Asymmetric Cross-Attention Hierarchical Network Based on CNN and Transformer for Bitemporal Remote Sensing Images Change Detection

Xiaofeng Zhang, Shuli Cheng, Liejun Wang, and Haojin Li

***Abstract*— As an important task in the field of remote sensing (RS) image processing, RS image change detection (CD) has made significant advances through the use of convolutional neural networks (CNNs). The transformer has recently been introduced into the field of CD due to its excellent global perception capabilities. Some works have attempted to combine CNN and transformer to jointly harvest local-global features; however, these works have not paid much attention to the interaction between the features extracted by both. Also, the use of the transformer has resulted in significant resource consumption. In this article, we propose the Asymmetric Cross-attention Hierarchical Network (ACAHNet) by combining CNN and transformer in a series-parallel manner. The proposed Asymmetric Multiheaded Cross Attention (AMCA) module reduces the quadratic computational complexity of the transformer to linear, and the module enhances the interaction between features extracted from the CNN and the transformer. Different from the early and late fusion strategies employed in previous work, the effectiveness of the mid-term fusion strategy employed by ACAHNet shows a new choice of timing for feature fusion in the CD task. Our experiments on the proposed method on three public datasets show that our network has a better performance in terms of effectiveness and computational resource consumption compared to other comparative methods.**

***Index Terms*— Asymmetric cross-attention, change detection (CD), convolutional neural network (CNN), deep learning (DL), transformer.**

I. Introduction

REMOTE sensing (RS) image data has gained tremendous growth with the development of different types of sensors. This growth in data has driven the geoscience and RS community to apply deep learning (DL) algorithms to solve different RS tasks [1]. Change detection (CD) is receiving increasingly widespread attention in these tasks. CD is defined in [2] as "The process of identifying differences in the state of the same object or phenomenon by observing it at different times." Specifically, the CD system aims to compare two coregistered RS images taken at different times to assign a binary label to each pixel [3]. The label indicates whether the region corresponding to that pixel has changed between shots. As shown in Fig. 1, in the change map, white indicates changed and black indicates unchanged. CD techniques have many applications in areas, such as deforestation assessment [2], urbanization monitoring [4], [5], disaster assessment [6], [7], environmental monitoring [8], agricultural monitoring [9], and many more. Developments in CD technology have made ground monitoring more efficient, reducing labor costs and time consumption.

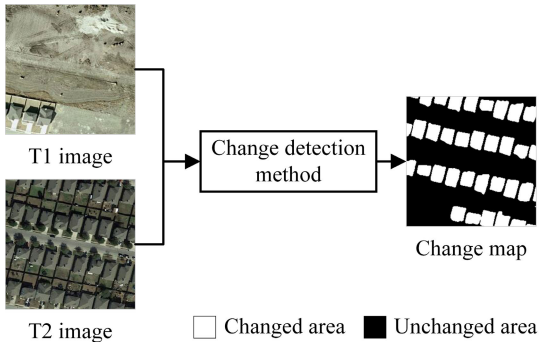

Fig. 1. Graphical illustration of the CD task.

The rapid growth of RS image data has placed new demands on CD techniques. First, during the acquisition of bitemporal RS imagery, changes in light angle or intensity, changes in seasons, sensor type, and other imaging conditions can affect the representation of the same semantic object in a given image pair. In Fig. 2(a), we can see a clear difference in appearance between the two images under different lighting conditions. Second, as high-resolution RS images have complex texture information and limited spectral information [10], this can make it more difficult to align RS images to geometry, leading to some registration errors. These registration errors can have an impact on the reliability of the CD results. The presence of significant registration errors can be observed in Fig. 2(b). Third, the definition of change varies depending on the application scene. For example, in Fig. 2(c), it can be observed that changes in temporary objects, such as cars, are

Manuscript received 30 November 2022; revised 17 January 2023; accepted 13 February 2023. Date of publication 15 February 2023; date of current version 27 February 2023. This work was supported in part by the Scientific and Technological Innovation 2030 Major Project under Grant 2022ZD0115802, in part by Xinjiang Uygur Autonomous Region Key Laboratory Open Project under Grant 2022D04028, in part by the Natural Science Foundation of Xinjiang Uygur Autonomous Region under Grant 2022D01C82, and in part by the National Science Foundation of China under Grant U1903213. *(Corresponding author: Shuli Cheng.)*

Xiaofeng Zhang, Liejun Wang, and Haojin Li are with the College of Information Science and Engineering, Xinjiang University, Ürümqi 830046, China (e-mail: zxf1332@stu.xju.edu.cn; wljxju@xju.edu.cn; lhj96599@stu.xju.edu.cn).

Shuli Cheng is with the College of Information Science and Engineering and the College of Mathematics and System Science, Xinjiang University, Ürümqi 830046, China (e-mail: cslxju@xju.edu.cn).

Digital Object Identifier 10.1109/TGRS.2023.3245674

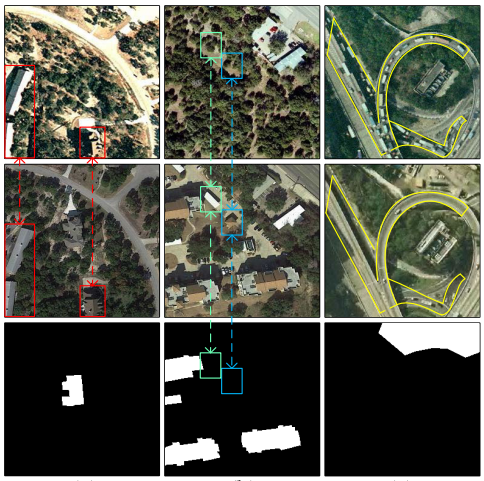

Fig. 2. Variety of irrelevant changes affect change detection results. (a) Light causes a change in roof color. (b) Some registration errors. (c) Moving cars changes.

not labeled as changes in the label. These uncorrelated changes can interfere with the detection of changes of interest. Finally, the number of pixels in the unchanged region of the sample is much greater than the number of pixels in the changed region. This is the problem of sample imbalance. In general, with the problems mentioned above, learning the different information of sample features effectively and improving the accuracy and efficiency of CD are the problems that need to be solved for RS CD at present.

In response to the above problems in CD, researchers have proposed a number of methods from various perspectives. As far as traditional methods are concerned, they can be divided into three categories: Methods based on algebraic operations on images [11], methods based on image transformations [12], and postclassification-based approach [13]. Among the methods based on algebraic operations on images, change vector analysis (CVA) [11] uses the direction and magnitude of the change vector (often expressed as a Euclidean distance) to analyze the type of change. In this process, a threshold is set to distinguish between changed and unchanged regions. Image transformation-based methods use feature representations extracted from the original image to distinguish between changed and unchanged pixels. Seo et al. [14] used random forest regression to build an SAR image incorporating coarse-grained surface features of synthetic aperture radar (SAR) images and spectral features of multispectral (MS) images for CD. Ahlqvist [15] used a postclassification approach to land cover change analysis using category semantics as a basis; however, the effectiveness of this approach is highly dependent on the effectiveness of prelearned features. In addition, traditional machine learning methods, such as K-nearest neighbors (K-NN) [16], support vector machines (SVMs) [17], [18], Decision Trees (DTs) [19], and Markov random field models [20], [21] have also been used to implement CD.

Traditional CD methods require tedious preprocessing of the images to be detected and the construction of complex supervised or unsupervised classification algorithms. Methods constructed in this case are less robust. In contrast, DL methods have end-to-end characteristics and nonlinear mapping capabilities, avoiding the tedious phase of manual feature design. This provides ideas for exploring new CD methods. Examples include stacked autoencoders (SAEs) [22] deep belief networks (DBNs) [23], and convolutional neural networks (CNNs) [24]. Among them, the mechanism of weight sharing and local connectivity of CNN gives them the ability to extract refined and multilevel abstract features, which are widely used in both computer vision [25] and RS [26]. As the CD network requires two images as input, the difference in the way the two images are processed will also affect the final detection result. Some of the earlier approaches connected coregistered bitemporal images as a whole as input to the network. For example, FC-EF, proposed by Daudt et al. [3], concatenates bitemporal images and extracts features through convolutional networks to detect changes. This type of method is called an "early fusion networks" [27], [28], [29]. As early fusion networks processed images after cascading, the networks were not able to extract the respective single-temporal features of the bitemporal images well, and fine single-temporal features are clearly important for the later detection of feature differences. Late fusion networks based on weight sharing [30], [31], [32], on the other hand, use two branches of the network that extract features to obtain single-temporal features separately, which are fused by various strategies and then used as input to the CD branch. For example, Chen and Shi [33] used two branches of a conjoined network to extract single-temporal features of a bitemporal images in parallel, using Euclidean distance to measure the differences between features to eventually generate a change map.

The features learned in the shallower layers of the late fusion network are, however, not well represented, and the information extracted in the deeper layers suffers from a loss of contextual relevance due to the local connectivity nature of the CNN. The CD network proposed based on UNet [34] uses skip connections to communicate deep and shallow information, reducing the loss of detail to a certain extent. Based on the idea of dense connectivity in UNet^{++} [35], the SNUNet proposed by Fang et al. [28] improves the ability to discriminate between changed and unchanged regions by combining long- and short-connected architectures to superimpose and fuse features at different levels, thereby reducing the semantic differences in the feature maps during fusion. Recurrent neural network (RNN) excels at handling sequential relationships and can exploit the temporal dependence between bitemporal images [36], [37] for CD. Mou et al. [36] first combined CNN and RNN for CD in MS images. The insufficient temporal information of bitemporal high-resolution RS images, however, leads to the limited temporal information used by these methods.

The fully CNN-based model attempts to capture long-range dependencies using methods that include deepening the network structure [28] and using dilated convolution [38]. Various forms of attention mechanisms [28], [38], [39], [40] have also been introduced to the CD task to reduce the interference of redundant features and to enhance global information

dependence. Channel attention [28], [41] and spatial attention [39] were used to weight the channel and spatial representations of fused features to capture features with more discriminative properties. Liu et al. [39] introduced a dual-attention module with channel attention and spatial attention in the network to enhance the feature representation. Approaches that simply use channel attention or spatial attention, however, still fall short in terms of long-range contextual associations in the spatiotemporal dimension due to the inherent limitations of the local nature of convolutional operations. This deficiency may lead to uncertainty in edge pixels and missed detection of tiny objects. The emergence of self-attention [33], [29] mechanisms with nonlocal attention capabilities provides new ideas for capturing long-range dependencies and thus obtaining more representative feature representations. Chen and Shi [33] used a self-attention module to process feature representations at multiple scales in order to better adapt to the scale variations of the object to be examined, capturing the spatiotemporal dependencies at different scales.

