






# Detail Enhanced Change Detection in VHR Images Using a Self-Supervised Multiscale Hybrid Network

Dalong Zheng , Zebin Wu , Senior Member, IEEE, Jia Liu , Member, IEEE, Chih-Cheng Hung , Member, IEEE, and Zhihui Wei , Member, IEEE

**Abstract**—The integration of the transformer and convolutional neural network (CNN) has become a useful method for change detection in remote sensing images. The main function of the transformer is to capture the global features, while the CNN is more for obtaining the local features. However, such an integration is not efficient for change detection in the very-high-resolution (VHR) remote sensing images with fine surface detail information. Hence, to improve this traditional construction of the transformer and CNN, we propose a dense Swin-Transformer-V2 (DST) and VGG16, coined as DST-VGG, for extracting the discriminatory features for change detection. The difference between our proposed network and other networks is that the output of the VGG16 encoders will be used in the DST in which more Swin-V2 blocks are added for fine feature extraction. The learning model in the VGG16 encoders employs a self-supervised method, which is guided through the change in details. Our network not only inherits the advantages of the integration of the transformer and CNN, but also captures the features of change relationship through the DST and catches the primitive features in both prechanged and postchanged regions through the VGG16. In addition, we design a mixed feature pyramid within the DST, which provides interlayer interaction information and intralayer multiscale information for a more complete feature learning within the new network. Furthermore, we impose a self-supervised strategy to guide the VGG16 provide the semantic change information from the output features of the encoder. We compared our experimental results with those of the state-of-the-art methods on four commonly used public VHR remote sensing datasets. It shows that our network performs better, in terms of F1, IoU, and OA, than those of the existing networks for change detection.

**Index Terms**—Change detection, mixed feature pyramid (MFP), self-supervised learning (SSL), Swin transformer V2, VGG16.

Manuscript received 2 November 2023; revised 10 December 2023; accepted 20 December 2023. Date of publication 1 January 2024; date of current version 18 January 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62071233, Grant 61971223, Grant 62276133, and Grant 61976117, in part by the Jiangsu Provincial Natural Science Foundations of China under Grant BK20211570, Grant BK20180018, and Grant BK20191409, in part by the Fundamental Research Funds for the Central Universities under Grant 30917015104, Grant 30919011103, Grant 30919011402, and Grant 30921011209, in part by the Key Projects of University Natural Science Fund of Jiangsu Province under Grant 19KJA360001, and in part by the Qinglan Project of Jiangsu Universities under Grant D202062032. (Corresponding author: Zebin Wu.)

Dalong Zheng, Zebin Wu, Jia Liu, and Zhihui Wei are with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: zhengdl@njust.edu.cn; wuzb@njust.edu.cn; omegaliuj@njust.edu.cn; gswei@njust.edu.cn).

Chih-Cheng Hung is with the Center for Machine Vision and Security Research, Kennesaw State University, Kennesaw, GA 30144 USA (e-mail: chung1@kennesaw.edu).

Code is available on-line at <https://github.com/DalongZ/DST-VGG>.  
Digital Object Identifier 10.1109/JSTARS.2023.3348630

## I. INTRODUCTION

CHANGE detection is one of the earliest and important remote sensing tasks, which have been studied by many researchers for a long period of time [1], [2], [3], [4]. Change detection is defined as the identification of changes in the surface area found in images over time. It is used in many scenarios, including disaster assessment [5], urban planning, and land surface change [6], [7]. With the rapid development of satellites and sensors, very-high-resolution (VHR) remote sensing images have gradually become one of the mainstream remote sensing images in research. These images have a very high spatial resolution, ranging from 0.03 to 1 m per pixel, and provide rich spatial information and fine surface details. However, one of the main challenges faced by VHR images for change detection is high intraclass variation and low interclass variance of the targets being detected [8]. The pseudochanges are the significant contributor to this challenge, which are caused by different lighting or shadows. Therefore, it has been the focus of research on how to design a stable network and provide comprehensive and diverse feature information to distinguish the pseudochanges in change detection as shown in Fig. 1.

Traditional change detection algorithms, according to different detection units, can be divided into pixel-based algorithms and object-based algorithms. The detection results of pixel-based algorithms are obtained through feature extraction, and then, threshold segmentation, which include methods based on arithmetic operations (band difference [9] and spectral angle mapper [10]), methods based on transformation (change vector analysis [11], [12], principal component analysis [13], independent component analysis [14]), postclassification change detection [15], and slow feature analysis [16]). Object-based algorithms segment the images, and then, compare the classification results to get the change detection results [17]. Pixel-based algorithms are trapped by the interference of small noise regions and the choice of segmentation threshold. Meanwhile object-based algorithms often get stuck in the accumulation of multiple classification errors that affect the detection accuracy [1]. Both of these traditional algorithms require prior knowledge and manual design, and are easily affected by sensor noise.

With the availability of VHR remote sensing data, deep learning has also shown outstanding detection ability in the field of remote sensing. The CNN converts the input images into the high-dimensional deep features, and combines the targets and background to extract semantic information for achieving the detection effect beyond many traditional methods.

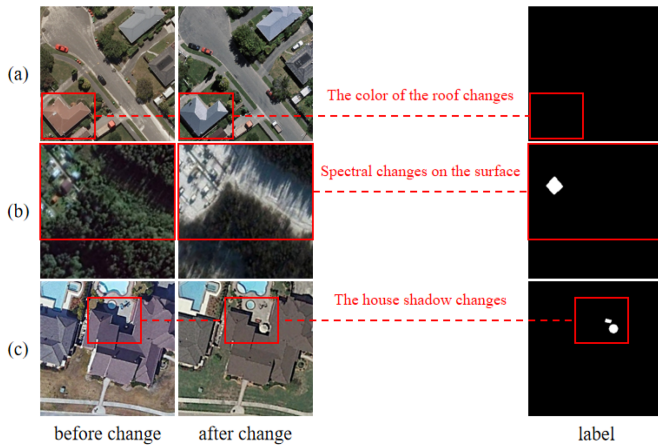


Fig. 1. Variety of pseudochanges become the challenges in change detection. (a) Roof color changes. (b) Spectral changes on the surface. (c) House shadow changes.

Daut et al. [18] provided the three most common baseline networks for change detection. The architecture [19], which combines the CNN and conditional random field refines the edges of detection areas, but its training is slow. The CNN is limited by the narrow receptive field of local information. The transformer rises rapidly due to the capability of modeling global information. However, with the transformer only, it does not work well for change detection due to the lack of low-level details [20]. Therefore, the key to this research is the ability to build an efficient architecture through the combination of a transformer and CNN.

In general, network architectures for change detection can be divided into early fusion (EF) [18], [21], [22], [23] and late fusion (LF) [18], [24], [25], [26], [27], [28], [29], [30] networks. The EF network works by stitching two images together and feeding them into the network as a single input. By concatenating two three-channel images into a six-channel image, both Alcantarilla et al. [21] and Zhang et al. [22] fed the stack into full convolutional neural network (FCN) and UNet++, respectively, and output the change map after training the network. The disadvantage of this method is that the network lacks the deep features of single images, resulting in fractured edges and broken structures in change map. In the LF network, the features are extracted from the prechanged and postchanged images, respectively, by using the dual-input structure, and are fused in the second half of the network. The Siamese network, which is the most prominent LF network consists of two subnets with shared weights. The siamese network was first used for change detection in [24]. The use of convolutional block attention module (CBAM) and deep supervision for the siamese network, respectively, alleviates the problem of heterogeneous features fusion and deep features migration in the training process [25]. However, for the LF network, the contradiction between the dual-stream input of the encoder and the single output of the decoder often results in the disappearance of gradient propagation and affects the low-level features learning of two original images [25]. As a consequence, it is another problem worth our exploration on how to overcome the respective disadvantages of these two network

architectures and provide the complete and diverse features for change detection.

In addition to the design of the overall network architectures for change detection, researchers are also pushing forward the elaboration of the functional modules in the network. The attention modules introduced in change detection include squeeze-and-excitation attention (SE) [31], efficient channel attention (ECA) [32], CBAM [33], and cross-attention [29]. As the ground objects have different scales in the VHR images, it is essential to extract diverse detail information for a network, which is still robust for change detection. Multiscale features of deep learning generally can be divided into three categories: multiscale features between different layers, multiscale interaction features between different layers, and multiscale features from different convolution units. The first type of multiscale features was embedded in the common U-Net network [18]. The second type of multiscale features typically interacts and fuses using the transformer or CNN. The third type of multiscale features is provided by a variety of convolution units, such as inception [34], dilated convolution [35], res2net convolution (Res2Net-Conv) [36], and selective kernel convolution (SK-Conv) [37]. We should consider the integration and utilization of these three multiscale features. Other researchers have proposed by combining with interaction feature [38], generative adversarial network (GAN) [39], [40], or self-supervised learning (SSL) [41], [42], [43] to obtain more discriminative features. These deep learning technologies are aimed at solving the problem of high intraclass variation and low interclass variance by mining the different features of change detection data.

Motivated by the aforementioned concerns, this study proposes a new end-to-end network, coined as DST-VGG, by combining dense Swin-Transformer-V2 and VGG16. More Swin-V2 blocks are used to build the UNet++ type main network, and the VGG16 encoder is used to build the CNN auxiliary network. The DST-VGG overcomes the defect of only local information in the CNN and the insufficient interpretation of low-level details in the transformer in most integrated models. On the other hand, the Swin-V2 main network belongs to the EF network, and the CNN branch belongs to LF. This structure provides the prechanged features, postchanged features, and change relation features (namely, the six-channel concatenation from the prechanged and postchanged images) for the accurate acquisition of change detection results. The CBAM and deep supervision also promote the fusion of the heterogeneous features and the rapidly stable convergence of the network, respectively. To have a better integration of the transformer and CNN, we propose a new multiscale module, mixed feature pyramid (MFP), which provides interlayer multiscale interaction information and intralayer multiscale information to supplement the UNet++ main network, which only captures interlayer multiscale information. We design a new decoder for the CNN branch with the VGG16 encoder and use the self-supervised strategy to train the encoder for extracting features so that the CNN branch can provide learnable and more discriminant semantic information. To sum up, the main contributions of this study are fourfold.

- 1) We propose an end-to-end hybrid network DST-VGG that possesses both advantages of the transformer and CNN

and overcomes respective disadvantages of the EF and LF network. This is a new deep learning paradigm of the integration of dense global features and detail local features for change detection in VHR images.

- 2) An MFP is proposed, for the first time, to provide interlayer interaction information and intralayer multiscale information. It is a plug-and-play module that has been experimentally proven to be effective not only in our proposed network but also with other change detection networks.
- 3) We design a new decoder for the CNN branch with the VGG16 encoder and impose the self-supervised strategy to train the VGG16 for extracting the features to provide more discriminative semantic information for the main network.
- 4) Compared with the existing state-of-the-art networks for change detection, our change detection scores and the elaborate change maps are better on four common public VHR datasets.

The rest of this article is organized as follows. Section II gives the review of the related work. Section III elaborates the proposed DST-VGG network. The experimental evaluations and ablation studies are given in Section IV. Finally, Section V concludes this article.

## II. RELATED WORK

### A. Transformer

Transformer is a popular deep learning method in natural language processing due to its capability of modeling global information [44]. The vision transformer (ViT) was proposed in [45] to bring the spirit of self-attention to image classification, but this huge model cannot be directly applied to other detection tasks in computer vision. Swin transformer adopts the inspiration of architectural hierarchy refinement and local information interaction, which reduces the heavy computation involved in modeling global information [46]. Liu et al. [47] proposed Swin-V2 that stabilizes the training process caused by the increase of model parameters and mitigates the resolution difference between the upstream and downstream tasks to obtain the advanced detection performance. At the same time, various transformer models were rapidly developed for remote sensing images processing [48], [49], [50], [51], [52], [53], [54].

