





MLCNet: Multitask Level-Specific Constraint Network for Building Change Detection

Taoyuan Liu , Jiepan Li , *Graduate Student Member, IEEE*, Weinan Cao , *Student Member, IEEE*, Minghao Tang, and Guangyi Yang 

Abstract—Change detection is an essential fundamental task in remote sensing image analysis. Owing to powerful deep abstract feature extraction ability, many deep learning-based change detection methods have emerged recently. Although previous works have recognized the characteristics and advantages of multilevel deep features and attempted to integrate them, the utilization process lacks a clear emphasis on the advantages aspect of different-level features. One-size-fits-all fusion treats all levels equally, neglecting the spatial advantage and semantic advantage of low and high-level features, respectively. This leads to deficiencies in the integrity and edge accuracy of the final change predictions. To address these issues, we propose a multitask level-specific constraint network, named MLCNet, which addresses the issues by optimizing the advantages of features at different levels. MLCNet comprises a Siamese encoder, fusion modules tailored for features at different levels, and a decoding mechanism that effectively combines semantic and spatial information of features at adjacent levels. In addition, by reconstructing the original ground truths (GTs) into the semantic, binary, and edge GTs, multitask learning constraints are established during network training, compelling the network to enhance targeted emphasis on the characteristics of features at different levels. Experimental results on the three building change detection datasets validate the practicality of MLCNet.

Index Terms—Change detection (CD), deep supervision, level-specific constraint, multilevel features, multitask learning.

I. INTRODUCTION

CHANGE detection in remote sensing is defined as identifying alterations in land cover types across distinct time intervals by analyzing two or more remote sensing images captured over the same geographical area at different times [1], [2], [3]. Very-high-resolution (VHR) image change detection is a fundamental and critically important remote sensing interpretation technique that plays a pivotal role in the remote sensing

community. Its applications span across various domains including land cover analysis [4], [5], urban expansion analysis [6], [7], environmental change detection [8], [9], natural disaster monitoring [10], and more.

Over the past few decades, numerous methods have been developed for remote sensing image change detection [11], [12]. Traditional methods can be classified into two categories: algebraic methods, such as change vector analysis [13], identify changed pixels by performing pixelwise algebraic calculations in the original image space, and transformation-based methods, such as principal component analysis [14], [15], transform original images into appropriate feature spaces and then determine changed pixels through algebraic calculations. Although these early approaches met specific application needs in low-resolution images, they did not perform well on VHR images. As the resolution increases, the variations of illumination, contrast, and noise in bitemporal images also increase. These variations can impact the feature similarity of bitemporal remote sensing images, making change detection in VHR remote sensing images a challenging research area. Traditional change detection methods that rely on manually designed features are becoming less effective in meeting accuracy requirements [16]. In recent years, deep learning technology has gained attention for its powerful ability to extract representative abstract deep features, leading many researchers to utilize this technology in VHR image change detection. To fully leverage different level deep features' information, inspired by UNet [17] and its consequence works, some researchers use skip-connections to complement less localized information with spatial details, such as three architectures (FC-EF, FC-Siam-conc, FC-Siam-diff) proposed in [18], or dense connection, such as SNUNet [19]. With the rise of attention mechanisms in recent years, many scholars have applied these mechanisms to integrate multilevel feature information, allowing the network to selectively focus on regions relevant to changes, with notable examples including MSPSNet [20], STANet [21], and A2Net [22]. Later, the introduction of transformers in the visual domain [23] also led to their extensive use in VHR image change detection. Several recent works, such as BIT [24], ChangeFormer [25], EATDer [26], and MDAFormer [27], demonstrate the unique advantage of transformer in long-range information extraction and preservation by applying it to the interaction of same-level bitemporal features or hierarchical feature extraction and decoding.

Compared to traditional methods, change detection methods based on deep learning have made significant progress. However,

Manuscript received 31 March 2024; revised 20 May 2024; accepted 10 June 2024. Date of publication 18 June 2024; date of current version 1 July 2024. This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2042022rc0027, in part by the Key R&D Program of Zhejiang Province under Grant 2022C03107, and in part by the Open Funding of Key Laboratory of Marine Environmental Survey Technology and Application, Ministry of Natural Resources under Grant MESTA-2021-B008. (Corresponding author: Guangyi Yang.)