Based on self-attention, the transformer [42] adds feedforward networks (FFNs), weight tying and positional encoding to compute global contextual relations. Transformers have gained wide application in tasks, such as natural language modeling [43] and speech recognition [44]. Vision Transformer (ViT) [45] directly feeds image patch sequences into a pure transformer applied to image classification tasks. This approach, however, requires a large number of computational resources. Swin transformer [46] improves computational efficiency using a shifted window mechanism that allows cross-window connections. Recently, transformers have also been introduced into the CD field in various forms. Some CD networks rely solely on transformers for their construction. For example, Zhang et al. [47] designed a pure transformer CD network, SwinSUNet, based on the Swin transformer module to build the codec. The good performance of the model shows that solving CD tasks can be done without relying on convolution operations. ChangeFormer [48] extracts coarse-grained and fine-grained features from bitemporal images by constructing a hierarchical transformer encoder for twin networks. A lightweight MLP decoder is then used to fuse the multiscale difference features computed by the difference module. Hybrid models using CNN in combination with transformers also have great potential for solving CD tasks. The Bit-CD proposed by Chen et al. [49] uses transformers as encoders for contextual modeling in the space-time domain. The features extracted by the convolutional network are effectively enhanced. The Bit-CD sequential use of convolutional operations with the transformer module is designed to ensure the efficiency of the network. TransUNetCD [10] introduces a transformer module in the UNet architecture to form a CNN-transformer encoder to encode the tokenized image blocks. Most CD networks combining CNN and transformer use a CNN network as the backbone to extract semantic features and then use transformer blocks to encode or decode them. This combination of CNN and transformer in series ensures efficiency but lacks the mutual information interaction between CNN and transformer. ICIF-Net [13] combines a CNN with a transformer in parallel. ICIF-Net extracts and fuses the semantic features of a bitemporal image in parallel and then performs an intrascale cross-interaction module and an interscale feature fusion module on the dual-branch features, which together provide fine-grained detail and target-level interest for the CD of RS image pairs.

Based on the above, we can realize that purely solving CD tasks using CNN is not conducive to long-range information interaction. Pure transformer CD networks are able to capture remote dependency features to some extent, but the global attention has quadratic computation complexity associated with the size of the input. Therefore, the combination of CNN and transformer may have more potential in solving CD tasks. Our CD network chose to combine CNN and transformer in its basic idea. The way of combining the two can affect the inference efficiency and performance of the network, and we have drawn on the idea of Bit-CD to avoid feeding high-resolution images directly into the transformer module by using a series connection between modules. As ICIF-Net demonstrates the advantage of interacting with the information extracted from both by connecting CNN and transformer in parallel, we combine the CNN with the transformer in parallel within the module. In addition, we have introduced convolution into the transformer module as a convolutional projection and feedforward network. This combination of CNN and transformer multilayered series-parallel connection balances the performance and efficiency of the network.

The main contributions of this article are as follows.

1) We propose a hybrid end-to-end CD network based on a multilevel series-parallel combination of CNN and transformer. This network combines the global attention of the transformer with the local sensing capability of the CNN. The complementary strengths of the two enable the network to effectively eliminate the effects caused by pseudo-changes in the CD scene due to various factors.
2) The proposed asymmetric multihead crossover attention (AMCA) module retains the most representative tokens by introducing an adaptively updated semantic map using a nonlinear mapping. The quadratic computational complexity of self-attention is reduced to linear computational complexity. The module deeply integrates the respective advantages of CNN and transformer.
3) Our three publicly benchmarked CD datasets were fully experimented with, and the quantitative and visualization results show that our model achieves better results than other related methods, striking a balance between efficiency and performance.

The rest of the article is structured as follows. Section II describes our proposed method in detail. Section III presents the experimental setup and analyses the results in detail. Section IV discusses the performance of the network, the effectiveness of the modules, the impact of sample imbalance on the network, and the fusion strategy, and then visualizes the network structure. Finally, the conclusion of the article is drawn in Section V.

II. Methodology

In this section, we first describe the overall flow of our proposed method, followed by the asymmetric cross-attention

TABLE I
CONFIGURATIONS OF DIFFERENT SCALES OF ACAHNET

Model	C1	C2	C3	C4	C5	Params. (M)	FLOPs(G)
ACAHNet /8	8	16	32	64	128	11.30	2.98
ACAHNet /16	16	32	64	128	256	44.44	11.00
ACAHNet /24	24	48	96	192	384	99.42	27.03

module and the semantic generation module, respectively, and finally, the loss function used.

A. Overall Architecture

The overall structure of the Asymmetric Cross-attention Hierarchical Network (ACAHNet) is shown in Fig. 3. ACAHNet takes three channels of bitemporal RS images as input and outputs two channels of change prediction maps. The bitemporal RS images input to ACAHNet are progressively downsampled by a dual-branch encoder with shared weights. The downsampling operation is implemented via the patch merging [46] layer. After the spatial resolution has been reduced to half of its original size, the feature maps are fed into the semantic map generation module to generate low spatial resolution semantic maps by nonlinear projection. Both are fed simultaneously into the AMCA module for feature encoding work. After the spatial resolution has been reduced to a quarter of its original size, the feature maps and semantic maps extracted from the dual-branch part of the encoder are each concatenated and channel-adjusted to perform the fusion of the dual-branch features. The fused features continue through two stages of encoding before moving on to the decoding stage. The features go through the AMCA module and the convolution module, in turn, to gradually recover the change information. The low-level features from the two-branch part of the encoder are fused with the features from the decoder output via a three-branch aggregation (TBA) module. The semantic and feature maps from the single-branch part of the encoder and from the decoder are fused respectively through the dual-branch aggregation (DBA) module. After the change map has been restored to the same spatial resolution as the input image, the channels are adjusted to two using convolution to output the final change map.

ACAHNet first extracts low-level features from RS images by convolutional blocks and then reduces the spatial resolution by downsampling. The lower spatial resolution reduces the computational effort of the AMCA module. The semantic map is a highly representative feature matrix generated by nonlinear projection. The spatial resolution of the semantic map is set to a constant, half the minimum spatial resolution of the feature map. This reduces the computational complexity of self-attention from quadratic to linear, still providing efficient global information capture and fine-grained error correction. The values of C_i in Fig. 3 represent the channel values for the different stages of the feature and semantic maps. We arrived at three models with different scales of channel variation by setting the value of C_i. The specific values are shown in Table I.

B. Asymmetric Multihead Crossover Attention

The main structure of the AMCA module is shown in Fig. 4. Like the traditional transformer block, the AMCA is built upon the multiheaded self-attention (MHSA) and the FFN. In vision tasks, in order to conform to the transformer's calculation, each pixel of the feature map $X_f \in \mathbb{R}^{\mathbb{H}\times\mathbb{W}\times\mathbb{d}}$ is usually taken as a token and then flattened into a sequence $\mathbb{X}_\mathbb{t} \in \mathbb{R}^{\mathbb{m}\times\mathbb{d}}$ as input to the transformer block, where $\mathbb{H}$ denotes the height of the feature map, $\mathbb{W}$ denotes the width of the feature map, $\mathbb{d}$ denotes the number of channels of the feature map and tokens, and $\mathbb{m} = \mathbb{H}\mathbb{W}$ denotes the length of the token sequence. The token sequence $\mathbb{X}_\mathbb{t}$ input to the transformer is projected to the query ($\mathbb{Q} \in \mathbb{R}^{\mathbb{m}\times\mathbb{d}}$), key ($\mathbb{K} \in \mathbb{R}^{\mathbb{m}\times\mathbb{d}}$), and value ($\mathbb{V} \in \mathbb{R}^{\mathbb{m}\times\mathbb{d}}$) embeddings. The self-attention of the transformer is calculated by the following scaled dot product formula:

$$\text{Attention}(Q, K, V) = \underbrace{\text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)}_{P} V. \tag{1}$$

Transformer uses MHSA for better capacity and to reduce overfitting. The query, key, and value embeddings are linearly projected into multiple representation subspaces via a parameter matrix. By computing each head attention separately, the results of the multihead attention are eventually concatenated together as the final output of the MHSA module

$$\begin{aligned}\text{MHSA}(Q, K, V) &= \text{Concat}(\text{head}_1, \ldots, \text{head}_\text{h})\text{W}^\text{O} \\ \text{where head}_\text{i} &= \text{Attention}\left(Q W_\text{i}^\text{Q}, \text{KW}_\text{i}^\text{K}, \text{VW}_\text{i}^\text{V}\right)\end{aligned} \tag{2}$$

where $\mathbb{W}_\mathbb{i}^\mathbb{Q} \in \mathbb{R}^{\mathbb{d}_\mathbb{m}\times\mathbb{d}_\mathbb{w}}$, $\mathbb{w}_\mathbb{i}^\mathbb{K} \in \mathbb{R}^{\mathbb{d}_\mathbb{m}\times\mathbb{d}_\mathbb{k}}$, $\mathbb{W}_\mathbb{i}^\mathbb{V} \in \mathbb{R}^{\mathbb{d}_\mathbb{m}\times\mathbb{d}_\mathbb{v}}$, $\mathbb{W}^\mathbb{O} \in \mathbb{R}_\mathbb{v}^{\mathbb{h}\mathbb{d}} \times \mathbb{C}$ are learnable matrices, $\mathbb{d}_\mathbb{k}$, $\mathbb{d}_\mathbb{v}$are the hidden dimensions of the projection subspace, $\mathbb{h}$ is the number of heads, and $\mathbb{d}_\mathbb{m}$ is the embedding dimension.

The matrix $\mathbb{P} \in \mathbb{R}^{\mathbb{m}\times\mathbb{m}}$ in (2) is often referred to as the context mapping matrix. Transformer uses $\mathbb{P}$ to measure the similarity between the tokens; however, computing $\mathbb{P}$ requires the dot product of two $\mathbb{m} \times \mathbb{c}$ matrices, which results in a computational complexity of $\mathbb{O}(\mathbb{m}^2\mathbb{c})$. When the spatial resolution of the image is high, the computational efficiency of the transformer block decreases dramatically as $\mathbb{m}$ becomes too large. This quadratic computation complexity dependence on sequence length has become a major bottleneck for the transformer in vision tasks.

In image data, neighboring pixels in the same semantic object have similar semantic information. This means that performing self-attention operations directly on the input image is inefficient and redundant in terms of efficiency, although it improves accuracy. It has been shown theoretically and empirically that the context mapping matrix $\mathbb{P}$ is a low-rank matrix [50]. This indicates that most of the information is concentrated on the maximum singular value. Inspired by this, our proposed AMCA uses a low-rank matrix to approximate the self-attention matrix $\mathbb{P}$, thus reducing the computational complexity of self-attention to $\mathbb{O}(\mathbb{m})$, while the original feature matrix is processed by a parallel convolution module (PCM) and then fused with the features generated by AMCA to achieve a balance between efficiency and performance.