Due to the lack of detection capability on low-level details in the pure transformer model [20], [55], the transformer is generally combined with the CNN for change detection in VHR images. There exists some combination methods. Chen et al. [26] proposed a bitemporal image transformer (BIT) that embeds the transformer to the CNN to enhance the capability of modeling contexts within the spatial-temporal domain. Based on the BIT, the TransUNetCD [56] was proposed to augment low-level detail information. However, it still has some defects because the serial combination of the transformer with the CNN is insufficient for generating discriminative features. The main reason for these defects is that both transformer features and CNN features at different stages are very important for change detection, while the single transformer in the series architecture is not enough

to provide rich global information [26], [56]. In the parallel associative approach, Feng et al. [57] used the transformer and CNN, respectively, to extract the pair change features for feature fusion, but they ignored the supplement of low-level details at the decoder stage. Our study adopts a new parallel approach with the global and detail features association to avoid the defect of the aforementioned models and obtain the better detection result.

### B. Multiscale Information

Multiscale information is an important approach in image processing. In deep learning, multiscale features can be divided into three categories: multiscale features between different layers, multiscale interaction features between different layers, and multiscale features from different convolution units. The first category is often implicit in a variety of classical networks, such as ResNet, FCN, and U-Net [58], [59], [60]. A feature pyramid transformer (FPT) uses two different transformers to interfuse the features between different layers to generate the second type of multiscale features [61]. The third type of multiscale features is provided by various convolution units, such as inception, dilated convolution, Res2Net-Conv, and SK-Conv.

In VHR images change detection, Feng et al. [30] extracted the multilevel intertemporal features through the double branches of shared weights, and then, performed the information fusion and features difference to obtain the robust change features. The multiscale decoupled convolution was constructed by using atrous convolutions with different dilation rates. The researchers embedded several of these convolutions in different layers to acquire the two types of multiscale features between and within layers [62]. Inspired by the FPT, we propose an MFP that provides interlayer multiscale interaction information and intralayer multiscale information to supplement the main network, which only gives interlayer multiscale information.

### C. Self-Supervised Learning (SSL)

Since the labeled data are labor intensive, SSL, in which a large amount of unlabeled data can be used to train networks and extract knowledge that are then transferred to downstream tasks, has recently flourished as a deep learning technology. Contrast learning is one of the representative SSL models [63], in which the image sample pairs are generated through the network, and the cost function is set to shorten the distance between the two positive samples as far as possible, and expand the distance between the positive and negative samples. The mask autoencoder is another important branch of the SSL [64].

In change detection, Chen and Bruzzone [65] used the bootstrap your own latent framework to pretrain the heterogeneous remote sensing images, and then, transferred it to the downstream change detection task. The use of the ViT encoder and random masking as the data enhancement technique further advances the SSL architecture [66]. The SSL has been shown to provide the discriminative semantic information when combined with the supervised technique to train the network [41]. Different from their work in [41], we design a new encoder–decoder architecture with VGG16 as the encoder, and apply the SSL to this

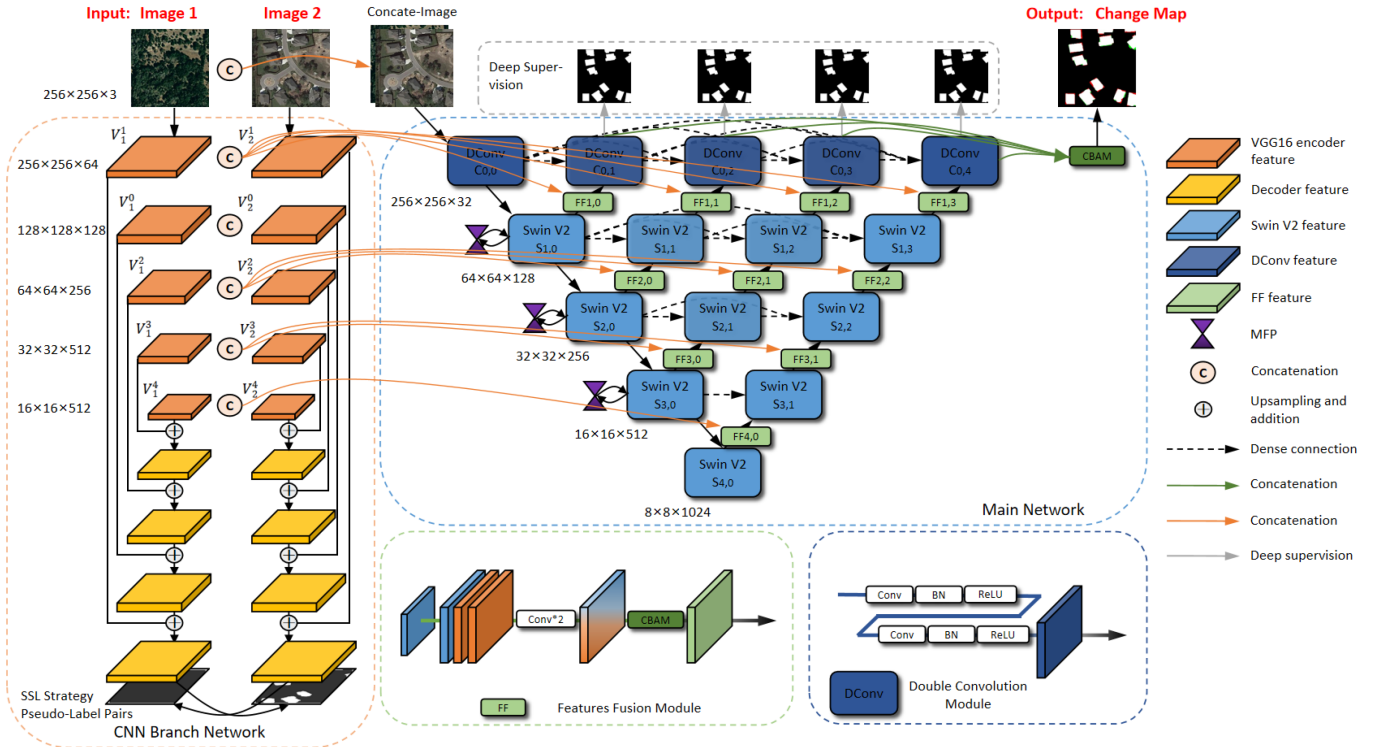


Fig. 2. Overall architecture of the DST-VGG. The output features of the VGG16 encoders are used in the DST (namely the main network) in which more Swin-V2 blocks are added for extracting discriminative features. In particular, the output features that have the same length and width as the input images are crucial for change detection in VHR images. The MFP provides diverse multiscale information. The SSL is imposed to guide the VGG16 to provide the semantic change information for the output features of the encoders.

branch architecture in providing the self-supervised semantic features for the main network.

### III. PROPOSED METHOD

In this section, we first introduce the DST-VGG architecture. Then, the Swin-V2 block, MFP module, CNN branch network, and SSL strategy are described in the order. The loss function of the model is then defined.

#### A. Proposed DST-VGG

The overall architecture of the DST-VGG is shown in Fig. 2. In conjunction with Algorithm 1, we elaborate on the details of the network. First of all, we initialize the parameters  $i$  and  $j$  to satisfy the following conditions:

$$\begin{aligned}
 &1 \leq i \leq 4, \quad i \in \mathbb{N} \\
 &0 \leq j \leq 3, \quad j \in \mathbb{N} \\
 &\text{when } j = 0, \quad i \in \{1, 2, 3, 4\} \\
 &\text{when } j = 1, \quad i \in \{1, 2, 3\} \\
 &\text{when } j = 2, \quad i \in \{1, 2\} \\
 &\text{when } j = 3, \quad i \in \{1\}
 \end{aligned} \tag{1}$$

where  $\mathbb{N}$  is the set of natural numbers,  $i$  is the number of layers of DST or VGG16, and  $j$  is the number of columns of the DST.

The CNN branch network consists of VGG16 as the encoder and our designed decoder. Then, we obtain the encoder features of the CNN branch network to compensate for the lack of low-level details and single image convolution features of the DST

$$V_1^i, V_2^i = \text{CNN}_{\text{SSL}}(I_1, I_2) \tag{2}$$

where  $\text{CNN}_{\text{SSL}}()$  refers to CNN branch network trained using the SSL strategy, and  $V_1^i, V_2^i$  and  $I_1, I_2$  represent the encoder feature pair and the original image pair, respectively, in this article. On another path in parallel, the features in the first column of the DST are generated

$$C_{0,0}, S'_{i,j=0} = \text{SwinV2}_{\text{backbone}}(\text{DConv}(\text{Cat}(I_1, I_2))) \tag{3}$$

where  $\text{SwinV2}_{\text{backbone}}()$ ,  $\text{DConv}()$ , and  $\text{Cat}()$  refer to Swin-V2 backbone [47], double convolution module (DConv), and the concatenation in the channel dimension, respectively;  $C$  denotes the feature from DConv; and  $S'$  denotes the feature in column 1 from Swin-V2 backbone. The features estimated by MFP that provides interlayer multiscale interaction information and intralayer multiscale information are given as follows:

$$S_{1,0}, S_{2,0}, S_{3,0} = \text{MFP}(S'_{1,0}, S'_{2,0}, S'_{3,0}); S_{4,0} = S'_{4,0} \tag{4}$$

where  $S$  refers to the feature that comes from MFP or Swin-V2 block.

**Algorithm 1:** Inference of DST-VGG Model for Change Detection

- Input:** Image 1  $I_1$ , Image 2  $I_2$ .  
**Output:** Change map CM.
- 1 **Initialization:** Set parameters  $i, j$  satisfying (1).
  - 2 **Acquisition of encoder features for CNN branch network:** Update  $V_1^i, V_2^i$  using  $I_1, I_2$  via (2).
  - 3 **Acquisition of features in column 1 for DST:** Update  $C_{0,0}, S'_{i,j=0}$  using  $I_1, I_2$  via (3).
  - 4 **Estimation of MFP features:** Update  $S_{1,0}, S_{2,0}, S_{3,0}$  and  $S_{4,0}$  by solving (4).
  - 5 **Acquisition of Swin-V2 and feature fusion features in rows 1 to 4 for DST:**
  - 6 **for**  $j = 0$  to 3 **do**
  - 7     **if**  $j == 1$  **then**
  - 8         Compute  $S_{i,1}$  by  $S_{i,0}, FF_{i+1,0}$  via (5);
  - 9     **else if**  $j == 2$  **then**
  - 10         Compute  $S_{i,2}$  by  $S_{i,0}, S_{i,1}$ , and  $FF_{i+1,1}$  via (6);
  - 11     **else if**  $j == 3$  **then**
  - 12         Compute  $S_{i,3}$  by  $S_{i,0}, S_{i,1}, S_{i,2}$ , and  $FF_{i+1,2}$  via (7).
  - 13     **end**
  - 14     Update  $FF_{i,j}$  using  $S_{i,j}, V_1^i$ , and  $V_2^i$  via (8).
  - 15 **end**
  - 16 **Estimation of other DConv features:** Compute  $C_{0,1}, C_{0,2}, C_{0,3}$  and  $C_{0,4}$  by solving (9)
  - 17 **Generation of change map:** Update CM via (10).

