Dual-Attention-Guided Multiscale Feature Aggregation Network for Remote Sensing Image Change Detection

Hongjin Ren, Min Xia, *Member, IEEE*, Liguo Weng, Kai Hu, and Haifeng Lin

***Abstract*—Remote sensing image change detection plays an important role in urban planning and environmental monitoring. However, the existing change detection algorithms have limited ability in feature extraction, feature relationship understanding, and capture of small target features and edge detail features, which leads to the loss of some edge detail information and small target features. To this end, a new dual-attention-guided multiscale feature aggregation network is proposed. In the encoding stage, the fully convolutional dual-branch structure is used to extract the semantic features of different scales, and then, the multiscale adjacent semantic information aggregation module is used to aggregate the adjacent semantic features at different scales, which can better capture and fuse the features of different scales, thereby improving the accuracy and robustness of change detection. In the decoding stage, the dual-attention fusion module is proposed to guide and fuse the features extracted from different scales along the spatial and channel directions and reduce the background noise interference. In addition, this article also proposes a three-branch feature fusion module and a global semantic information enhancement module to make the network better integrate global semantics and differential semantics and further integrate high-level semantic features. We also introduce an auxiliary classifier in the decoding stage to provide additional supervision signals and fuse the output of the three auxiliary classifiers with the output of the main decoder to further achieve multiscale feature fusion. The comparative experiments on three remote sensing datasets show that the proposed method is superior to the existing change detection methods.**

***Index Terms*—Change detection, deep learning, multiscale fusion, remote sensing image.**

I. Introduction

REMOTE sensing image change detection refers to comparing the differences between two remote sensing images at different times in the same area to detect the changes of the surface in time. In this process, each picture is assigned a binary label, namely, label 0 (unchanged) and label 1 (changed). It is an important application of remote sensing technology in the monitoring and analysis of surface changes. It can help people understand the dynamic changes of the surface and has many applications in our lives, such as urban expansion [1], [2], land management [3], [4], environmental monitoring [5], [6], disaster assessment [7], [8], and other fields [9], [10], [11]. Of course, remote sensing image change detection technology usually faces many interference factors and challenges, such as the speed of ground object change, the occlusion and occlusion of ground objects, the change of illumination conditions, and the change of land cover types [12]. Therefore, how to identify the actual changes of remote sensing images with interference factors becomes extremely challenging.

With the continuous improvement of satellite resolution and the development of unmanned aerial vehicle remote sensing technology, the resolution and spatial–temporal resolution of remote sensing images have also been greatly improved. The acquisition and use of high-resolution remote sensing images and time-series remote sensing images provide more data support and technical support for remote sensing image change detection and also greatly promote the development of change detection technology. Nowadays, many change detection methods have been proposed, but they can be roughly divided into two categories: traditional change detection methods and change detection methods based on deep learning. Before the rise of deep learning methods, traditional remote sensing image change detection methods dominated the mainstream, including methods based on pixel comparison, methods based on feature extraction, and methods based on spatiotemporal models. Pixel comparison method is a common detection method, including simple pixel comparison method [13] and ratio pixel comparison method [14], [15]. This kind of method is easy to operate. It only needs to compare the images of the two phases pixel by pixel and judge whether there is a change according to the difference between the pixels. Among them, the simple pixel comparison method is the most basic method. It is simple to calculate by directly comparing the pixel gray value difference between the two phases, but it is prone to errors in the case of inaccurate image registration and large noise interference. The ratio pixel comparison method can better reflect the real change degree of the change target by calculating the ratio of the pixel gray values of the two time phases. However, these methods cannot deal with complex remote sensing images, such as different

Manuscript received 31 October 2023; revised 8 December 2023 and 17 January 2024; accepted 25 January 2024. Date of publication 6 February 2024; date of current version 22 February 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 42075130. *(Corresponding author: Min Xia.)*

Hongjin Ren, Min Xia, Liguo Weng, and Kai Hu are with the Collaborative Innovation Center on Atmospheric Environment and Equipment Technology, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: xiamin@nuist.edu.cn).

Haifeng Lin is with the College of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China.

Digital Object Identifier 10.1109/JSTARS.2024.3362370

lighting conditions and different shooting time, so there will be false detection and missed detection. The method based on feature extraction is based on the direct comparison method, and the detection accuracy can be significantly improved by using the related algorithms of machine learning. In 2009, Celik [16] used principal component analysis and K-means clustering to distinguish between changed pixels and unchanged pixels in remote sensing images. In 2016, Jia et al. [17] used near-infrared channels and red channels to calculate the normalized vegetation index and monitor small changes in vegetation. This kind of method can overcome the limitations of direct comparison method, but it needs professional knowledge and large computing resources. The method based on spatiotemporal model can overcome the limitations of the method based on feature extraction. This method mainly uses the temporal properties of remote sensing images to establish a change detection model. For example, Botsch and Nossek [18] used a feature selection method for change detection in multivariate time series to analyze remote sensing image data at different time points to improve the accuracy and robustness of change detection. However, these methods require high temporal and spatial resolution and require a lot of computing resources and professional knowledge support. With the continuous development of remote sensing technology and the rise of deep learning methods, the limitations of traditional change detection methods have gradually emerged. Traditional methods mainly rely on manual feature extraction and threshold setting, which are sensitive to data quality and environmental changes. At the same time, there are also problems such as large computational complexity and false detection rate.

Therefore, in recent years, deep learning methods have gradually become a research hot spot in remote sensing image change detection [19]. The deep learning method can automatically learn the features in the image without manual feature extraction [20], [21], [22] and has strong nonlinear modeling ability, which can better adapt to the change detection task. From the early proposed fully convolutional neural network (FCN) [23], [24], [25], to the typical convolutional neural network (CNN) [26], [27], [28], and then to the recently emerging Transformer network [29], [30], these methods have been widely used in remote sensing image change detection and achieved good results. Ronneberger et al. [31] proposed a UNet network for medical image segmentation tasks. UNet gradually restores image space and edge detail information through skip connection and layer-by-layer upsampling and has achieved excellent results in medical image segmentation tasks. Because medical images and remote sensing images have similar characteristics, UNet is also used in remote sensing image change detection tasks and has achieved good results. For example, Lv et al. [32] embedded multiscale information attention in the backbone network of UNet to realize the multiscale information fusion task of bit-time images. Zhang et al. [33] combined the features of different stages of the conjoined feature extractor to improve the ability of the encoder and the feature extractor to extract the characterization features. Fu et al. [34] proposed a dual-attention scene parsing network DANet, which uses a self-attention mechanism to model global semantic information. Wang et al. [35] proposed the HRNet network, which improves the network's ability to capture details through high-resolution feature fusion and obtains the best performance in multiple image segmentation tasks. Ma et al. [36] used multiscale banded convolution to extract multiscale features of images, realized the fusion of multiple features, and obtained a finer-grained feature representation. Xu et al. [37] used the feature pyramid to make the model extract features at different levels and strengthen fusion to understand the semantic features of images more comprehensively. The above methods all use multiscale information for semantic fusion to a certain extent. However, when fusing features of different scales, if there is no appropriate attention mechanism or weight adjustment, some irrelevant information may be introduced into the model, thus reducing the performance of the model. In addition, the CNN method is limited in processing long-range context information, so, in some cases, the global context may be ignored and the change area cannot be dynamically focused. In this case, the introduction of attention mechanism to improve the ability of global context modeling and local feature attention ability, for example, is a feasible way, such as spatial attention [38] and channel attention [39]. For example, Song et al. [40] used the spatial attention module (SAM) in the encoding and decoding stages, so that the network assigns more weight to the region of interest, but ignores the global context information to a certain extent. Choi and Kim [41] consider the channel correlation between dual-time images and weight the features of different channels, so that the model can better understand the global context information. However, channel attention usually ignores the importance difference of different positions in the image and cannot deal with the change detection task that needs to focus on specific position information. Therefore, Woo et al. [42] unified channel attention and spatial attention, not only paying attention to important information between channels but also dynamically adjusting attention allocation at different locations. Because the convolutional block attention module (CBAM) is too simple for the combination of channel attention and spatial attention, the simple CBAM has been unable to adapt to some complex image tasks, such as remote sensing image change detection tasks. Previous deep learning methods have been exploring multiscale fusion of low-level to high-level features for many years. For example, Zhu et al. [43] insert channel attention into spatial attention, which combines spatial advantages and multichannel advantages and extracts deeper features from the fused features for classification. Ma et al. [44] used channel attention and densely connected atrous spatial pyramid pooling to enhance the feature extraction ability of the network in the encoding stage. Yan et al. [45] reconstructed the original data into a multiscale layout from the data end and then combined the channel information and angle information to construct an attention mechanism for multiscale fusion. These methods do not take into account the interaction between the adjacent semantics of low-level features to high-level features, and these methods do not perform well in attention fusion strategies. Our network not only makes full use of the multiscale time features extracted by the backbone network in the encoding stage but also enhances the representation ability of the bitemporal features. In the

decoding stage, we also model the spatial information and channel information and perform weighted fusion of these two attention-guided features, so that the network can better pay attention to and guide the location information and channel information of the bitemporal features and better perform multilevel feature fusion. And we also perform multiscale fusion on the high-level semantic features extracted from the backbone network to make the model better understand the global context semantic information. In summary, our model is more reasonable for the fusion strategy from low-level features to high-level features, and we take into account the multiscale fusion strategy in the encoding–decoding stage to better perform multilevel feature fusion.

In general, the above described change detection methods have some problems. First, previous change detection methods based on deep learning either focus on multiscale feature fusion strategies or focus on attention allocation strategies. We know that relying only on multiscale feature fusion will make the model enter some irrelevant information, and these irrelevant noises will affect the recognition of the final change region. If only attention allocation is used, the feature extraction will not be rich enough, and sufficient attention guidance cannot be performed. Of course, change detection methods that simultaneously consider multiscale feature extraction and attention allocation also follow. For example, Song et al. [40] and Ma et al. [36] used spatial attention allocation while using multiscale features, so that the location information of the two-time image was fully allocated to a certain extent, but the channel information was ignored, thus ignoring the context modeling. Wang et al. [46] introduced the CBAM on the basis of multiscale features, so that the network can simultaneously model spatial information and channel information. As mentioned above, the CBAM is too simple to allocate spatial attention and channel attention and cannot adapt to remote sensing image change detection tasks, especially in the case of unbalanced samples.

Therefore, this article proposes a dual-attention-guided multiscale feature aggregation network (DAMFANet) to solve the above problems. We designed four modules to improve the accuracy and robustness of the algorithm and also introduced an auxiliary classifier to help network training. First, we select a pretrained ReaNet34 to extract bitemporal features. Then, in order to make full use of the dual-time features extracted by the backbone network, we propose a multiscale adjacent semantic information aggregation module (MASAM), which is used to integrate semantic information at different scales to achieve multiscale feature fusion and sharing, so as to obtain more discriminative feature representation. At the same time, in order to make better use of the multiscale information extracted in the encoding stage and allocate more reasonable attention to the bitemporal features, we propose a dual-attention fusion module (DAFM). While modeling the spatial information and channel information, the two attention-guided features are weighted and fused, so that the network can better pay attention to and guide the location information and channel information of the bitemporal features and reduce the interference of irrelevant noise. In addition, a three-branch feature fusion module (TBFFM) is designed to fuse the global semantic information and differential semantic information of the dual-time remote sensing image, while retaining the original information to avoid information loss. At the same time, considering the importance of global semantic information, we propose a global semantic information enhancement module (GSEM), which performs multiscale fusion of high-level semantic features to make the model better understand the global context semantic information. The main contributions of this article can be summarized as follows.

1) A DAMFANet framework is proposed. Previous deep learning methods have some problems in multiscale feature fusion and attention allocation strategies, especially the loss of feature boundary information and small target information. This method makes full use of the rich feature information in the remote sensing image through the cross-fusion of different scales, uses the unique dual attention to guide the fusion in space and channel information at the same time, restores the target area, edge details, and small target features in the process of dual-time remote sensing image change as much as possible, and also effectively avoids the occurrence of missed detection and false detection. The network is end-to-end trainable, making our network training simpler.
2) The MASAM, DAFM, TBFFM, and GSEM are proposed. The MASAM can integrate the adjacent semantic information of different scales in the feature extractor to achieve multiscale feature fusion and sharing. The DAFM can adaptively capture the changing information on the spatial and channel dimensions and can better extract the correlation information of the input feature map and suppress the information of the unchanged area. The TBFFM can fuse the global information and difference information of the input, which can improve the representation ability of the model to the input features and refine the edge texture features better. The GSEM can aggregate and refine features at different scales and can improve the expression ability and discrimination of global semantic features, thereby enhancing the network's ability to extract and recognize global semantic information.
3) We test our proposed DAMFANet on three datasets, including our own proposed dataset BICDD, public dataset CDD [47], and public dataset LEVIR-CD [48]. The test results show that compared with the previous change detection algorithm based on deep learning, the DAMFANet is a new algorithm with higher accuracy and stronger robustness.