The authors are with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, Hubei 430070, China, also with Electronic Engineering, Wuhan University, Wuhan, Hubei 430072, China, and also with Zhejiang Academy of Emergency Management Science, Hangzhou, Zhejiang 310007, China (e-mail: taoyuanliu1022@gmail.com; jiepanli@whu.edu.cn; jackzhang@whu.edu.cn; tongcody@outlook.com; ygy@whu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2024.3415171

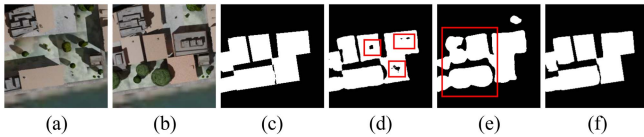


Fig. 1. Shortcomings of CNN and transformer methods. (a) T1 images. (b) T2 images. (c) Ground truth (GT). (d) Prediction result of the FC-EF model. (e) Prediction result of the ChangeFormer model. (f) Prediction result of our MLCNet model. The red boxes indicate the main error regions of each network.

CNN methods can be limited when recognizing long-range dependencies [28]. When two change regions in an image are far apart, it becomes difficult for CNNs to establish a relationship between them through long-range feature interaction. Consequently, the features extracted from these two change clusters may differ significantly. This often results in a lack of intraclass consistency and gaps in the detected change clusters, even if each cluster belongs to the same category, as shown in Fig. 1(d). Although transformer-based methods are skilled at capturing global dependencies, they may sometimes overlook local details critical for precise change detection. This oversight can lead to blurry misalignment along the edges of detected changes, as shown in Fig. 1(e). Despite attempts in previous studies to integrate these two approaches, such as [29], [30], [31], these methods also failed to address the abovementioned issue. Insufficient purposeful learning of the distinct characteristics of different level features is likely the cause. Treating all levels of features uniformly within a single architecture, be it CNNs, transformers, or a hybrid thereof, overlooks the nuanced emphasis required to leverage the distinct advantages offered by features at different levels. Studies by [32], [33], [34] have highlighted the importance of simultaneously leveraging the advantages of features at different levels. However, they still employ the same operation to cluster features at different levels, resulting in an even utilization of spatial details and semantic information for each level feature, lacking focus on their distinct advantages. Consequently, these operations are hard to address the issues we mentioned above. Some works attempt to enhance the accuracy of change edges and intraclass consistency by adding extra edge change labels or utilizing deep supervision. However, they do not take the correlation between these constraints and the characteristics of different level features into account. Adding supervision with the extra edge change labels to the final output [35], [36] or higher level features [37] may come too late in the training process, as the spatial information in higher level features is already blurred. Therefore, adding these additional constraints requires considering the advantages of different-level features and implementing them purposefully. Consequently, the current change detection networks face the following challenges:

- 1) As network features transition from shallow to deep layers, the feature representation gradually shifts from spatial details to semantics [38], [39]. Low-level features easily suffer from imaging differences, leading to unclear semantic information representation about changes, while high-level features tend to represent low-resolution spatial detail information. However, current networks often employ the same structure when utilizing features at different levels, lacking targeted emphasis on the advantages

of each level feature. Thus, low-resolution information of high-level features may be incorrectly emphasized to distinguish the edge of landcover, and ambiguous semantic information of low-level features may be wrongly used to classify the pixels' category.

- 2) Most current methods lack specific constraint measures tailored to the advantages of different-level features, resulting in a lack of emphasis when utilizing the advantages aspect of different-level features in the network.

