

With the successful application of deep learning techniques in other vision fields, it has been gradually extended to the field of change detection and become a mainstream method in recent years. After full convolutional networks (FCNs) [42] are proposed, many scholars apply FCNs to change detection tasks. U-Net [43] is first proposed for change detection, and then, Siamese network is constructed for extracting bitemporal features. Due to the inherent advantages of Siamese networks in extracting bitemporal features, they are widely used in change detection tasks [29], [31], [44]. Daudt et al. [29] propose three different networks for the change detection task, namely FE-EF, FC-Siam-diff and FC-Siam-conc. FE-EF takes the bitemporal images as a whole, and then, inputs them into change detection networks. Both FC-Siam-diff and FC-Siam-conc construct Siamese networks with shared weights to extract bitemporal features, the difference being that the former fuses bitemporal information through feature differences, while the latter does so through feature concatenation. Since different layers of the CNN contain different information, the shallow layer contains precise location information and the deep layer contains rich semantic information, a feature fusion strategy is proposed for combining multilevel features of images. Peng et al. [45] propose U-Net++, which adopts a multiside output fusion strategy to combine CMs containing different semantic information to generate the final CM. In [32], Qian et al. combine multilevel features extracted from different convolution stages to adapt to scale changes of changing regions in the labels. The complexity of remote sensing image scenes is increased by factors such as sensors, lighting, and seasonal changes, so context modeling is important to identify real changes in the images. To expand the receptive field and capture more levels of features, scholars have proposed several strategies, including the use of deeper network models [31], [46], [47], the use of dilated convolutions [47], and attention mechanisms [17], [31], [46]. Zhang et al. [47] use deeper Siamese networks for feature extraction and multiple dilation convolutions with different rates to expand the receptive field. Since the attention mechanism can highlight the effective information in the features, it is widely introduced to refine the extracted bitemporal features, such as spatial attention [17], [48], [49], channel attention [17], [48], [49], [50], self-attention [46], and dual attention [31]. Chen et al. [31] use spatial-channel attention on the extracted features to emphasize the changed information in the images, resulting in a better feature representation. Chen and Shi [46] integrate the self-attention module into the network model to strengthen the spatiotemporal relationship of bitemporal features to generate more discriminative features. Although deep learning-based methods achieve good results due to their excellent modeling ability, traditional convolution kernels in CNNs reduce spatial details when dealing with changed regions of different size and shape, resulting in the missing of small changes and blurring of sharp edges.

III. Methodology

In this section, the overall framework of the proposed method is introduced, and detailed descriptions are given for the FADL and DR modules.

A. *Framework*

Fig. 1 shows the overall framework of the adaptive spatial and difference learning method. The proposed change detection method consists of encoder and decoder. Specifically, ResNet-18 with four residual blocks is extended to a Siam structure as the encoder part to extract shallow and deep features. T_1 and T_2 are bitemporal images that serve as inputs to the ResNet-18 network. The bitemporal features output by each residual block are used as the input of the FADL module. Consisting of KASL part and JFR part, the FADL module is used to acquire adaptive spatial knowledge and perform difference learning. Finally, the four multiscale difference features output by the four FADL modules are used as the input of the decoder.

The decoder mainly contains a multilayer perceptron (MLP), a classifier, and a DR module. The DR module further recalibrates spatial and difference details using shallow features. The classifier is used to generate the final DI. The weighted batch contrastive loss (WBCL) function is used to train the network model. Finally, the final CM is obtained from the DI by threshold segmentation.

B. *FADL Module*

The FADL module is used to acquire adaptive spatial knowledge and perform difference learning. In FADL, a KASL part is constructed to capture varying spatial information, which selects proper kernels to adapt different size and shape of changed

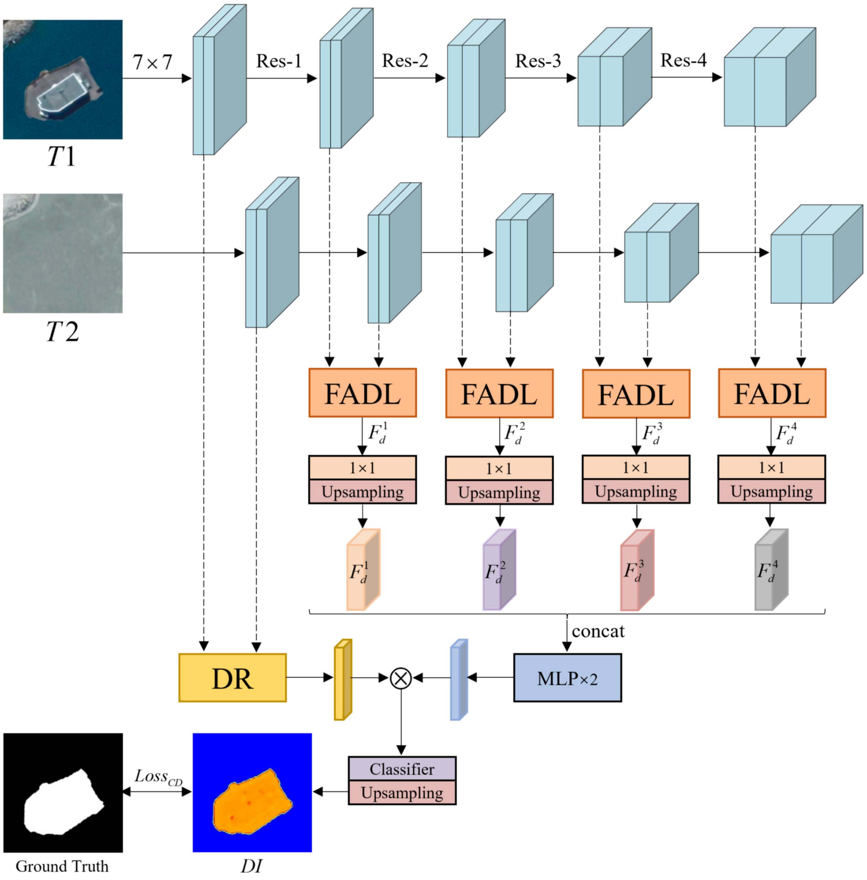


Fig. 1. Framework of adaptive spatial and difference learning.

regions; a JFR part is employed to learn comprehensive features between bitemporal data, which are further enhanced by difference information.

Fig. 2 shows the structure of FADL. Features F_1 and F_2 are bitemporal features obtained from each residual block in the ResNet-18 network. First, the features F_1 and F_2 are input to the KASL part to obtain the adaptive features A_1, B_1, and C_1 of the feature F_1, and the adaptive features A_2, B_2, and C_2 of the feature F_2, respectively.

$$\begin{aligned} A_1, B_1, C_1 &= \text{KASL}\left(F_1\right) \\ A_2, B_2, C_2 &= \text{KASL}\left(F_2\right). \end{aligned} \tag{1}$$

With KASL, the varying spatial information can be captured, which alleviates the situation of missing small changes due to the loss of useful information. Then, calculate the Euclidean distances of A_1 and A_2, B_1 and B_2, and C_1 and C_2, respectively, to get the DIs D_1, D_2, and D_3.

At the same time, F_1 and F_2 are input into the JFR part to get the refined feature F_3. Here, F_3 contains both changed and unchanged features, and the changed features are enhanced after refinement by JFR. F_3 is multiplied by D_1, D_2, and D_3, respectively, for difference learning to obtain D'_1, D'_2, and D'_3, as shown by the orange arrow in Fig. 2. Difference learning further enhances the changed regions of different size and shape in D'_1, D'_2, and D'_3, and also enhances the distinguishability between the changed and unchanged regions, which facilitates the formation of sharp edges. Concatenate D'_1, D'_2, and D'_3 in the channel dimension, and then, obtain F_d through two 1×1

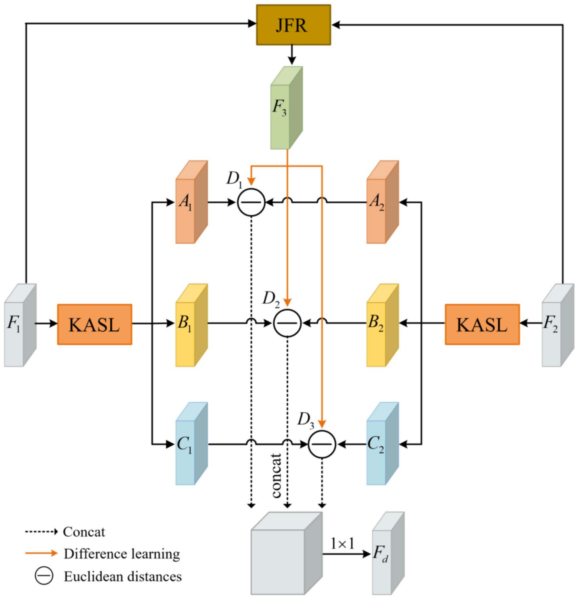


Fig. 2. FADL module.