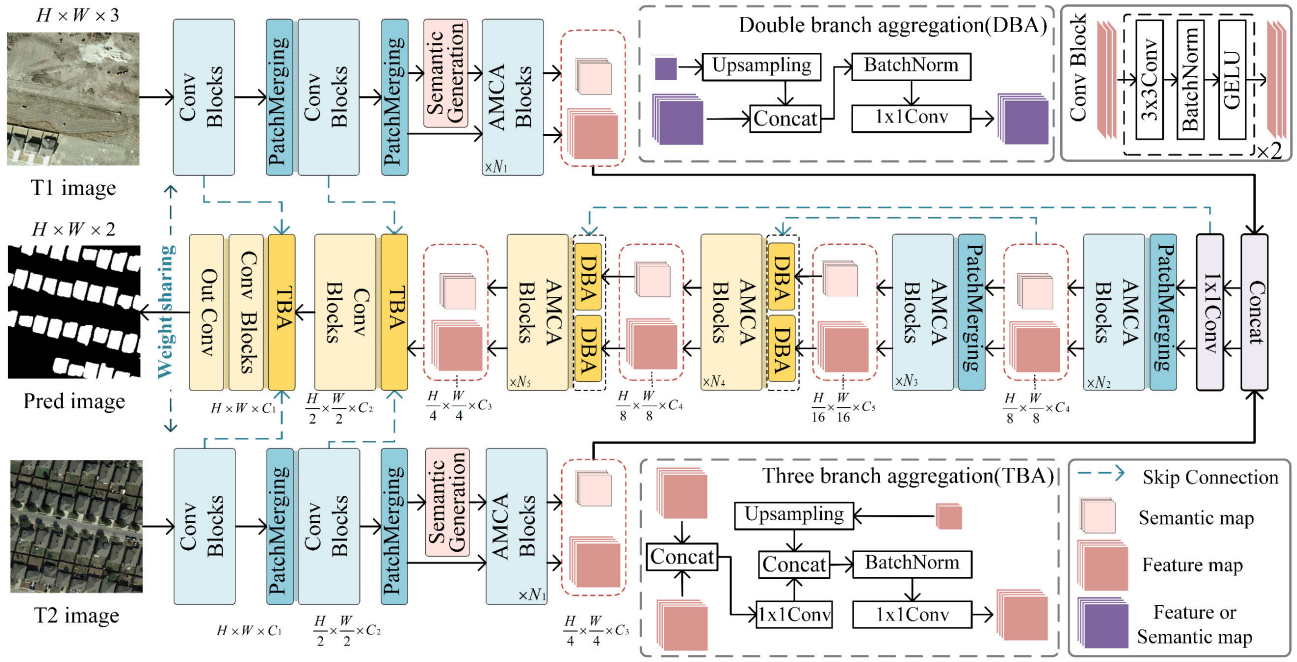

Fig. 3. Overall structure of the proposed ACAHNet.

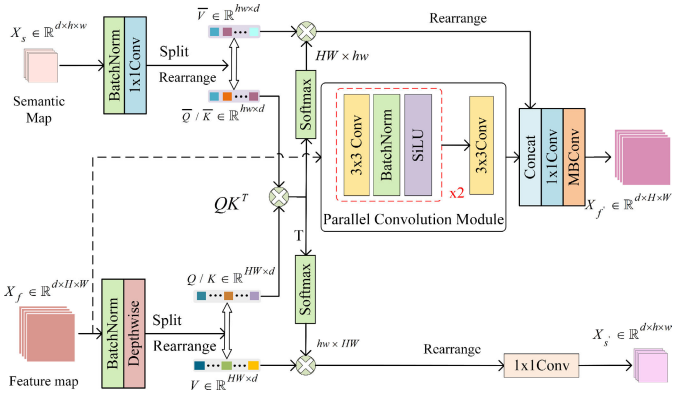

Fig. 4. AMCA module.

Specifically, we use nonlinear projection to transform features $\mathbb{X}_{\mathbb{f}} \in \mathbb{R}^{\mathbb{d}\times\mathbb{H}\times\mathbb{W}}$ into semantic features $\mathbb{X}_{\mathbb{s}} \in \mathbb{R}^{\mathbb{d}\times\mathbb{h}\times\mathbb{w}}$ with a high degree of information aggregation, where $\mathbb{h}, \mathbb{w} < \mathbb{H}, \mathbb{W}$. The AMCA module receives $\mathbb{X}_{\mathbb{s}}$ and $\mathbb{X}_{\mathbb{f}}$ as inputs and projects each of them to $\mathbb{Q}, \mathbb{K}, \mathbb{V} \in \mathbb{R}^{\mathbb{m}\times\mathbb{d}}$ and $\overline{\mathbb{Q}}, \overline{\mathbb{K}}, \overline{\mathbb{V}} \in \mathbb{R}^{\mathbb{l}\times\mathbb{d}}$, where $\mathbb{l} = \mathbb{hw} \ll \mathbb{m} = \mathbb{HW}$. Instead of all-to-all attention, we perform cross-attention on two sets of token sequences of different lengths. The $\mathbb{P}$ matrix constructed in this way requires only an $\mathbb{m}\times\mathbb{d}$ and an $\mathbb{l}\times\mathbb{d}$ matrix for the dot product operation. If $\mathbb{l}$ is fixed, then the computational complexity of attention is reduced to $\mathbb{O}(\mathbb{mld})$, which grows linearly with the length $\mathbb{m}$ of the token sequence. The crossed self-attention is calculated as follows:

$$\text{Cross-Attention}_f\left(Q, \overline{K}, \overline{V}\right) = \underbrace{\text{softmax}\left(\frac{Q\overline{K}^T}{\sqrt{d}}\right)}_{\mathbb{R}^{n\times l}}\overline{V}$$

$$\text{Cross-Attention}_s\left(\overline{Q}, K, V\right) = \underbrace{\text{softmax}\left(\frac{\overline{Q}K^T}{\sqrt{d}}\right)}_{\mathbb{R}^{l\times n}}V. \quad (3)$$

To further reduce computational consumption and memory usage, queries and keys for both $\mathbb{X}_{\mathbb{s}}$ and $\mathbb{X}_{\mathbb{f}}$ are shared so that computation can be reused through transposition operations as follows:

$$\overline{Q}K^T = \left(Q\overline{K}^T\right)^{\mathrm{T}}. \quad (4)$$

We introduced depth-wise separable convolution [51] into the AMCA module to project features into different spaces. Depth-wise separable convolution collects spatial and dimensional information through depth convolution and point convolution, respectively. The input features are then divided into three sets of feature matrices in the channel direction and then flattened for subsequent attention calculations. As the semantic features $\mathbb{X}_s$ are already highly aggregated, a $\mathbb{3}\times\mathbb{3}$ convolution and padding operation would cause noise effects. Therefore, a $\mathbb{1}\times\mathbb{1}$ convolution is used to project the semantic map.

The PCM in Fig. 4 consists of three stacked convolutional layers. The PCM and AMCA are processed simultaneously, with the features extracted by the latter being reshaped into 2-D and then fused with those extracted by the former. The fused features carry both local and global contexts, which means that the AMCA module has both local intrinsic bias and scale invariant bias. The fused features were processed by FFN. Considering the need to preserve the structural information of the features, we replaced the original two-layer perceptron with MBConv [52] as an FFN. It adds an activation layer between deep convolution and point convolution to perform the convolution operation. The AMCA module is formulated

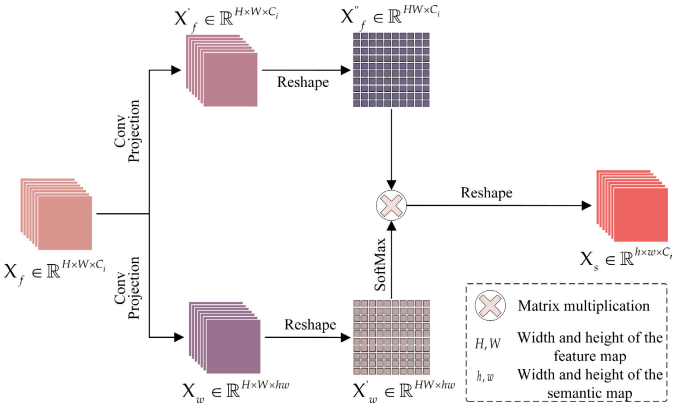

Fig. 5. Semantic generation module.

as follows:

$$\begin{cases} Q/K/V = F\text{latten}\left(D\text{wConv}\left(\text{BatchNorm}\left(X_{f_{i-1}}\right)\right)\right) \\ \overline{Q}/\overline{K}/\overline{V} = F\text{latten}\left(\text{Conv}\left(\text{BatchNorm}\left(X_{s_{i-1}}\right)\right), 1\right) \end{cases}$$
$$\begin{cases} X'_{f_i} = \text{MCSA}_f\left(Q, \overline{K}, \overline{V}\right) \\ X'_{s_i} = \text{MCSA}_s\left(\overline{Q}, K, V\right) \end{cases}$$
$$\begin{cases} X_{f_i} = \text{MBConv}(\text{Conv}(\text{Concat}(P\text{CM}\left(X_{f_{i-1}}\right) \\ \text{Reshape}\left(X'_{f_i}\right))), 1) \\ X_{s_i} = \text{Conv}\left(\text{Reshape}\left(X'_{s_i}\right), 1\right) \end{cases} \tag{5}$$

where $MCSA$ denotes the multihead computation of asymmetric crossover self-attention, which is the same as the traditional transformer block in the multihead computation. $\mathbb{D}_{\mathbb{W}}\mathbb{C}\mathbb{O}\mathbb{m}_{\mathbb{V}}$ denotes the depth-wise separable convolution. $\mathbb{C}\mathbb{O}\mathbb{m}_{\mathbb{V}}(\mathbb{X}, \mathbb{I})$ denotes the convolution of $\mathbb{I} \times \mathbb{I}$ on feature $\mathbb{X}$.

The nonlinear mapping of the AMCA module enables better maintenance and updating of the semantic information of the semantic maps, thus ensuring an approximation to traditional self-attention when using low spatial resolution semantic and feature maps for cross-attention. The addition of PCM and convolutional projection enables the AMCA module to capture long-range and local information in a complementary way.