To address the issues mentioned above, we propose a multi-task level-specific constraint network, named MLCNet. The proposed MLCNet can apply tailored constraints and optimizations to features at different levels, allowing the network to leverage each level's strengths without being disrupted by weaknesses, ensuring accurate and robust change detection results. In detail, the MLCNet adopts a Siamese encoder to extract the bitemporal multilevel features. To leverage spatial contextual details of low-level features and achieve clear change edges, we introduce an edge detail preservation module (EDPM) to fuse low-level features. Regarding the fusion of high-level features, we introduce a cross-temporal semantic attention module (CSAM) with a novel cross-attention mechanism to enhance the change in the semantic representation aspect. During the decoding stage, the goal is to comprehensively integrate both lower and higher level features while performing upsampling decoding. The semantic information from higher layers is utilized for accurate classification, while the precise spatial information from low-level layers is leveraged for accurate edge delineation. A specialized multilevel interactive fusion decoder (MLIFD) has been devised to achieve this goal. The main contributions of our proposed method are presented as the following:

- 1) We classify the hierarchical features generated by the Siamese backbone into two categories: low-level features that contain rich spatial information to represent land cover edge, and high-level features that contain rich semantic information to represent land cover categories. In addition, we assign unique tasks to features at different levels to enable multitask learning.
- 2) We develop two separate fusion modules tailored to distinct level features according to their unique representations: the EDPM for low-level features and the CSAM for high-level features.
- 3) We deconstruct the initial GTs into "Semantic," "Binary," and "Edge" GTs, with corresponding losses computed during network training. Our method demonstrates superior performance compared to current state-of-the-art (SOTA) methods across diverse change detection datasets.

We organize the remaining part of this article as follows. In this article, we present an overview of the relevant literature in Section II. Section III elaborates on the proposed MLCNet architecture and its modules. The experimental results will be discussed in Section IV. Finally, Section V concludes the article.

II. RELATED WORK

A. CNN-Based CD Models

Lately, convolutional neural networks (CNNs) have gained widespread adoption owing to their adeptness in extracting

features efficiently for understanding intricate imaging scenarios. Change detection tasks demand pixel-level prediction results that align well with the characteristics of CNN. Therefore, early deep learning change detection networks mostly used CNN models. These CNN-based models can be categorized as either metric based or classification based.

Metric-based models transform original pixel values from bitemporal images into a high-dimensional embedding distance space to measure change probability for each pixel. In this space, distances between unchanged pixel pairs decrease, while distances between changed pixel pairs increase. Han et al. [40] proposed a method that combines a Siamese convolutional network with threshold segmentation for change detection, using contrastive loss for model training. Chen et al. [21] introduced STANet, employing self-attention modules to establish spatiotemporal dependencies between same-level bitemporal features, outputting feature maps for computing pixel pair distance. Liu et al. [41] proposed a deep convolutional coupling network that uses coupling algorithms to transform bitemporal features, resulting in a more consistent feature space and reducing the differences in unchanged land features while highlighting differences in changed land features.

Classification-based deep learning models extract change features from bitemporal images and classify each pixel as change or not to obtain the predicted change map. Typically, classification-based deep learning algorithms use functions like softmax or sigmoid to transform the final output of the model into change probabilities. Daudt et al. [18] innovatively applied the UNet architecture [17] to change detection tasks. They constructed three types of UNet networks with different skip connection configurations, effectively utilizing multilevel features. Fang et al. [19] proposed SNUNet-CD, which employs a densely connected Siamese UNet++ architecture [42], effectively preserving multilevel feature information. Through subsequent attention mechanism fusion operations, it reallocates weights to the multilevel information for comprehensive utilization. Lei et al. [43] proposed an innovative approach that enhances change detection results by adding boundary constraints. Their method extracted predicted change boundaries from the network outputs and compared them with edge change GT to calculate the loss for network training.

Despite the significant advancements enabled by CNN-based change detection methods, they still exhibit limitations in recognizing long-range dependencies, resulting in significant differences in extracted features and a lack of intraclass consistency [44]. Some works try to solve this problem by adding deep supervision to guide the feature learning, such as IFNet [45], which applies binary change labels to multilevel features in the decoder stage for deep supervision. However, as previously analyzed, constraints like deep supervision of multilevel features may necessitate more targeted measures to emphasize their distinct advantages, applying the same labels across all levels potentially suboptimal. Some other research enhances the internal consistency of change categories with superpixel algorithms or graph-CNN, such as ESCNet [46], WNet [36], and PGCFNet [47]. These methods' accuracy may be influenced by the results of superpixel or graph clustering, leading to the generation of smooth and blurred edges. Some other works

utilized specific attention mechanisms to enhance features in specific directions. For example, HFA-Net [48] employs spatial and high-frequency attention to highlight targeted landcover's features, while HDANet [49] uses differential attention to optimize detection results. The uniform treatment of multilevel features indeed enhances the feature representation in the aspects they designed but fails to differentiate attention directions for different levels of features, thus being unable to address the aforementioned issue.