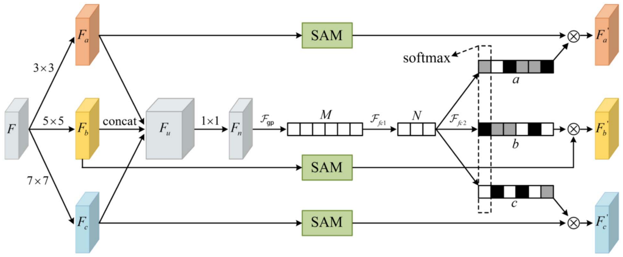


Fig. 3. KASL part.

convolutions, shown as follows:

$$F_d = \delta\left(\mathcal{B}\left(f^{1\times 1}\left(\delta\left(\mathcal{B}\left(f^{1\times 1}\left([D_1', D_2', D_3']\right)\right)\right)\right)\right)\right) \quad (2)$$

where δ denotes the ReLU function, $\mathcal{B}$ denotes the batch normalization, $f^{1\times 1}(\cdot)$ represents a convolution operation with 1×1, and $[\cdot]$ represents the concatenation operation. F_d is used as the output of FADL.

As illustrated in the framework shown in Fig. 1, FADL module is introduced in the encoder and follows each residual block of the ResNet-18 network. Four pairs of bitemporal features extracted by residual blocks are, respectively, put into the FADL module, and then, four multiscale difference features F_d^1, F_d^2, F_d^3, and F_d^4 are obtained and serve as the input of the decoder.

1) KASL Part: As shown in Fig. 3, the receptive fields corresponding to different sizes of convolution kernels are different, the extracted spatial information is also different when convolution operations are performed on the same object. In order to capture varying spatial information, the proposed KASL part selects proper convolution kernels to adapt different size and shape of changed regions, which is achieved by weighting the convolution kernels of different sizes.

Fig. 3 shows the structure of KASL. The feature F is obtained from each residual block. The $F \in \mathbb{R}^{C\times H\times W}$, where C, H, and W denote the channel, height, and width of F, respectively. F is passed through 3×3, 5×5 and 7×7 convolution layers to obtain three new sets of features with different spatial information, named F_a, F_b, and F_c, where $\{F_a, F_b, F_c\} \in \mathbb{R}^{C\times H\times W}$. In order to reduce the number of parameters, the traditional 5×5 and 7×7 convolutional kernels are replaced with 3×3 dilated convolutions with dilation sizes of 2 and 3, respectively. Concatenate F_a, F_b, and F_c to obtain the feature F_u, where $F_u \in \mathbb{R}^{3C\times H\times W}$. F_u contains the spatial information of F_a, F_b, and F_c. The feature F_n is obtained by passing F_u through two 1×1 convolutional layers, where $F_n \in \mathbb{R}^{C\times H\times W}$. Note that after each convolution operation, the batch normalization and ReLU functions are performed accordingly.

Then, global average pooling is performed on F_n to obtain global information to generate M, where $M \in \mathbb{R}^C$. N is obtained by M reducing its number of channels through the first fully connected layer, where $N \in \mathbb{R}^d$. d is the number of channels of N, expressed as

$$d = \max(C/r, L) \quad (3)$$

where r is the reduction rate and L represents the minimum value of d; in our experiment, $r = 16$, $L = 32$. Pass N through the second fully connected layer, increase its number of channels, and then, divide it into three blocks in the channel dimension to obtain A, B, and C, where $\{A, B, C\} \in \mathbb{R}^C$, shown as follows:

$$[A, B, C] = \mathcal{F}_{fc2}(N). \quad (4)$$

The softmax operations are performed at the corresponding positions of A, B, and C to obtain the weights of the branches of the convolution kernel of different sizes

$$a_i = \frac{e^{A_i}}{e^{A_i} + e^{B_i} + e^{C_i}}$$
$$b_i = \frac{e^{B_i}}{e^{A_i} + e^{B_i} + e^{C_i}}$$
$$c_i = \frac{e^{C_i}}{e^{A_i} + e^{B_i} + e^{C_i}} \quad (5)$$

where a, b, and c denote the attention weights of F_a, F_b, and F_c, respectively, a_i is the ith element of a, likewise b_i and c_i, and A_i is the ith row of A, likewise B_i and C_i. Note that $a_i + b_i + c_i = 1$. Attention weights a, b, and c can be adaptively adjusted to different size and shape of objects, allowing varying spatial information to be captured.

In addition to strengthen the attention to feature spatial information, F_a, F_b, and F_c are input to the spatial attention module (SAM) separately. Fig. 4 shows the structure of the SAM. The max pooling and average pooling operations are performed on the feature F, respectively, concatenated in the channel dimension, and then, the spatial attention weights are obtained by 1×1 convolution and sigmoid function. Finally, the original features F are multiplied with the spatial attention weights to obtain more discriminative features. The SAM process is as follows:

$$\mathrm{SAM}(F) = \sigma\left(f^{1\times 1}([\mathrm{AvgPool}(F); \mathrm{MaxPool}(F)])\right) * F \quad (6)$$

where σ represents the sigmoid function. After F_a passes through SAM, it is multiplied by a to get the adaptive spatial feature map F_a', likewise F_b and F_c, shown as follows:

$$F_a' = a * \mathrm{SAM}(F_a)$$
$$F_b' = b * \mathrm{SAM}(F_b)$$
$$F_c' = c * \mathrm{SAM}(F_c). \quad (7)$$

By applying different weights to the three convolutional branches, objects of different size and shape in F_a', F_b', and F_c' are made to be captured.

2) JFR Part: The JFR part aims to refine the combined bitemporal features, highlighting representative features through increased attention to space and channels.

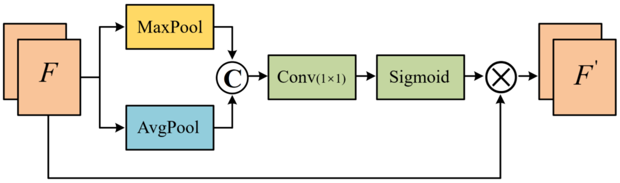


Fig. 4. SAM structure.

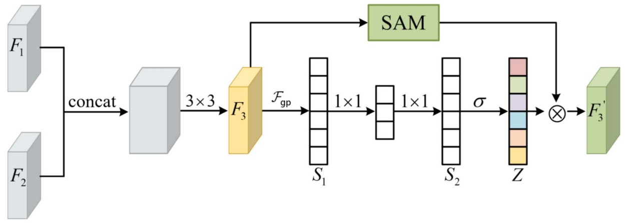


Fig. 5. JFR part.

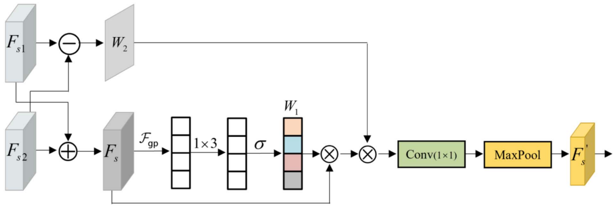


Fig. 6. DR module.

The structure of JFR is shown in Fig. 5. F_1 and F_2 represent bitemporal features acquired after each residual block, where $\{F_1, F_2\} \in \mathbb{R}^{C\times H\times W}$. F_1 and F_2 are concatenated in the channel dimension, and then, feature F_3 is obtained through a 3×3 convolutional layer, where $F_3 \in \mathbb{R}^{C\times H\times W}$. Here, the concatenation operation is used to preserve bitemporal information in order to prevent the loss of useful information. Since feature concatenation tends to cause information redundancy, the change features are subsequently enhanced by adding spatial as well as channel attention.

Global average pooling is performed on F_3 to obtain global information and generate channel-wise statistics S_1, where $S_1 \in \mathbb{R}^C$. S_1 is passed through two 1×1 convolutional layers to obtain S_2, where $S_2 \in \mathbb{R}^C$. The channel attention weight Z of feature F_3 is obtained by performing the sigmoid operation on S_2. Its process is as follows:

$$Z = \sigma\left(f^{1\times1}\left(\delta\left(\mathcal{B}\left(f^{1\times1}\left(\mathcal{F}_{\text{gp}}\left(F_3\right)\right)\right)\right)\right)\right) \tag{8}$$

where $\mathcal{F}_{\text{gp}}(\cdot)$ represents the global average pooling operation.

Finally, feature F_3 is passed through SAM to increase spatial attention, and then multiplied by Z to obtain F_3', where $F_3' \in \mathbb{R}^C$, shown as follows:

$$F_3' = Z * \text{SAM}\left(F_3\right). \tag{9}$$

Note that F_3' is the refined feature of F_1 and F_2 and is used as the output of the JFR part.

C. DR Module

The DR module uses the precise location information of the shallow features to further recalibrate the spatial and difference details of the deep difference features.