The rest of this article is organized as follows. In Section II, we introduce each module in the model in detail. In Section III, we introduce the composition of the dataset. In Section IV, we test the performance of the model through experiments. Finally, Section V concludes this article.

II. Methodology

Our research aims to solve the change detection task, that is, to accurately detect change regions and invariant regions in two dual-time remote sensing images. Previous studies have shown that the CNN has been widely used in the field of

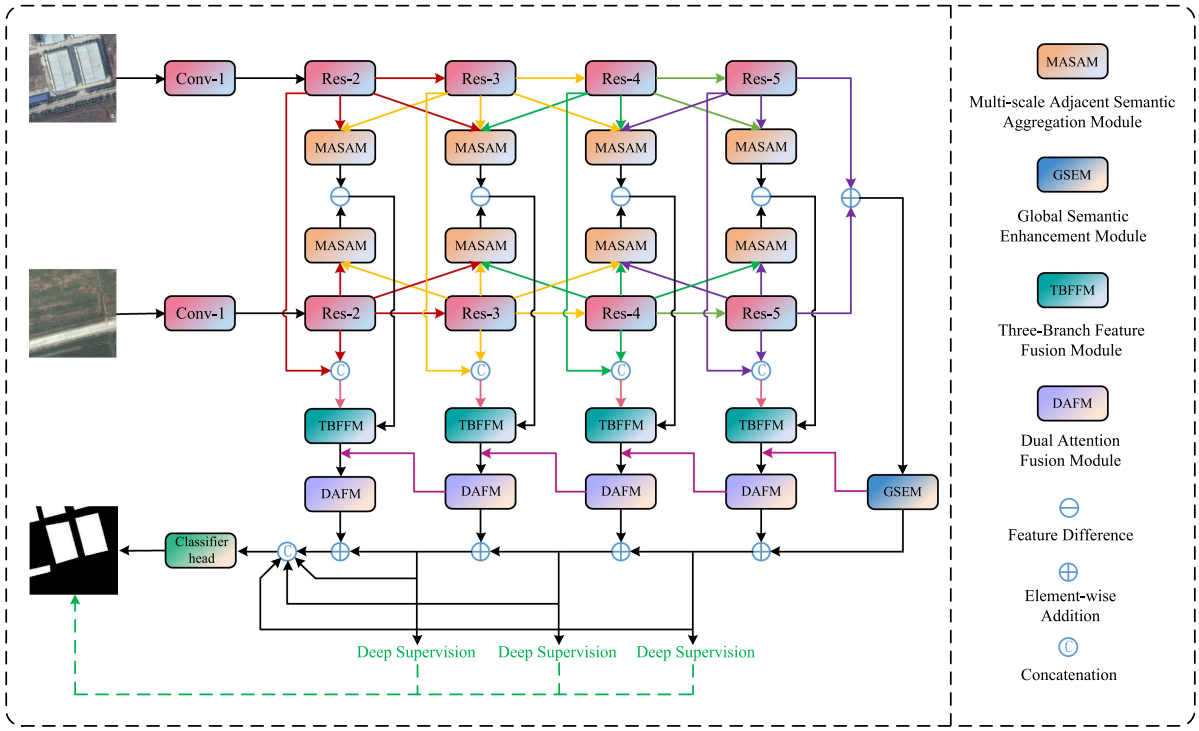

Fig. 1. Proposed DAMFANet framework. First, the temporal feature is extracted from a registered pair of images by the weight-shared ResNet-34 in the encoding stage. Then, we use MASAM to aggregate the temporal features of adjacent scales to enhance its feature representation ability. TBFFM aims to fuse global features and difference features to better capture the change area of dual-time features. In the decoding stage, DAFM can adaptively capture the changing information in the spatial and channel dimensions, which can better extract the correlation information of the input feature map and suppress the information of the unchanged region. GSEM aims to aggregate and refine features at different scales to improve the expression ability and discrimination of global semantic features. Finally, a change map is obtained by gradually aggregating multiscale temporal difference features. And we introduce deep supervision signals into the model to help the model converge better.

semantic segmentation [49], [50], because they can effectively classify images at the pixel level. Therefore, we choose the CNN as the basic framework of the algorithm, and we believe that this method can also be applied to change detection tasks. Our goal is to design an end-to-end learning model that can automatically extract features from two remote sensing images and classify change regions and invariant regions. We believe that the Siamese network [51], [52] can be well applied to the change detection task by extracting the features of the two images by sharing the weights and comparing the differences between them. Therefore, we can use the Siamese network to compare two photos taken at the same place and at different times and detect differences between them. We believe that this method can effectively solve the change detection task and provide a new idea and method for change detection research.

Our change detection model adopts a U-shaped structure based on the Siamese network, and its overall architecture is composed of an FCN composed of an encoder and a decoder. Among them, the encoder is divided into two branches with shared weights. We use the pretrained ResNet-34 [38] network to extract global semantic information and differential semantic information of dual-temporal remote sensing images, while the decoder is used to perform multiscale fusion and upsampling of the extracted feature maps to obtain change detection results. In order to further improve the performance of the model, we also introduce four auxiliary modules and three auxiliary classifiers. In the encoding stage, in order to make full use of the semantic information of different scales extracted by the two encoder branches, the MASAM we designed can integrate the adjacent semantic information extracted by the feature extractor at different scales, realize the information interaction and enhancement of multiscale features, and help the network to better capture the key features in the image. In the decoding stage, the DAFM can adaptively capture the changing information in the spatial and channel dimensions and can better extract the correlation information of the input feature map and suppress the information of the unchanged region. The TBFFM can fuse the global information and difference information of the input, which can improve the representation ability of the model to the input features and refine the edge texture features better. The GSEM can aggregate and refine features at different scales and can improve the expression ability and discrimination of global semantic features, thereby enhancing the network's ability to extract and recognize global semantic information. The complete architecture of the model is shown in Fig. 1.

A. Backbone

We propose a change detection network based on the Siamese network and the U-shaped structure, in which we use pretrained ResNet-34 as the backbone network. It consists of five convolution blocks, which are named Conv-1, Res-2, Res-3,

Res-4, and Res-5. We only use the four convolution blocks from Res-2 to Res-5 and use them as a connected network to generate four different scale outputs, so that we can cross-integrate multiscale semantic information later. In the task of change detection, the choice of encoder is very critical, because the quality of the encoder network directly affects the performance of change detection. Therefore, in general, we choose networks that perform well in large-scale image classification tasks as encoders, such as VGG [39], ResNet, DenseNet [53], etc. These networks have proven to have strong feature expression capabilities, but the computational efficiency of the encoder should also be considered. We know that as the depth of the network becomes deeper and deeper, the features learned by the network are more comprehensive, but the training difficulty of the network also becomes larger. Moreover, when the network is too deep, the disappearance of the gradient in the back propagation becomes a problem. In order to solve these problems, we use the residual structure of ResNet, which realizes the direct connection between the front and back network layers through skip connection, so as to avoid the problem of gradient disappearance or gradient explosion in the training process of deep neural network. Jump connection allows information to flow more easily in the network, making training easier to converge. Specifically, the input and output of the residual structure are feature maps, in which the output is obtained by a series of convolution, batch normalization, and activation operations on the input, and the jump connection operation is also performed, that is, the input feature map is added directly to the output feature map. In this way, the output feature map can contain both the information from the input feature map and the output feature map of the previous layer, which makes the network deeper and ensures the stability of the training. In ResNet, the core idea of the residual structure is $H(X) = F(X) + X$, where X is the input, $F(\mathrm{X})$ is the output, and $F(\cdot)$ represents a series of convolution, batch normalization and activation operations. Therefore, when a layer is determined to be a redundant layer during the network training process, learning makes $F(\mathrm{X}) = 0$, that is, $H(\mathrm{X}) = 0$, so that the input of the layer network is exactly the same as the input of the previous layer, thus avoiding the problem of network degradation to a certain extent.

B. Multiscale Adjacent Semantic Information Aggregation Module

Although ResNet has been proved to have good feature representation ability as an encoder in change detection tasks, it is important to note that the semantic features extracted by the last four convolution blocks of ResNet34, at different scales, are independent of each other. This means that there may be information islands between the semantic information of different scales extracted by the last four convolution blocks of ResNet34, resulting in insufficient feature representation and affecting the performance of the model. Therefore, in order to make full use of the semantic information of different scales extracted by the encoder at different stages and strengthen the information interaction of multiscale semantic features, we design a MASAM to aggregate semantic information of different scales, realize the information interaction and enhancement of multiscale features, and help the network better capture the key features in the image.

The structure of the module is shown in Fig. 2. Since we only output the features extracted from the four convolution blocks after the encoder, the adjacent aggregation between the four scales has only two adjacent scales and three adjacent scales. Therefore, we use three branches to form the whole module. The sizes of feature maps f_a, f_b, and f_c are $(C/2) \times (2 \times H) \times (2 \times W)$, $C \times H \times W$, and $(2 \times C) \times (H/2) \times (W/2)$, respectively. When the two adjacent scales are aggregated, a 1×1 convolution is first performed on f_b to compress the number of feature channels to half of the original, extract more useful channel information, and output to ${f_b}'$. Then, a 3×3 convolution is performed to further extract and enhance the features, and the output is ${f_b}''$. If it is aggregated with the next scale, a 1×1 convolution is performed on f_c first, and then, a 3×3 convolution is performed to compress the number of channels to half of the original and further extract and enhance the features, and then, it is upsampled to restore the same size as the feature map f_b, and the output is ${f_c}'$. Then, the two features are spliced on the channel dimension to obtain a richer feature map, and then, it is first subjected to a 1×1 convolution, and then, a 3×3 convolution is performed. The channel is compressed to C and the feature information is further extracted, and the output is ${f_d}'$. Then, add operations are performed on ${f_d}'$, ${f_b}'$, and ${f_c}'$, and the feature maps of these two scales are fused to improve the feature expression ability of the model. The final output is f_{out}, and the size is $C \times H \times W$. If it is aggregated with the previous scale feature map, a maximum pooling downsampling is performed on the feature map f_a, and then, a 3×3 convolution is performed to effectively extract the locally strongest feature, and the output is ${f_a}'$. Then, the remaining operation is similar to the previous one, and the feature maps of these two scales are fused. The final output is f_{out}, and the size is also $C \times H \times W$. When the semantic information of three adjacent scales is aggregated, ${f_a}'$, ${f_b}''$, and ${f_c}'$ are spliced along the channel dimension to obtain a feature map with more channels and richer features. Then, it is first subjected to a 1×1 convolution, and then, a 3×3 convolution is performed. The channel is compressed to C, and the feature information is further extracted to obtain ${f_d}'$. Finally, the addition operation is performed on ${f_d}'$, ${f_b}'$, ${f_a}'$, and ${f_c}'$, and the feature maps of the three scales are fused. The final output is f_{out}, and the size is also $C \times H \times W$. In general, the module can effectively integrate the adjacent semantic information extracted by the feature extractor at different scales, realize the information interaction and enhancement of multiscale features, and help the network better capture the key information and features in the image. The calculation formula of the above process is as follows:

$$ {f_a}' = f^{3\times3}(Maxpool(f_a)) \tag{1} $$

$$ {f_b}' = f^{1\times1}(f_b) \tag{2} $$

$$ {f_b}'' = f^{3\times3}\left(f^{1\times1}(f_b)\right) \tag{3} $$

$$ {f_c}' = Upsample(f^{3\times3}(f^{1\times1}(f_c))) \tag{4} $$

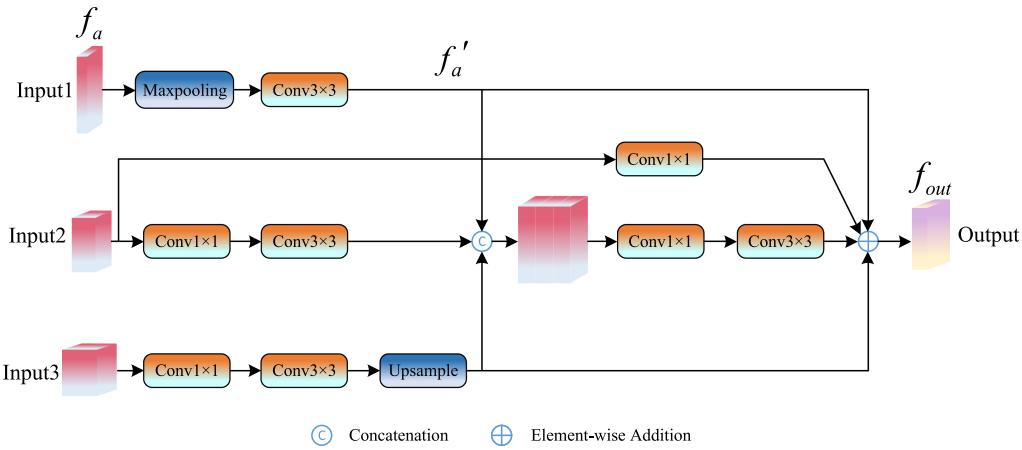

Fig. 2. Structure of the MASAM. f_a, f_b, and f_c represent the characteristics of different scales from the backbone network. f_d represents the features obtained by splicing the features of different scales on the channel dimension after a series of operations.