C. Semantic Generation Module

The AMCA module requires two matrices with different spatial resolutions as input, so a semantic generation module is added between the convolution block and the AMCA module to generate the initial semantic map. After the initial feature extraction of the input bitemporal RS image by the convolution block, the initial semantic map is generated by the semantic generation module using linear mapping, both of which are used as the initial input to the AMCA module. The structure of the semantic generation module is shown in Fig. 5. First, the feature $\mathbb{X}_{\mathbb{f}}$ is projected into two different subspaces $\mathbb{X}'_{\mathbb{f}}$C $\mathbb{X}_{\mathbb{w}}$ using convolution. The spatial resolution of both is kept the same as the original feature, where the number of channels of $\mathbb{X}_{\mathbb{w}}$ is adjusted to be the same as the product of the length and width of the set semantic map. Then, the reshaped $\mathbb{X}_{\mathbb{w}}$ is subjected to a Softmax operation, which is used as the weight matrix for the matrix operation with the reshaped $\mathbb{X}'_{\mathbb{f}}$. In this way, the information in feature $\mathbb{X}_{\mathbb{f}}$ is re-weighted to generate a low-resolution semantic map for information aggregation. It should be noted that the semantic map is not downsampled when they pass through the patch merging module; only convolutional updates are performed.

D. Loss Function

The CD task is an intensive pixel-level prediction task. We choose a weighted cross-entropy (WCE) loss function [28] and a dice loss [53] function to jointly compute the losses for training. The cross-entropy (CE) loss function is used to eliminate the problem of imbalance between the number of change and invariant samples. We describe the predicted change map $\hat{\mathbb{Y}} \in \mathbb{R}^{\mathbb{H}\times\mathbb{W}}$ as a set of pixel points as follows:

$$\widehat{Y} = \left\{\widehat{y}_i | i = 1, 2, 3, \ldots, H \times W\right\}. \tag{6}$$

Then the WCE loss function can be defined in the following form:

$$L_{\text{wce}} = \frac{1}{H \times W} \sum_{i=1}^{H\times W} \text{weight[class]} \cdot \left(\log\left(\frac{e^{\widehat{y}_i[\text{class}]}}{e^{\widehat{y}_i[0]} + e^{\widehat{y}_i[1]}}\right)\right) \tag{7}$$

where the value field of class is $\{\mathbb{0}, \mathbb{1}\}$, corresponding to the predicted value of the pixel as changed and unchanged, respectively.

The Dice coefficient describes the similarity of the set and takes values in the range $[\mathbb{0}, \mathbb{1}]$. $\mathbb{Y}$ denotes the set of pixels of the ground truth. $\hat{Y}$ denotes the set of the prediction map. The Dice coefficients are described as follows:

$$\text{DiceCoefficient} = \frac{2|Y \cap \widehat{Y}|}{|Y| + |\widehat{Y}|} \tag{8}$$

where $|Y|$ $|\hat{Y}|$ denote the number of pixels in the set of ground truth and predicted pixels, and $|Y \cap \hat{Y}|$ denotes the number of pixels in the intersection of the two. Dice loss is defined as follows:

$$L_{\text{Dice}} = 1 - \text{DiceCoefficient}. \tag{9}$$

The smaller the value of Dice loss, the more similar $\hat{Y}$与 ^{Y} are. The total loss of the training is defined as follows:

$$L_{\text{loss}} = L_{\text{wce}} + L_{\text{dice}}. \tag{10}$$

III. Experiments and Results Analysis

To validate the performance of our proposed ACAHNet, we conducted experiments on three public CD datasets: Synthetic Images and Real Seasonal-Varying CD Dataset (CDD) [54], Vision and RS-CD (LEVIR-CD) [33], and Sun Yat-sen University Dataset (SYSU-CD) [30]. In this section, we first describe the datasets used and the comparison methods. Then the evaluation metrics for the network are described. Finally, the experimental setup and results are introduced.

A. Dataset Settings

CDD [54]: This dataset is a realistic seasonal RS variation dataset of Google Earth images, containing changes in, e.g., buildings, roads, and vehicles. The CDD dataset ignores variation caused by seasonal differences and brightness, which makes it more difficult for DC algorithms to eliminate pseudo change.

LEVIR-CD [33]: This dataset is a building dataset of Google Earth imagery covering a variety of building types, such as high-rise flats, villas, small garages, and large warehouses. The dataset pays attention to changes caused by seasonality and changes in light. We cut 637 high-resolution Google Earth image pairs with a pixel size of 1024 × 1024 into sample pairs of 256 × 256 pixels without overlap, resulting in 7120/1024/2048 image pairs for training/validation/test.

SYSU-CD [30]: This dataset includes changes in buildings, vegetation changes, road changes, and changes in marine targets. The dataset was first divided into a training set, a validation set, and a test set from the original 800 image pairs in a 6:2:2 ratio. Then, 25 sample pairs of 256 × 256 size were randomly collected from each image pair, and the data was enhanced using random flips and rotations. A total of 20 000 pairs of 256 × 256 sized blocks of aerial images were obtained, with the number of image pairs in the training/validation/test sets being 12 000/4000/4000. Details of the three datasets described above are shown in Table II.

B. Comparison Methods

We chose different types of DL DC models as a comparison method. The methods used for comparison were as follows.

FC-EF [3]: This method uses an early fusion strategy, and the network architecture is based on UNet. The original bitemporal images are concatenated as network input and processed by a single-stream convolutional network to detect changes.

FC-Siam-Diff [3]: The method uses a postfusion strategy based on the FC-EF network. Multiscale features are extracted from the twin convolutional network for bitemporal images, and algebraic operations are used to obtain disparity features to detect changes.

FC-Siam-Conc [3]: The method is a multiscale feature-level concatenation method. The method differs from FC-Siam-Diff networks in that multiscale features extracted from twin convolutional networks are concatenated in the channel dimension to perform DC.

CDNet [55]: The method is based on an inverse convolutional network for DC. It uses an early fusion strategy, takes an image pair as input, and obtains a pixel-level classification map of structural changes.

STANet [33]: The method is built on top of ResNet18, which uses a self-attention mechanism to calculate spatial and temporal correlations to detect changes.

SNUNet/48 [28]: The method network structure is based on UNet^{++}. It uses a densely connected twin network for feature extraction, reducing the loss of deep-level location information. One of the proposed integrated channel attention modules (ECAM) can refine the most representative features at different semantic levels. The model with an initial number of 48 channels is the best performing [28].

Bit-CD [49]: The method uses a late fusion strategy, adding transformer as an encoder on top of the convolutional network to model the spatial-temporal feature to better simulate the context. The learned context-rich tokens are fed back into the pixel space to detect changes.

ChangeFormer [48]: The method is a purely transformer-based feature-level DC method that does not use a convolutional network but rather performs the DC task directly through a transformer encoder–decoder network.

SwinSUNet [47]: This method uses the Swin transformer blocks as the basic unit to design a Siamese U-shaped structure to solve the CD problem. It is the first pure transformer network for CD tasks. The excellent performance of the model proves the potential of the transformer in the field of CD.

C. Evaluation Metrics

To quantitatively characterize the performance of the model, we have followed the mainstream metrics in the field of DC and reported the precision (Pre.) and recall (Rec.) values for the change categories and the overall accuracy (OA) performance for the DC task. We use F1 and intersection over union (IoU) scores for the change categories as the main quantitative metrics. The metrics are calculated as follows:

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

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

$$\text{OA} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FN}+\text{TN}+\text{FP}} \tag{13}$$

$$\text{IoU} = \frac{\text{TP}}{\text{TP}+\text{FP}+\text{FN}} \tag{14}$$

$$\text{F1} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{15}$$

where TP represents the number of true positives. True positive indicates that changed pixels were correctly predicted. FP represents the number of false positives. False positives indicate that unchanged pixels were incorrectly predicted as changed. TN represents the number of true negatives. True negatives indicate that unchanged pixels were correctly predicted. FN represents the number of false negatives. False negatives indicate that changed pixels were incorrectly predicted as unchanged. The values of F1 and IoU range from 0 to 1, with higher values indicating a better performance of the network. To provide a more complete picture of the model's performance, we also report the number of parameters (Params.) and the inference time of the model.

D. Training Details

We trained the proposed model using CE loss and a combination with Dice loss. We trained the model using the PyTorch framework on an NVIDIA TITAN RTX (24 GB) GPU, using the AdamW optimizer for exponential learning rate decay. The initial learning rate was set to 0.0001, and the initial learning rate was reached within five epochs using linear warm-up,

TABLE II
INFORMATION ON THE MAIN PARAMETERS OF THE THREE DATASETS WE USED

Datasets	Spatial Resolution	Size/Image	Number of Pixels in Datasets			Number of Images in Datasets		
			Changed	Unchanged	Ratio	Train	Validation	Test
CDD	0.03-1m/pixel	256 × 256	134,068,750	914,376,178	1:6.82	10,000	3,000	3,000
LEVIR-CD	0.5m/pixel	256 × 256	30,913,975	637,028,937	1:20.61	7120	1024	2048
SYSU-CD	0.5m/pixel	256 × 256	286,092,024	1,024,627,976	1:3.58	12000	4000	4000

TABLE III
COMPARISON RESULTS ON THE THREE CD TEST SETS. THE HIGHEST SCORE IS MARKED IN BOLD. ALL THE SCORES ARE DESCRIBED IN PERCENTAGE (%)

Model	Params. (M)	Flops (G)	CDD Pre. / Rec. / F1 / IoU / OA	LEVIR-CD Pre. / Rec. / F1 / IoU / OA	SYSU-CD Pre. / Rec. / F1 / IoU / OA
FC-EF	1.29	2.92	76.56/43.49/55.47/38.38/91.76	82.27/66.28/73.41/58.00/97.55	75.97/70.80/73.29/57.85/87.83
FC-Siam-Conc	1.93	4.55	88.00/53.58/66.61/49.93/93.66	86.81/67.66/76.05/61.36/97.83	76.41/76.17/76.29/61.67/88.83
FC-Siam-Diff	1.75	3.99	88.49/51.53/65.14/48.30/93.49	86.55/74.38/80.00/66.68/98.11	**88.05**/55.29/67.92/51.43/87.68
CDNet	1.36	21.52	91.93/84.87/88.26/78.98/97.34	88.38/85.08/86.70/76.52/98.67	81.09/78.38/79.72/66.27/90.59
SNUNet/48	28.34	97.87	96.82/96.72/96.77/93.74/99.24	91.66/88.48/90.04/81.89/99.00	83.31/74.00/78.38/64.45/90.37
STANet	16.93	6.58	88.97/94.31/91.56/84.44/97.95	80.99/**91.21**/85.79/75.12/98.46	82.36/74.30/78.12/64.10/90.18
Bit-CD	3.55	10.60	95.86/94.59/95.22/90.88/98.88	91.95/88.57/90.23/82.19/99.02	79.04/76.71/77.86/63.75/89.71
ChangeFormer	267.90	129.27	95.27/93.82/94.54/89.65/98.72	91.53/88.86/90.17/82.10/99.01	84.99/70.93/77.33/63.04/90.19
SwinSUNet	40.95	11.19	95.96/94.58/95.26/90.96/98.89	90.51/89.72/90.11/82.00/98.99	84.09/72.67/77.97/63.89/90.31
ACAHNet/8	11.30	2.98	96.18/96.43/96.30/92.87/99.14	91.67/89.74/90.70/82.98/99.06	84.29/80.54/82.37/70.03/91.87
ACAHNet/16	44.44	11.00	97.22/97.66/97.44/95.01/99.39	92.08/90.09/91.08/83.62/99.10	84.39/80.66/82.48/70.19/91.92
ACAHNet/24	99.42	24.07	**97.54**/**97.89**/**97.72**/**95.54**/**99.46**	**92.36**/90.68/**91.51**/**84.35**/**99.14**	83.96/**81.54**/**82.73**/**70.55**/**91.97**