At this point, the rest of Swin-V2 and feature fusion features in rows 1 to 4 of main network can be generated

$$S_{i,1} = \text{SwinV2}(\text{Conv}(\text{Cat}(S_{i,0}, \text{Up}(FF_{i+1,0})))) \quad (5)$$

$$S_{i,2} = \text{SwinV2}(\text{Conv}(\text{Cat}(S_{i,0}, S_{i,1}, \text{Up}(FF_{i+1,1})))) \quad (6)$$

$$S_{i,3} = \text{SwinV2}(\text{Conv}(\text{Cat}(S_{i,0}, S_{i,1}, S_{i,2}, \text{Up}(FF_{i+1,2})))) \quad (7)$$

$$FF_{i,j} = \text{FFM}(S_{i,j}, V_1^i, V_2^i) \quad (8)$$

where  $\text{FFM}()$  represents feature fusion module,  $FF$  represents the feature fusion features,  $\text{SwinV2}()$  represents Swin-V2 block, and  $\text{Conv}()$  and  $\text{Up}()$  are used to adjust the spatial and channel resolutions of the features. Then, other DConv features are acquired as follows:

$$\begin{aligned} C_{0,1} &= \text{DConv}(\text{Cat}(C_{0,0}, FF_{1,0})) \\ C_{0,2} &= \text{DConv}(\text{Cat}(C_{0,0}, C_{0,1}, FF_{1,1})) \\ C_{0,3} &= \text{DConv}(\text{Cat}(C_{0,0}, C_{0,1}, C_{0,2}, FF_{1,2})) \\ C_{0,4} &= \text{DConv}(\text{Cat}(C_{0,0}, C_{0,1}, C_{0,2}, C_{0,3}, FF_{1,3})). \end{aligned} \quad (9)$$

Finally, we generate change map CM via CBAM [33] and sigmoid function

$$\text{CM} = \text{Sig}(\text{Conv}_{1 \times 1}(\text{CBAM}(\text{Cat}(C_{0,1}, C_{0,2}, C_{0,3}, C_{0,4})))) \quad (10)$$

where  $\text{Sig}()$  and  $\text{Conv}_{1 \times 1}()$  are sigmoid function and  $1 \times 1$  convolution, respectively. It should also be noted that dense connectivity allows the network to extract more diverse features. Deep supervision solves deep network feature shifting during

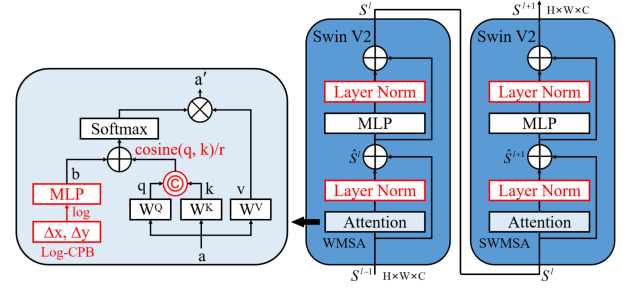


Fig. 3. Swin-V2 block. The improvements of Swin-V2 compared to Swin-V1 are marked in red. The notations and operations of Swin-V2 block are explained in Section III-B in detail.

training. The combination of these two techniques allows the quick and stable training and convergence of the network to take place. The FFM concatenates, nonlinearizes, and uses the CBAM to process the heterogeneous features for effective fusion. DConv is the union of convolution, batch normalization (BN), and ReLU function.

To summarize, the detection performance of the DST-VGG, which is a parallel hybrid architecture, can be attributed to the following reasons: the DST consisting of more Swin-V2 blocks is responsible for extracting the change relationship features. The CNN branch network complements the DST by providing the prechanged and postchanged detail features necessary for accurate detection. In particular, the detail features are preserved in the same size as the input images. Moreover, the incorporation of the multiscale features and SSL semantic features further enhances the overall feature learning of the network. These combined factors contribute to the DST-VGG for achieving the state-of-the-art detection performance.

### B. Swin-V2 Block

Swin-type transformers reduce the number of model parameters while modeling global information through shifted window and hierarchical mechanism [46]. Swin-V2 [47] further employs the postnormalization and scaled cosine attention techniques to improve the stability of the large vision model. At the same time, the log-spaced continuous position bias method is used to alleviate the problem of transferring the model trained on low-resolution images to high-resolution images. So, we use Swin-V2 as the base block to build the DST. This is illustrated in Fig. 3. Swin-V2 splits the image into the patches, and then, models the patches to generate the features. Its input and output are represented by  $S^{l-1}$  and  $S^l$ , respectively. We place layer normalization (LN) after multihead self-attention (MSA) and multilayer perceptron (MLP) to improve the stability of the network. The MSA includes window MSA (WMSA) and shifted window MSA (SWMSA): the former extracts the image features in the windows to reduce the computational cost, and the latter maintains the interaction of global information by sliding windows. The MLP is used to enhance the nonlinear capability of the block. The residual connection ensures the effective dissemination of information in the deep network. The

specific cooperation of these components is as follows:

$$\begin{aligned}\hat{S}^l &= \mathbf{WMSA}(\mathbf{LN}(S^{l-1})) + S^{l-1} \\ S^l &= \mathbf{MLP}(\mathbf{LN}(\hat{S}^l)) + \hat{S}^l \\ \hat{S}^{l+1} &= \mathbf{SWMSA}(\mathbf{LN}(S^l)) + S^l \\ S^{l+1} &= \mathbf{MLP}(\mathbf{LN}(\hat{S}^{l+1})) + \hat{S}^{l+1}\end{aligned}\quad (11)$$

where  $S$  and  $\hat{S}$  are the image features, and  $l$  is the number of Swin-V2 block layers.

Attention is the core of the entire block. The image features are first mapped into the three vectors: query ( $Q$ ), key ( $K$ ), and value ( $V$ ). Then, we use Sim function [namely (14)] to calculate the correlation weight matrix coefficients of  $Q$  and  $K$ , and normalize these weight matrix by Softmax. Finally, the dot product between the weight coefficients and  $V$  is obtained to form the self-attention features

$$\mathbf{Attention}(Q, K, V) = \mathbf{Softmax}(\mathbf{Sim}(Q, K))V \quad (12)$$

where  $Q$ ,  $K$ , and  $V$  are  $N \times d$  matrices, and  $N$  and  $d$  severally represent the number of patches and the dimension of single head self-attention. Instead of applying a single head self-attention, the MSA of Swin-V2 computes each head self-attention separately and concatenates these head self-attentions that represent different subspaces

$$\mathbf{MSA}(Q, K, V) = \mathbf{Cat}(\text{head}_1, \dots, \text{head}_H)W^O \quad (13)$$

where  $\text{head}_p = \mathbf{Attention}(QW_p^Q, KW_p^K, VW_p^V)$

where  $W_p^Q \in \mathbb{R}^{D \times d_k}$ ,  $W_p^K \in \mathbb{R}^{D \times d_k}$ ,  $W_p^V \in \mathbb{R}^{D \times d_v}$ , and  $W^O \in \mathbb{R}^{hd_v \times D}$  are parameter matrices, and  $h$  and  $D$ , respectively, represent the number of self-attention heads and the dimension of embedding layers. We set  $d_k = d_v = D/h$ .

Swin-V2 computes the attention logit of a pixel pair  $m$  and  $n$  by a scaled cosine function

$$\mathbf{Sim}(q_m, k_n) = \cos(q_m, k_n)/r + \mathbf{B}_{mn} \quad (14)$$

where  $\mathbf{B}_{mn}$  is the relative position bias between pixel  $m$  and  $n$ , and  $r$  is a learnable scalar, nonshared across heads and layers. Furthermore, Swin-V2 uses the log-spaced coordinates instead of the original linear-spaced ones to facilitate the transferring of the model between different resolution images. The log-spaced coordinates are then used as the input of  $\Phi$  to generate the bias values

$$\begin{aligned}\widehat{\Delta x} &= \text{sign}(x) \cdot \log(1 + |\Delta x|) \\ \widehat{\Delta y} &= \text{sign}(y) \cdot \log(1 + |\Delta y|)\end{aligned}\quad (15)$$

$$\mathbf{B}(\Delta x, \Delta y) = \Phi(\Delta x, \Delta y) \quad (16)$$

where  $\Phi$  is the two-layer MLP with an ReLU activation, while  $\Delta x$  and  $\Delta y$  are linear-scaled coordinates and  $\widehat{\Delta x}$  and  $\widehat{\Delta y}$  are the log-spaced coordinates. Read Swin-V2 [47] for more details of this block.

### C. Mixed Feature Pyramid (MFP)

The FPT utilizes different transformers to provide the self-attention features of different layers and the interaction features

between layers [61]. Self-transformer provides the self-attention features of three layers. Rendering transformer provides a down-top interaction feature between layers, and grounding transformer (GT) provides a top-down interaction feature. Inspired by the FPT and the requirements of the existing task, we construct a new multiscale module, MFP, by improving the FPT.

Since the basic blocks of current networks have evolved from convolutions to transformers, we remove the self-transformer. Moreover, we delete rendering transformer because it does not play a critical role on VHR images change detection experimentally. Thus, we only impose the GT to provide the interaction information between layers. However, in addition to multiscale features of different layers and multiscale interaction features between layers, multiscale information also includes multiscale features modeling within layers. So, we use the SK-Conv [37] and Res2Net-Conv [36] to provide the multiscale features inside different layers, respectively. Of course, other multiscale convolutions, for example, inception, dilated convolution, can be used here. We use the SK-Conv and Res2Net-Conv as an example to experimentally validate our proposed concepts. The MFP, FPT, SK-Conv, and Res2Net-Conv are shown in Fig. 4.

The SK-Conv and Res2Net-Conv are severally used to mine the intralayer multiscale information of the input features  $a_m$ ,  $b_m$ , and  $c_m$ . At the same time, the GT is utilized to provide the multiscale interaction features between different layers. These features are then rearranged according to different spatial resolutions. After residual blocks are added to the different layers, feature fusion is performed. Subsequently, using convolution, CBAM, and dropout in turn, we get the output features with intralayer multiscale information and multiscale interaction information between layers. The roles of the CBAM and dropout are to enhance the effective fusion of different features and prevent overfitting, respectively. The MFP is a plug-and-play module. It has also been shown effective in a variety of change detection models in our experiments.

We present the three extractors used for the multiscale features. The core operation of the GT is

$$\mathbf{GT}(Q_g, K_g, V_g) = \mathbf{Softmax}(Q_g K_g^T) V_g \quad (17)$$

where  $Q_g$ ,  $K_g$ , and  $V_g$  are the nonlinear transformations of  $X_g$ , and  $K_g^T$  is the transposition matrix of  $K_g$ . We define that  $X_g$ ,  $\bar{X}_g$ , and  $\tilde{X}_g$  are the input feature, intermediate variable, and output feature of GT, respectively. Take the layer  $a_m$  and  $b_m$  in Fig. 4 as an example to explain the complete operation of GT

$$\begin{aligned}\bar{X}_g^a &= \mathbf{Up}(\mathbf{BN}(\mathbf{Conv}(X_g^a))) \\ \bar{X}_g^b &= \mathbf{BN}(\mathbf{Conv}(X_g^b)) \\ \tilde{X}_g^{ab} &= \mathbf{GT}(\mathbf{Cat}(\bar{X}_g^a, \bar{X}_g^b))\end{aligned}\quad (18)$$

where  $\mathbf{BN}()$  stands for batch normalization, and the other symbols are represented as aforementioned. Both SK-Conv and Res2Net-Conv are the multiscale convolutions. SK-Conv obtains the features  $U_s$  and  $V_s$  by a  $3 \times 3$  convolution and a  $5 \times 5$  convolution, respectively. A joint SE attention is then applied for  $U_s$  and  $V_s$  to obtain  $\bar{U}_s$  and  $\bar{V}_s$ . In the end,  $\bar{U}_s$  plus  $\bar{V}_s$  gives the SK-Conv feature  $\tilde{X}_s$ . The multiscale component of