$$f_{\text{out}} = f^{3\times3}\left(f^{1\times1}\left(\left[f_b'';f_c'\right]\right)\right) + f_b' + f_c' \tag{5}$$

$$f_{\text{out}} = f^{3\times3}\left(f^{1\times1}\left(\left[f_b'';f_a'\right]\right)\right) + f_b' + f_a' \tag{6}$$

$$f_{\text{out}} = f^{3\times3}\left(f^{1\times1}\left(\left[f_b'';f_a';f_c'\right]\right)\right) + f_b' + f_a' + f_c'. \tag{7}$$

In the formula, $f^{1\times1}(\cdot)$ represents the 2-D convolution, batch normalization, and ReLU activation function with convolution kernel size of 1, and $f^{3\times3}(\cdot)$ represents the 2-D convolution, batch normalization, and ReLU activation function with convolution kernel size of 3. $MaxPool(\cdot)$ denotes maximum pooling, $Upsample(\cdot)$ denotes bilinear interpolation upsampling, and $[;]$ denotes splicing operation. Among them, f_a, f_b, and f_c represent the feature maps of three different scales generated by the encoder.

C. Three-Branch Feature Fusion Module

Considering that simple addition, splicing and convolution operations on two different feature information cannot make full use of these two types of feature information and even destroy the integrity and diversity of these two types of information, resulting in information redundancy. Therefore, for the global semantic information of the original image and the difference information after MASAM, we propose a new TBFFM to aggregate these two feature information, which can improve the model's ability to represent the input features and refine the edge texture features better.

The structure of the module is shown in Fig. 3. The module consists of three branches. The left branch is used to extract global features from the original image, the right branch is used to extract the difference features after the MASAM, and the middle branch is used to fuse and enhance the global information of f_{cat} and the difference information of f_{sub}. The size of the feature graph f_{cat} is $(2 \times C) \times H \times W$, and the size of the feature graph f_{sub} is $C \times H \times W$. For the left branch, it is divided into two branches. One branch performs global average pooling on f_{cat} to obtain the feature map of $(2 \times C) \times 1 \times 1$ and then performs two 1×1 convolutions to compress the number of feature channels to half of the original and extract more useful channel information. The other branch performs 1×1 convolution on f_{cat} first and, then, 3×3 convolution to further process and enhance the global features. For the two branches of the right branch, one performs two 1×1 convolutions on f_{sub}, and the other performs two 3×3 convolutions on f_{sub} to extract and emphasize the difference features of the image. For the intermediate branch, f_{cat} and f_{sub} are first spliced along the channel dimension, and then, 1×1 convolution and 3×3 convolution are performed. Then, the output of the three branches is added and fused, and then, a 3×3 convolution is performed to obtain the output f_{out}. The module can effectively

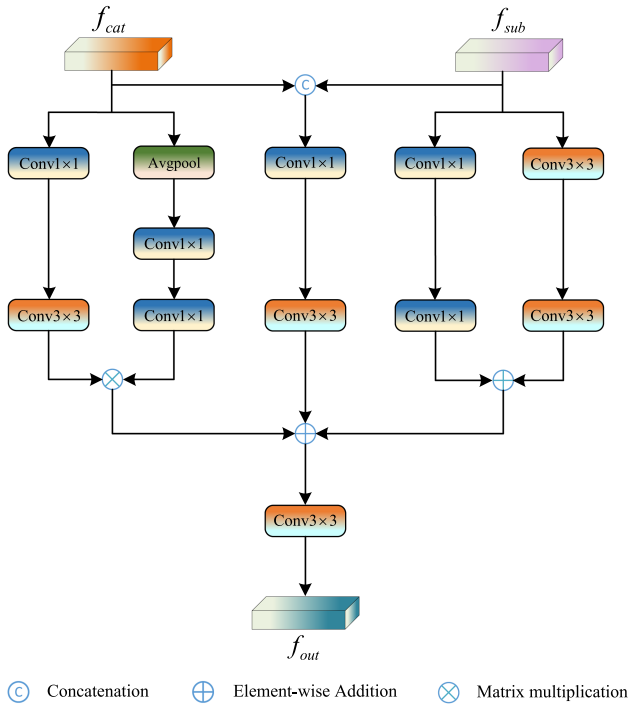

Fig. 3. Structure of the TBFFM.

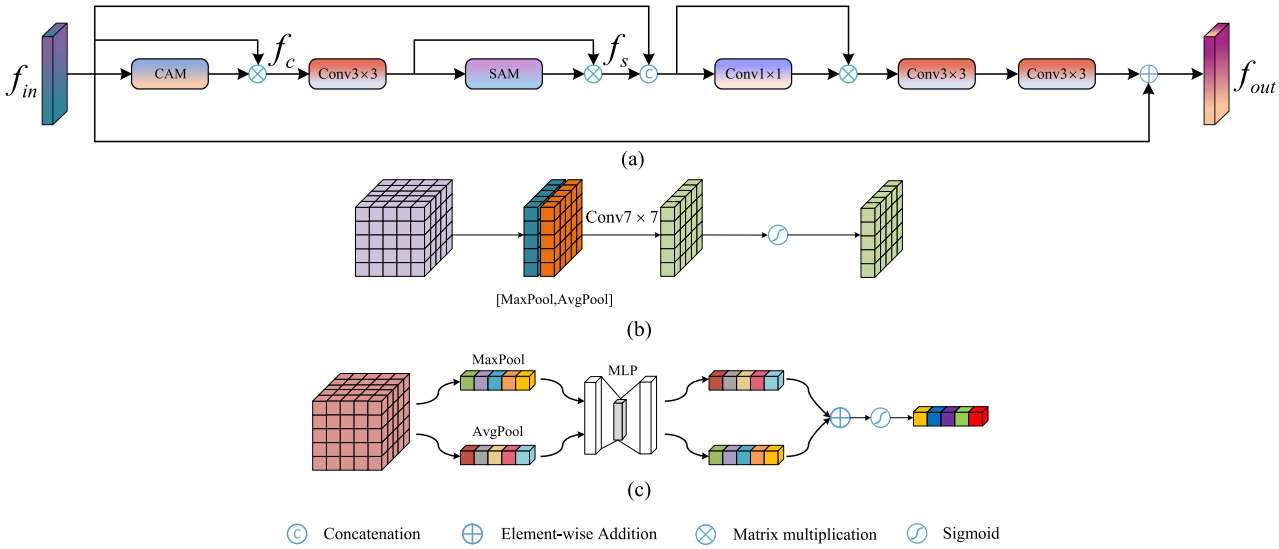

Fig. 4. Structure of the DAFM. (a) Proposed DAFM. (b) Detailed structure of the SAM. (c) Detailed structure of the CAM.

fuse the global features extracted from the original image and the difference features after the MASAM, so that the model can understand the image more comprehensively and capture the key features in the image. The calculation formula of the above process is as follows:

$$f_{\text{out1}} = f^{3\times3}\left(f^{1\times1}\left(f_{\text{cat}}\right)\right) \otimes f^{1\times1}(f^{1\times1}(Avgpool(f_{\text{cat}}))) \quad (8)$$

$$f_{\text{out2}} = f^{1\times1}\left(f^{1\times1}\left(f_{\text{sub}}\right)\right) + f^{3\times3}(f^{3\times3}(f_{\text{sub}})) \quad (9)$$

$$f_{\text{out3}} = f^{3\times3}\left(f^{1\times1}\left([f_{\text{cat}}; f_{\text{sub}}]\right)\right) \quad (10)$$

$$f_{\text{out}} = f^{3\times3}\left(f_{\text{out1}} + f_{\text{out2}} + f_{\text{out3}}\right). \quad (11)$$

In the formula, $AvgPool(\cdot)$ denotes the global average pooling, and $\otimes$ denotes the element-by-element multiplication. $f^{1\times1}(\cdot)$ represents a 2-D convolution, batch normalization, and ReLU activation function with a convolution kernel size of 1, and $f^{3\times3}(\cdot)$ represents a 2-D convolution, batch normalization, and ReLU activation function with a convolution kernel size of 3. f_{out1}, f_{out2}, and f_{out3} represent the outputs of the left branch, right branch, and middle branch, respectively. Let f_{R1} and f_{R2} represent the original feature map generated by the two branches of the encoder, and f_{S1} and f_{S2} represent the feature map of the original feature map after the MASAM. The calculation formulas of global feature map f_{cat} and difference feature map f_{sub} are as follows:

$$f_{\text{cat}} = [f_{R1}; f_{R2}] \quad (12)$$

$$f_{\text{sub}} = abs\left(f_{S1} - f_{S2}\right). \quad (13)$$

where $[;]$ denotes the splicing operation and $abs(\cdot)$ denotes the absolute difference operation.

D. Dual Attention Fusion Module

Remote sensing images usually contain a large amount of background information and noise, and key change information may be submerged in these data. Therefore, if the change areas that need to be paid attention to are not clearly distinguished, it will be difficult for the network to assign accurate labels to each pixel. The CBAM is a lightweight attention module [42]. It consists of two submodules, channel attention module (CAM) and SAM, which perform channel and spatial attention, respectively. Inspired by the CBAM, we propose a new DAFM, which can automatically learn and assign the weights of the changed region and the unchanged region, so that the network can give greater weight to the pixels of the changed region and suppress the interference noise of the unchanged region, so as to better capture the change information.