As shown in Fig. 1, before introducing the DR module, four multiscale difference features F_d^1, F_d^2, F_d^3, and F_d^4 are first processed. The number of their channels is adjusted to C_d by a 1×1 convolutional layer, where $C_d = 96$. Then, F_d^2, F_d^3, and F_d^4 are upsampled to the same size as F_d^1 by bilinear interpolation. The four processed multiscale difference features are concatenated, and then, put into two MLPs, adjusting the number of channels to C_m to obtain F_m, where $C_m = 32$, shown as follows:

$$F_m = \text{MLP}\left(\text{MLP}\left(\left[Up\left(f^{1\times1}\left(F_d^n\right)\right)\right]\right)\right), n = 1, 2, 3, 4 \tag{10}$$

where $Up(\cdot)$ represents the upsampling operation. MLP is composed of two convolutional layers with a kernel size of 3×3, with batch normalization and ReLU activation functions after each 3×3 convolutional operation.

Fig. 6 shows the structure of DR. Features F_{s1} and F_{s2} are obtained from the first 7×7 convolutional layer of ResNet-18. The fusion feature F_s is obtained by element-wise summation of F_{s1} and F_{s2}. Global average pooling is executed for F_s, and then, the channel attention weights W_1 of the feature F_s are obtained by 1×3 1-D convolution and sigmoid function, shown as follows:

$$W_1 = \sigma\left(f^{1\times3}\left(\mathcal{F}_{gp}\left(F_{s1} + F_{s2}\right)\right)\right). \tag{11}$$

The output W_1 learns features of interest in shallow features. Calculate the Euclidean distance of F_{s1} and F_{s2} to obtain W_2 as follows:

$$W_2 = E\left(F_{s1}, F_{s2}\right) \tag{12}$$

where $E(\cdot)$ represents Euclidean distance and W_2 learns difference information in shallow features. W_1 and W_2 are multiplied with F_s and then F_s' is obtained by 1×1 convolutional layer and max pooling, shown as follows:

$$F_s' = \text{MaxPool}\left(f^{1\times1}\left(F_s * W_1 * W_2\right)\right). \tag{13}$$

The weighting of W_1 and W_2 makes the difference information in the shallow features enhanced. Finally, the shallow feature F_s' and the deep difference feature F_m are multiplied to get $F_{\text{i n}}$, which acts as the input of the classifier. By using the precise location information of the shallow features, it makes the difference details in the deep difference features complementary, and also further enhances the distinguishability between the changed and unchanged regions.

D. Classifier and Loss Function

The classifier is used to produce a one-channel DI as the final result. The classifier is composed of two convolutional layers, one with a kernel size of 3×3 and the other with 1×1. After F_{in} passes the classifier, it is upsampled to a size of 256×256 to obtain the DI F_{out}. The final CM is obtained by performing threshold segmentation on the F_{out}.

Because of the class imbalance between the changed and unchanged regions, the WBCL is used as the loss function to minimize the distance between the unchanged regions and maximize the distance between the changed regions. The expression of the loss function is as follows:

$$\begin{aligned}\text{Loss}_{\text{CD}} = \sum_{i,j=0}^{M} \frac{1}{2}\Big[&x_1\left(1 - gt_{i,j}\right) dt_{i,j}^2 \\ &+ x_2 gt_{i,j} \max\left(m - dt_{i,j}, 0\right)^2\Big]\end{aligned} \tag{14}$$

$$x_1 = \frac{1}{p_u}, x_2 = \frac{1}{p_c} \tag{15}$$

where M denotes the size of dt; x_1 and x_2 denote the weights of unchanged pixels and changed pixels, respectively; $gt_{(i,j)}$ and $dt_{(i,j)}$ denote the ground truth and DI values at point (i, j), where $i, j \in [0, M)$; m denotes the margin, which is used to filter pixels whose value exceeds it and is set to 1.5; and p_u and p_c denote unchanged and changed pixel counts, respectively.

IV. Experiments

In this section, three public datasets, evaluation metrics, some settings of the experiments, and the eight comparison methods chosen are first presented. Then, the experimental analysis is performed.

A. Datasets

To validate the effectiveness of our proposed change detection method, experiments are conducted on three public change detection datasets: synthetic images and real season-varying change detection dataset (CDD) [51], Sun Yat-sen University change detection dataset (SYSU-CD) [32], and Learning Vision and Remote Sensing Laboratory building change detection dataset (LEVIR-CD) [46].

1) CDD Dataset: The CDD dataset is taken by Google Earth and consists of 11 pairs of multispectral (composed of R, G, and B) images, of which seven pairs have a size of 4725 × 2200 pixels and four pairs have a size of 1900 × 1000 pixels. Its spatial resolution ranges from 3 to 100 cm/pixel. The CDD dataset contains change information of objects such as buildings, cars, and roads. This places a higher demand on the change detection algorithm by ignoring the changes caused by seasonal factors. The 11 pairs of original images are rotated and cropped to obtain 16000 pairs of 256 × 256 pixel images, which are classified into 10 000/3000/3000 pairs of images for training, validation, and testing, respectively.

2) SYSU-CD Dataset: The SYSU-CD dataset consists of 20 000 pairs of 256 × 256 pixel aerial images taken in Hong Kong between 2007 and 2014. Its spatial resolution is 0.5 m and the band number is 3. The dataset contains change information of objects such as vegetation, buildings, roads, water, and ships. In our experiments, the 20 000 pairs of images in the SYSU-CD dataset are classified into 12 000/4000/4000 pairs of images for training, validation, and testing, respectively.

3) LEVIR-CD Dataset: The LEVIR-CD dataset consists of 637 pairs of 1024 × 1024 pixel high-resolution images. The dataset is taken by Google Earth and has a spatial resolution of 0.5 m. The images, taken between 2002 and 2018, show land-use change in 20 different areas across multiple cities in Texas, USA. It focuses on buildings and contains information on the changes in buildings such as high-rise apartments, villa houses, large warehouses, and small garages. In our experiments, images of 1024 × 1024 pixels are cropped into 256 × 256 pixel nonoverlapping patches, resulting in 7120/1024/2048 pairs of images that are used for training, validation, and testing, respectively.

B. Evaluation Metrics

To validate the performance of the proposed method, we employ four evaluation metrics, namely precision (Pre), recall (Rec), F1-score (F1), and intersection over union (IoU) to evaluate the change detection results. The expressions of Pre, Rec, F1, and IoU are shown as follows:

$$\begin{aligned} \text{Pre} &= \frac{\text{TP}}{\text{TP}+\text{FP}} \\ \text{Rec} &= \frac{\text{TP}}{\text{TP}+\text{FN}} \\ F1 &= \frac{2\text{Pre}\cdot\text{Rec}}{\text{Pre}+\text{Rec}} \\ \text{IoU} &= \frac{\text{TP}}{\text{TP}+\text{FP}+\text{FN}} \end{aligned} \tag{16}$$

where TP is true positive, FP is false positive, and FN is false negative, and they denote the number of pixels correctly predicted as changed, incorrectly predicted as changed, and incorrectly predicted as unchanged, respectively. Notably, we focus more on F1 and IoU values in the experiments since they better reflect the performance of the model.

C. Implementation Details

Our proposed change detection model is implemented under the Pytoch framework, using an NVIDIA Titan V for training and testing. In the training phase, some important parameters in the experiments are set as follows: Adam [52] is used as the optimizer, the learning rate is setup to 0.0001 and the batch size is setup to 8. The same training and testing settings are adopted for all datasets. After each training epoch, the IoU values are computed on the validation set. After training is completed, the model which has the highest IoU value is selected to evaluate the test set.

D. Comparative Methods

To validate the effectiveness of our method, eight representative change detection algorithms are selected for comparison. Namely, fully convolutional Siamese-concatenation (FC-Siam-Conc) [29], fully convolutional Siamese-difference (FC-Siam-Diff) [29], super-resolution-based change detection network (SRCDNet) [53], deeply supervised attention metric-based network (DSAMNet) [32], bitemporal image transformer (BIT) [54], CNN-transformer network with multiscale context aggregation (MSCANet) [55], difference-enhancement dense-attention convolutional neural network (DDCNN) [17], and dual-branch multilevel intertemporal network (DMINet) [56]. A brief description of the aforementioned eight algorithms is given as follows:

1) FC-Siam-Conc [29]: A Siamese structure-based fully convolutional network that extracts multilevel features of bitemporal images and fuses bitemporal information using feature concatenation.

2) FC-Siam-Diff [29]: FC-Siam-Diff has the same feature extractor as FC-Siam-Conc, with the difference that it uses feature differences to fuse bitemporal information.