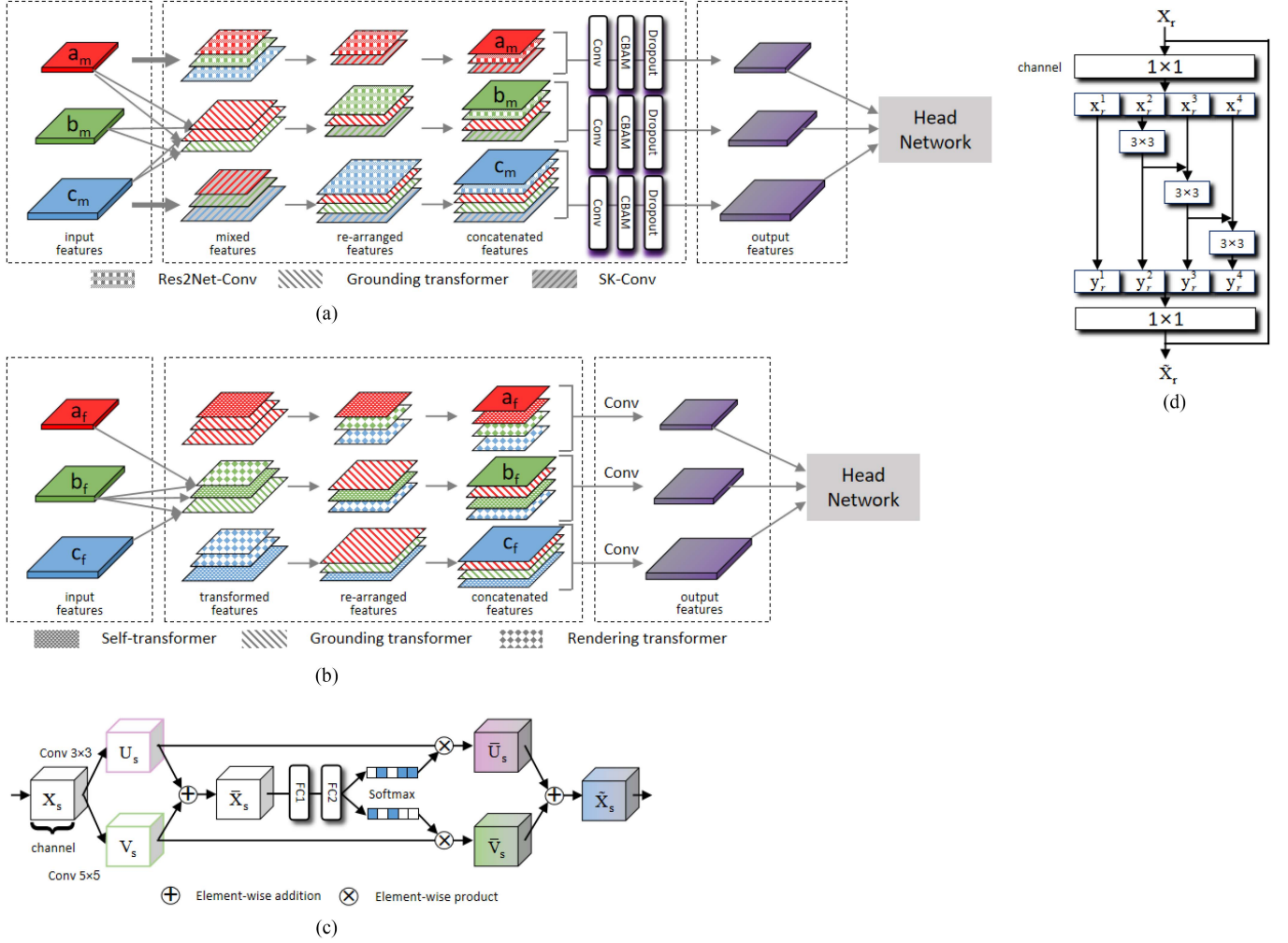


Fig. 4. (a) MFP, (b) FPT, (c) SK-Conv, and (d) Res2Net-Conv. Compared with the FPT, we remove self-transformer and rendering transformer in the MFP. We use GT, SK-Conv, and Res2Net-Conv to construct the MFP that provides interlayer interaction information and intralayer multiscale information.

Res2Net-Conv can be expressed as

$$y_r^z = \begin{cases} x_r^z & z = 1 \\ \text{Conv}(x_r^z) & z = 2 \\ \text{Conv}(x_r^z + y_r^{z-1}) & 2 < z \leq 4. \end{cases} \quad (19)$$

As shown in Fig. 4, we pass the input  $X_r$  through  $1 \times 1$  convolution (19), and  $1 \times 1$  convolution in turn, to get the Res2Net-Conv feature  $\tilde{X}_r$ .

#### D. CNN Branch Network and SSL Strategy

We design a new decoder and impose the SSL strategy [41] to optimize the low-level features of the VGG16 encoder. The detailed implementation is depicted in Fig. 5. We compose the decoder base block with convolution, BN, and ReLU. The five base blocks are arranged in order of the increasing spatial resolutions. The decoder features and the corresponding encoder features then are fused using upsampling and addition. Furthermore, the  $1 \times 1$  convolution is used to output the probability map of change detection.

For the SSL strategy, we generate the pseudolabels based on the probability maps through the CNN branch network

$$\text{PL}_{1,u} = \begin{cases} 0 & \text{PM}_{1,u} < 0.5 \\ 1 & \text{PM}_{1,u} \geq 0.5 \end{cases} \quad (20)$$

$$\text{PL}_{2,u} = \begin{cases} 0 & \text{PM}_{2,u} < 0.5 \\ 1 & \text{PM}_{2,u} \geq 0.5 \end{cases}$$

where  $u$  refers to each pixel of the probability maps or pseudolabels, and  $\text{PM}_{1,u}$  and  $\text{PM}_{2,u}$  are the two probability maps that are the outputs from the branch network. Since  $\text{PM}_{1,u}$  and  $\text{PM}_{2,u}$  are normalized to  $[0,1]$  by sigmoid, we can obtain the pseudolabels  $\text{PL}_{1,u}$  and  $\text{PL}_{2,u}$  via (20). After that, we determine the changed and unchanged regions based on the labels from the change detection datasets. In the unchanged regions, we use the pseudolabels generated by the one branch to supervise the other branch. In the changed regions, we use the opposite results of the pseudolabels from the one branch, to supervise the other branch. The goal is to keep the unchanged features as close as possible and the changed features as far away as possible. The

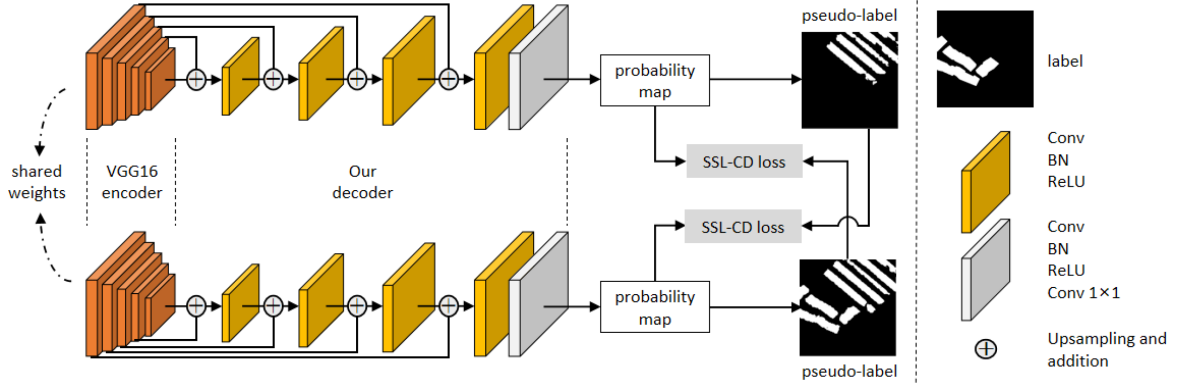


Fig. 5. CNN branch network and SSL strategy.

SSL-CD loss is defined as follows:

$$\begin{aligned}
 L_{SSL1} &= \mathbf{F}(\text{PM}_{1,u}, \text{PL}_{2,u} | u \in U_\alpha) \\
 &\quad + \mathbf{F}(\text{PM}_{1,u}, 1 - \text{PL}_{2,u} | u \in C_\alpha) \\
 L_{SSL2} &= \mathbf{F}(\text{PM}_{2,u}, \text{PL}_{1,u} | u \in U_\alpha) \\
 &\quad + \mathbf{F}(\text{PM}_{2,u}, 1 - \text{PL}_{1,u} | u \in C_\alpha) \quad (21)
 \end{aligned}$$

where  $U_\alpha$  and  $C_\alpha$  represent, respectively, the unchanged and changed regions,  $L_{SSL1}$  and  $L_{SSL2}$  are the two SSL-CD loss, and  $\mathbf{F}(\cdot)$  is a metric function. We choose  $L_{BCE}$  as  $\mathbf{F}(\cdot)$ . Therefore, the encoder features provided by the CNN branch network are rich in semantic information and have more discrimination ability for change detection.

### E. Loss Function

Bitemporal change detection is fundamentally a binary classification task, so binary cross entropy (BCE) loss is usually used as in the following:

$$L_{BCE} = -(t \log(\hat{t}) + (1 - t) \log(1 - \hat{t})) \quad (22)$$

where  $t$  and  $\hat{t}$  denote the predicted change confidence and the label in the corresponding position, respectively. For change detection tasks, however, the changed regions are far less than the unchanged regions, so there is a serious class imbalance problem in change detection. For example, the ratio of changed pixels to unchanged pixels in the season-varying change detection dataset (SVCD) is 0.046. To mitigate this problem, Dice loss is often used

$$L_{Dice} = 1 - \frac{2\hat{t}t + \sigma}{\hat{t} + t + \sigma}. \quad (23)$$

Here, adding  $\sigma$  avoids the case where the denominator is zero, and  $t$  and  $\hat{t}$  are similarly defined as in (22).

The loss function used in our model is a combination of BCE and Dice loss. Moreover, to address the features shift during the training of the deep network, we use the deep supervision strategy. Specifically, the deep supervision strategy uses the same labels as those used for the network outputs. The labels are replicated in four copies for the four deep supervision interfaces. DConvs  $C_{0,1}$ ,  $C_{0,2}$ ,  $C_{0,3}$ , and  $C_{0,4}$  output the probability

maps through convolution and sigmoid function. At last, we use  $\sum_{m=1}^4 L_{BCE}^m + \lambda_1 L_{Dice}^m$  to measure and optimize these probability maps by the deep supervision labels. The total loss function is expressed as follows:

$$L_{Total} = \sum_{m=1}^5 L_{BCE}^m + \lambda_1 L_{Dice}^m + \lambda_2 L_{SSL1} + \lambda_2 L_{SSL2} \quad (24)$$

where  $\sum_{m=1}^5 L_{BCE}^m + \lambda_1 L_{Dice}^m$  represents the output loss and the four deep supervision losses of the DST,  $L_{SSL1}$  and  $L_{SSL2}$  are the self-supervised losses of the CNN branch network, and  $\lambda_1$  and  $\lambda_2$  are the weight coefficients. We set  $\lambda_1$  and  $\lambda_2$  to 0.5 and 0.25, respectively.

## IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we briefly describe our experimental configuration and elaborate our experimental results analysis, ablation study, and network visualization in turn.

### A. Experimental Configurations

1) *Datasets*: Our model is tested on the four publicly available change detection datasets, achieving the state-of-the-art results. Due to GPU memory limitations, we crop the images into the nonoverlapping patches of size  $256 \times 256$  for the four datasets.

- 1) Learning, vision, and remote sensing change detection dataset (LEVIR-CD) [67] is a publicly available change detection resource for large buildings. It comprises 637 pairs of high-resolution (0.5 m) remote sensing images, each measuring  $1024 \times 1024$  pixels. We crop these images, and then, obtain 7120/1024/2048 pairs as the training/validation/test data.
- 2) SVCD [68] contains 11 pairs of multispectral images obtained from Google Earth with spatial resolutions ranging from 0.03 to 1 m. Following the dataset partitioning, we apply the data augmentation methods, image rotation, and image flipping, to the training set. As a result, we obtain a total of 60 000/3000/3000 training/validation/test pairs.
- 3) Wuhan university change detection dataset (WHU-CD) [69] focuses on the buildings change detection. It



features a pair of high-resolution (0.2 m) aerial images, each measuring  $32\,507 \times 15\,354$  pixels. Since there is not a general data partitioning scheme for WHU-CD, we cut these images into the nonoverlapping segments of size  $256 \times 256$  and randomly divide them into 6096/764/764 pairs for the training/validation/testing sessions, respectively.