B. Transformer-Based CD Models

The remarkable ability of transformers to model long-range dependencies led to their predominant use in natural language processing tasks and computer vision domains. Recently, transformers have also been increasingly utilized for model development in remote sensing change detection. These approaches can be broadly categorized into two sets: exclusively employ the transformer architecture or integrate transformers with CNNs. For the first class, Bandara et al. [25] proposed ChangeFormer, which introduced a hierarchical encoder based on the transformer architecture. This network employed differential modules to obtain change maps at each scale. Subsequently, it utilized multilayer perceptrons (MLPs) to decode these change maps, effectively leveraging multilevel information of global features for change map prediction. Zhang et al. [50] proposed SwinSUNet, a transformer-based U-shape structure model. This network utilizes the Swin Transformer (SwinT) module instead of the traditional transformer module. The traditional transformer module conducts multihead self-attention directly on input features. In contrast, the SwinT module uses shifted windows to divide features into different regions and perform multihead self-attention on each region. This reduces the computational burden and enhances the local attention capability of the transformer module.

About the second class, Chen et al. [24] introduced a model called bitemporal image transformer (BIT) for change detection tasks. The model used two weight-shared CNNs to transform the images into feature maps at a high level, which were then refined with two transformer encoders. This helped capture contextual relations across spatial and temporal domains, enhancing understanding of global context. Li et al. [31] proposed TransUNet, which combined the strengths of UNet and transformers to construct an encoder-decoder model for remote sensing change detection (RSCD). UNet effectively captured intricate local details, while the transformer modeled global contextual relations [51]. The model predicted positive change maps with enhanced performance by integrating these components.

However, transformer-based methods are also susceptible to the issue of insufficient spatial details, which may constrain the model's capacity to detect subtle changes and accurate edges. The works integrating transformers with CNNs have partly addressed this issue by leveraging CNNs' strong capability to capture local spatial details. However, methods utilize both transformers and CNNs for all level features, like SUT [52], which perform a parallel encoder separately consisting of transformers and CNNs, simultaneously enhancing the feature representation of spatial details and semantics but lacking emphasis. The