TABLE I
QUANTITATIVE RESULTS ON THE CDD DATASET

Methods	Pre(%)	Rec(%)	F1(%)	IoU(%)
FC-Siam-Conc	80.30	85.79	82.96	70.88
FC-Siam-Diff	84.01	83.00	83.50	71.68
SRCDNet	90.94	94.70	92.78	86.54
DSAMNet	92.69	93.55	93.12	87.12
BIT	**97.99**	91.31	94.53	89.63
MSCANet	95.22	93.46	94.33	89.27
DDCNN	97.22	93.28	95.21	90.86
DMINet	96.81	94.68	95.74	91.82
Ours	96.97	**96.91**	**96.94**	**94.06**

TABLE II
QUANTITATIVE RESULTS ON THE SYSU-CD DATASET

Methods	Pre(%)	Rec(%)	F1(%)	IoU(%)
FC-Siam-Conc	74.92	82.44	78.50	64.61
FC-Siam-Diff	87.49	59.96	71.15	55.22
SRCDNet	73.57	78.60	76.00	61.29
DSAMNet	74.88	82.16	78.35	64.41
BIT	**88.33**	66.82	76.09	61.40
MSCANet	81.33	75.70	78.42	64.49
DDCNN	83.64	75.16	79.17	65.52
DMINet	84.13	79.76	81.89	69.33
Ours	82.10	**82.71**	**82.41**	**70.08**

TABLE III
QUANTITATIVE RESULTS ON THE LEVIR-CD DATASET

Methods	Pre(%)	Rec(%)	F1(%)	IoU(%)
FC-Siam-Conc	75.72	**95.44**	84.44	73.08
FC-Siam-Diff	78.74	93.74	85.59	74.81
SRCDNet	81.49	90.71	85.85	75.21
DSAMNet	80.23	89.09	84.43	73.05
BIT	89.44	89.63	89.53	81.05
MSCANet	90.41	89.10	89.75	81.40
DDCNN	**93.19**	87.34	90.17	82.10
DMINet	92.64	89.00	90.78	83.12
Ours	92.38	90.77	**91.57**	**84.45**

3) SRCDNet [53]: SRCDNet integrates five convolutional block attention modules (CBAMs) into the feature extractor to obtain more discriminative features.

4) DSAMNet [32]: A deeply supervised attention metric-based network that uses a metric module to learn CM and introduces a deeply supervised module to strengthen the supervision of the hidden layer to generate more useful features.

5) BIT [54]: A transformer-based network that uses a transformer encoder–decoder that efficiently models the context in the spatial-temporal domain, and then, obtains the CM by feature differencing.

6) MSCANet [55]: MSCANet is also a transformer-based method that utilizes the CNN to extract rich multiscale features and employs transformers to aggregate contextual information. Additionally, it adopts a multibranch prediction strategy to add supervision to deep layers.

7) DDCNN [17]: A difference-enhancement dense-attention convolutional neural network that uses dense attention to model correlations between features at different levels, while introducing a difference-enhancement unit that weights each pixel to highlight representative features.

8) DMINet [56]: A dual-branch multilevel intertemporal network, which implements interactions between bitemporal features by combining self-attention and cross attention. In addition, an incremental aggregation strategy is used to implement multilevel feature aggregation.

E. Comparative Experiments

The CDD, SYSU-CD, and LEVIR-CD datasets are detected using the proposed method, and the results are compared with selected eight representative change detection algorithms. The quantitative comparison of the three datasets is shown in Tables I–III, and the visualization comparison is shown in Figs. 7–9.

1) Comparisons on the CDD Dataset: Table I summarizes the quantitative results of Pre, Rec, F1, and IoU for all methods on the CDD dataset. As shown in Table I, F1 and IoU of FC-Siam-Conc are the lowest with 82.96% and 70.88%, respectively, followed by FC-Siam-Diff with F1 and IoU of 83.50% and 71.68%, respectively. The F1 and IoU of DSAMNet are 93.12% and 87.12%, respectively, which are 0.34% and 0.58% higher than those of SRCDNet, verifying the effectiveness of increasing supervision on the hidden layer. BIT and MSCANet outperform the aforementioned four methods on the CDD dataset, indicating the effectiveness of enhancing the contextual information of the bitemporal features through the transformer. The DDCNN is ranked second among the eight compared methods, which indicates that guiding the learning of low-level features by high-level features is helpful to improve the detection accuracy. DMINet is second only to our method, and its F1 and IoU are lower than our method by 1.2% and 2.24%, respectively, which indicates that the interaction between the bitemporal features is feasible before obtaining the difference features. Our method outperform the eight contrasting methods, obtaining the highest Rec, F1, and IoU of 96.91%, 96.94%, and 94.06%, respectively.

The visualization comparison further validates that our method obtains the best results on the CDD dataset. As shown in Fig. 7, our method achieves the best visualization in detecting changes in roads, vehicles, and buildings. FC-Siam-Diff and FC-Siam-Conc can only detect some large and obvious changes. For some small changes, they have many false detections and missed detections. SRCDNet and DSAMNet can detect some small changes such as vehicles, but the generated edges are blurred and there are missed detections. MSCANet also suffers from edge blurring, but its missed detection is lower than that of SRCDNet and DSAMNet. The remaining three methods improve the edge blur problem, but still have the problem of missed detection. When detecting snow-covered roads, FC-Siam-Diff and FC-Siam-Conc can only detect some obvious road changes. BIT, MSCANet, and DMINet can detect some narrow road changes, but the detected changes are not continuous. DDCNN has the problem of misdetected roads. The road changes detected by SRCDNet and DSAMNet are more continuous, but the detected road are larger than the actual ones. When detecting vehicle changes, for obvious vehicles,

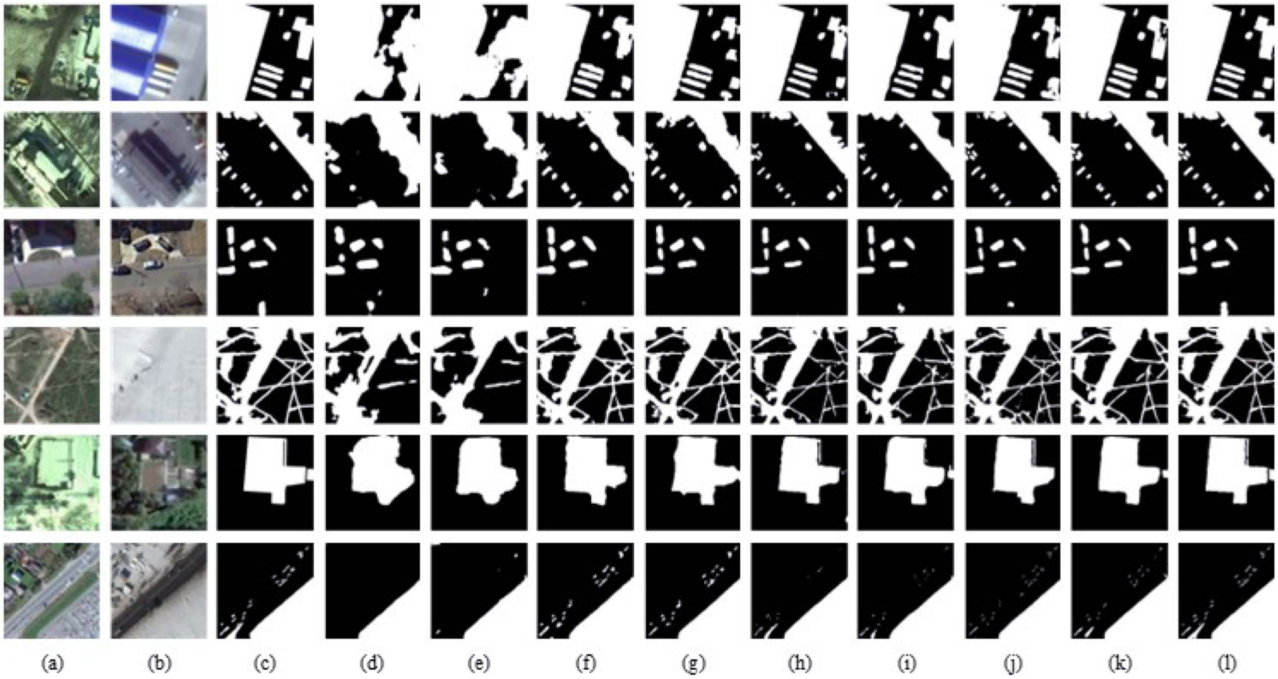


Fig. 7. Visualization results on the CDD dataset. (a) Image T1. (b) Image T2. (c) Ground truth. (d) FC-Siam-Conc. (e) FC-Siam-Diff. (f) SRCDNet. (g) DSAMNet. (h) BIT. (i) MSCANet. (j) DDCNN. (k) DMINet. (l) Ours.