Fig. 4 describes the structure of this module, where the size of the input feature graph f_{in} is $C \times H \times W$. First, the channel refinement of the input features is performed by the CAM, that is, the average pooling operation and the maximum pooling operation are performed on the input features, and the feature map is compressed into two tensors of $C \times 1 \times 1$ size. Then, these two tensors are input into the multilayer perceptron (MLP), and the output of MLP is combined by summing the elements. Finally, a channel weight vector W_c is obtained by the Sigmoid function, and then, the channel weight vector W_c is weighted by the input feature f_{in}, which can strengthen the channels that are useful for the change detection task and suppress the unimportant channels. Then, the feature map refined by the CAM is first subjected to a 3×3 convolution and then further refined by the SAM in the spatial dimension. First, $AvgPool$ and $MaxPool$ operations are performed on the f_c after a 3×3 convolution along the channel direction. At this time, the feature map is compressed into two tensors of size $1 \times H \times W$, and then, the two tensors are spliced, and 7×7 convolution is used to capture a wider range of spatial information and convert it into a higher level feature representation. Finally, a spatial weight vector W_s is obtained by the Sigmoid function, and then, an element-by-element multiplication operation is performed on the f_c after a 3×3 convolution and the spatial weight vector W_s, which can emphasize the changing region and suppress the

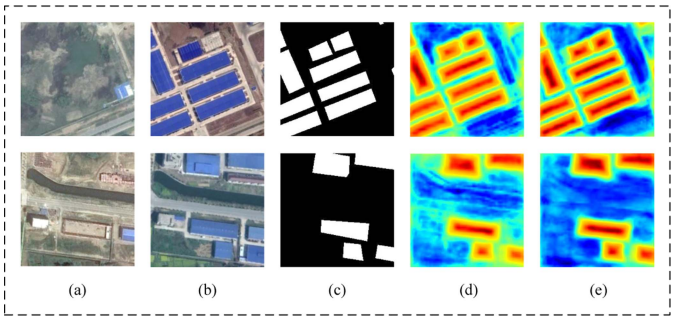

Fig. 5. DAFM heat map comparison. (a) and (b) Bitemporal remote sensing images. (c) Label. (d) Feature heat map without DAFM in the network. (e) Feature heat map with DAFM in the network.

invariant region. The calculation formula of the above process is as follows:

$$W_c = \sigma\left(\text{MLP}(Avgpool(f_{\text{in}})) + \text{MLP}\left(Maxpool(f_{\text{in}})\right)\right) \tag{14}$$

$$f_c = f_{\text{in}} \otimes W_c \tag{15}$$

$$W_s = \sigma(f^{7\times7}([Avgpool(f^{3\times3}(f_c)); Maxpool(f^{3\times3}(f_c))])) \tag{16}$$

$$f_s = f^{3\times3}(f_c) \otimes W_s \tag{17}$$

where $\sigma(\cdot)$ represents the Sigmoid activation function, and $\otimes$ represents the element-by-element multiplication. $f^{3\times3}(\cdot)$ denotes 2-D convolution, batch normalization, and ReLU activation function with convolution kernel size of 3, and $f^{7\times7}(\cdot)$ denotes 2-D convolution and batch normalization with convolution kernel size of 7. f_c represents the feature map after CAM refinement, and f_s represents the feature map after SAM refinement.

Then, f_s is spliced with the original feature map, and then, the feature map after 1×1 convolution is multiplied by the previous spliced feature map to obtain f_d element by element, so that the global semantic information of the original feature map can be retained and the important features of channel direction and spatial direction can be fused. Then, two 3×3 convolutions of f_d are added to the input features to obtain the final output. The calculation formula of the above process is as follows:

$$f_d = f^{1\times1}([f_s; f_{\text{in}}]) \otimes [f_s; f_{\text{in}}] \tag{18}$$

$$f_{\text{out}} = f^{3\times3}(f^{3\times3}(f_d)) + f_{\text{in}}. \tag{19}$$

Fig. 5 shows the heat map of our DAFM. Fig. 5(a) and (b) shows bitemporal remote sensing images, Fig. 5(c) shows labels, and Fig. 5(d) and (e) shows the heat maps without DAFM and with DAFM, respectively. It can be seen that for the changed regions that have not been paid much attention before, after adding the DAFM, the network assigns a larger weight to the pixels in these changed regions, that is, the red region in the heat map, and the network assigns a smaller weight to the pixels in the invariant region, that is, the blue region in the heat map.

E. Global Semantic Enhancement Module

We know that the change detection task is a pixel-level prediction task. In the classification of distinguishing the change region from the invariant region, the global semantic details are often not taken into account, resulting in the omission of some small target features. In some change detection tasks, pyramid pooling module (PPM) is usually used to extract the context information of the feature map by using pooling modules of different sizes, and finally, the context information is spliced with the original input features . Although this effectively alleviates the problem of context semantic loss, it is not sensitive enough to the pixel-level classification of some detailed features. Therefore, in order to better adapt to the change detection task, we propose a GSEM.

Fig. 6 shows the structure of this module. First, for the high-level semantic features with rich category information extracted by the two encoder branches, we add them to obtain the input feature map f_{in} with a size of $C \times H \times W$. Then, it is input into four parallel branches, and the feature maps with rich semantic information are subjected to four global pooling layers of different scales to obtain a sub-region of size $s \times s$, where $s = \{1, 2, 4, 8\}$ defines four pyramid scales to realize the aggregation and refinement of semantic features of different scales. Then, a 1×1 convolution is performed separately, followed by bilinear interpolation upsampling to restore the resolution of the feature and increase the sensitivity of the detail feature. Then, the four features of different scales are spliced to provide a richer feature representation. Finally, a 1×1 convolution and a 3×3 convolution are added to the original features to obtain the output f_{out}, which can refine the rich semantic features after splicing and avoid the loss of original detail information. The calculation formula of the above process is as follows:

$$f_{\text{out}i} = Upsample(f^{1\times1}(Avgpool^{s\times s}(f_{\text{in}}))), \quad i = \{1, 2, 3, 4\} \tag{20}$$

$$f_{\text{out}c} = [f_{\text{out}1}; f_{\text{out}2}; f_{\text{out}3}; f_{\text{out}4}] \tag{21}$$

$$f_{\text{out}} = f^{3\times3}\left(f^{1\times1}\left(f_{\text{out}c}\right)\right) + f_{\text{in}}. \tag{22}$$

where $Avgpool^{s\times s}(\cdot)$ represents the adaptive average pooling, and the output size is $s \times s$. $f_{\text{out}i}$ represents the output of each parallel branch, and $f_{\text{out}c}$ represents the output of four parallel branches after splicing.

F. Loss Function

We use BCEWithLogitsLoss as the loss function of the model. In the change detection task, in most cases, the proportion of the changed region is much smaller than that of the unchanged region, which leads to the class imbalance problem. Therefore, it is necessary to reduce the impact of this imbalance, that is, to set the weight coefficient on the loss function to constrain the network, so that its training is more focused on the changing region. Therefore, during the training process, we conducted in-depth supervision of the network. The loss consists of four parts, namely, the loss of the main branch and three auxiliary losses. The weight coefficients are 1, 0.5, 0.4, and 0.3, respectively. The

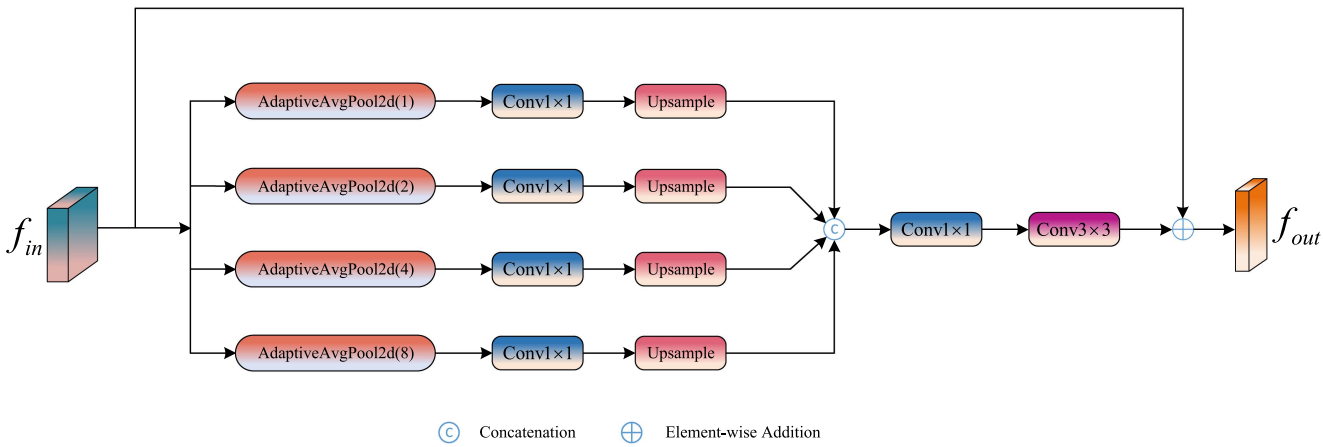

Fig. 6. Structure of the GSEM.

binary cross-entropy (BCE) loss function can be expressed as

$$L_{\text{BCE}}(c, g) = g \cdot \log c + (1 - g) \cdot \log(1 - c) \tag{23}$$

where $\cdot$ is the dot product operation, and c and g are the predicted change graph and the corresponding real label graph, respectively. The total training loss is expressed as

$$\begin{aligned} L_{\text{total}} = {} & \lambda_1 L_{\text{BCE}}(c_1, g) + \lambda_2 L_{\text{BCE}}(c_2, g) \\ & + \lambda_3 L_{\text{BCE}}(c_3, g) + \lambda_4 L_{\text{BCE}}(c_4, g) \end{aligned} \tag{24}$$

where λ_1 represents the weight coefficient of the main loss, and λ_2, λ_3, and λ_4 are the weight coefficients of the three auxiliary losses, respectively. c_1 represents the predicted change graph of the main branch, and c_2, c_3, and c_4 are the predicted change graphs of the three auxiliary branches, respectively.

III. Datasets

Data play an important role in deep learning. The quality of the dataset determines the training results of the model. A high-quality dataset can usually improve the quality of model training and the accuracy of prediction. Since there are few public datasets in the field of change detection, we construct a bitemporal remote sensing image change detection dataset (BICDD) to train the model. In order to fully verify the effectiveness of the DAMFANet proposed in this article, we train and test on the three datasets of BICDD, CDD [47], and LEVIR-CD [48].

A. BICDD

Since there are few public datasets in the field of remote sensing image change detection, we established a BICDD to verify the effectiveness of the algorithm. The BICDD contains 6840 pairs of high-resolution dual-temporal remote sensing images; each image size is 256 × 256 pixels, of which 5472 pairs of images are used as training sets, 684 pairs of images are used as validation sets, and 684 pairs of images are used as test sets, which are divided according to the ratio of 8:1:1. All images are images of different regions in China from 2010 to 2020.

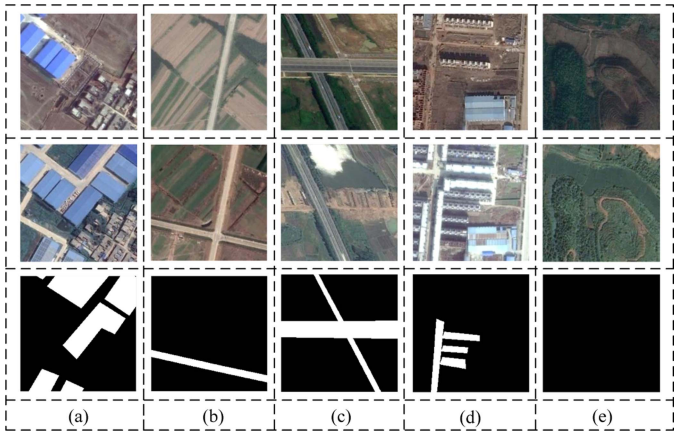

Fig. 7. BICDD diagram. Each column represents a sample. The first and second lines represent the two-phase Google Earth image, and the third line represents the label (black represents the invariant area, and white represents the change area). (a)–(e) represent factories, farmland, roads, buildings, and unchanged areas, respectively.

The types of change areas include factories, farmland, roads, buildings, mining areas, etc. As shown in Fig. 7, we select some different types of samples from the dataset. It can be seen that our dataset contains many common scenarios. In the production of the dataset, we deliberately added some image pairs with large deviation of shooting angle to simulate the actual application scene as much as possible. In addition, we also selected some image pairs taken in different seasons.

B. CDD

The CDD dataset is an open remote sensing image change detection dataset composed of seven pairs of 4725 × 2700 pixels and four pairs of 1900 × 1000 pixels of dual-temporal remote sensing images. We cut 11 pairs of images into 16 000 pairs of images with a size of 256 × 256 pixels, of which 10 000 pairs constitute the training set, 3000 pairs constitute the verification set, and the remaining 3000 pairs constitute the test set. Fig. 8 shows a schematic diagram of some samples in the CDD dataset.

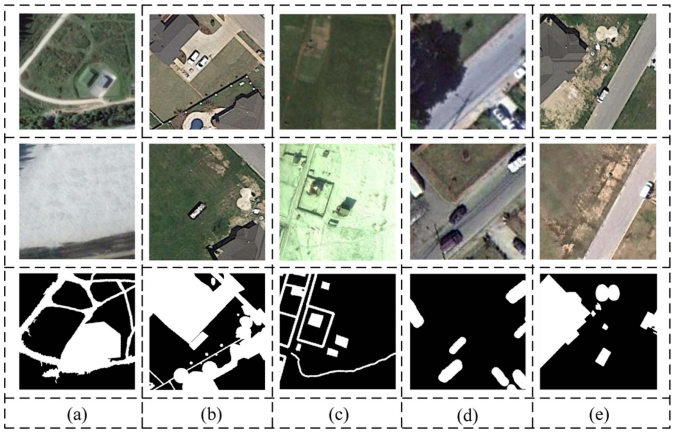

Fig. 8. CDD dataset diagram. Each column represents a sample. The first line and the second line represent the dual-temporal remote sensing image, and the third line represents the label (black represents the invariant area, and white represents the changing area). (a)-(e) represent roads, villas, factories, cars, and buildings, respectively.