TABLE IV
ABLATION EXPERIMENTS RESULTS ON SYSU-CD AND LEVIR-CD. [ALL VALUES ARE REPORTED AS PERCENTAGES (%)]

Model	Params.(M)	Flops(G)	SYSU-CD Pre. / Rec. / F1 / IoU / OA	LEVIR-CD Pre. / Rec. / F1 / IoU / OA
Base_Conv	4.21	1.78	80.83 / 81.20 / 81.02 / 68.09 / 91.02	91.29 / 89.03 / 90.14 / 82.06 / 99.00
Base_Transformer	4.99	5.16	84.61 / 79.88 / 82.18 / 69.75 / 91.83	92.03 / 89.63 / 90.81 / 83.17 / 99.07
w/o PCM	5.50	1.95	84.60 / 79.49 / 81.97 / 69.45 / 91.75	91.05 / 89.35 / 90.19 / 82.14 / 99.01
w/o FFN	7.98	2.64	83.12 / 79.76 / 81.40 / 68.64 / 91.40	90.83 / 90.09 / 90.46 / 82.58 / 99.03
w/o ConvProject	11.14	2.92	84.07 / 77.70 / 80.76 / 67.74 / 91.27	90.53 / 89.40 / 89.96 / 81.76 / 98.98
ACAHNet/8	11.30	2.98	84.29 / 80.54 / 82.37 / 70.03 / 91.87	91.67 / 89.74 / 90.70 / 82.98 / 99.06

followed by an exponential decay. The training period was set to 200, and the batch was set to 16. It is worth noting that in order to ensure the fairness of comparison with other methods, we reimplement all the CD networks used for comparison using their public codes with default hyperparameters.

E. Comparative Experiments

We report in Table III the overall comparison results of ACAHNet with the comparison methods on the CDD, LEVIR-CD, and SYSU-CD test sets. The quantitative results show that the comprehensive performance of ACAHNet on these datasets is significantly better than other comparative methods. On the CDD dataset, the F1 metric of ACAHNet/24 is 0.95 points higher, and the IoU metric is 1.8 points higher than the second place SNUNet/48. On the LEVIR-CD dataset, ACAHNet/24 is higher than the nearest BIT-CD by 1.28 and 2.16 points. ACAHNet/8 outperforms Bit-CD by 0.47 and 0.79 points on the F1 metric and IoU metric. ACAHNet/8 has only 28% of the floating point operations of Bit-CD, which means that ACAHNet/8 has a faster inference speed. On the SYSU-CD dataset, the F1 metrics and IoU metrics of ACAHNet/24 are 3.12 and 4.44 points higher than those of CDNet. ACAHNet/8 has only 13% of CDNet's floating point operations, but the F1 and IoU metrics are 2.65 and 3.76 points higher than CDNet. It is worth noting that, benefiting from the constructed asymmetric cross-attention mechanism, ACAHNet/8 has excellent DC capability while maintaining a low number of floating-point operations and parameters.

The visualization comparison of the detection results of each method is shown in Fig. 6. In order to show the detection results of each method more visually, we use different colors

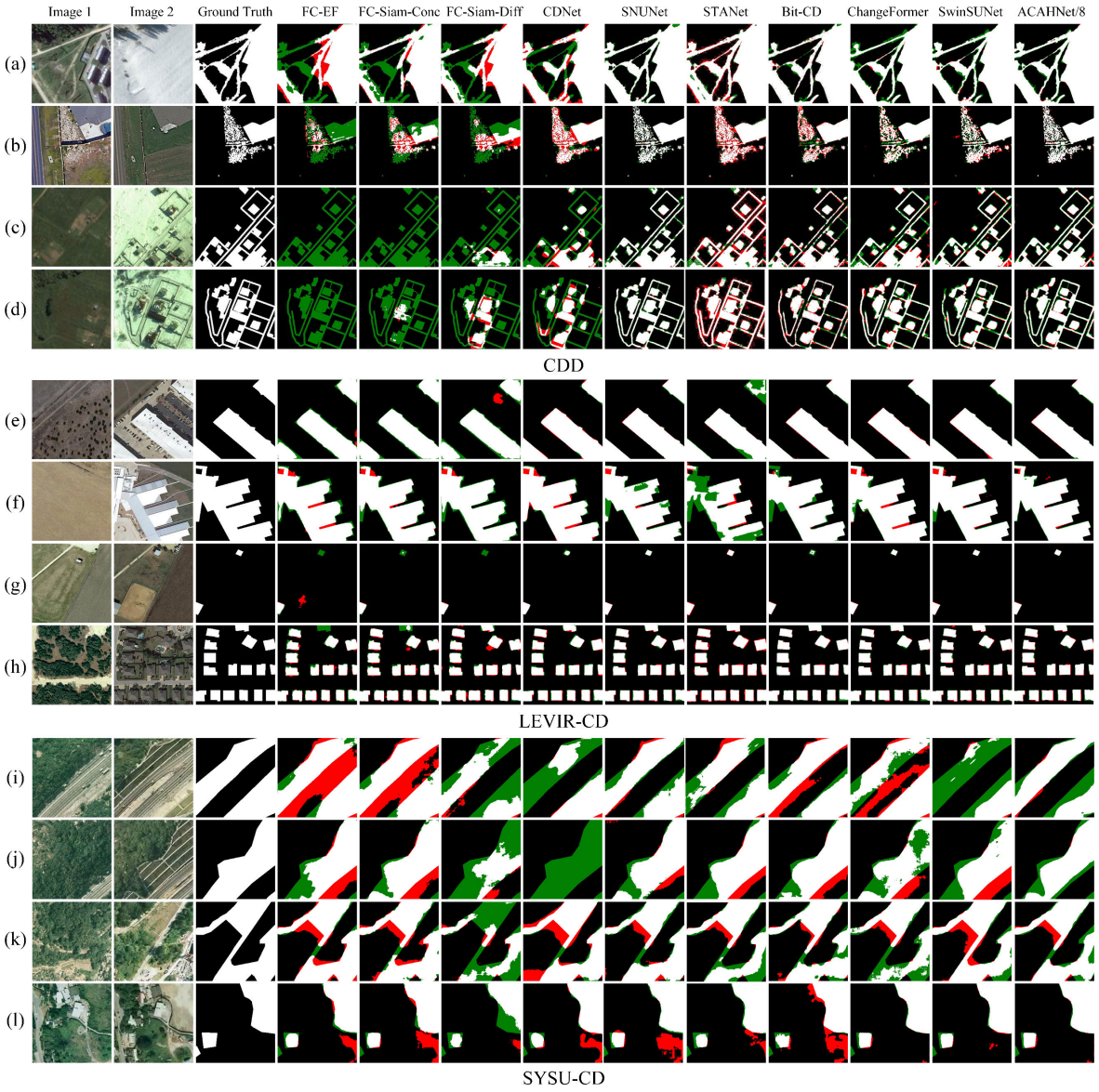

Fig. 6. Visualization results of different methods on the CDD, LEVIR-CD, and SYSU-CD test sets. (a)–(l) denote prediction results of all the compared methods for different samples, respectively.

to indicate TP (white), TN (black), FP (red), and FN (green). ACAHNet/8 performs well on the detection of complex, sparse, dense, and small changes [e.g., Fig. 6(a), (b), and (g)] and more accurately represents the real semantic changes. As seen in Fig. 6(h), Bit-CD and ACAHNet/8 are able to accurately detect changes in dense buildings and attenuate the effects of noise. Compared with BIT-CD, ACAHNet/8 has a more complete detection of building edges [e.g., Fig. 6(c), (d), (e), and (f)]. This is due to the fact that the Bit-CD model upsamples directly to full resolution resulting in a loss of detail in the region of variation. ACAHNet, however, uses a multilevel string-parallel combination of transformer and CNN to enable the network to learn long-range contexts while possessing inductive bias, thus retaining much detailed information. CDNet's missed detection of changes (green) is more severe [e.g., Fig. 6(i) and (j)], a phenomenon also observed on FC-EF. Both use an early fusion strategy that connects dual-time images as the input to the network. This leads to the fact that the features obtained by upsampling cannot be effectively recovered by jumping over the connection due to the lack of deep features of the original image. Fig. 6(e)–(h) shows the completeness of the detection of boundaries by ACAHNet, ChangeFormer, and SwinSUNet. The dense connectivity mechanism of SNUNet makes its detection of boundaries also more complete and more accurate within the changed objects. The phenomenon of false detection (red) is, however, more prominent due to the lack of global information capture [e.g., Fig. 6(i), (k), and (l)]. SwinSUNet and ChangeFormer are also more complete in detecting boundaries due to the use of transformer, but they cannot detect some fine-grained changes and misjudgments of small irrelevant changes due to the lack of complementary local information. With the long-range information captured by the transformer and the fine-grained local features extracted by the CNN, ACAHNet maintains the internal compactness of the object and the integrity of the detection of changes on the boundary as much as possible.