- 4) Sun Yat-Sen University change detection dataset (SYSU-CD) [27] contains 20000 pairs of orthographic aerial images with the spatial resolution of 0.5 m taken in Hong Kong. Each image is  $256 \times 256$  pixels. We use the 12000/4000/4000 training/validation/testing pairs based on the dataset provider splitting. It is worth noting that SYSU-CD presents multiple types of changed objects in the more complex scenario, making it a particularly challenging dataset.

2) *Baseline and State-of-the-Art Methods*: We compare the DST-VGG with the baseline and state-of-the-art methods as follows. The first three methods serve as the baselines, while the last seven methods represent the advanced networks developed over the past three years. We implement these change detection networks using the publicly available codes and default hyper-parameters.

- 1) *FC-EF* [18]: Bitemporal change detection images are concatenated as a single input to FCN.
- 2) *FC-Siam-Diff* [18]: A siamese FCN is employed to extract the multilevel features, utilizing the differences of these features to detect the changed information.
- 3) *FC-Siam-Conc* [18]: The multilevel features are extracted and fused using a Siamese FCN with the cascaded architecture.
- 4) *IFNet* [25]: The CBAM is applied to the heterogeneous features at each level of the cascaded decoder, and deep supervision is used for the improved training of intermediate layers.
- 5) *SNUNet-CD* [28]: A combination of siamese structure and UNet++ is utilized to extract the high-level features.
- 6) *BIT* [26]: A serial hybrid network that embeds transformer into ResNet.
- 7) *DCFF-Net* [70]: A parallel pure CNN that combines VGG16 with UNet++, and integrates the CBAM and deep supervision.
- 8) *TransUNetCD* [56]: A serial cascaded hybrid network that embeds transformer into UNet.
- 9) *ICIF-Net* [57]: A parallel hybrid network focusing on the interaction and fusion of transformer and CNN.
- 10) *FCCDN* [41]: A supervised model with self-supervised strategy, producing the features rich in semantic information.
- 11) *P2V* [71]: A method for mining spatio-temporal features from two change images.

3) *Implementation Details*: We implement our model using PyTorch and train it on a single NVIDIA GeForce RTX 3090 GPU. During the training, we optimize the model with Adam optimizer. The batch size is set to 4. The learning rate is initially set to  $5 \times 10^{-5}$  and linearly decays to 0 over the course of 200 epochs.

4) *Evaluation Metrics*: F1-score is an index used to measure the performance of binary classification models, taking into account both precision and recall. So, we primarily employ F1-score with respect to the change category as the main evaluation metric. F1-score is shown as follows:

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN}. \quad (25)$$

In addition, we also report precision, recall, intersection over union (IoU) for the change category, and overall accuracy (OA). These metrics are defined as follows:

$$\text{precision} = \frac{TP}{TP + FP} \quad (26)$$

$$\text{recall} = \frac{TP}{TP + FN} \quad (27)$$

$$\text{IoU} = \frac{TP}{TP + FP + FN} \quad (28)$$

$$\text{OA} = \frac{TP + TN}{TP + TN + FP + FN} \quad (29)$$

where TP, TN, FP, and FN represent the number of true positive, true negative, false positive, and false negative, respectively.

## B. Experimental Results

1) *Results Analysis and Comparison*: Tables I and II present the overall comparison results for the test sets: LEVIR-CD, SVCD, WHU-CD, and SYSU-CD. Through quantitative analysis, our model has demonstrated the obvious improvements over other methods in the three key indicators, F1 score, IoU, and OA, across these datasets. This shows the significant advantage of our model over other models. In some scenarios, our model is worse than the IFNet and FC-Siam-Diff on the precision, and is worse than the DCFF-Net on the recall, because these models favor changed or unchanged regions. However, change detection is a pixel-level classification problem that considers both changed regions and unchanged regions, so the bias for a certain class is not conducive to the overall detection results.

Notably, our method outperforms the recent DCFF-Net by the F1 score improvement of 0.78/0.31/ 0.91/0.83 on the four datasets, underscoring the significance of both global information provided by transformer and local information represented by the CNN for change detection. Furthermore, our approach also shows a performance advantage over the two serial networks, BIT and TransUNetCD, reinforcing the superiority of the parallel combination of the transformer and CNN. In summary, our proposed model achieves the state-of-the-art results by leveraging a parallel architecture of transformer and CNN with the multiscale and self-supervised features that enhance its discrimination capabilities (refer to the analysis of Table III).

The comparison of visualization results on the four datasets is shown in Figs. 6 and 7. We use different colors to represent TP(white), TN(black), FP(green), and FN(red) in the change maps. It is observed that our model maintains the best structured state compared to other models from (a) and (c) of LEVIR-CD and (a), (b), and (e) of WHU-CD. In addition, our model demonstrates the superior performance in detecting dense small

TABLE I  
COMPARISON RESULTS ON THE THREE CHANGE DETECTION DATASETS

Method	LEVIR-CD					SVCD					WHU-CD				
	Pre. / Rec. / F1 / IoU / OA					Pre. / Rec. / F1 / IoU / OA					Pre. / Rec. / F1 / IoU / OA				
FC-EF	86.16 / 86.20 / 86.18 / 76.16 / 98.59					85.35 / 77.56 / 81.27 / 42.14 / 95.59					86.13 / 86.01 / 86.07 / 75.67 / 98.82				
FC-Siam-Diff	90.36 / 84.81 / 87.50 / 81.06 / 98.77					92.28 / 78.70 / 84.95 / 48.87 / 96.56					81.40 / 89.11 / 85.08 / 71.79 / 98.68				
FC-Siam-Conc	87.30 / 87.81 / 87.55 / 75.09 / 98.73					92.04 / 81.94 / 86.70 / 52.95 / 96.90					79.98 / 90.94 / 85.11 / 66.25 / 98.65				
IFNet	<b>93.73</b> / 87.31 / 90.40 / 84.04 / 99.06					97.71 / 93.64 / 95.63 / 81.31 / 98.94					<b>98.51</b> / 82.46 / 89.77 / 90.28 / 99.21				
SNUNet-CD	91.00 / 88.30 / 89.63 / 79.18 / 98.96					98.13 / 97.62 / 97.87 / 88.24 / 99.48					88.50 / 90.31 / 89.40 / 73.50 / 99.09				
BIT	91.81 / 88.00 / 89.86 / 79.35 / 98.99					97.07 / 96.43 / 96.75 / 83.52 / 99.20					92.10 / 92.41 / 92.26 / 78.56 / 99.34				
DCFF-Net	92.96 / 89.83 / 91.37 / 82.80 / 99.14					98.75 / 98.94 / 98.84 / 92.79 / 99.71					96.51 / 93.11 / 94.78 / 88.55 / 99.57				
TransUNetCD	90.62 / 88.44 / 89.52 / 79.63 / 98.94					97.44 / 96.52 / 96.98 / 84.81 / 99.26					94.72 / 91.21 / 92.93 / 83.82 / 99.41				
ICIF-Net	92.23 / 88.53 / 90.34 / 81.54 / 99.04					97.61 / 97.04 / 97.32 / 85.98 / 99.34					93.61 / 89.69 / 91.61 / 82.61 / 99.31				
FCCDN	92.10 / 84.86 / 88.33 / 80.48 / 98.86					95.96 / 95.56 / 95.76 / 79.79 / 98.96					92.65 / 90.44 / 91.53 / 79.94 / 99.29				
P2V	91.91 / 88.98 / 90.42 / 82.05 / 99.04					98.32 / 97.99 / 98.16 / 89.72 / 99.55					94.79 / 90.75 / 92.72 / 86.09 / 99.40				
Ours	92.98 / <b>91.33</b> / <b>92.15</b> / <b>85.44</b> / <b>99.21</b>					<b>99.18</b> / <b>99.12</b> / <b>99.15</b> / <b>94.02</b> / <b>99.79</b>					96.75 / <b>94.65</b> / <b>95.69</b> / <b>90.34</b> / <b>99.64</b>				

The best values are highlighted in bold font. All results are expressed as percentages (%).

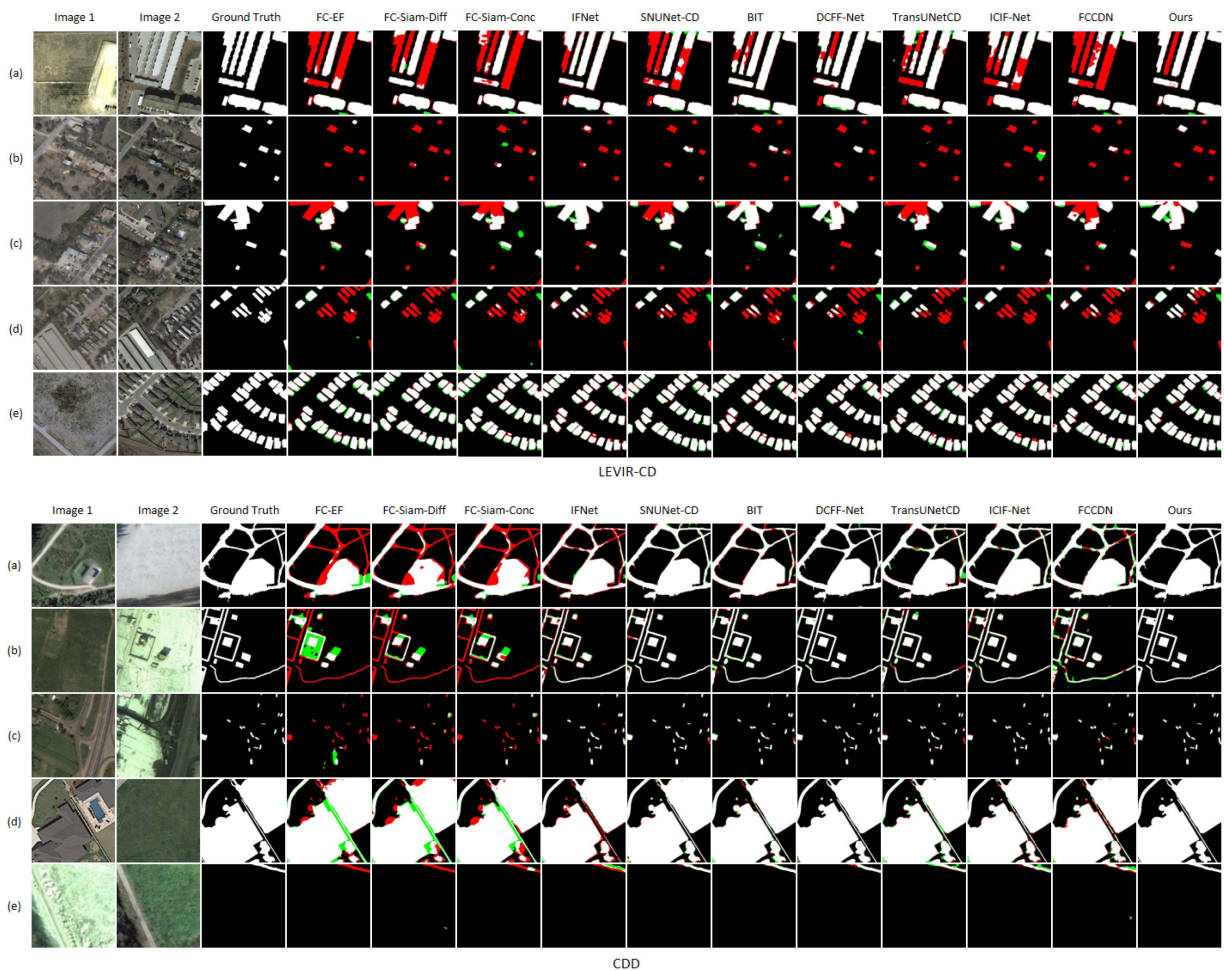


Fig. 6. Visualization results of various methods on the LEVIR-CD and SVCD test sets. We use different colors to represent TP(white), TN(black), FP(green), and FN(red) in the change maps. (a) and (e) Prediction results of these methods for different samples, respectively.

objects and edge objects, as evidenced by LEVIR-CD (d) and WHU-CD (d).