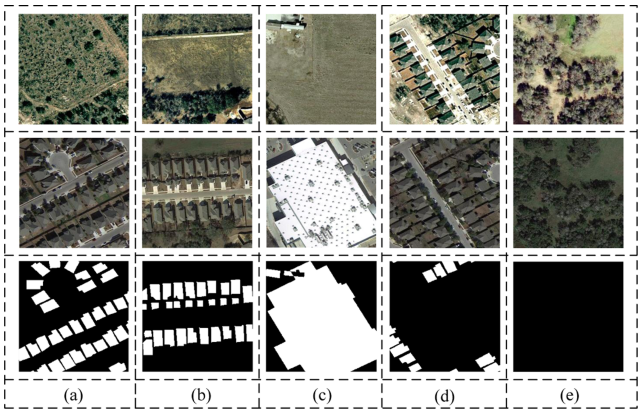

Fig. 9. LEVIR-CD dataset diagram. Each column represents a sample. The first line and the second line represent the dual-temporal remote sensing image, and the third line represents the label (black represents the invariant area, and white represents the change area). (a)-(e) represent villas, apartments, large warehouses, garages, and unchanged areas, respectively.

C. LEVIR-CD

The LEVIR-CD dataset is a large-scale building change detection dataset, including 637 pairs of ultra-high-resolution Google Earth images with a size of 1024 × 1024. All images were taken in 20 different regions of Texas from 2002 to 2018. The dataset focuses on significant changes in buildings, including villas, apartments, large warehouses, garages, etc. In addition, the dataset takes into account seasonal changes and light changes. The LEVIR-CD dataset includes 7120, 1024, and 2048 pairs of images, which are divided at a ratio of 7:1:2. Some samples of the LEVIR-CD dataset are shown in Fig. 9.

IV. Experiments

A. Evaluation Indicators

In Section III, we have randomly divided the dataset into training set, validation set, and test set according to a specific proportion. In this section, we comprehensively evaluate the performance of DAMFANet in change detection tasks. First, we use the training set to train the model, evaluate the performance of the model on the validation set, and adjust the hyperparameters of the model to obtain the best performance. Finally, we use an unprecedented test set to verify the generalization ability of the model and ensure the reliability of the model in practical applications. The efficiency and advancement of our proposed method are verified by ablation experiments and comparative experiments. Our experiments are carried out on BICDD, CDD, and LEVIR-CD datasets, using four evaluation indicators, namely precision (PR), recall (RC), MIoU, and $F1$. The mathematical expression of the evaluation index is as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{25}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{26}$$

$$\text{MIoU} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{27}$$

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{28}$$

In the above formula, TP represents true positive, which refers to the part that is correctly predicted as a change area; FP represents false positive, which refers to the part that is wrongly predicted as a change area; TN indicates true negative, that is, the correct prediction is the part of the unchanged area; and FN represents false negative, referring to the part of the region that is incorrectly predicted to be unchanged.

TABLE I
Comparative Experiments of the DAMFANet Under Different Backbone Networks

Backbone	PR (%)	RC (%)	MIoU (%)	$F1$ (%)
VGG19	78.53	76.79	76.16	77.21
VGG16	81.36	77.68	78.54	79.56
ResNet18	87.26	83.16	85.23	86.02
ResNet50	87.69	84.35	85.80	86.66
ResNet34	**89.70**	**85.85**	**86.79**	**87.73**

Bold numbers represent the best results.

B. Experimental Details

All the experiments in this article are completed on GeForce RTX 3080 based on PyTorch. We use BCEWithLogitsLoss as the loss function of the model and Adam as the optimizer for model training. In view of the effectiveness of dynamically adjusting the learning rate, we adopt the Poly learning rate strategy, and the learning rate of each epoch is $lr \times (1 - \frac{\text{epoch}}{\max\_\text{epoch}})^{\text{power}}$. The batch size is set to 16, the maximum number of epochs ($\max\_\text{epoch}$) is set to 200, the initial learning rate (lr) is set to 0.0001, and the power is set to 0.9. During the training process, we conducted in-depth supervision of the network.

C. Network Backbone Selection

Before starting the experiment, we need to select a backbone network. We used ResNet18, ResNet34, ResNet50, VGG16, and VGG19 for experiments. Table I shows the experimental results.

TABLE II
ABLATION EXPERIMENT OF DAMFANET

Method	PR (%)	RC (%)	MIoU (%)	$F1$ (%)
Backbone	87.38	82.62	84.14	84.93
Backbone+MASAM	88.28	84.30	85.22	86.24
Backbone+MASAM+CBAM	88.36	84.42	85.33	86.46
Backbone+MASAM+DAFM	88.54	84.64	85.65	86.68
Backbone+MASAM+DAFM+TBFFM	88.87	84.86	85.97	87.05
Backbone+MASAM+DAFM+TBFFM+PPM	88.92	84.96	86.12	87.28
Backbone+MASAM+DAFM+TBFFM+GSEM	89.46	85.52	86.43	87.49
Backbone+MASAM+DAFM+TBFFM+GSEM+Aux	**89.70**	**85.85**	**86.79**	**87.73**

Bold numbers represent the best results.

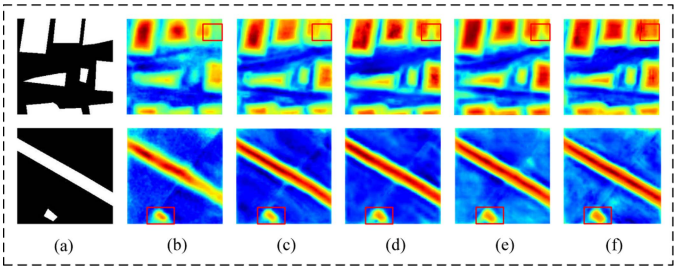

Fig. 10. Heat map under the action of different modules. (a) True label. (b) Heat map of Backbone + MASAM. (c) Heat map of Backbone + MASAM + DAFM. (d) Heat map of Backbone + MASAM + DAFM + TBFFM. (e) Heat map of Backbone + MASAM + DAFM + TBFFM + GSEM. (f) Heat map of Backbone + MASAM + DAFM + TBFFM + GSEM + Aux. As shown in the figure, as the modules are superimposed, the model's attention to the changing regions, especially the edge details and small targets, gradually increases, which verifies the effectiveness of each of our modules in improving the model's feature expression ability.

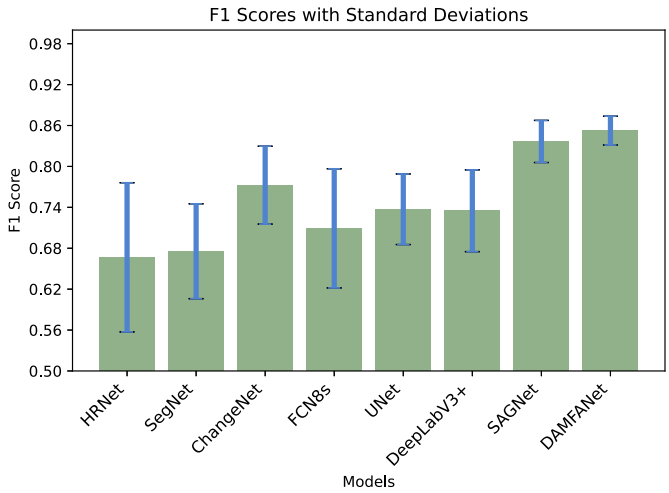

Fig. 11. Comparison on the BICDD. The bar graph represents the $F1$ value, and the standard deviation is displayed at the top.

The best scores are shown in bold. It can be seen from the table that ResNet34 has the best overall performance.

D. Ablation Experiments

In this section, we conduct ablation experiments on the BICDD to evaluate the effectiveness of each module in our network. For relatively complex neural networks, it is necessary to evaluate the network performance by adding or deleting some networks, which helps us to understand our network. The training strategies of all models are the same. In this experiment, we add or delete the proposed modules on the backbone network to verify the effectiveness of each module. Table II shows the results of ablation experiments. We mainly focus on MIoU and $F1$ to verify our module.

1) Ablation Experiment of the MASAM: The MASAM can aggregate semantic information at different scales, realize information interaction and enhancement of multiscale features, and help the network better capture key features in the image. This method can make full use of the features of different scales extracted by the encoder. The results in Table II show that the MASAM increases MIoU score and $F1$ score by 1.08% and 1.31%, respectively.

2) Ablation Experiment of the DAFM: The DAFM can automatically learn and allocate the weights of the changed region and the unchanged region, so that the network can give greater weight to the pixels of the changed region and suppress the interference noise of the unchanged region, so as to better capture the changed information. The results of Table II showed that DAFM increased MIoU and $F1$ by 0.43% and 0.44%, respectively. We also compare our proposed DAFM with the CBAM. The results show that our DAFM is superior to the CBAM, and MIoU and $F1$ are increased by 0.32% and 0.22%, respectively.

3) Ablation Experiment of the TBFFM: Our proposed TBFFM can better integrate global information and difference information and can help the network to better refine edge texture features. The results in Table II show that the TBFFM increases MIoU score and $F1$ score by 0.32% and 0.37%, respectively.

4) Ablation Experiment of the GSEM: Our proposed GSEM can aggregate and refine features at different scales and can improve the expression ability and discrimination of global semantic features, thereby enhancing the network's ability to extract and recognize global semantic information. The results in Table II show that our proposed GSEM increases MIoU and $F1$ by 0.46% and 0.44%, respectively. At the same time, we also compare our proposed GSEM with the PPM module. The experimental results from Table II show that our GSEM is superior to PPM, and MIoU and $F1$ are increased by 0.31% and 0.21%, respectively.

5) Ablation Experiment of Auxiliary Classifiers: In order to alleviate the problem of gradient disappearance, we introduce auxiliary classifiers in the network decoding stage and generate independent auxiliary losses for each auxiliary classifier. This can provide additional supervision signals for the network,