TABLE V
ANALYSIS OF FUSION STRATEGIES. [ALL VALUES ARE REPORTED AS PERCENTAGES (%)]

	Params. (M)	Flops (G)	CDD Pre. / Rec. / F1 / IoU / OA	LEVIR-CD Pre. / Rec. / F1 / IoU / OA	SYSU-CD Pre. / Rec. / F1 / IoU / OA
Fusion Point1	10.89	2.58	96.09/95.57/95.83/92.00/99.02	90.79/89.40/90.09/81.97/98.99	83.07/80.33/81.68/69.03/91.50
Fusion Point2	11.30	2.98	96.18/96.43/96.30/92.87/99.14	91.67/89.74/**90.70**/**82.98**/99.06	84.29/**80.54**/**82.37**/**70.03**/**91.87**
Fusion Point3	13.04	3.35	**96.74**/**97.13**/**96.93**/**94.05**/**99.27**	90.91/**90.12**/90.51/82.67/99.03	**85.04**/78.79/81.79/69.20/91.73
Fusion Point4	19.92	3.71	96.57/96.45/96.51/93.26/99.17	**91.96**/89.31/90.62/82.85/99.05	83.94/79.09/81.44/68.70/91.50

IV. DISCUSSION

A. Effects of the Modules on the Network Performance

By removing modules one by one, we validated the effectiveness of each key component used in ACAHNet on the SYSU-CD and LEVIR-CD datasets. The experimental results are shown in Table IV, where "w/o" is an abbreviation for "without." To verify the effectiveness of the AMCA module, we replace all the AMCA modules with the first convolutional block as the Base_Conv model. We use the traditional self-attention method to calculate the attention weights to obtain the Base_Transformer model. The designed ablation experiments were performed on the ACAHNet/8 model. As we can see in Table IV, the Base_Conv model obtained by replacing the AMCA module with the convolutional module still ranks highly in performance with the comparison model mentioned above. This is due to the fact that we have only replaced the AMCA module, which leaves the network with some loss of long-range information capture capability, but the overall structure of the network is still maintained, so the F1 metrics and IoU metrics of the Base model are still excellent. Compared to Base_Conv, the F1 metric and IoU metric of ACAHNet/8 on SYSU-CD improved by 1.35% and 1.94%.

In contrast to ACAHNet/8, Base_Transformer uses a traditional self-attention mechanism, where features are projected directly onto $Q/K/V$ tokens of the same size, and then attention values are calculated. The comparison of the ACAHNet/8 and Base_Transformer data in Table IV shows that such-high resolution features with redundant information can be replaced by a low-resolution semantic map with highly aggregated information to keep performance while reducing the number of floating point operations and thus improving efficiency.

Maintaining and updating semantic maps with highly aggregated information in the AMCA mechanism is, however, critical and difficult. Inside the AMCA module, we use a PCM to supplement the local information. As we can see from the comparison of the data before and after the removal of the PCM, the PCM can effectively reduce the effect of losing highly representative information during the maintenance and updating of the semantic map.

We used MBConv as an FFN for the AMCA module. As can be seen in Table IV, when the FFN was removed, the local features extracted by parallel convolution could not effectively complement the features extracted by the AMCA mechanism with different information. This resulted in the F1 values of the model decreasing by 0.97% and 0.24% on the two datasets, respectively, and the IoU values of the model decreasing by 1.39% and 0.4% on the two datasets, respectively. After we replaced the depth-wise separable convolutional projection with a linear projection, the semantic map could not be effectively updated because the simple linear projection approach could not capture the structural information of the image. As shown in Table IV, the F1 values of the model decreased by 1.61% and 1.08% on the two datasets before and after replacing the convolutional projection, and the IoU values of the model decreased by 2.29% and 1.78% on the two datasets, respectively.

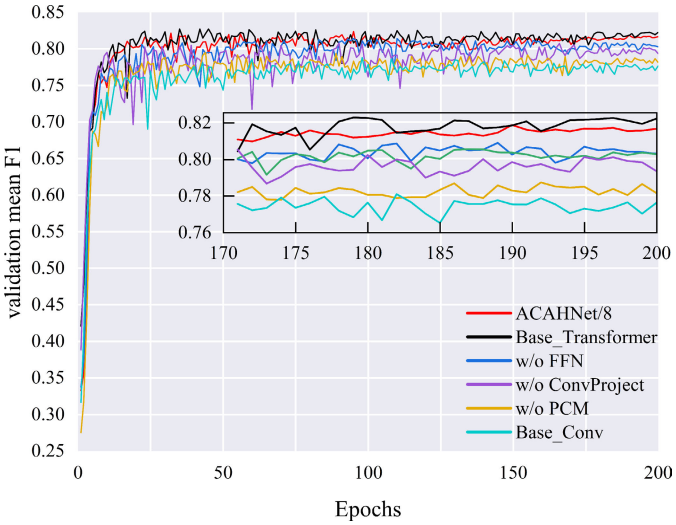

Fig. 7. Line chart of the ablation experiment on the SYSU-CD dataset, with a small window showing details of the last 30 epochs of the line chart.

As shown in Fig. 7, to more visually demonstrate the role of each module, we plotted line plots of the F1 metrics for the validation set during the training of each ablation test on the SYSU-CD dataset. From the line plot, we can see that because we performed a linear warm-up of five epochs, each model was able to reach near the highest of its respective F1 values by 20 epochs. We can see from Fig. 7 that while the difference in metrics between Base_Transformer and ACAHNet on the test set is smaller, the fluctuations in data are more pronounced for Base_Transformer than ACAHNet on the validation set. This implies that ACAHNet possesses stronger robustness.

B. Effects of Fusion Strategy on Network Performance

Depending on the dual-branch feature fusion strategy, it can be divided into early fusion and late fusion. We divide the encoder into four stages, where the convolution part is the

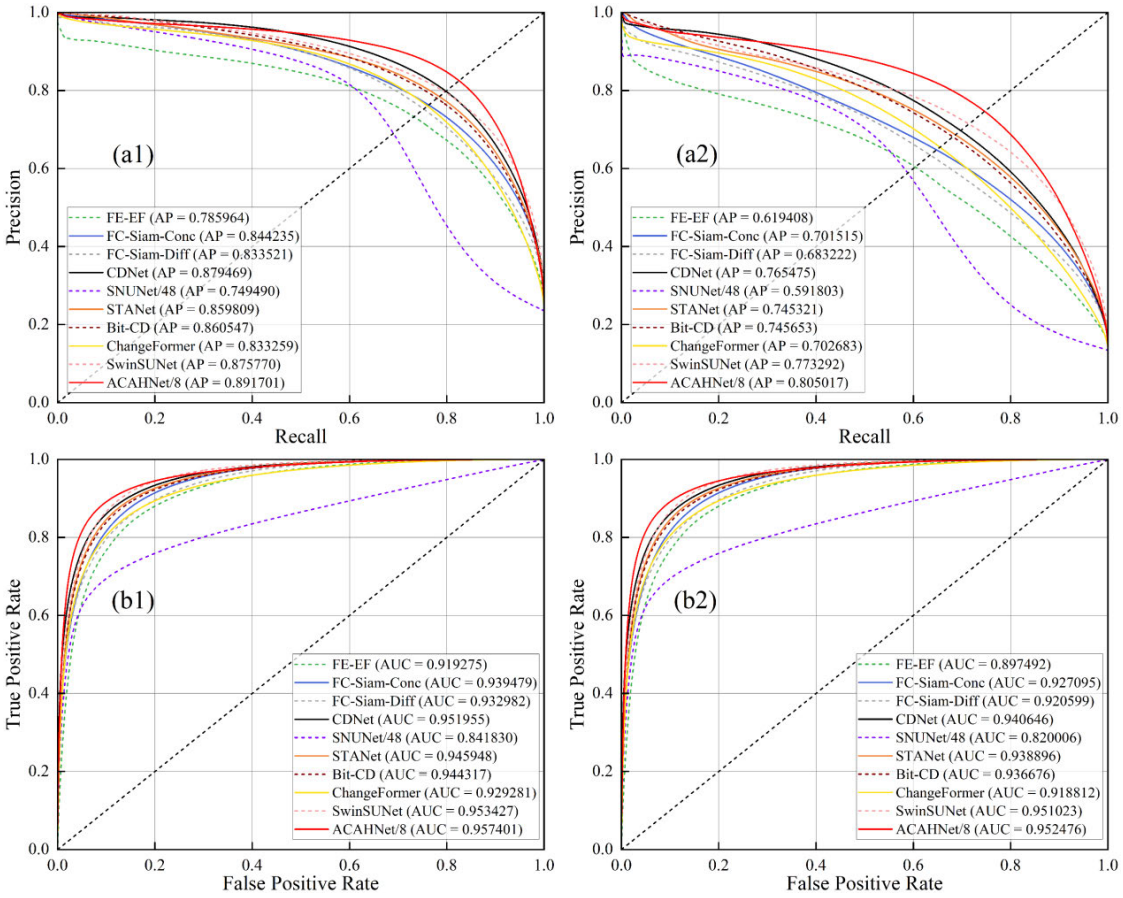

Fig. 8. PR and ROC curves of SYSU-CD results. (a1) and (a2) are the PR curves before and after the adjustment of the distribution of the SYSU-CD dataset. (b1) and (b2) are the ROC curves before and after the adjustment of the distribution of the SYSU-CD dataset.

TABLE VI

ANALYSIS OF THE IMPACT OF SAMPLE IMBALANCE ON THE NETWORK. [ALL VALUES ARE REPORTED AS PERCENTAGES (%)]

Loss function	SYSU-CD Pre. / Rec. / F1 / IoU / OA	LEVIR-CD Pre. / Rec. / F1 / IoU / OA
Dice	83.76 / 80.51 / 82.10 / 69.64 / 91.72	88/86 / 88.37 / 88.61 / 79.56 / 98.84
Bce	84.43 / 78.36 / 81.28 / 68.47 / 91.49	92.75 / 85.04 / 88.73 / 79.75 / 98.89
Wce	83.20 / 81.13 / 82.15 / 69.72 / 91.69	91.17 / 88.46 / 89.80 / 81.48 / 98.97
Wce+Dice	84.29 / 80.54 / **82.37** / 70.03 / 91.87	91.67 / 89.74 / **90.70** / 82.98 / 99.06

TABLE VII

INFORMATION ON THE SYSU-CD TEST SET

SYSU-CD	Test Set	
	Original	After adjustment
Number of Images	4000	2500
Changed Pixels	61,820,917	21,989,546
Unhanged Pixels	200,323,083	141,850,454
Ratio	1:3.24	1:6.45

first stage, and the last three AMCA modules are the other three stages. By fusing features extracted at different stages, we designed four sets of experiments targeting the fusion strategy. The results of the experiments are shown in Table V, where we can see that both earlier and later fusion of the dual-branch features results in lower model metrics.