The SVCD dataset poses the unique challenges due to its varying illumination conditions and high occurrence of pseudo-changes, such as the snow cover is added in image 2 of (a),

and illumination differences are visible in (b) and (e) image pairs. However, our model outperforms other methods in these difficult scenes, particularly for the three detection requirements of structured changed regions, small targets, and edge targets. The challenge of the SYSU-CD dataset lies also in the high

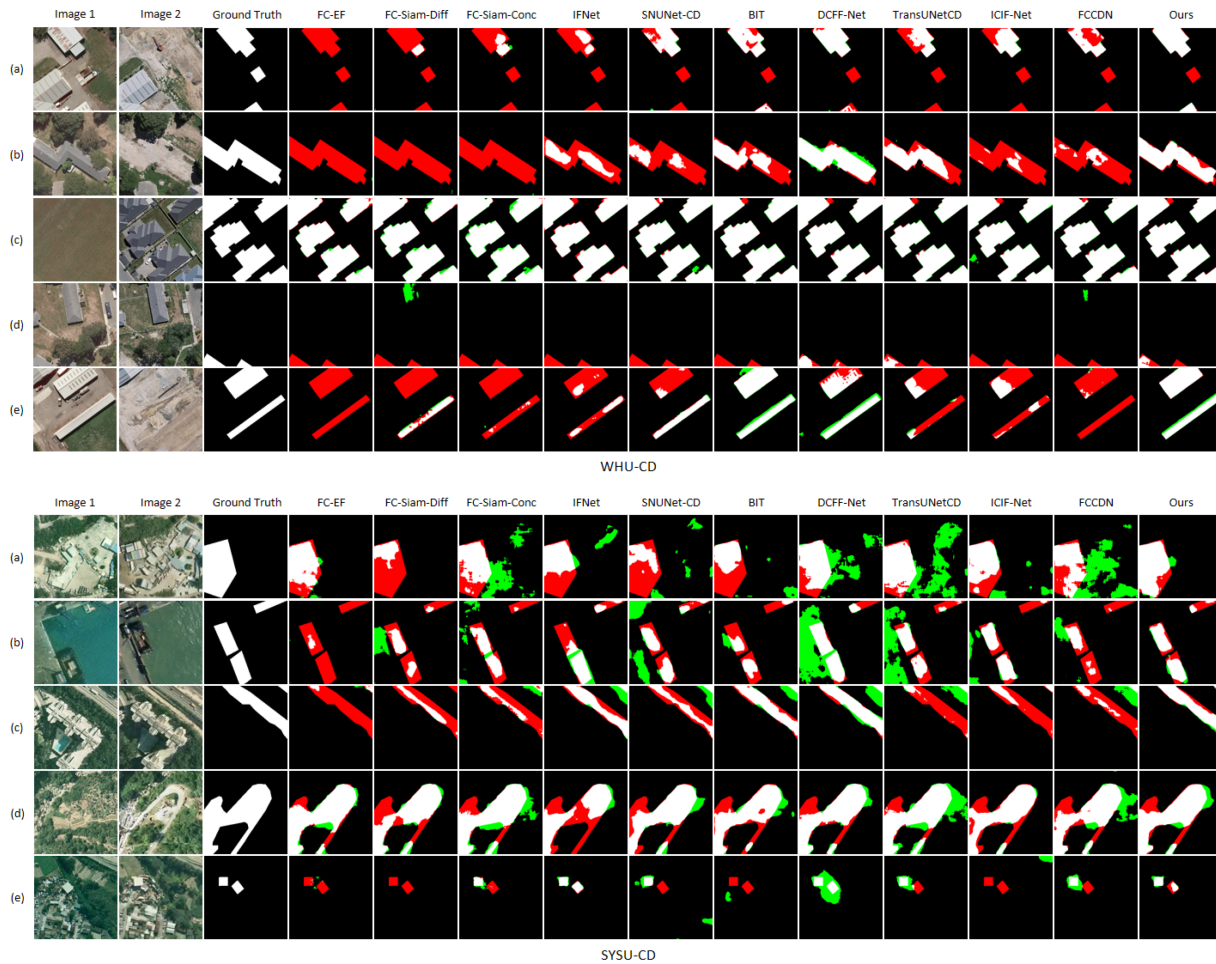


Fig. 7. Visualization results of various methods on the WHU-CD and SYSU-CD test sets. We use different colors to represent TP(white), TN(black), FP(green), and FN(red) in the change maps. (a) and (e) Prediction results of these methods for different samples, respectively.

TABLE II  
COMPARISON RESULTS ON SYSU-CD DATASET

Method	SYSU-CD				
	Pre. /	Rec. /	F1 /	IoU /	OA
FC-EF	79.30 /	68.84 /	73.70 /	44.64 /	88.41
FC-Siam-Diff	<b>89.80</b> /	58.49 /	70.84 /	42.37 /	88.64
FC-Siam-Conc	82.31 /	73.52 /	77.67 /	50.33 /	90.03
IFNet	85.16 /	75.36 /	79.96 /	57.22 /	91.09
SNUNet-CD	80.03 /	76.62 /	78.29 /	52.99 /	89.98
BIT	81.67 /	76.52 /	79.01 /	52.41 /	90.41
DCFF-Net	78.71 /	<b>86.24</b> /	82.30 /	62.05 /	91.25
TransUNetCD	77.25 /	80.17 /	78.68 /	54.72 /	89.75
ICIF-Net	78.53 /	78.89 /	78.71 /	53.80 /	89.94
FCCDN	78.57 /	78.14 /	78.36 /	53.33 /	89.82
P2V	80.76 /	76.77 /	78.71 /	54.79 /	90.21
Ours	83.46 /	82.81 /	<b>83.13</b> /	<b>62.90</b> /	<b>92.08</b>

The best values are highlighted in bold font. All results are expressed as percentages (%).

intra-class variation and low inter-class variance of background and targets. Our model is still able to maintain the structure of targets relatively well in the scenes where the targets are cluttered and close to the background. The detection results are well understood. Most of structured regions are obtained from the dense global information of transformer, and small targets

and edges are captured with the low-level detail information of the CNN. This once again validates the importance of our parallel hybrid architecture.

2) *Training Processes Analysis and Efficiency Comparison:* We evaluate the performance of the DST-VGG on the four datasets by tracking the two metrics, F1 score and loss, as depicted in Fig. 8. The F1 score and loss curves provide an intuitive understanding that our model is stably convergent and efficient. The peak values on the F1 curves are observed at the points 0.9212(LEVIR-CD), 0.9765(SVCD), 0.9544(WHU-CD), and 0.8100(SYSU-CD) for four datasets, indicating that our model requires a training process of just 35 epochs. Given the SVCD dataset’s abundance of small targets and intricate labelings, the DST-VGG exhibits a slight growth even after 35 epochs. Comparing the datasets, the SYSU-CD appears to be the most challenging and prone to the rapid overfitting. This could potentially be attributed to the different distribution between training data and validation data, wherein the overfitting on training data impairs the generalization capacity of the DST-VGG. Currently, no relevant studies have been conducted to explain this phenomena.

The further conjoint analysis with the F1 values of 0.9215(LEVIR-CD), 0.9771(SVCD), 0.9569(WHU-CD), and

TABLE III  
ABLATION STUDIES FOR THE OVERALL NETWORK ON THE THREE DATASETS

Overall Network			LEVIR-CD	SVCD	WHU-CD
Swin-V2	MFP	SSL	F1 / OA	F1 / OA	F1 / OA
			<b>91.37 / 99.14</b>	<b>94.81 / 98.73</b>	<b>94.78 / 99.57</b>
✓			91.85 / 99.18	97.58 / 99.41	95.34 / 99.61
	✓		91.47 / 99.15	96.15 / 99.06	94.80 / 99.57
		✓	91.38 / 99.13	96.66 / 99.18	95.25 / 99.60
✓	✓		91.97 / 99.19	97.66 / 99.43	95.10 / 99.59
✓		✓	92.00 / 99.19	97.60 / 99.41	95.31 / 99.61
	✓	✓	91.40 / 99.13	96.69 / 99.19	94.82 / 99.57
✓	✓	✓	<b>92.15 / 99.21</b>	<b>97.71 / 99.44</b>	<b>95.69 / 99.64</b>

Their F1 and OA scores are listed in the table. The base and best results are annotated in blue and red, respectively. All results are expressed as percentages (%).

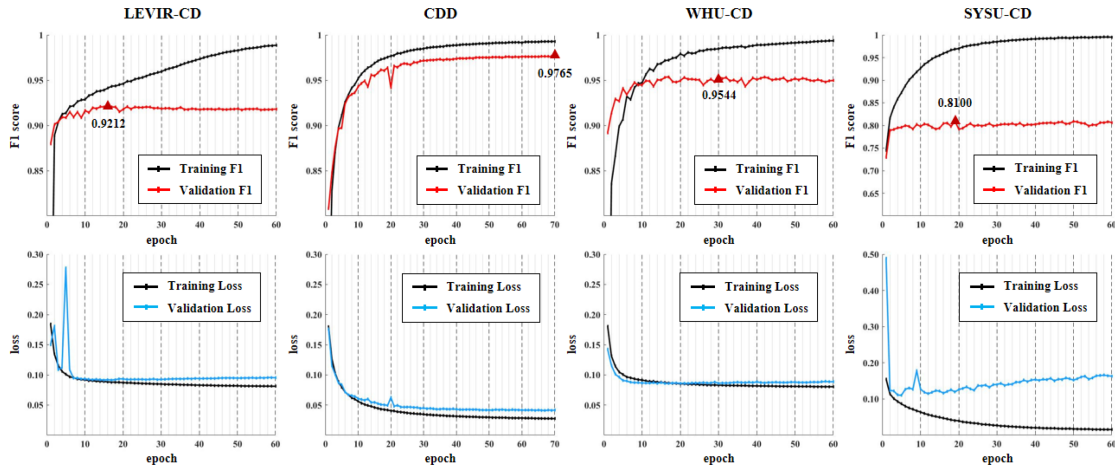


Fig. 8. Training processes analysis of the DST-VGG.

0.8313(SYSU-CD) in Tables I and II reaffirms the strong generalization ability of our model. With 0.9771(SVCD) value, it reveals that the F1 performance for the SVCD test set without applying any data augmentations. Based on the hybrid features, dense connection, and deep supervision in our model, the proposed network greatly contributes to the stability and efficiency of training processes. Among them, the combination of the VGG16 and Swin-V2 architecture plays a key role for the stable convergence of the model.

We report the number of parameters, FLOPs, F1, and OA of the compared models and our model on the SVCD dataset in Table IV. Although our model is expensive in the number of parameters and FLOPs, our detection results are the best. And remote sensing change detection does not have strict requirements for real-time performance and model size. So, it is necessary to pay a certain computational cost to obtain more accurate detection results in practical applications.

### C. Ablation Studies and Parameter Analysis

1) *Ablation Study of the Overall Network*: For the overall network architecture, our contributions consist of the three parts: Swin-V2 main network, MFP, and CNN branch network trained using the SSL strategy. As shown in Table III, we perform the ablation study for these three contributions on the LEVIR-CD,

TABLE IV  
EFFICIENCY COMPARISON OF METHODS

Method	Params.(M)	FLOPs(G)	SVCD F1 / OA
FC-EF	5.15	7.15	81.27 / 95.59
FC-Siam-Diff	6.98	9.43	84.95 / 96.56
FC-Siam-Conc	7.73	10.66	86.70 / 96.90
IFNet	197.41	164.51	95.63 / 98.94
SNUNet-CD	50.40	109.65	97.87 / 99.48
BIT	26.37	21.23	96.75 / 99.20
DCFF-Net	226.71	551.10	98.84 / 99.71
TransUNetCD	464.18	93.04	96.98 / 99.26
ICIF-Net	90.95	50.59	97.32 / 99.34
FCCDN	44.52	24.76	95.76 / 98.96
Ours	918.54	1430.43	99.15 / 99.79

We report the number of parameters (Params.), FLOPs, F1, and OA on the SVCD dataset. The results of F1 and OA are expressed as percentages (%).