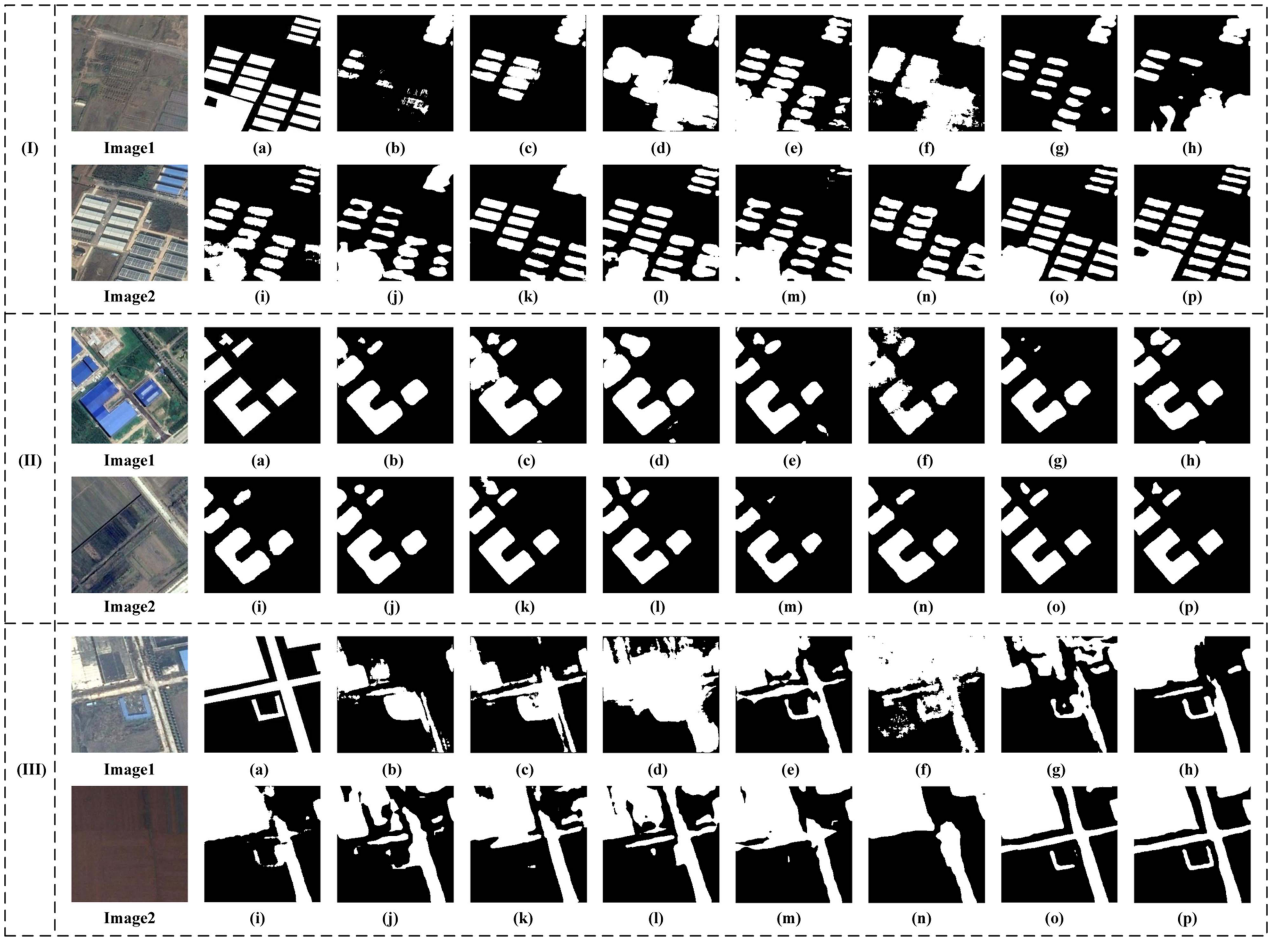

Fig. 12. Comparison of prediction maps of different algorithms on the BICDD. (I)–(III) are the comparative experiments of three pairs of dual-temporal remote sensing images. Image1 and Image2 represent dual-temporal remote sensing images. (a)–(p) represent the prediction graphs of label, FC-Siam-Diff, FC-EF, FC-Siam-Conc, PVT, SegNet, SegFormer, HRNet, FCN-8 s, PSPNet, UNet, DeepLab V3+, BiseNet, ChangeNet, SAGNet, and our network DAMFANet, respectively.

supervise at different levels of the network, make the gradient spread better, and help the network better learn and optimize the model parameters. In addition, we also fuse the output of the three auxiliary classifiers with the output of the main decoder to realize the fusion of multiscale features, which can further improve the performance of change detection. It can be seen from the results of Table II that the introduction of the auxiliary classifier increases MIoU and $F1$ by 0.36% and 0.24%, respectively.

The performance of the backbone network is further optimized by the proposed MASAM, DAFM, TBFFM, GSEM, and the introduction of auxiliary classifiers. With the introduction of these modules, MIoU increased by 1.08%, 0.43%, 0.32%, 0.46%, and 0.36%, respectively, and $F1$ increased by 1.31%, 0.44%, 0.37%, 0.44%, and 0.24%, respectively. It can be seen that under the synergy of these modules, our final model is 2.65% and 2.8% higher than the MIoU and $F1$ of the base network, respectively. Therefore, for the change detection task, the four modules plus the introduction of auxiliary classifiers can assist the basic network to reduce the occurrence of misclassification and missed classification in the prediction stage, so as to effectively detect the change area. In addition, we perform visual feature extraction at each stage of the model. The specific results are shown in Fig. 10.

E. Comparative Experiments

1) Comparative Experiments on the BICDD: In order to test our model more comprehensively, we compare our proposed method with many cutting-edge change detection techniques and other semantic segmentation techniques on the BICDD, including CNN-based methods and Transformer-based methods. In order to ensure the fairness of the comparison test, the training strategies of all models remain unchanged. The experimental results are shown in Table III. It can be seen from the table that the detection result of FC-Siam-Diff is the worst, and its MIoU and $F1$ scores are only 60.81% and 50.48%, respectively. SAGNet is superior to other networks, with MIoU and $F1$ scores of only 85.30% and 86.19%, respectively. Our model DAMFANet is superior to other algorithms in four indicators. MIoU and $F1$ scores are improved by 1.49% and 1.54%, respectively, on the basis of SAGNet. Visually, Fig. 11 shows the $F1$ value and its standard deviation on BICDD between our network and several selected competitors. Obviously, the performance of our

TABLE III
COMPARATIVE EXPERIMENTS ON THE BICDD

Method	PR (%)	RC (%)	MIoU (%)	$F1$ (%)	Params (M)	FLOPs (G)	Time (ms)
FC-Siam-Diff [54]	86.00	35.72	60.81	50.48	1.35	4.26	5.13
FC-EF [54]	79.74	39.51	61.80	52.84	1.35	3.12	7.59
FC-Siam-Conc [54]	79.79	50.06	66.65	61.53	1.55	4.87	5.22
PVT [55]	79.35	70.17	75.24	74.48	61.18	12.86	30.96
SegNet [56]	77.52	72.15	75.33	74.74	29.48	42.6	5.43
SegFormer [57]	79.17	72.46	76.12	75.67	81.54	15	44.36
HRNet [35]	79.77	74.21	77.09	76.89	65.86	23.5	54.75
FCN-8s [58]	83.49	74.37	78.68	78.67	18.65	20.17	8.16
PSPNet [59]	82.58	76.11	79.07	79.22	67.94	64.03	9.15
UNet[31]	83.48	75.57	79.20	79.33	13.4	31.05	3.34
DeepLab V3+ [60]	83.01	78.52	80.31	80.70	54.93	45.77	9.01
BiseNet [61]	85.62	77.24	80.85	81.22	50.25	11.14	5.49
ChangeNet [62]	89.03	75.33	81.31	81.64	23.52	10.68	17.01
SAGNet [63]	87.78	84.67	85.30	86.19	32.23	28.94	25.32
DAMFANet (Ours)	**89.70**	**85.85**	**86.79**	**87.73**	76.81	21.07	40.22

Bold numbers represent the best results.

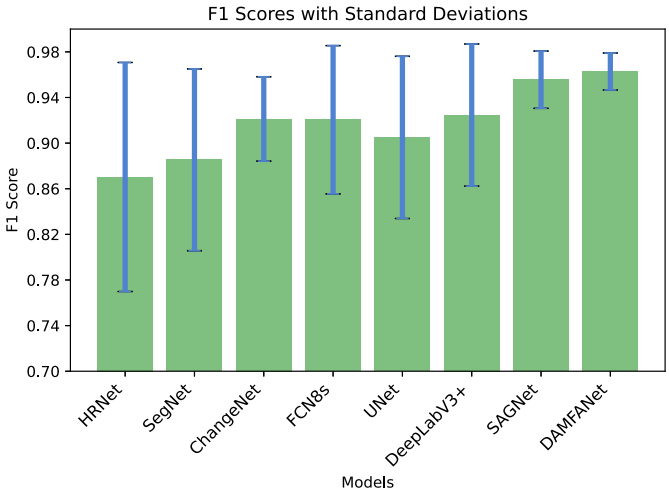

Fig. 13. Comparison on the CDD dataset. The bar graph represents the $F1$ value, and the standard deviation is displayed at the top.

TABLE IV
COMPARATIVE EXPERIMENTS ON CDD

Method	PR (%)	RC (%)	MIoU (%)	$F1$ (%)
FC-EF [54]	84.72	55.17	71.63	66.82
FC-Siam-Conc [54]	91.09	56.73	73.78	69.91
FC-Siam-Diff [54]	88.36	62.59	76.01	73.27
HRNet [35]	94.16	94.33	93.88	94.24
SegNet [56]	95.16	93.79	94.03	94.47
ChangeNet [62]	96.90	92.42	94.19	94.61
BiseNet [61]	95.30	94.29	94.36	94.79
UNet[31]	97.22	93.95	95.17	95.56
SegFormer [57]	97.59	94.30	95.54	95.92
PSPNet [59]	96.54	95.93	95.87	96.24
PVT [55]	96.91	95.79	95.99	96.34
FCN-8s [58]	97.01	96.84	96.61	96.93
DeepLab V3+ [60]	96.93	97.01	96.66	96.97
SAGNet [63]	97.21	97.11	96.86	97.16
DAMFANet (Ours)	**97.62**	**97.67**	**97.39**	**97.65**

Bold numbers represent the best results.

TABLE V
COMPARATIVE EXPERIMENTS ON LEVIR-CD

Method	PR (%)	RC (%)	MIoU (%)	$F1$ (%)
FC-EF [54]	87.32	75.88	83.21	81.20
FC-Siam-Diff [54]	90.16	77.57	84.92	83.39
FC-Siam-Conc [54]	87.61	82.54	86.15	85.00
BiseNet [61]	89.39	85.49	88.13	87.40
PSPNet [59]	89.34	85.70	88.20	87.48
PVT [55]	90.25	85.49	88.48	87.80
SegFormer [57]	90.85	87.23	89.24	88.67
ChangeNet [62]	92.57	85.25	89.30	88.76
FCN-8s [58]	91.58	86.71	89.56	89.07
DeepLab V3+ [60]	90.88	87.35	89.57	89.08
SegNet [56]	91.57	87.69	90.01	89.59
SAGNet [63]	92.29	87.42	90.19	89.79
HRNet [35]	91.62	88.71	90.49	90.14
UNet[31]	92.07	89.67	91.12	90.86
DAMFANet (Ours)	**93.05**	**89.91**	**91.66**	**91.45**

Bold numbers represent the best results.

DAMFANet is superior to others. It should be pointed out that although our algorithm is superior to other algorithms in performance, the parameters of our model are a little large and the time complexity is high. We will solve this problem in the future work. On the premise of ensuring performance, we will reduce the complexity and parameters of the model, improve the training speed, and reduce the training cost.

The comparison of the prediction maps of various algorithms is shown in Fig. 12. By comparing the prediction maps on three different pairs of dual-temporal Google Earth images, we tested the effect of our method more comprehensively. In Fig. 12, (a) represents the label graph and (b)–(p) represent the prediction graph of each algorithm. It can be seen from the diagram that the FC-Siam-Diff change detection method has poor prediction effect, it is difficult to identify the change area, and there is a large area of missed detection. The prediction maps of other deep learning algorithms are also rough and can only approximately estimate the location of the changing region. There is a large degree of false detection and missed detection of edge detail information. The prediction map of our algorithm can not only locate the change area but also be more accurate than other deep learning algorithms in edge details. Our algorithm adds MASAM, DAFM, TBFFM, and GSEM between the Siamese network structures and introduces an auxiliary classifier, which can clearly distinguish the changing region from the unchanged region and use the U-shaped structure to continuously fuse

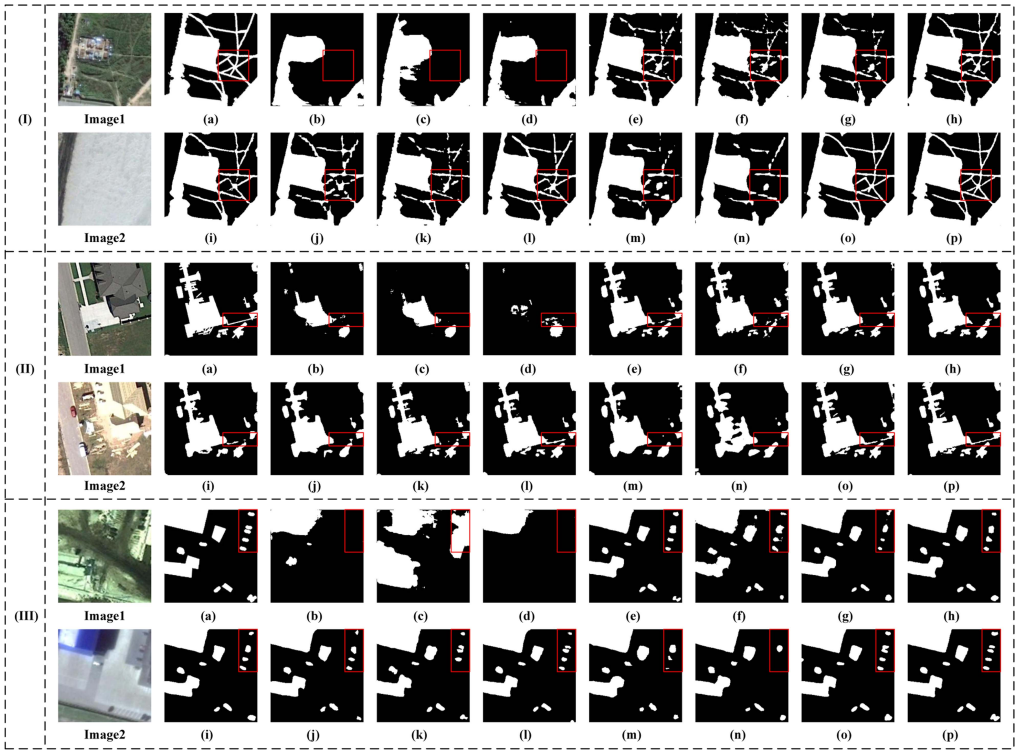

Fig. 14. Comparison of prediction maps of different algorithms on CDD. (I)–(III) are the comparative experiments of three pairs of dual-temporal remote sensing images. Image1 and Image2 represent dual-temporal remote sensing images. (a)–(p) represent the prediction graphs of label, FC-Siam-Diff, FC-EF, FC-Siam-Conc, PVT, SegNet, SegFormer, HRNet, FCN-8 s, PSPNet, UNet, DeepLab V3+, BiseNet, ChangeNet, SAGNet, and our network DAMFANet, respectively.