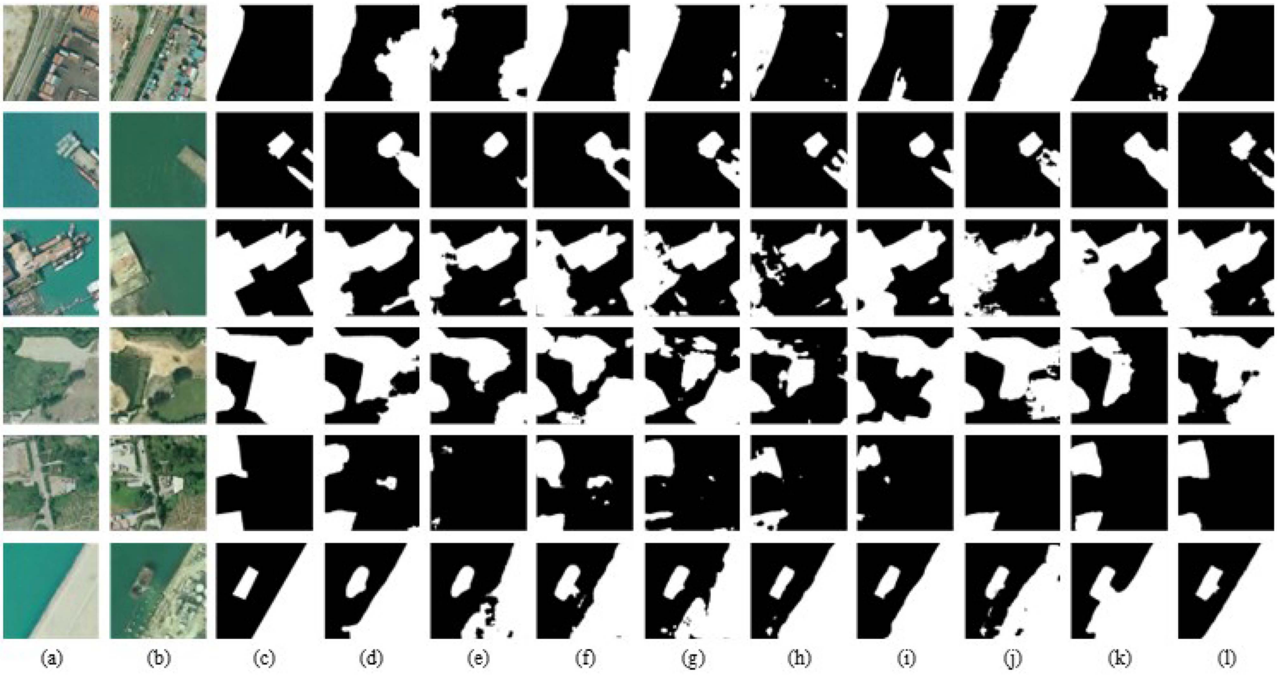


Fig. 8. Visualization results on the SYSU-CD dataset. (a) Image T1. (b) Image T2. (c) Ground truth. (d) FC-Siam-Conc. (e) FC-Siam-Diff. (f) SRCDNet. (g) DSAMNet. (h) BIT. (i) MSCANet. (j) DDCNN. (k) DMINet. (l) Ours.

all eight contrasting methods can detect them, but for nonobvious vehicles, only our method can detect vehicle changes completely, as shown in the third row of Fig. 7. For building detection, our method is the only one that can detect building changes intact and maintain sharp edges, as shown in the fifth row of Fig. 7. Due to the ability of the FADL module to capture varying spatial information, their changes can be completely detected and produce sharp edges when detecting changed regions of different size and shape, such as roads, vehicles, and buildings.

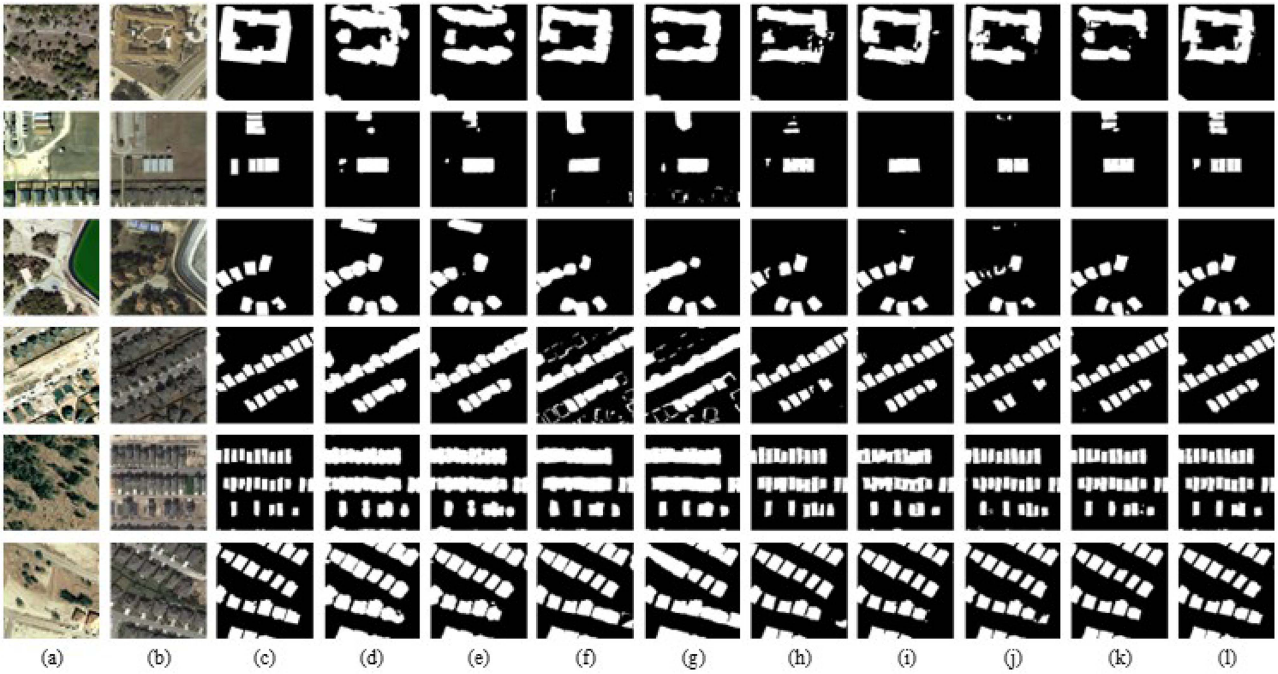


Fig. 9. Visualization results on the LEVIR-CD dataset. (a) Image T1. (b) Image T2. (c) Ground truth. (d) FC-Siam-Conc. (e) FC-Siam-Diff. (f) SRCDNet. (g) DSAMNet. (h) BIT. (i) MSCANet. (j) DDCNN. (k) DMINet. (l) Ours.

2) Comparisons on the SYSU-CD Dataset: Compared to the eight contrasting methods, our method also achieves the best results on the SYSU-CD dataset. As shown in Table II, our method obtains the highest Rec, F1, and IoU scores of 82.71%, 82.41%, and 70.08%, respectively. Among the eight contrasting methods, DMINet performs the best with an F1 of 81.89% and an IoU of 69.33%, and DDCNN is second only to DMINet with an F1 of 79.17% and an IoU of 65.52%. The performance of FC-Siam-Conc and MSCANet on the SYSU-CD dataset is similar, with a difference of 0.08% in their F1 values and 0.12% in their IoU values. The F1 and IoU of DSAMNet are 78.35% and 64.41%, which are 2.26% and 3.01% higher than BIT, respectively. The Pre of BIT is the highest at 88.33%. SRCDNet has an F1 of 76.00% and an IoU of 61.29%, which is only better than FC-Siam-Diff. The F1 and IoU of FC-Siam-Diff are the lowest with 71.15% and 55.22%, respectively, which may be due to feature differences that cause loss of useful information during the feature fusion process.

Fig. 8 further presents the visualization comparison on the SYSU-CD dataset. Our method also achieves optimal visualization results compared to the eight contrasting methods. As shown in the first row of Fig. 8, when detecting changes before and after construction, all eight contrasting methods have false detections, and DDCNN has the most serious false detection, and the changes detected by FC-Siam-Diff are incomplete. When detecting ship changes, MSCANet's visualization is second only to our method, FC-Siam-Diff misses some ship changes, and the remaining six contrasting methods suffer from false detection problems, incorrectly detecting docks as changes, as shown in the second row of Fig. 8. The eight contrasting methods underperform in complex scenes, while our method is able to extract relatively complete changes, as shown in the fourth row of Fig. 8. For the detection of sea construction, among the eight contrasting methods, FC-Siam-Diff, DSAMNet, and DMINet have more serious false detection problems, FC-Siam-Conc has edge blur problem, and the edges generated by SRCDNet, BIT, and DDCNN are rough. MSCANet is able to produce a relatively good visualization, but it does not do well in terms of edge detail, as shown in the sixth row of Fig. 8. As shown in the fifth row of Fig. 8, compared with the eight contrasting methods, only our method can detect all changes relatively completely and maintain sharp edges, and DMINet is second only to our method.

3) Comparisons on the LEVIR-CD Dataset: The quantitative comparison of the different methods on the LEVIR-CD dataset is displayed in Table III. F1 and IoU of DSAMNet are the lowest with 84.43% and 73.05%, respectively, followed by FC-Siam-Conc with F1 of 84.44% and IoU of 73.08%. FC-Siam-Conc achieves the highest Rec of 95.44%. FC-Siam-Diff has an F1 of 85.59% and an IoU of 74.81%, which suggests that difference holds more useful information and transfers it to the decoder compared to concatenation. SRCDNet has an F1 of 85.85% and IoU of 75.21%. The F1 and IoU values of MSCANet are slightly higher than the BIT, which indicates that it is feasible to add supervision to the deeper layers. DDCNN ranks second among the eight contrasting methods, obtaining the highest Pre of 93.19% with an F1 of 90.17% and an IoU of 82.10%, indicating that using high-level features to guide low-level feature learning is effective. Likely to the results on the first two datasets, DMINet performs second only to our method on the LEVIR-CD dataset with 90.78% and 83.12% F1 and IoU, respectively. Compared with the eight contrasting methods, our method performs the best, obtaining the highest F1 and IoU of 91.57% and 84.45%, respectively, which are 0.79% and 1.33% higher than the figures obtained by DMINet.