SVCD, and WHU-CD datasets. According to the data analysis of Table III, Swin-V2 main network has the largest effect on the overall performance improvement, and the latter two have the similar effects. Especially on the SVCD dataset, the Swin-V2 main network shows an improvement of F1 score 2.77% compared to the CNN main network. The different combinations of two contributions occasionally occur the mutually exclusive phenomena. However, our model achieves the better results

TABLE V  
LONGITUDINAL DISSECTION (LEFT) AND HORIZONTAL PROMOTION (RIGHT) FOR MFP ON THE LEVIR-CD DATASET

GT	MFP		FC-Siam-Diff	Ours
	Res2Net-Conv	SK-Conv	F1 / OA	F1 / OA
			<b>87.50 / 98.77</b>	<b>91.845 / 99.177</b>
✓			87.49 / 98.74	91.955 / 99.192
	✓		87.89 / 98.79	92.057 / 99.191
		✓	87.66 / 98.74	91.915 / 99.186
✓	✓		87.78 / 98.78	91.975 / 99.190
✓		✓	87.50 / 98.75	91.866 / 99.178
	✓	✓	87.92 / 98.78	91.843 / 99.179
✓	✓	✓	<b>88.20 / 98.81</b>	<b>92.060 / 99.200</b>

The base and best results are annotated in blue and red, respectively. All results are expressed as percentages (%).

TABLE VI  
ABLATION STUDIES FOR SWIN-V2 ON THE THREE DATASETS

Swin-V2		LEVIR-CD	SVCD	WHU-CD
Pre-trained	Conf.	F1 / OA	F1 / OA	F1 / OA
	Conf.1	91.66 / 99.17	97.41 / 99.36	95.32 / 99.61
✓	Conf.1	91.77 / 99.17	97.46 / 99.38	<b>95.50 / 99.62</b>
✓	Conf.2	<b>91.85 / 99.18</b>	<b>97.58 / 99.41</b>	95.34 / 99.61
✓	Conf.3	91.79 / 99.18	97.42 / 99.37	95.26 / 99.60

We report F1 and OA scores. The best values are highlighted in bold font. All results are expressed as percentages (%).

than the baseline on the three datasets, using the three contributions together. Our ablation experiment uses F1 score as the main evaluation metric that is roughly positively correlated with OA. This also verifies the importance of our parallel architecture consisting of Swin-V2 main network and CNN branch network. The three types of multiscale features and the encoder semantic features guided by the SSL strategy also have clear guiding effects on the model performance.

2) *Parameter Analysis of Swin-V2*: We mainly perform the ablation study of the Swin-V2 blocks about the pretrained weights and number of blocks as described in Table VI. We only use two Swin-V2 blocks for  $S_{1,1}$ ,  $S_{1,2}$ , and  $S_{2,1}$ . For the U-shaped structure formed by  $[S_{1,0}, S_{2,0}, S_{3,0}, S_{4,0}, S_{3,1}, S_{2,2}, S_{1,3}]$ , we try the three configurations of Swin-V2 blocks:  $[2, 2, 2, 2, 2, 2]$ ,  $[2, 2, 6, 2, 6, 2, 2]$ , and  $[2, 2, 18, 2, 18, 2, 2]$ . The ablation results on the three datasets show that the combination of the pretrained weights and configuration of  $[2, 2, 6, 2, 6, 2, 2]$  has the advanced detection performance and robust application scenarios. This is due to the facts that the pretrained weights usually contain the prior information of upstream tasks, and too many Swin-V2 blocks maybe lead to the underfitting of some intermediate layer parameters of the model.

3) *Ablation Study of the Swin-V2 Architecture*: By performing the ablation study for the Swin-V2 architecture, we further have a clear understanding of the detection performance of different Swin-V2 architectures from Table VII. Similar to the UNet and UNet++, the performance of the Swin-V2++ is superior to the ability of the pure Swin-V2 architecture. The dense connection undoubtedly makes the extracted change features more diverse. However, the emphasis on detail features from the VGG16 is significantly more important for change detection. The combination of the VGG16 and Swin-V2++ greatly

Method	MFP	LEVIR-CD			
		Pre. / Rec. / F1 / IoU / OA			
FC-Siam-Conc		87.30 / 87.81 / <b>87.55</b> / <b>75.09</b> / <b>98.73</b>			
FC-Siam-Conc	✓	87.91 / 88.73 / <b>88.32</b> / <b>77.04</b> / <b>98.80</b>			
IFNet		93.73 / 87.31 / <b>90.40</b> / <b>84.04</b> / <b>99.06</b>			
IFNet	✓	93.68 / 88.02 / <b>90.76</b> / <b>84.44</b> / <b>99.09</b>			
SNUNet-CD		91.00 / 88.30 / <b>89.63</b> / <b>79.18</b> / <b>98.96</b>			
SNUNet-CD	✓	90.35 / 89.07 / <b>89.70</b> / <b>76.64</b> / <b>98.96</b>			
FCCDN		92.10 / 84.86 / <b>88.33</b> / <b>80.48</b> / <b>98.86</b>			
FCCDN	✓	89.95 / 87.66 / <b>88.79</b> / <b>79.91</b> / <b>98.87</b>			

TABLE VII  
ABLATION STUDIES FOR SWIN-V2 ARCHITECTURE ON THE TWO DATASETS

Swin-V2 Architecture	LEVIR-CD	WHU-CD
	F1 / IoU / OA	F1 / IoU / OA
Swin-V2	88.87 / 78.95 / 98.86	91.79 / 84.62 / 99.31
Swin-V2++	89.16 / 80.25 / 98.91	92.47 / 80.92 / 99.36
VGG16+Swin-V2++	<b>91.85 / 83.90 / 99.18</b>	<b>95.34 / 89.96 / 99.61</b>

We report F1, IoU and OA scores. The best values are highlighted in bold font. All results are expressed as percentages (%).

TABLE VIII  
ABLATION STUDY FOR THE SUPERVISED STRATEGIES OF CNN BRANCH NETWORK ON THE THREE DATASETS

Strategy	LEVIR-CD	SVCD	WHU-CD
	F1 / OA	F1 / OA	F1 / OA
Unsupervised	91.97 / 99.19	97.66 / 99.43	95.10 / 99.59
Supervised	91.97 / 99.19	97.68 / 99.43	95.39 / 99.62
Self-supervised	<b>92.15 / 99.21</b>	<b>97.71 / 99.44</b>	<b>95.69 / 99.64</b>

We report F1 and OA scores. The best values are highlighted in bold font. All results are expressed as percentages (%).

improves the change discrimination capability of the model on VHR remote sensing images.

4) *Ablation Study of the MFP*: We propose the MFP using the combination of GT, Res2Net-Conv, and SK-Conv. In the left subtable of Table V, we present a detailed ablation analysis for these three modules with FC-Siam-Diff and our model as the base lines on the LEVIR-CD dataset. In the right subtable of Table V, we further test the plug-and-play performance of the MFP based on the other four models. The data of the left subtable support the detection role of each module and the improvement of overall performance using the MFP. Specifically, the MFP improves F1 scores by 0.70% and 0.215% on FC-Siam-Diff and our model (our model has a high complexity), respectively. Through the analysis of the right subtable, FC-Siam-Conc, IFNet, SNUNet-CD, and FCCDN achieve the improvements of 0.77%, 0.36%, 0.07%, and 0.46% in F1 score, respectively. The experiments on the longitudinal dissection and horizontal promotion support that the MFP effectively improves the performance of change detection models by providing the interlayer interaction information and intralayer multiscale information.

5) *Ablation Study of the SSL Strategy*: We compare the performance gains brought by guiding the CNN branch network under the three strategies of unsupervised, supervised, and self-supervised, as shown in Table VIII. On the LEVIR-CD, SVCD,

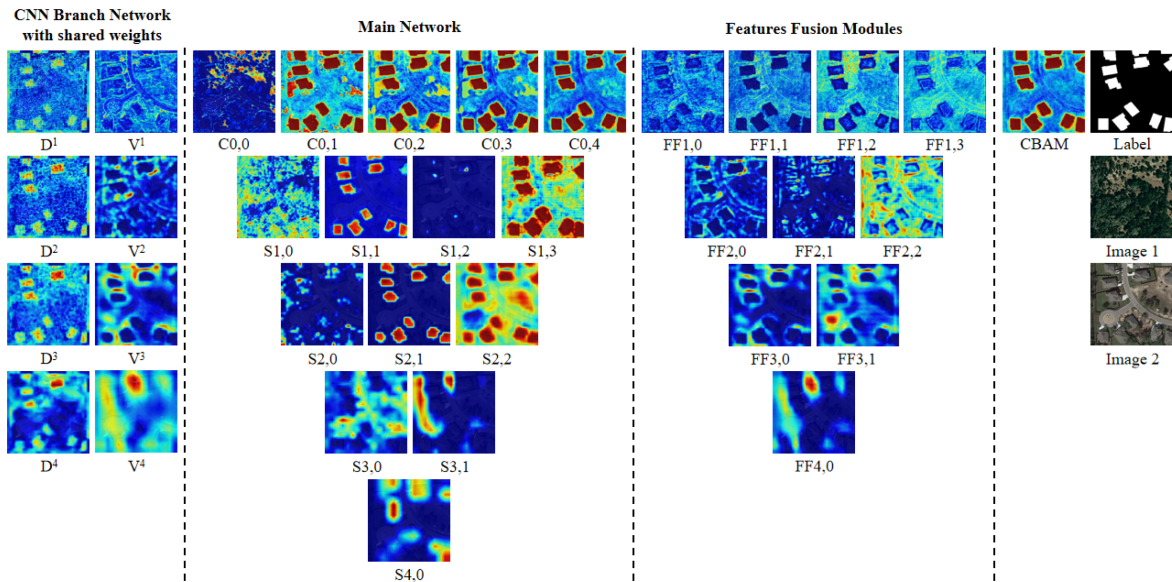


Fig. 9. Example for the visualization of the DST-VGG. It is obvious that the CNN branch network focuses on detailed information, while the main network focuses on structured global information for change detection. Feature fusion module fuses these two kinds of information, and then, obtains the change probability map by the CBAM.

and WHU-CD datasets, the self-supervised strategy achieves the best results. The reason for this phenomenon is that SSL provides the deep features with semantic information for change detection tasks [41].

#### D. Network Visualization

To further elucidate the practical effect of each network module, we conduct the analysis of network visualization. As depicted in Fig. 9, we broadly segment the network into three components: CNN branch network, main network, and feature fusion modules. Given that the CNN branch network is a dual-branch network with shared weights, we only present the activation maps for one branch. It is distinctly seen that the encoder layers furnish attention to the low-level detail information for main network. Conversely, the DST-VGG provides the structured abstract features. Feature fusion modules concentrate on detailing the differential specifics of remote sensing objects while preserving the structural information. The consolidation and retention of these two types of information are pivotal for change detection in VHR images. By contrasting the label and change activation map acquired through the CBAM, it becomes evident that our model exhibits the robustness in the intricate scene and different lighting condition.

#### V. CONCLUSION

In this article, we propose a new end-to-end hybrid network DST-VGG for change detection in VHR remote sensing images. The difference between our proposed network and other networks is that the output features of the VGG16 encoders in our model are used in the DST in which more Swin-V2 blocks are added for extracting discriminative features. Our network is specifically designed for change detection in VHR images.

It not only integrates the benefits of both the transformer and deep convolutional networks, but also successfully captures the features of change relationship via the DST and catches the detailed features in both prechanged and postchanged regions using the VGG16. Furthermore, the integration of detail features from the encoders of VGG16 and global features of the DST presents a new deep learning paradigm that can be effective for other image detection tasks, such as classification and object detection, in remote sensing. We also design an MFP module that can be seamlessly integrated with different image detection networks to provide the diverse multiscale features, bridging the feature gap between local and global aspects. We employ SSL policy that guides the VGG16 in delivering the encoder semantic features for the main network. Compared with other existing networks, our network results in the rapid but stable convergence and state-of-the-art performance on four commonly utilized public datasets of change detection. Moving forward, we will primarily focus on lightening the transformers in an effort to reduce the model complexity.