Premature fusion of dual-branch feature results in a lack of deeper features from a single original image, and the boundaries of the change map are broken and do not facilitate the recovery of change information lost due to upsampling when performing skip connections. Late fusion strategies typically fuse features at the last stage of the dual-branch encoder to feed into the decoder for processing. Because of the low spatial resolution of the features at the end of the encoder, the late fusion strategy may result in inadequate learning and low representation of the original image features. This causes more noise to be introduced when upsampling to recover spatial resolution. After comparing the number of floating-point operations, the number of model parameters, and F1,

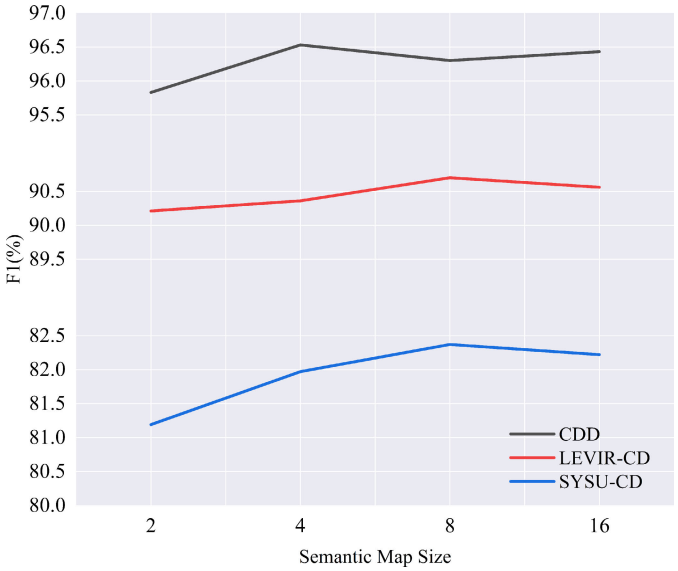

Fig. 9. Effects of the size of the semantic map on the effectiveness of the model in three different datasets.

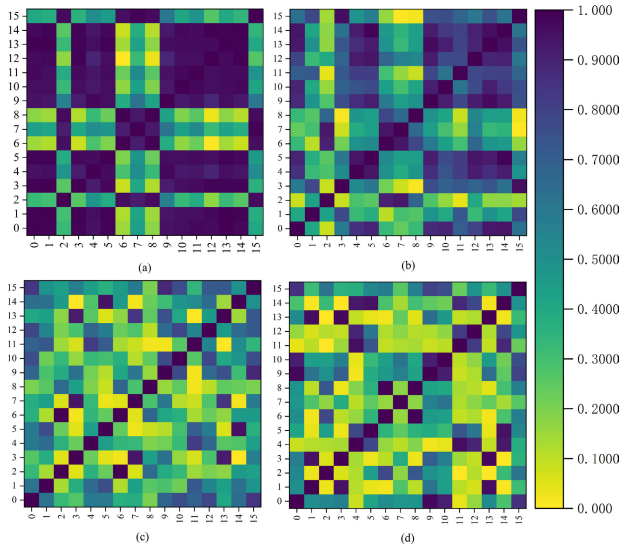

Fig. 10. Correlation of token pairs in semantic maps of size 4 for different models. (a)-(d) are the correlation heatmaps of the token pairs in the initial semantic maps from four different models with F1 values of 77.93%, 87.40%, 91.07%, 96.50% on the CDD dataset.

the data presented by Fusion Point2 is the best among the four groups of experiments. In this article, we choose to fuse the dual-branch features in the second stage of the encoder so that the encoder is divided into dual-branch and single-branch parts. This mid-term fusion strategy combines the advantages of both early and late fusion and allows the network to learn deeper information at both lower and higher levels more fully.

C. Effects of Sample Imbalance on Network Performance

To illustrate the impact of the sample imbalance problem on the network performance, we optimize ACAHNet/8 by using different loss functions. Examples include CE loss, WCE loss [3], and dice loss [53]. As can be seen in Table II, the degree of sample imbalance differs significantly between the SYSU-CD dataset and the LEVIR-CD dataset, facilitating experimental comparisons. Table VI shows the experimental results. On the LEVIR-CD dataset, the F1 metric decreased by up to 2.09%, while on the SYSU-CD dataset, the F1 indicator decreased by up to 1.09%. In this article, we use both WCE loss and dice loss to optimize the network. This experiment illustrates the effectiveness of the loss function we use to solve the sample imbalance.

To further compare the ability of the proposed ACAHNet with the comparison models in dealing with the real sample imbalance problem, we plotted Precision-Recall (PR) curves and Receiver Operating Characteristic (ROC) curves. In the case of sample imbalance, Area Under Curve (AUC) on the ROC curve provides a more balanced assessment of the overall performance of the model [56]. We use PR curves to compare the ability of each model to detect changes in the face of real sample imbalance. When dealing with highly unbalanced datasets, PR curves can demonstrate differences between models that are not obvious in the ROC space [57]. In order to better evaluate the performance of each model under the actual sample imbalance, after plotting the PR and ROC curves of each model on the SYSU-CD test set [see Fig. 8(a1) and (b1)], we adjusted the distribution of the samples by removing some samples from the SYSU-CD test set. The information on the test set before and after adjustment is shown in Table VII. Then, we plotted PR and ROC curves on the adjusted dataset [see Fig. 8(a2) and (b2)].

From Fig. 8(a1), we can see that our proposed ACAHNet/8 has higher Average Precision (AP), and the PR curves of ACAHNet/8 basically cover the PR curves of each of the other models. Comparing the AP values of each model on the PR curves before and after the test set adjustment, it can be seen that ACAHNet/8 is least affected, and FE-EF is most affected by the increasing degree of sample imbalance. Fig. 8(b1) shows that the AUC of ACAHNet/8 is the highest among the comparison models, and the ROC curve of ACAHNet/8 basically covers the ROC curve of each other model. It can be observed that the trend and AUC values of the ROC curves remain almost unchanged before and after the test set adjustment. This is due to the fact that the ROC curve eliminates the effect of sample imbalance on the evaluation and provides a more balanced assessment of the overall performance of each model. These PR curves and ROC curves can illustrate the relatively good performance of ACAHNet in dealing with multiple degrees of sample imbalance.

D. Semantic Map Analysis

We conducted experiments on the size of semantic map, and the results are shown in Fig. 9. It can be seen that the model works best on the LEVIR-CD dataset and the SYSU-CD dataset when the semantic map size is 8. The model works best on the CDD dataset with a semantic map size of 4. The semantic map size set in this article is 8, which is half of the minimum feature space resolution in ACAHNet. It can be seen that the effectiveness of the model is not significantly reduced on the three datasets when the size of the semantic maps is 2. This is due to the fact that for the CD task, there are only two categories at the pixel level: changed and unchanged. Whereas

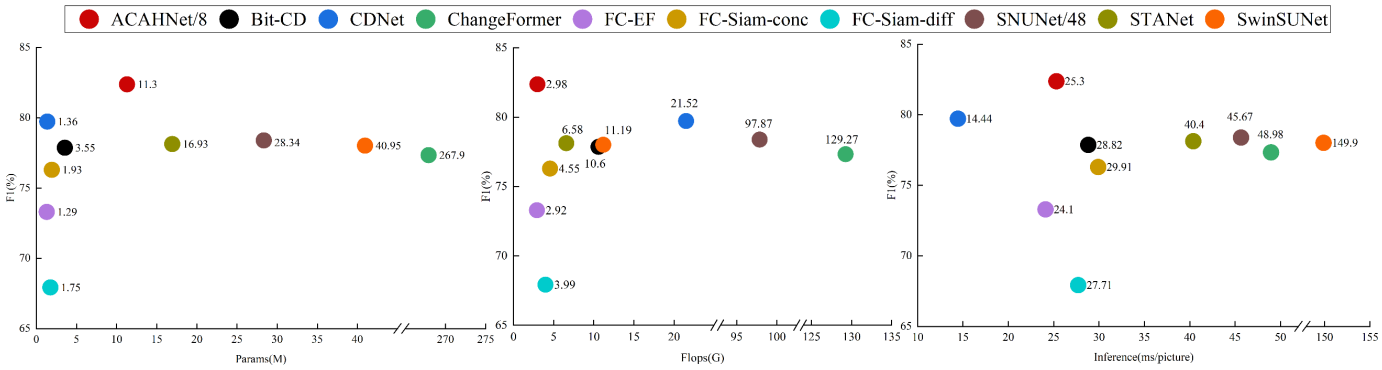

Fig. 11. Floating point plot of efficiency metrics for ACAHNet and other CD models.

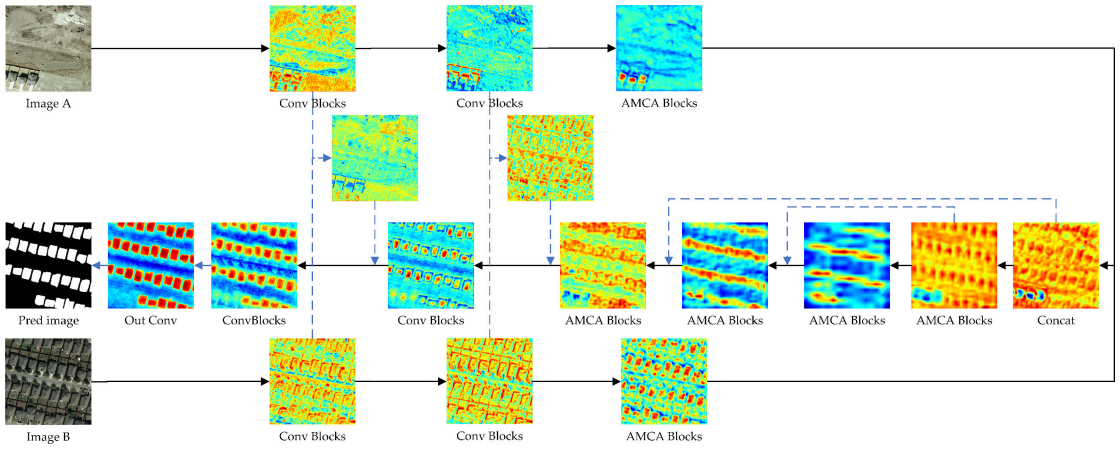

Fig. 12. Network visualization taking bitemporal RS images from LEVIR-CD as an example.

each head in AMCAs can focus on a different region of the image. So even a semantic map of size 2 is sufficient to retain valid information about the features.

To further illustrate that the most representative feature information is retained in the semantic map, we selected four models with different effects and extracted their initial semantic maps extracted by the semantic generation module separately. From Fig. 10(a) to (d), the performance of the models used rises one by one. For presentation purposes, the semantic map size of the chosen model is 4. We calculated the similarity between the tokens in the extracted semantic maps separately using the cosine similarity. We took the absolute value of the cosine similarity and then visualized the cosine similarity matrix using a heat map. From Fig. 10(a) to (d), we can see that the better the performance of the model, the lower the similarity between the tokens of the initial semantic map extracted by the semantic generation module. This experiment shows that the less redundant and more representative the information retained in the initial semantic map is, the better the model will be. The initial semantic map with a high degree of information aggregation will be further updated and improved in subsequent AMCA blocks.