Fig. 9 provides a more intuitive comparison of different methods on the LEVIR-CD dataset. In terms of detecting changes in large buildings, except our method, the buildings detected by the eight contrasting methods are incomplete. FC-Siam-Conc and FC-Siam-Diff have false detection and edge blurring problems; SRCDNet and DSAMNet also have edge blurring problems, and BIT; DDCNN and DMINet are affected by noise. MSCANet is the method that can detect the building outline relatively completely in addition to our method, as shown in the first row of Fig. 9. In terms of detecting changes in small buildings, our method is the only one that can detect nearly all changes and generate sharp edges. Several other methods have different degrees of false detections, missed detections, and edge blurring problems. As shown in Fig. 9(d)–(g), FC-Siam-Conc, FC-Siam-Diff, SRCDNet, and DSAMNet all suffer from edge blurring and false detection problems. The false detection problem is more serious for SRCDNet and DSAMNet compared to FC-Siam-Conc and FC-Siam-Diff, as shown in the second, fourth, and sixth rows of Fig. 9. Although BIT and DDCNN can generate relatively sharp edges, but there are missed detections and false detections, as shown in the fourth and fifth rows of Fig. 9. MSCANet has fewer false detection problems, but misses some unobvious buildings, as shown in Fig. 9(i). DMINet does a relatively good job of maintaining the details of the building edges, but also has the problem of missing some unobvious buildings, as shown in Fig. 9(k). For unobvious building changes, all eight contrasting methods perform poorly except our method, as shown in the second row of Fig. 9. In conclusion, compared to the eight contrasting methods, our method is the only one that can detect all changes more completely and maintain sharp edges, whether for larger or smaller buildings, obvious or not.

F. Ablation Study

Ablation experiments are performed on the CDD, SYSU-CD, and LEVIR-CD datasets to validate the effectiveness of FADL module and DR module. In detail, five sets of experiments are set up to validate the effectiveness of each individual module and the effectiveness of combinations between different modules. The "Base" baseline means that the FADL module and DR module are not included, and the multiscale difference features acquired from each residual block are fed directly to the decoder. In the "Base" model, KASL, JFR, and DR are introduced. The "Base + KASL" model is used as the second baseline, the "Base + JFR" model is used as the third baseline, the "Base + DR" model is used as the fourth baseline, and the "Base + FADL (KASL + JFR)" model is used as the fifth baseline. The "Base + FADL + DR" model is our proposed method.

1) Ablation Study on the CDD Dataset: The quantitative results of the ablation study on the CDD dataset are presented in Table IV. The "Base" model has an F1 of 95.32% and an IoU of 91.06% on the CDD dataset. The "Base + KASL" model improves F1 from 95.32% to 95.41% and IoU from 91.06% to 91.23%, validating the effectiveness of KASL in capturing varying spatial information. The "Base + JFR" model improves F1 from 95.32% to 96.40% and IoU from 91.06% to 93.05%, validating the effectiveness of JFR in learning comprehensive features between bitemporal data. Compared to introducing JFR alone in the "Base" model, the "Base + FADL" model improves F1 from 96.40% to 96.82% and IoU from 93.05% to 93.83%, which demonstrates that the combination of KASL and JFR can achieve a gain effect. Compared to the "Base" model, the "Base + DR" model improves F1 from 95.32% to 95.57% and IoU from 91.06% to 91.51%, which validates the effectiveness of using shallow features to recalibrate spatial and difference details. F1 and IoU of the "Base + FADL + DR" model are the highest with 96.94% and 94.06%, respectively, demonstrating that combining FADL and DR is feasible.

TABLE IV
ABLATION STUDY ON THE CDD DATASET

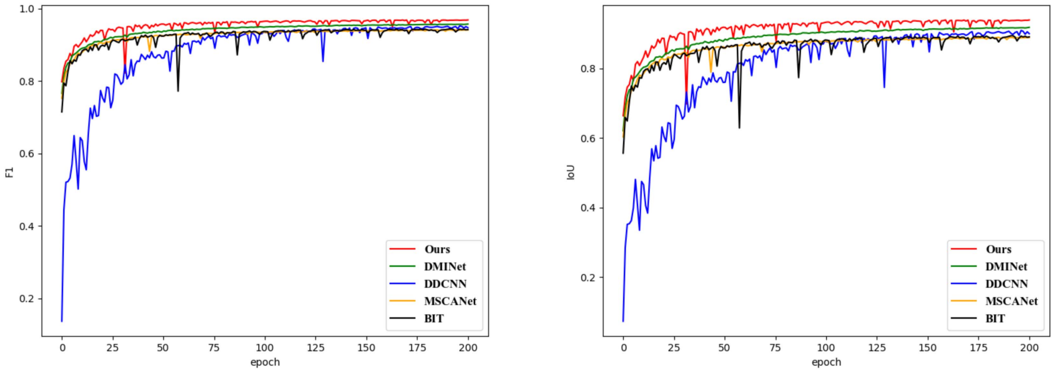
	KASL	JFR	DR	Pre(%)	Rec(%)	F1(%)	IoU(%)
Base				95.31	95.32	95.32	91.06
Base	√			95.57	95.25	95.41	91.23
Base		√		96.29	96.51	96.40	93.05
Base			√	95.83	95.30	95.57	91.51
Base	√	√		96.63	97.01	96.82	93.83
Base	√	√	√	96.97	96.91	96.94	94.06

TABLE V
ABLATION STUDY ON THE SYSU-CD DATASET

	KASL	JFR	DR	Pre(%)	Rec(%)	F1(%)	IoU(%)
Base				77.44	85.70	81.36	68.57
Base	√			81.94	82.14	82.04	69.55
Base		√		83.86	79.72	81.74	69.12
Base			√	83.50	80.45	81.95	69.42
Base	√	√		83.83	80.62	82.19	69.77
Base	√	√	√	82.10	82.71	82.41	70.08

2) Ablation Study on the SYSU-CD Dataset: The quantitative results of the ablation study on the SYSU-CD dataset are presented in Table V. The introduction of KASL, JFR, and DR in the "Base" model, respectively, all increase F1 and IoU values, verifying the effectiveness of each individual part on the SYSU-CD dataset. The "Base" model has an F1 of 81.36% and an IoU of 68.57%. The "Base + KASL" model improves F1 from 81.36% to 82.04% and IoU from 68.57% to 69.55%. The "Base + JFR" model improves F1 from 81.36% to 81.74% and IoU from 68.57% to 69.12%. The "Base + DR" model improves F1 from 81.36% to 81.95% and IoU from 68.57% to 69.42%. Compared to introducing KASL and JFR alone in the "Base" model, F1 and IoU of the "Base + FADL" model are the highest with 82.19% and 69.77%, respectively, validating the effectiveness of the combination of KASL and JFR. Compared with other baselines, F1 and IoU of the "Base + FADL + DR" model are the highest with 82.41% and 70.08%, respectively, which further illustrates the gain effect of the combination of FADL and DR.

3) Ablation Study on the LEVIR-CD Dataset: As shown in Table VI, the results are similar to those of the ablation experiments on the first two datasets. The introduction of KASL, JFR, and DR in the "Base" model, respectively, all increase F1 and IoU values, validating the effectiveness of each individual part on



(a) (b)

Fig. 10. Learning curve visualization results. (a) F1 value of CDD validation set. (b) IoU value of CDD validation set.

TABLE VI
ABLATION STUDY ON THE LEVIR-CD DATASET

	KASL	JFR	DR	Pre(%)	Rec(%)	F1(%)	IoU(%)
Base				92.57	89.77	91.15	83.73
Base	√			92.25	90.51	91.37	84.12
Base		√		91.79	91.14	91.46	84.27
Base			√	91.75	90.90	91.32	84.03
Base	√	√		92.22	90.83	91.52	84.37
Base	√	√	√	92.38	90.77	91.57	84.45

the LEVIR-CD dataset. The "Base" model has an F1 of 91.15% and an IoU of 83.73%. Compared with the "Base" model, F1 and IoU increase by 0.22% and 0.39% for the "Base + KASL" model, 0.31% and 0.54% for the "Base + JFR" model, and 0.17% and 0.3% for the "Base + DR" model, respectively. Compared with introducing KASL, JFR, and DR alone in the "Base" model, the "Base + FADL" model and "Base + FADL + DR" model further improve the F1 and IoU values, which validates the effectiveness of combining KASL, JFR, and DR. The F1 and IoU of the "Base + FADL" model are 91.52% and 84.37%, respectively. The F1 and IoU of the "Base + FADL + DR" model are the highest with 91.57% and 84.45%, respectively.