TABLE VI
COMPARATIVE EXPERIMENTS OF OUR METHOD BEFORE AND AFTER PARAMETER PRUNING

Dataset	Methods	MIoU (%)	$F1$ (%)	Params (M)	FLOPs (G)
BICDD	DAMFANet	86.79	87.73	76.81	21.07
	DAMFANet*	86.55	87.46	48.67	16.38
CDD	DAMFANet	97.39	97.65	76.81	21.07
	DAMFANet*	97.02	97.38	48.67	16.38
LEVIR-CD	DAMFANet	91.66	91.45	76.81	21.07
	DAMFANet*	91.34	91.22	48.67	16.38

"*" denotes the pruned model.

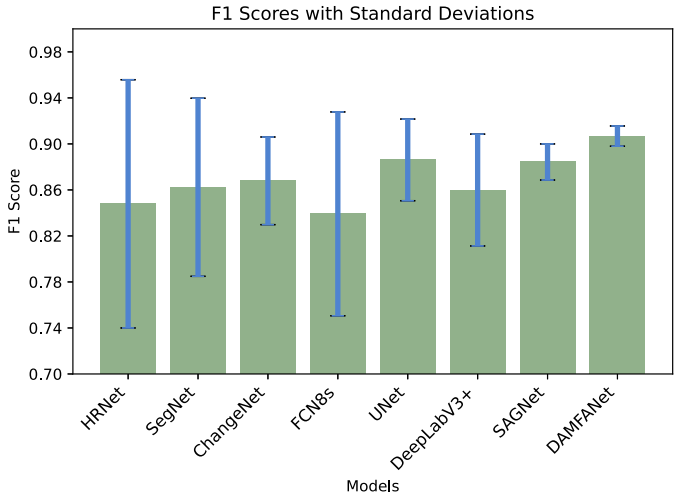

Fig. 15. Comparison on the LEVIR-CD dataset. The bar graph represents the $F1$ value, and the standard deviation is displayed at the top.

multiscale feature information to correct edge details. Therefore, the prediction graph of our algorithm is closer to the real label.

2) Comparative Experiments on CDD: A single dataset is not enough to fully examine the performance of the model. Therefore, we also test our model on the CDD dataset. Like the previous experiments, the comparison of all algorithms is carried out in the same environment. Table IV shows our experimental results on the CDD dataset. As can be seen from the table, compared with other deep learning algorithms, our algorithm has obviously reached the optimal value on all four indicators. The MIoU and $F1$ scores have increased by 0.53% and 0.49%, respectively, on the basis of SAGNet. Visually, Fig. 13 shows the $F1$ value and its standard deviation on CDD between our network and several selected competitors. Obviously, the performance of our DAMFANet is superior to others.

The comparison of the prediction maps of various algorithms is shown in Fig. 14. By comparing three groups of prediction maps selected from 3000 pairs of dual-temporal remote sensing images in the CDD dataset, we tested the effect of our method more comprehensively. In Fig. 14, (a) represents the label graph,

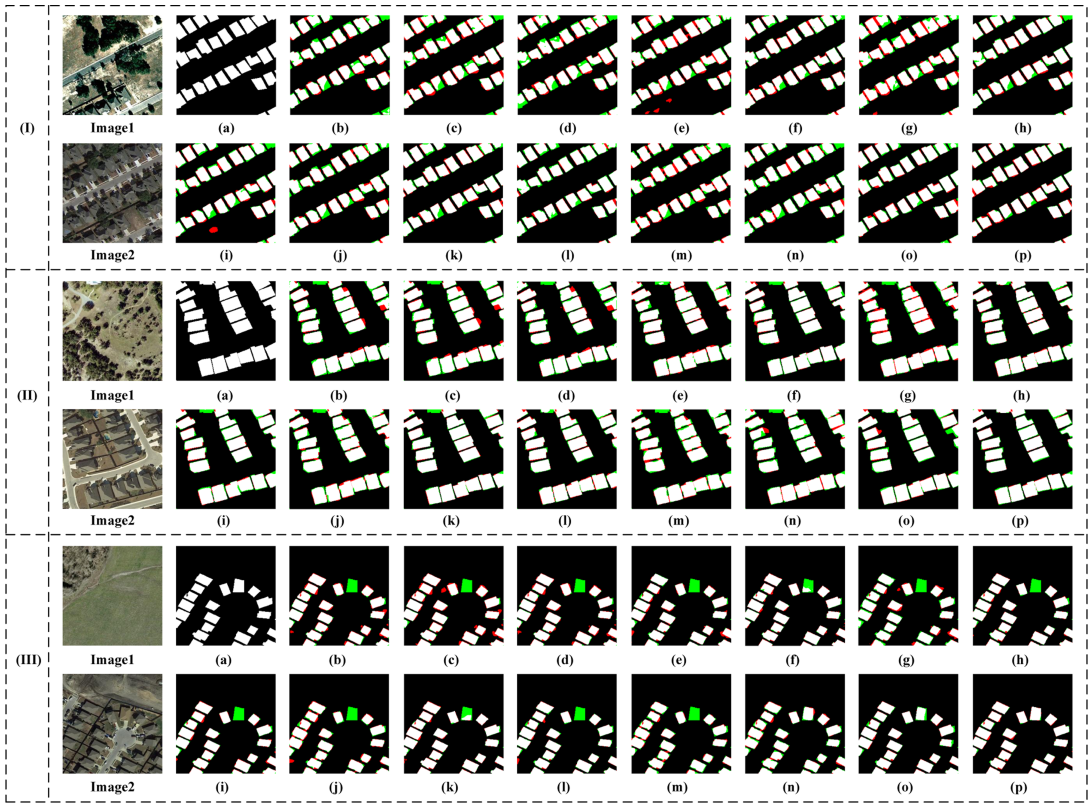

Fig. 16. Comparison of prediction maps of different algorithms on LEVIR-CD. (I)–(III) are the comparative experiments of three pairs of dual-temporal remote sensing images. Image1 and Image2 represent dual-temporal remote sensing images. (a)–(p) represent the prediction graphs of label, FC-Siam-Diff, FC-EF, FC-Siam-Conc, PVT, SegNet, SegFormer, HRNet, FCN-8 s, PSPNet, UNet, DeepLab V3+, BiseNet, ChangeNet, SAGNet, and our network DAMFANet, respectively. Red indicates false detection, and green indicates missed detection.

and (b)–(p) represent the prediction graph of each algorithm. It can be seen from the figure that for some small targets or slender targets, that is, the area marked by the red box in the figure, other algorithms either cannot identify the change area or can only roughly identify the change area, and the identification of small targets and slender targets is extremely vague. Our algorithm performs best in this respect. It can not only accurately identify all the changed regions but also accurately segment the boundaries of the changed regions. In particular, there is basically no adhesion between small target features. Therefore, the prediction graph of our algorithm is closer to the real label.

F. Generalization Experiment on LEVIR-CD

We also conducted comparative experiments on the LEVIR-CD dataset. Considering the limitations of GPU, we cut the original image pair of 1024×1024 pixels into 256×256 pixels. The training set includes 7120 pairs of images, and the test set includes 2048 pairs of images. Like the previous experiments, the comparison of all algorithms is carried out in the same environment. The experimental results are shown in Table V. It can be seen from the table that compared with other deep learning algorithms, our algorithm has obviously reached the optimal value on all four indicators, and the MIoU and $F1$ scores have increased by 0.54% and 0.59%, respectively, on the basis of UNet. This fully proves the effectiveness of our algorithm. Visually, Fig. 15 shows the $F1$ value and its standard deviation on LEVIR-CD between our network and several selected competitors. Obviously, the performance of our DAMFANet is superior to others.

The comparison of the prediction graphs of various algorithms on the LEVIR-CD dataset is shown in Fig. 16. The three groups of prediction images were all from 2048 groups of prediction images in the test set. In Fig. 16, (a) represents the label graph, and (b)–(p) represent the prediction graph of each algorithm. It can be seen from the graph that when detecting the changes of multiple adjacent buildings, the existing deep-learning-based algorithms predict that the boundaries are mostly jagged, and there may be adhesions, resulting in false detection and missed detection (represented by red and green, respectively). Our algorithm can clearly distinguish each changing building and predict the boundary of the changing area more smoothly, thus effectively reducing the occurrence of false detection and missed detection.

G. Parameter Pruning Experiments and Results

As our approach consists of a backbone network ResNet34 and four auxiliary modules, our model exhibits a multilevel multiscale feature representation. While achieving commendable performance, our model incurs significant computational overhead. Therefore, we aim to compress our model to reduce

parameter count. Parameter pruning, as a model compression technique, efficiently reduces the number of parameters, enhancing the inference efficiency while preserving model performance as much as possible. The core idea of parameter pruning is to eliminate weights or neurons in the model that contribute minimally to task performance.

Due to the complex structure of our model, adopting the fine-grained pruning method allows for flexible pruning across different levels, channels, or modules. This approach enables us to reduce the parameter count while retaining crucial information for the change detection task. Considering that the TBFFM aims to fuse global semantic information and differential semantic information, these two types of semantic information share some similarities. The intermediate branch in the TBFFM concatenates these two types of semantic information in the channel dimension, introducing some redundant channels. Therefore, we performed channel pruning on the TBFFM by removing channels that contribute minimally to the model's performance, reducing both parameter count and computational burden. We also applied channel pruning to the GSEM. In addition, we evaluated the importance of certain layers in our model using metrics such as gradients, activation values, and weight magnitudes. Layers with small gradients, activation values, and weight magnitudes were considered less contributive and were pruned. However, we still need to evaluate the performance of the pruned model through experiments. After multiple experiments and comprehensive evaluations, we determined the redundant channels removed through channel pruning and the redundant layers removed through layer pruning. We evaluated the performance of the pruned model on three datasets, comparing it with the original model in terms of performance, parameter count, and FLOPs. As shown in Table VI, after pruning, our model's parameter count and FLOPs were reduced by 28.14 M and 4.69 G, respectively. Through fine-tuning, the performance of the pruned model experienced a slight decrease but remained relatively unaffected.

V. Conclusion

This article proposes a DAMFANet. Aiming at the problems of multiscale feature fusion and attention allocation strategy in previous deep learning methods, this algorithm makes full use of the rich feature information in remote sensing images through cross-fusion of different scales and uses unique dual attention to guide fusion in space and channel information at the same time. The target area, edge details, and small target features in the process of dual-time remote sensing image change are restored as much as possible, and the occurrence of missed detection and false detection is also effectively avoided. Specifically, four modules are designed to improve the accuracy and robustness of the algorithm, and an auxiliary classifier is introduced to help network training. We propose a MASAM to integrate semantic information at different scales and strengthen the information interaction of multiscale semantic features, so as to obtain more discriminative feature representation. In addition, we also propose a DAFM, a TBFFM, and a GSEM. While modeling spatial information and channel information, the DAFM also weightedly fuses the two attention-guided features, so that the network can better pay attention to and guide the location information and channel information of dual-temporal features and reduce the interference of irrelevant noise. The TBFFM combines the global semantic information and the difference semantic information of the dual-time remote sensing image, while retaining the original information to avoid information loss. The GSEM obtains semantic information of different scales and integrates them to make the model better understand the global context semantic information. In addition, we also introduce an auxiliary classifier, which can not only provide additional deep supervision signals for the network but also realize multiscale feature fusion and further improve the performance of the model. The experimental results show that the DAMFANet is superior to other deep learning algorithms on BICDD, CDD, and LEVIR-CD datasets. However, the algorithm still has room for improvement. Under the premise of ensuring the detection accuracy, the complexity and parameter quantity of the model are reduced, the training speed is improved, and the training cost is reduced.