E. Comparison of Model Comprehensive Performance

We compared the combined performance of ACAHNet/8 and other models based on performance metrics, such as the number of parameters, number of floating point operations, and inference time. Data presented in Fig. 11 were all obtained on a server with an Intel Xeon Gold 5120 @ 2.20 GHz CPU and an NVIDIA TITAN RTX (24 GB) GPU. All F1 values were taken from the performance of each model on the SYSU-CD dataset. We evaluated the model inference time using 500 pairs of 256 × 256 pixel images at a time and averaged the results of 10× as the final result. ACAHNet/8 outperformed the other models using the self-attention mechanism in all three metrics (ACAHNet/8 had a larger number of parameters than Bit-CD). Compared to models using only convolution, ACAHNet/8 offers a significant improvement in model effectiveness without a large increase in computational and storage costs. Overall, our model strikes a better balance between accuracy and cost of use than other comparison methods.

F. Network Visualization

In Fig. 12, to better understand the ACAHNet model, we visualize the feature maps of the various stages of the model. For ease of presentation, we zoomed in on the deeper features with a lower spatial resolution to 256 × 256 by linear interpolation. In the visualization diagram, red indicates regions of greater concern to the model, and blue indicates regions of lesser concern to the model. As the semantic map is not available for visualization, it is not shown in Fig. 12.

From the visualization, we can clearly see the various processes of feature extraction, feature fusion, and DC that

the input RS image undergoes. In the two-branch part of the encoder, the model recognizes the semantic objects in each image. In the single-branch part of the encoder, the fused features begin to focus on the semantic objects that may have changed. In this process, skip connections play an important role in recovering and correcting the change information. In Fig. 12, the fused features from the two-branch part of the encoder then suppress possible misdetection by the decoder through skip connections. The localization of the semantic targets and the contours of the semantic targets in each RS image, taken as a whole, are progressively refined with the depth of the model, demonstrating the effectiveness of our proposed modeling strategy.

V. Conclusion

In order to increase the interaction between local and global features, in this article, we propose a hybrid CNN-transformer model ACAHNet for RS images DC using a design paradigm that combines series-parallel association. The AMCA provides the global feature extraction capability of the network, and the PCM provides local features to complement the information that may be lost due to the aggregated semantic map. In addition, unlike any previous network in the field of DC, we broke the conventional fusion strategy restrictions in this field, did not adopt early or late fusion strategies, but explored and revealed another mid-term fusion strategy in the experiment. We have conducted extensive experiments on three publicly available datasets, namely CDD, LEVIR-CD, and SYSU-CD. The experimental data show that our proposed ACAHNet performs well in terms of comprehensive performance compared to other comparative methods. Although we can obtain more outstanding DC by upgrading the initial channels, this will lead to a rapid increase in the number of parameters and computational complexity of the model, and future work will focus more on achieving model performance improvements with less computational and storage cost investment.

References

[1] L. Khelifi and M. Mignotte, "Deep learning for change detection in remote sensing images: Comprehensive review and meta-analysis," *IEEE Access*, vol. 8, pp. 126385–126400, 2020.

[2] A. Singh, "Review article digital change detection techniques using remotely-sensed data," *Int. J. Remote Sens.*, vol. 10, no. 6, pp. 989–1003, 1989.

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

[4] C. Marin, F. Bovolo, and L. Bruzzone, "Building change detection in multitemporal very high resolution SAR images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 5, pp. 2664–2682, May 2015.

[5] X. Huang, L. Zhang, and T. Zhu, "Building change detection from multitemporal high-resolution remotely sensed images based on a morphological building index," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 7, no. 1, pp. 105–115, Jan. 2014.

[6] S. Mahdavi, B. Salehi, W. Huang, M. Amani, and B. Brisco, "A PolSAR change detection index based on neighborhood information for flood mapping," *Remote Sens.*, vol. 11, no. 16, p. 1854, Aug. 2019.

[7] M. Gong et al., "Change detection in synthetic aperture radar images based on deep neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 125–138, Jan. 2016.

[8] 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, Nov. 2013.

[9] T. Adão et al., "Hyperspectral imaging: A review on UAV-based sensors, data processing and applications for agriculture and forestry," *Remote Sens.*, vol. 9, no. 11, p. 1110, 2017.

[10] 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.

[11] W. A. Malila, "Change vector analysis: An approach for detecting forest changes with Landsat," in *Proc. LARS Symposia*, 1980, p. 385.

[12] B. Bai, W. Fu, T. Lu, and S. Li, "Edge-guided recurrent convolutional neural network for multitemporal remote sensing image building change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5610613.

[13] 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.

[14] D. K. Seo, Y. H. Kim, Y. D. Eo, M. H. Lee, and W. Y. Park, "Fusion of SAR and multispectral images using random forest regression for change detection," *ISPRS Int. J. Geo-Inf.*, vol. 7, no. 10, p. 401, Oct. 2018.

[15] O. Ahlqvist, "Extending post-classification change detection using semantic similarity metrics to overcome class heterogeneity: A study of 1992 and 2001 U.S. National land cover database changes," *Remote Sens. Environ.*, vol. 112, no. 3, pp. 1226–1241, Mar. 2008.

[16] G. Verdier and A. Ferreira, "Adaptive Mahalanobis distance and k-nearest neighbor rule for fault detection in semiconductor manufacturing," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 1, pp. 59–68, Feb. 2011.

[17] H. Nemmour and Y. Chibani, "Multiple support vector machines for land cover change detection: An application for mapping urban extensions," *ISPRS J. Photogram. Remote Sens.*, vol. 61, no. 2, pp. 125–133, Nov. 2006.

[18] J. J. Gapper, H. El-Askary, E. Linstead, and T. Piechota, "Coral reef change detection in remote Pacific islands using support vector machine classifiers," *Remote Sens.*, vol. 11, no. 13, p. 1525, Jun. 2019.

[19] 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, Nov. 2005.

[20] T. Kasetkasem and P. K. Varshney, "An image change detection algorithm based on Markov random field models," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 8, pp. 1815–1823, Aug. 2002.

[21] K. Zong, A. Sowmya, and J. Trinder, "Building change detection from remotely sensed images based on spatial domain analysis and Markov random field," *J. Appl. Remote Sens.*, vol. 13, no. 2, p. 1, May 2019.

[22] G. Liu, L. Li, L. Jiao, Y. Dong, and X. Li, "Stacked Fisher autoencoder for SAR change detection," *Pattern Recognit.*, vol. 96, Dec. 2019, Art. no. 106971.

[23] F. Samadi, G. Akbarizadeh, and H. Kaabi, "Change detection in SAR images using deep belief network: A new training approach based on morphological images," *IET Image Process.*, vol. 13, no. 12, pp. 2255–2264, May 2019.

[24] M. Zhang and W. Shi, "A feature difference convolutional neural network-based change detection method," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 10, pp. 7232–7246, Oct. 2020.

[25] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[26] X. X. Zhu et al., "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[27] 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, p. 1382, Jun. 2019.

[28] 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, 2022.

[29] 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, Nov. 2021, Art. no. 2159049.

[30] 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, 2022, Art. no. 5604816.

[31] 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, p. 3394, Aug. 2021.

[32] 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, 2021.
[33] 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, p. 1662, May 2020.
[34] O. Ronneberger, P. Fischer, and T. J. A. Brox, "U-Net: Convolutional networks for biomedical image segmentation," 2015, *arXiv:1505.04597*.
[35] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-Net architecture for medical image segmentation," in *Proc. Int. Workshop Multimodal Learn. Clin. Decis. Support*, in Lecture Notes in Computer Science, vol. 11045. Granada, Spain, Jun. 2018, pp. 3–11.
[36] L. Mou, L. Bruzzone, and X. X. Zhu, "Learning spectral–spatial–temporal features via a recurrent convolutional neural network for change detection in multispectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 924–935, Feb. 2019.
[37] H. Chen, C. Wu, B. Du, L. Zhang, and L. Wang, "Change detection in multisource VHR images via deep Siamese convolutional multiple-layers recurrent neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2848–2864, Jul. 2020.
[38] 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.
[39] 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.
[40] C. X. Zhang et al., "A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 183–200, Aug. 2020.
[41] J. Zheng, Y. Feng, C. Bai, and J. Zhang, "Hyperspectral image classification using mixed convolutions and covariance pooling," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 522–534, May 2020.
[42] A. Vaswani et al., "Attention is all you need," 2017, *arXiv:1706.03762*.
[43] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[44] Z. Tian et al., "Synchronous transformers for end-to-end speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 7884–7888.
[45] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[46] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," 2021, *arXiv:2103.14030*.
[47] 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.
[48] W. G. C. Bandara and V. M. Patel, "A transformer-based Siamese network for change detection," 2022, *arXiv:2201.01293*.
[49] H. Chen, Z. Qi, and Z. Shi, "Remote sensing image change detection with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5607514.
[50] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," 2020, *arXiv:2006.04768*.
[51] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 1800–1807.
[52] M. Sandler et al., "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2018, pp. 4510–4520.
[53] F. Milletari, N. Navab, and S. A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis.*, 2016, pp. 565–571.
[54] 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. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 42, no. 2, pp. 1–7, 2018.
[55] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi, "Street-view change detection with deconvolutional networks," *Auto. Robots*, vol. 42, no. 7, pp. 1301–1322, 2018.
[56] C. Ling, J. Huang, and H. Zhang, "AUC: A better measure than accuracy in comparing learning algorithms," in *Proc. 16th Conf. Can. Soc. Comput. Stud. Intell.*, 2003, pp. 329–341.
[57] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 233–240.

Xiaofeng Zhang received the B.E. degree from the Henan University of Technology, Zhengzhou, China, in 2021. He is currently pursuing the master's degree with the School of Information Science and Engineering, Xinjiang University, Ürümqi, China.

His research interests include computer vision and remote sensing image change detection.

Shuli Cheng received the Ph.D. degree from the School of Information and Engineering, Xinjiang University, Ürümqi, China, in 2021.

He is currently working with the School of Information Science and Engineering, Xinjiang University, where he as a Post-Doctoral Researcher with the College of Mathematics and System Science. His research interests include computer vision, natural language processing, multimodal processing, and remote sensing images change detection.

Liejun Wang received the Ph.D. degree from the School of Information and Communication Engineering, Xi'an Jiaotong University, Xi'an, China, in 2012.

He is currently a Professor with the School of Information Science and Engineering, Xinjiang University, Ürümqi, China. His research interests include wireless sensor networks, computer vision, and natural language processing.

Haojin Li received the B.E. degree from the North China University of Water Resources and Electric Power, Zhengzhou, China, in 2020. He is currently pursuing the master's degree with the School of Information Science and Engineering, Xinjiang University, Ürümqi, China.

His research interests include computer vision and remote sensing images change detection.