G. Learning Curve Comparison

Fig. 10 compares the learning curve visualization results of our method with DMINet, DDCNN, MSCANet, and BIT on the CDD validation set. The figure shows the F1 and IoU values for 200 epochs. Through the comparison of F1 and IoU, our method achieves good performance in terms of model stability and convergence speed, and obtains the highest accuracy rate. DMINet has the best model stability and its F1 and IoU values are second only to our method. Both DDCNN and BIT have weak model stability. In addition, DDCNN has the slowest convergence speed, but its final F1 and IoU values are higher than those of MSCANet and BIT. BIT performs well in terms of convergence speed, with slightly higher F1 and IoU values than MSCANet. MSCANet performs well in terms of model stability and convergence speed, but its F1 and IoU values are the lowest among these methods.

V. DISCUSSION

In this section, we discuss both the model efficiency and the comparison between different loss functions.

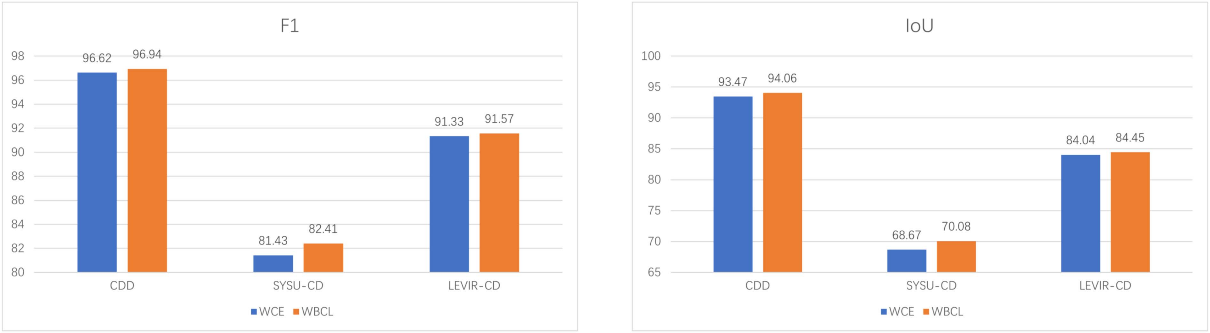
A. Model Efficiency

To understand the practical application requirements of different methods, this article evaluates model efficiency from three aspects: parameters (Params), floating-point operations (FLOPs), and inference time. Params represents the number of parameters that need to be learned in a model. It is commonly used to measure the complexity of a model, and its unit is 10^6 (M). FLOPs represents the total number of addition and multiplication operations performed by a model. It is commonly used to measure the computational requirements of a model, and its unit is 10^9 (G). Inference time represents the time taken by a model to generate output given a specific input. It is commonly used to measure the speed and real-time performance of a model and is typically measured in seconds (s).

Table VII shows a quantitative comparison of the Params, FLOPs, and inference time for the different methods, with the image size used being $3 \times 256 \times 256$. As can be seen from Table VII, FC-Siam-Conc and FC-Siam-Diff have lower Params, FLOPs, and inference time. DDCNN achieves the highest values in all three metrics. DMINet has relatively low Params and FLOPs, and has a performance second only to our method. SRCDNet and DSAMNet have relatively high FLOPs, and the performance gains they bring are not outstanding. BIT and MSCANet have relatively low inference time and the performance gains they bring are appreciable. Although our method has relatively high values for these three metrics, our method is able to deliver substantial performance improvements. As shown in the fourth and ninth rows of Table VII, the differences between our method and DSAMNet in terms of FLOPs and inference time are not significant, but the F1 and IoU values achieved by our method on the CDD dataset are 3.82% and 6.94% higher than those of DSAMNet, respectively. Therefore, although our method has relatively high values for these three metrics, it achieves the best performance, which is worthwhile.

TABLE VII
COMPARISON OF MODEL EFFICIENCY ACROSS DIFFERENT METHODS

Methods	Params(M)	FLOPs(G)	Inference Time(s)	F1(%)	IoU(%)
FC-Siam-Conc	1.55	5.32	12	82.96	70.88
FC-Siam-Diff	1.35	4.72	12	83.50	71.68
SRCDNet	16.98	71.68	22	92.78	86.54
DSAMNet	16.95	75.39	32	93.12	87.12
BIT	11.47	26.31	13	94.53	89.63
MSCANet	16.42	14.80	16	94.33	89.27
DDCNN	46.68	177.44	76	95.21	90.86
DMINet	6.24	14.55	20	95.74	91.82
Ours	30.58	94.90	33	96.94	94.06



(a) (b)

Fig. 11. Comparison between different loss functions. (a) F1 value. (b) IoU value.

B. Comparison of Loss Functions

We compare the weighted cross-entropy (WCE) loss with the WBCL used in this article. Fig. 11 shows the F1 and IoU values of the two loss functions on the three public datasets. As can be seen from the figure, WBCL achieves the best performance on all three datasets. On the CDD dataset, the F1 and IoU values of WBCL are 0.32% and 0.59% higher than those of WCE, respectively. Similar results are found on the other two datasets, which indicates the validity of the used WBCL. The role of WBCL is reflected in two aspects. One of them alleviates the category imbalance problem by applying different weights to the changed and unchanged pixels. The second is to enhance the separability between the changed and unchanged regions by increasing the distance of the changed regions and decreasing the distance of the unchanged regions.

VI. CONCLUSION

In this article, an adaptive spatial and difference learning method is proposed for the change detection task, which mainly contains two parts: FADL module and DR module. In FADL, a KASL part is constructed to capture varying spatial information; a JFR part is employed to learn comprehensive features between bitemporal data, which are further enhanced by difference information. The DR module further recalibrates the spatial and difference details by exploiting the precise location information of shallow features. Experimental results on three public datasets suggest that our proposed method is superior to the selected eight contrasting algorithms. Our method relieves the spatial ambiguity problem caused by traditional convolution kernels when dealing with changed regions of different size and shape. In change detection tasks, "nonsemantic changes" induced by light and seasonal changes can lead to pseudochange, which can affect the detection accuracy. Therefore, we will explore feature alignment methods to relieve the effects of "nonsemantic changes" in future studies.

REFERENCES

[1] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2019.
[2] L. Shen et al., "S2Looking: A satellite side-looking dataset for building change detection," *Remote Sens.*, vol. 13, no. 24, 2021, Art. no. 5094.
[3] F. Bovolo and L. Bruzzone, "A split-based approach to unsupervised change detection in large-size multitemporal images: Application to tsunami-damage assessment," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 6, pp. 1658–1670, Jun. 2007.
[4] M. M. Peña and F. Navarro, "An NDVI-data harmonic analysis to study deforestation in Peru's Tahuamanu province during 2001–2011," *Int. J. Remote Sens.*, vol. 37, no. 4, pp. 856–875, 2016.
[5] L. Liang, F. Chen, L. Shi, and S. Niu, "NDVI-derived forest area change and its driving factors in China," *PLoS One*, vol. 13, no. 10, 2018, Art. no. e0205885.
[6] C.-F. Chen et al., "Multi-decadal mangrove forest change detection and prediction in Honduras, Central America, with landsat imagery and a Markov chain model," *Remote Sens.*, vol. 5, no. 12, pp. 6408–6426, 2013.
[7] A. Mondini, F. Guzzetti, P. Reichenbach, M. Rossi, M. Cardinali, and F. Ardizzone, "Semi-automatic recognition and mapping of rainfall induced shallow landslides using optical satellite images," *Remote Sens. Environ.*, vol. 115, no. 7, pp. 1743–1757, 2011.

[8] D. Lu, P. Mausel, E. Brondizio, and E. Moran, "Change detection techniques," *Int. J. Remote Sens.*, vol. 25, no. 12, pp. 2365–2401, 2004.

[9] P. Du, X. Wang, D. Chen, S. Liu, C. Lin, and Y. Meng, "An improved change detection approach using tri-temporal logic-verified change vector analysis," *ISPRS J. Photogrammetry Remote Sens.*, vol. 161, pp. 278–293, 2020.

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

[11] M. H. Kesikoğlu, Ü. Atasever, and C. Özkana, "Unsupervised change detection in satellite images using fuzzy c-means clustering and principal component analysis," *Int. Arch. Photogrammetry Remote Sens. Spatial Inf. Sci.*, vol. 40, pp. 129–132, Oct. 2013.

[12] S. Marchesi and L. Bruzzone, "ICA and kernel ICA for change detection in multispectral remote sensing images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2009, pp. II–980.

[13] J. Rogan, J. Franklin, and D. A. Roberts, "A comparison of methods for monitoring multitemporal vegetation change using thematic mapper imagery," *Remote Sens. Environ.*, vol. 80, no. 1, pp. 143–156, 2002.

[14] L. Huang, G. Zhang, and Y. Li, "An object-based change detection approach by integrating intensity and texture differences," in *Proc. IEEE 2nd Int. Asia Conf. Inform. Control Autom. Robot.*, 2010, pp. 258–261.

[15] N. Wang, W. Li, R. Tao, and Q. Du, "Graph-based block-level urban change detection using Sentinel-2 time series," *Remote Sens. Environ.*, vol. 274, 2022, Art. no. 112993.

[16] L. Ru, B. Du, and C. Wu, "Multi-temporal scene classification and scene change detection with correlation based fusion," *IEEE Trans. Image Process.*, vol. 30, pp. 1382–1394, Nov. 2020.