#### REFERENCES

- [1] M. Hussain, D. Chen, A. Cheng, H. Wei, and D. Stanley, "Change detection from remotely sensed images: From pixel-based to object-based approaches," *ISPRS J. Photogrammetry Remote Sens.*, vol. 80, pp. 91–106, 2013.
- [2] J. Zhang, X. Jia, J. Hu, and K. Tan, "Moving vehicle detection for remote sensing video surveillance with nonstationary satellite platform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5185–5198, Sep. 2021.
- [3] L. Bruzzone and M. Marconcini, "Domain adaptation problems: A DASVM classification technique and a circular validation strategy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 770–787, May 2010.
- [4] C. Benedek, X. Descombes, and J. Zerubia, "Building development monitoring in multitemporal remotely sensed image pairs with stochastic birth-death dynamics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 33–50, Jan. 2012.

- [5] P. Lu, Y. Qin, Z. Li, A. C. Mondini, and N. Casagli, "Landslide mapping from multi-sensor data through improved change detection-based Markov random field," *Remote Sens. Environ.*, vol. 231, 2019, Art. no. 11235.
- [6] S. Jin, L. Yang, Z. Zhu, and C. Homer, "A land cover change detection and classification protocol for updating Alaska NLCD 2001 to 2011," *Remote Sens. Environ.*, vol. 195, pp. 44–55, 2017.
- [7] Z. Zhu and C. E. Woodcock, "Continuous change detection and classification of land cover using all available Landsat data," *Remote Sens. Environ.*, vol. 144, pp. 152–171, 2014.
- [8] Z. Lv, T. Liu, J. A. Benediktsson, and N. Falco, "Land cover change detection techniques: Very-high-resolution optical images: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 10, no. 1, pp. 44–63, Mar. 2022.
- [9] Z. Liangpei and W. Chen, "Advance and future development of change detection for multi-temporal remote sensing imagery," *Acta Geodaetica et Cartographica Sinica*, vol. 46, no. 10, 2017, Art. no. 1447.
- [10] H. Zhuang, K. Deng, H. Fan, and M. Yu, "Strategies combining spectral angle mapper and change vector analysis to unsupervised change detection in multispectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 5, pp. 681–685, May 2016.
- [11] W. A. Malila, "Change vector analysis: An approach for detecting forest changes with Landsat," in *Proc. LARS Symposia*, 1980, Art. no. 385.
- [12] F. Bovolo and L. Bruzzone, "A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 1, pp. 218–236, Jan. 2007.
- [13] J. Zhang and Y. Zhang, "Remote sensing research issues of the national land use change program of China," *ISPRS J. Photogrammetry Remote Sens.*, vol. 62, no. 6, pp. 461–472, 2007.
- [14] J. Zhong and R. Wang, "Multi-temporal remote sensing change detection based on independent component analysis," *Int. J. Remote Sens.*, vol. 27, no. 10, pp. 2055–2061, 2006.
- [15] G. Xian and C. Homer, "Updating the 2001 national land cover database impervious surface products to 2006 using Landsat imagery change detection methods," *Remote Sens. Environ.*, vol. 114, no. 8, pp. 1676–1686, 2010.
- [16] 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.
- [17] J. L. Gil-Yepes, L. A. Ruiz, J. A. Recio, Á. Balaguer-Beser, and T. Hermosilla, "Description and validation of a new set of object-based temporal geostatistical features for land-use/land-cover change detection," *ISPRS J. Photogrammetry Remote Sens.*, vol. 121, pp. 77–91, 2016.
- [18] 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.
- [19] D. Zheng, Z. Wei, Z. Wu, and J. Liu, "Learning pairwise potential CRFs in deep Siamese network for change detection," *Remote Sens.*, vol. 14, no. 4, 2022, Art. no. 841.
- [20] C. Zhang, L. Wang, S. Cheng, and Y. Li, "SwinSUNet: Pure transformer network for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5224713.
- [21] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi, "Street-view change detection with deconvolutional networks," *Auton. Robots*, vol. 42, no. 7, pp. 1301–1322, May 2018.
- [22] 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.
- [23] 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. 2021.
- [24] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu, "Change detection based on deep Siamese convolutional network for optical aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1845–1849, Oct. 2017.
- [25] 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.
- [26] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1301–1322, 2021.
- [27] 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, 2021, Art. no. 5604816.
- [28] S. Fang, K. Li, J. Shao, and Z. Li, "SUNet-CD: A densely connected Siamese network for change detection of VHR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2021, Art. no. 8007805.
- [29] X. Zhang, S. Cheng, L. Wang, and H. Li, "Asymmetric cross-attention hierarchical network based on CNN and transformer for bitemporal remote sensing images change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 2000415.
- [30] 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, 2023, Art. no. 4401015.
- [31] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.
- [32] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11534–11542.
- [33] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.
- [34] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1–9.
- [35] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Res.*, 2016.
- [36] 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.
- [37] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 510–519.
- [38] S. Fang, K. Li, and Z. Li, "Changer: Feature interaction is what you need for change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5610111.
- [39] B. Hou, Q. Liu, H. Wang, and Y. Wang, "From W-Net to CDGAN: Bitemporal change detection via deep learning techniques," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 1790–1802, Mar. 2020.
- [40] W. Zhao, L. Mou, J. Chen, Y. Bo, and W. J. Emery, "Incorporating metric learning and adversarial network for seasonal invariant change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2720–2731, Apr. 2020.
- [41] P. Chen, B. Zhang, D. Hong, Z. Chen, X. Yang, and B. Li, "FCCDN: Feature constraint network for VHR image change detection," *ISPRS J. Photogrammetry Remote Sens.*, vol. 187, pp. 101–119, 2022.
- [42] H. Chen, W. Li, S. Chen, and Z. Shi, "Semantic-aware dense representation learning for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5630018.
- [43] Y. Zhang, Y. Zhao, Y. Dong, and B. Du, "Self-supervised pretraining via multimodality images with transformer for change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5402711.
- [44] A. Vaswani et al., "Attention is all you need," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2017, vol. 30, pp. 6000–6010.
- [45] A. Dosovitskiy et al., "An image is worth 16 x 16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Res.*, 2020.
- [46] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10012–10022.
- [47] Z. Liu et al., "Swin transformer V2: Scaling up capacity and resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 12009–12019.
- [48] D. Hong et al., "Spectralformer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5518615.
- [49] L. Sun, G. Zhao, Y. Zheng, and Z. Wu, "Spectral-spatial feature tokenization transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5522214.
- [50] L. Gao, B. Liu, P. Fu, and M. Xu, "Adaptive spatial tokenization transformer for salient object detection in optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5602915.
- [51] X. He, Y. Zhou, J. Zhao, D. Zhang, R. Yao, and Y. Xue, "Swin transformer embedding UNet for remote sensing image semantic segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4408715.
- [52] X. Liao, B. Tu, J. Li, and A. Plaza, "Class-wise graph embedding-based active learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5522813.
- [53] B. Tu, X. Liao, Q. Li, Y. Peng, and A. Plaza, "Local semantic feature aggregation-based transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5536115.

- [54] B. Tu, Q. Ren, Q. Li, W. He, and W. He, "Hyperspectral image classification using a superpixel-pixel-subpixel multilevel network," *IEEE Trans. Instrum. Meas.*, vol. 72, 2023, Art. no. 5013616.
- [55] W. G. C. Bandara and V. M. Patel, "A transformer-based siamese network for change detection," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 207–210.
- [56] Q. Li, R. Zhong, X. Du, and Y. Du, "TransUNetCD: A hybrid transformer network for change detection in optical remote-sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5622519.
- [57] Y. Feng, H. Xu, J. Jiang, H. Liu, and J. Zheng, "ICIF-Net: Intra-scale cross-interaction and inter-scale feature fusion network for bitemporal remote sensing images change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4410213.
- [58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 770–778.
- [59] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 3431–3440, 2014.
- [60] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf., Medical Image Comput. Comput.-Assisted Interv.*, 2015.
- [61] D. Zhang, H. Zhang, J. Tang, M. Wang, X. Hua, and Q. Sun, "Feature pyramid transformer," in *Proc. Comput. Vision—ECCV 2020, 16th Eur. Conf.*, Aug. 2328, 2020, pp. 1–17.
- [62] T. Lei et al., "Ultralightweight spatial–spectral feature cooperation network for change detection in remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4402114.
- [63] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
- [64] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16000–16009.
- [65] Y. Chen and L. Bruzzone, "Self-supervised change detection in multi-view remote sensing images," 2021, *arXiv:2103.05969*.
- [66] Y. Zhang, Y. Zhao, Y. Dong, and B. Du, "Self-supervised pre-training via multi-modality images with transformer for change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5402711.
- [67] 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.
- [68] 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.
- [69] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multi-source 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.
- [70] F. Pan, Z. Wu, Q. Liu, Y. Xu, and Z. Wei, "DCFF-Net: A densely connected feature fusion network for change detection in high-resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 11974–11985, 2021.
- [71] M. Lin, G. Yang, and H. Zhang, "Transition is a process: Pair-to-video change detection networks for very high resolution remote sensing images," *IEEE Trans. Image Process.*, vol. 32, pp. 57–71, 2022.



**Dalong Zheng** received the B.S. degree in software engineering and M.S. degree in agricultural informatization from Inner Mongolia Agricultural University, Hohhot, China, in 2012 and 2017, respectively. He is currently working toward the Ph.D. degree in computer science and technology with the Nanjing University of Science and Technology, Nanjing, China. His research interests include change detection, deep learning, and conditional random fields.



**Zebin Wu** (Senior Member, IEEE) received the B.Sc. and Ph.D. degrees in computer science and technology from the Nanjing University of Science and Technology (NJUST), Nanjing, China, in 2003 and 2007, respectively.

He was a Visiting Scholar with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, Escuela Politécnica, University of Extremadura, Cáceres, Spain, from 2014 to 2015. He was a Visiting Scholar with the Department of Mathematics, University of California at Los Angeles, Los Angeles, CA, USA, from August 2016 to September 2016 and from July 2017 to August 2017. He was a Visiting Scholar with the GIPSA-Lab, Grenoble INP, the Université Grenoble Alpes, Grenoble, France, from August 2018 to September 2018. He is currently a Professor with the School of Computer Science and Engineering, NJUST. His research interests include hyperspectral image processing, parallel computing, and Big Data processing.



**Jia Liu** (Member, IEEE) received the B.S. and Ph.D. degrees in electronic engineering 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.



**Chih-Cheng Hung** (Member, IEEE) received the B.S. degree in business mathematics from Soochow University, Taipei, Taiwan, and the M.S. and Ph.D. degrees in computer science from the University of Alabama, Huntsville, AL, USA, in 1986 and 1990, respectively.

He is currently a Professor of computer science with Kennesaw State University (KSU), Kennesaw, GA, USA, where he is the Director with the Center for Machine Vision and Security Research. He also holds the position of YinDu Scholar with National Anyang University, Anyang, China. His research interests include image processing, pattern recognition, machine learning, neural networks, genetic algorithms, and artificial intelligence.

Dr. Hung served as the Conference Chair for the Association of Computing Machinery (ACM) Symposium on Applied Computing (SAC 2019) to be held in Limassol, Cyprus, in 2019.



**Zhihui Wei** (Member, IEEE) was born in Jiangsu, China, in 1963. He received the B.Sc. and M.Sc. degrees in applied mathematics and the Ph.D. degree in communication and information system from Southeast University, Nanjing, China, in 1983, 1986, and 2003, respectively.

He is currently a Professor and a Doctoral Supervisor with the Nanjing University of Science and Technology, Nanjing. His research interests include partial differential equations, mathematical image processing, multiscale analysis, sparse representation, and

compressive sensing.