References

[1] J. Chen, M. Xia, D. Wang, and H. Lin, "Double branch parallel network for segmentation of buildings and waters in remote sensing images," *Remote Sens.*, vol. 15, no. 6, 2023, Art. no. 1536.

[2] K. Sundarakumar, M. Harika, S. A. Begum, S. Yamini, and K. Balakrishna, "Land use and land cover change detection and urban sprawl analysis of Vijayawada city using multitemporal Landsat data," *Int. J. Eng. Sci. Technol.*, vol. 4, no. 1, pp. 170–178, 2012.

[3] L. Weng, K. Pang, M. Xia, H. Lin, M. Qian, and C. Zhu, "SGFormer: A local and global features coupling network for semantic segmentation of land cover," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 6812–6824, 2023.

[4] A. H. Chughtai, H. Abbasi, and I. R. Karas, "A review on change detection method and accuracy assessment for land use land cover," *Remote Sens. Appl.: Soc. Environ.*, vol. 22, 2021, Art. no. 100482.

[5] K. Hu, C. Weng, C. Shen, T. Wang, L. Weng, and M. Xia, "A multistage underwater image aesthetic enhancement algorithm based on a generative adversarial network," *Eng. Appl. Artif. Intell.*, vol. 123, 2023, Art. no. 106196.

[6] M. Hemati, M. Hasanlou, M. Mahdianpari, and F. Mohammadimanesh, "A systematic review of landsat data for change detection applications: 50 years of monitoring the earth," *Remote Sens.*, vol. 13, no. 15, 2021, Art. no. 2869.

[7] A. Mohammadi, S. Karimzadeh, K. Valizadeh Kamran, and M. Matsuoka, "Extraction of land information, future landscape changes and seismic hazard assessment: A case study of Tabriz, Iran," *Sensors*, vol. 20, no. 24, 2020, Art. no. 7010.

[8] M. Wieland and S. Martinis, "A modular processing chain for automated flood monitoring from multi-spectral satellite data," *Remote Sens.*, vol. 11, no. 19, 2019, Art. no. 2330.

[9] H. Ji, M. Xia, D. Zhang, and H. Lin, "Multi-supervised feature fusion attention network for clouds and shadows detection," *ISPRS Int. J. Geo-Inf.*, vol. 12, no. 6, 2023, Art. no. 247.

[10] Z. Ma, M. Xia, H. Lin, M. Qian, and Y. Zhang, "FENet: Feature enhancement network for land cover classification," *Int. J. Remote Sens.*, vol. 44, no. 5, pp. 1702–1725, 2023.

[11] C. Lu, M. Xia, M. Qian, and B. Chen, "Dual-branch network for cloud and cloud shadow segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5410012.

[12] D. Wang, L. Weng, M. Xia, and H. Lin, "MBCNet: Multi-branch collaborative change-detection network based on siamese structure," *Remote Sens.*, vol. 15, no. 9, 2023, Art. no. 2237.

[13] D. Stow, "Reducing the effects of misregistration on pixel-level change detection," *Int. J. Remote Sens.*, vol. 20, no. 12, pp. 2477–2483, 1999.

[14] H. Zhuang, M. Hao, K. Deng, K. Zhang, X. Wang, and G. Yao, "Change detection in SAR images via ratio-based Gaussian kernel and nonlocal theory," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5210215.

[15] L. Su, M. Gong, and B. Sun, "Change detection in synthetic aperture radar images based on non-local means with ratio similarity measurement," *Int. J. Remote Sens.*, vol. 35, no. 22, pp. 7673–7690, 2014.

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

[17] L. Jia, M. Li, P. Zhang, Y. Wu, L. An, and W. Song, "Remote-sensing image change detection with fusion of multiple wavelet kernels," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 9, no. 8, pp. 3405–3418, Aug. 2016.

[18] M. Botsch and J. A. Nossek, "Feature selection for change detection in multivariate time-series," in *Proc. IEEE Symp. Comput. Intell. Data Mining*, 2007, pp. 590–597.

[19] C. Ma, H. Yin, L. Weng, M. Xia, and H. Lin, "DAFNet: A novel change-detection model for high-resolution remote-sensing imagery based on feature difference and attention mechanism," *Remote Sens.*, vol. 15, no. 15, 2023, Art. no. 3896.

[20] K. Chen, X. Dai, M. Xia, L. Weng, K. Hu, and H. Lin, "MSFANet: Multi-scale strip feature attention network for cloud and cloud shadow segmentation," *Remote Sens.*, vol. 15, 2023, Art. no. 4853.

[21] Z. Wang, M. Xia, L. Weng, K. Hu, and H. Lin, "Dual encoder-decoder network for land cover segmentation of remote sensing image," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 2372–2385, 2024.

[22] L. Ding, M. Xia, H. Lin, and K. Hu, "Multi-level attention interactive network for cloud and snow detection segmentation," *Remote Sens.*, vol. 16, no. 1, 2023, Art. no. 112.

[23] W. Diao, X. Sun, X. Zheng, F. Dou, H. Wang, and K. Fu, "Efficient saliency-based object detection in remote sensing images using deep belief networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 2, pp. 137–141, Feb. 2016.

[24] P. Zhou, J. Han, G. Cheng, and B. Zhang, "Learning compact and discriminative stacked autoencoder for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4823–4833, Jul. 2019.

[25] Y. Qu, M. Xia, and Y. Zhang, "Strip pooling channel spatial attention network for the segmentation of cloud and cloud shadow," *Comput. Geosci.*, vol. 157, 2021, Art. no. 104940.

[26] Y. Dong, Q. Liu, B. Du, and L. Zhang, "Weighted feature fusion of convolutional neural network and graph attention network for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 31, pp. 1559–1572, 2022.

[27] Y. Feng, J. Zheng, M. Qin, C. Bai, and J. Zhang, "3D octave and 2D vanilla mixed convolutional neural network for hyperspectral image classification with limited samples," *Remote Sens.*, vol. 13, no. 21, 2021, Art. no. 4407.

[28] B. Chen, M. Xia, M. Qian, and J. Huang, "MANet: A multi-level aggregation network for semantic segmentation of high-resolution remote sensing images," *Int. J. Remote Sens.*, vol. 43, nos. 15/16, pp. 5874–5894, 2022.

[29] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10012–10022.

[30] C. Zhang, L. Weng, L. Ding, M. Xia, and H. Lin, "CRSNet: Cloud and cloud shadow refinement segmentation networks for remote sensing imagery," *Remote Sens.*, vol. 15, no. 6, 2023, Art. no. 1664.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput.-Assist. Interv. 18th Int. Conf.*, 2015, pp. 234–241.

[32] Z. Lv, P. Zhong, W. Wang, Z. You, and N. Falco, "Multi-scale attention network guided with change gradient image for land cover change detection using remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 2501805.

[33] M. Zhang, Z. Liu, J. Feng, L. Liu, and L. Jiao, "Remote sensing image change detection based on deep multi-scale multi-attention siamese transformer network," *Remote Sens.*, vol. 15, no. 3, 2023, Art. no. 842.

[34] J. Fu et al., "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3146–3154.

[35] J. Wang et al., "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.

[36] C. Ma, L. Weng, M. Xia, H. Lin, M. Qian, and Y. Zhang, "Dual-branch network for change detection of remote sensing image," *Eng. Appl. Artif. Intell.*, vol. 123, 2023, Art. no. 106324.

[37] X. Xu, Z. Yang, and J. Li, "AMCA: Attention-guided multi-scale context aggregation network for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5908619.

[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[39] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[40] L. Song, M. Xia, J. Jin, M. Qian, and Y. Zhang, "SUACDNet: Attentional change detection network based on siamese U-shaped structure," *Int. J. Appl. Earth Observ. Geoinformation*, vol. 105, 2021, Art. no. 102597.

[41] E. Choi and J. Kim, "Robust change detection using channel-wise co-attention-based siamese network with contrastive loss function," *IEEE Access*, vol. 10, pp. 45365–45374, 2022.

[42] 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.

[43] H. Zhu, W. Ma, L. Li, L. Jiao, S. Yang, and B. Hou, "A dual-branch attention fusion deep network for multiresolution remote-sensing image classification," *Inf. Fusion*, vol. 58, pp. 116–131, 2020.

[44] W. Ma, O. Karakuş, and P. L. Rosin, "AMM-FUSENet: Attention-based multi-modal image fusion network for land cover mapping," *Remote Sens.*, vol. 14, no. 18, 2022, Art. no. 4458.

[45] Q. Yan et al., "Cloud detection of remote sensing image based on multi-scale data and dual-channel attention mechanism," *Remote Sens.*, vol. 14, no. 15, 2022, Art. no. 3710.

[46] W. Wang, X. Tan, P. Zhang, and X. Wang, "A CBAM based multiscale transformer fusion approach for remote sensing image change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 6817–6825, 2022.

[47] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2019.

[48] 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.

[49] K. Chen, M. Xia, H. Lin, and M. Qian, "Multi-scale attention feature aggregation network for cloud and cloud shadow segmentation," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5612216.

[50] X. Dai, M. Xia, L. Weng, K. Hu, H. Lin, and M. Qian, "Multi-scale location attention network for building and water segmentation of remote sensing image," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5609519.

[51] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 539–546.

[52] X. Dai, K. Chen, M. Xia, L. Weng, and H. Lin, "LPMSNet: Location pooling multi-scale network for cloud and cloud shadow segmentation," *Remote Sens.*, vol. 15, 2023, Art. no. 4005.

[53] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[54] 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.

[55] W. Wang et al., "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 568–578.

[56] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[57] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2021, pp. 12077–12090.

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

[59] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2881–2890.

[60] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 801–818.

[61] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiseNet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 325–341.

[62] A. Varghese, J. Gubbi, A. Ramaswamy, and P. Balamuralidhar, "ChangeNet: A deep learning architecture for visual change detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, 2018, pp. 129–145.
[63] H. Yin et al., "Attention-guided siamese networks for change detection in high resolution remote sensing images," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 117, 2023, Art. no. 103206.

Hongjin Ren received the B.S. degree in automation from Chuzhou University, Chuzhou, China, in 2022.

He was a graduate student majoring in electronic information with the Nanjing University of Information Science and Technology, Nanjing, China. His research interests include deep learning and its applications.

Min Xia (Member, IEEE) received the Ph.D. degree in cybernetics control engineering from Donghua University, Shanghai, China, in 2009.

He is currently a Professor with the Nanjing University of Information Science and Technology, Nanjing, China, where he is also the Deputy Director of the Jiangsu Key Laboratory of Big Data Analysis Technology. His research interests include machine learning theory and its application.

Liguo Weng received the Ph.D. degree in electrical engineering from North Carolina A&T State University, Greensboro, NC, USA, in 2010.

He is currently an Associate Professor with the College of Automation, Nanjing University of Information Science and Technology, Nanjing, China. His main research interests include deep learning and its application in remote sensing image analysis.

Kai Hu received the bachelor's degree from the China University of Metrology, Hangzhou, China, in 2003, the master's degree from the Nanjing University of Information Science and Technology, Nanjing, China, in 2008, and the Ph.D. degree in instrument science and engineering from Southeast University, Nanjing, China, in 2015.

He is currently an Associate Professor with the Nanjing University of Information Science and Technology. His main research interests include deep learning and its applications in remote sensing images.

Haifeng Lin received the Ph.D. degree in forest engineering from Nanjing Forestry University, Nanjing, China, in 2019.

He is a Professor with the College of Information Science and Technology, Nanjing Forestry University, Nanjing, China. His main research interests include networking, wireless communication, deep learning, pattern recognition, and Internet of Things.