[17] X. Peng, R. Zhong, Z. Li, and Q. Li, "Optical remote sensing image change detection based on attention mechanism and image difference," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7296–7307, Sep. 2020.

[18] G. Alimjan, Y. Jiaermuhamaiti, H. Jumahong, S. Zhu, and P. Nurmamat, "An image change detection algorithm based on multi-feature self-attention fusion mechanism unet network," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 35, no. 14, 2021, Art. no. 2159049.

[19] Y. Jiang, L. Hu, Y. Zhang, and X. Yang, "WRICNet: A weighted rich-scale inception coder network for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–13, Jan. 2022, Art. no. 4705313.

[20] S. Fang, K. Li, J. Shao, and Z. Li, "SNUNet-CD: A densely connected siamese network for change detection of VHR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, Feb. 2021, Art. no. 8007805.

[21] H. Chen, Z. Qi, and Z. Shi, "Efficient transformer based method for remote sensing image change detection," 2021, *arXiv:2103.00208*.

[22] H. Chen, C. Wu, and B. Du, "Towards deep and efficient: A deep siamese self-attention fully efficient convolutional network for change detection in VHR images," 2021, *arXiv:2108.08157*.

[23] L. Yang, Y. Chen, S. Song, F. Li, and G. Huang, "Deep siamese networks based change detection with remote sensing images," *Remote Sens.*, vol. 13, no. 17, 2021, Art. no. 3394.

[24] Q. Ding, Z. Shao, X. Huang, and O. Altan, "DSA-Net: A novel deeply supervised attention-guided network for building change detection in high-resolution remote sensing images," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 105, 2021, Art. no. 102591.

[25] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.

[26] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[27] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2818–2826.

[28] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4278–4284.

[29] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional siamese networks for change detection," in *Proc. IEEE 25th Int. Conf. Image Process.*, 2018, pp. 4063–4067.

[30] S.-H. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. Torr, "Res2Net: A new multi-scale backbone architecture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 652–662, Feb. 2021.

[31] J. Chen et al., "DASNet: Dual attentive fully convolutional siamese networks for change detection in high-resolution satellite images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 1194–1206, Nov. 2020.

[32] Q. Shi, M. Liu, S. Li, X. Liu, F. Wang, and L. Zhang, "A deeply supervised attention metric-based network and an open aerial image dataset for remote sensing change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, Jun. 2021, Art. no. 5604816.

[33] L. Bruzzone and D. F. Prieto, "Automatic analysis of the difference image for unsupervised change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 3, pp. 1171–1182, May 2000.

[34] J. Deng, K. Wang, Y. Deng, and G. Qi, "PCA-based land-use change detection and analysis using multitemporal and multisensor satellite data," *Int. J. Remote Sens.*, vol. 29, no. 16, pp. 4823–4838, 2008.

[35] J. Im and J. R. Jensen, "A change detection model based on neighborhood correlation image analysis and decision tree classification," *Remote Sens. Environ.*, vol. 99, no. 3, pp. 326–340, 2005.

[36] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[37] A. A. Nielsen, K. Conradsen, and J. J. Simpson, "Multivariate alteration detection (MAD) and MAF postprocessing in multispectral, bitemporal image data: New approaches to change detection studies," *Remote Sens. Environ.*, vol. 64, no. 1, pp. 1–19, 1998.

[38] C. Wu, B. Du, and L. Zhang, "Slow feature analysis for change detection in multispectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2858–2874, May 2014.

[39] J. Chen, Z. Mao, B. Philpot, J. Li, and D. Pan, "Detecting changes in high-resolution satellite coastal imagery using an image object detection approach," *Int. J. Remote Sens.*, vol. 34, no. 7, pp. 2454–2469, 2013.

[40] B. Desclée, P. Bogaert, and P. Defourny, "Forest change detection by statistical object-based method," *Remote Sens. Environ.*, vol. 102, no. 1/2, pp. 1–11, 2006.

[41] S. Pang, X. Hu, M. Zhang, Z. Cai, and F. Liu, "Co-segmentation and superpixel-based graph cuts for building change detection from bi-temporal digital surface models and aerial images," *Remote Sens.*, vol. 11, no. 6, p. 729, Mar. 2019, doi: 10.3390/rs11060729.

[42] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.

[43] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Interv.*, 2015, pp. 234–241.

[44] F. Rahman, B. Vasu, J. Van Cor, J. Kerekes, and A. Savakis, "Siamese network with multi-level features for patch-based change detection in satellite imagery," in *Proc. IEEE Glob. Conf. Signal Inf. Process.*, 2018, pp. 958–962.

[45] D. Peng, Y. Zhang, and H. Guan, "End-to-end change detection for high resolution satellite images using improved unet," *Remote Sens.*, vol. 11, no. 11, 2019, Art. no. 1382.

[46] H. Chen and Z. Shi, "A spatial-temporal attention-based method and a new dataset for remote sensing image change detection," *Remote Sens.*, vol. 12, no. 10, 2020, Art. no. 1662.

[47] M. Zhang, G. Xu, K. Chen, M. Yan, and X. Sun, "Triplet-based semantic relation learning for aerial remote sensing image change detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 2, pp. 266–270, Feb. 2019.

[48] Y. Liu, C. Pang, Z. Zhan, X. Zhang, and X. Yang, "Building change detection for remote sensing images using a dual-task constrained deep siamese convolutional network model," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 5, pp. 811–815, May 2021.

[49] C. Zhang et al., "A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images," *ISPRS J. Photogrammetry Remote Sens.*, vol. 166, pp. 183–200, 2020.

[50] H. Jiang, X. Hu, K. Li, J. Zhang, J. Gong, and M. Zhang, "PGA-SiamNet: Pyramid feature-based attention-guided siamese network for remote sensing orthoimagery building change detection," *Remote Sens.*, vol. 12, no. 3, p. 484, Feb. 2020, doi: 10.3390/rs12030484.

[51] M. Lebedev, Y. V. Vizilter, O. Vygolov, V. Knyaz, and A. Y. Rubis, "Change detection in remote sensing images using conditional adversarial networks," *Int. Arch. Photogrammetry Remote Sens. Spatial Inf. Sci.*, vol. 42, no. 2, pp. 565–571, 2018.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[53] M. Liu, Q. Shi, A. Marinoni, D. He, X. Liu, and L. Zhang, "Super-resolution-based change detection network with stacked attention module for images with different resolutions," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–18, 2021.

[54] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–14, 2021.

[55] M. Liu, Z. Chai, H. Deng, and R. Liu, "A CNN-transformer network with multiscale context aggregation for fine-grained cropland change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4297–4306, 2022.

[56] Y. Feng, J. Jiang, H. Xu, and J. Zheng, "Change detection on remote sensing images using dual-branch multilevel intertemporal network," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, pp. 1–15, 2023.

Yangguang Liu received the bachelor's degree in software engineering from Henan University, Kaifeng, China, in 2021. He is currently working toward the master' degree in software engineering with the Nanjing University of Science and Technology, Nanjing, China.

His research interests include deep learning, remote sensing image change detection, and image segmentation.

Fang Liu (Member, IEEE) received the B.S. degree in information and computing science from Henan University, Kaifeng, China, in 2012 and the Ph.D. degree in intelligent information processing from Xidian University, Xi'an, China, in 2018.

She is currently an Associate Professor with the Nanjing University of Science and Technology, Nanjing, China. Her research interests include deep learning, object detection, polarimetric SAR image classification, and change detection.

Jia Liu (Member, IEEE) received the B.S. degree in intelligence science and technology and the Ph.D. degree in pattern recognition and intelligent systems from Xidian University, Xi'an, China, in 2013 and 2018, respectively.

He is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. His current research interests include computational intelligence and image understanding.

Xu Tang (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in electronic circuit and system from Xidian University, Xi'an, China, in 2007, 2010, and 2017 respectively, and the joint Ph.D. degree from the University of Colorado at Boulder, Boulder, CO, USA, in 2016, along with Prof. W. J. Emery.

He is currently an Associate Professor with the Key Laboratory of Intelligent Perception and Image Understanding, Ministry of Education, Xidian University. His research interests include remote sensing image content-based retrieval and reranking, hyperspectral image processing, remote sensing scene classification, and object detection.

Liang Xiao (Member, IEEE) received the B.S. degree in applied mathematics and the Ph.D. degree in computer science from the Nanjing University of Science and Technology (NJUST), Nanjing, China, in 1999 and 2004, respectively.

From 2006 to 2008, he was a Postdoctoral Research Fellow with the Pattern Recognition Laboratory, NJUST. From 2009 to 2010, he was a Postdoctoral Fellow with the Rensselaer Polytechnic Institute, Troy, NY, USA. Since 2013, he has been the Deputy Director of the Jiangsu Key Laboratory of Spectral Imaging Intelligent Perception, Nanjing. Since 2014, he has been the Vice-Director with the Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, NJUST, where he is currently a Professor with the School of Computer Science and Engineering. His research interests include remote sensing image processing, image modeling, computer vision, machine learning, and pattern recognition.