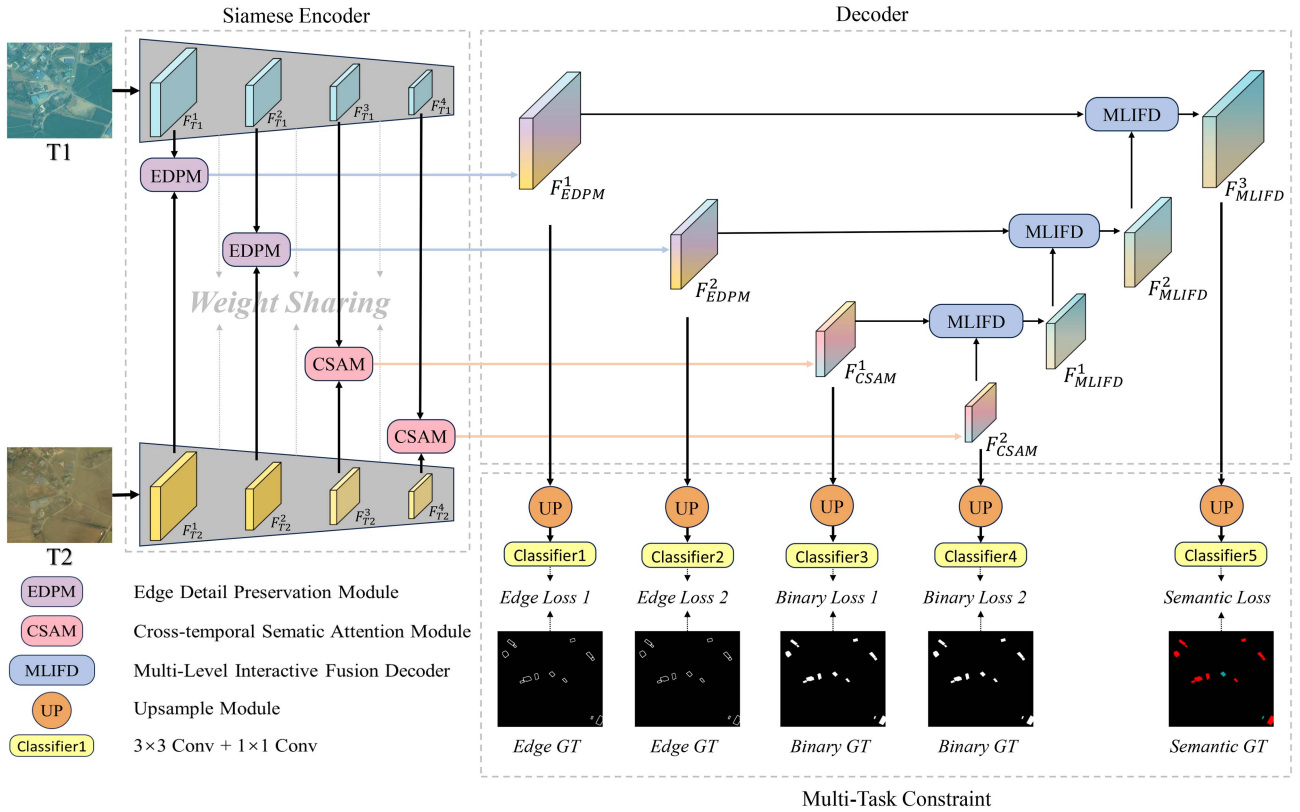


Fig. 2. Overall architecture of MLCNet.

blurred semantic information of low-level features and the lower spatial resolution of high-level features may negatively affect the final detection results. Conversely, methods like AMTNet [53] apply CNNs and transformers for feature extraction and feature refinement, respectively. This approach simultaneously utilizes three different level features to output three change detection prediction maps, these maps are then used for supervision to achieve multilevel feature utilization. However, this approach can still be somewhat simplistic and blunt for multilevel feature utilization, lacking emphasis on the advantages aspect of different level features. Besides, in the fusion of the same level or multilevel features, many existing methods still rely on simple channel concatenation [53] or direct fusion with MLPs [25], [54], these may still be insufficient in fully capitalizing on the advantages of different-level features.

III. METHODS

The definitions of the formulas, operators, and symbols used in this article are presented in Table I.

A. Overall Architecture

Fig. 2 illustrates the overall architecture of our proposed MLCNet, which comprises a Siamese encoder to extract the features of bitemporal images, two different fusion modules specifically designed for feature fusion at specific levels, and three MLIFD. We choose a Siamese network architecture with ResNet as the backbone of our encoder. To preserve high-precision spatial

TABLE I
DEFINITIONS OF FORMULAS, OPERATORS, AND SYMBOLS

name	operation
$Conv0$	Kernel size 1×1 convolutional layer.
$Conv1$	Kernel size 3×3 convolutional layer.
$Conv2$	Kernel size 7×7 convolutional layer.
$Conv3$	Kernel size 3×3 transposed convolutional layer.
$Conv^{dim}$	Kernel size 3×3 and dilation rate i convolutional layer.
$A \oplus B$	Concatenate feature A and feature B along dimension 1 (channel dimension).
$ReLU$	Rectified Linear Unit.
BN	Batch Normalization.
$Fuse$	$Fuse(A, B) = Conv1(A \oplus ReLU(BN(B)))$.

information and detailed features, we modified ResNet [55] by removing its average grouping and dense layers for feature extraction.

The last two layers involve more downsampling and channel increment operations than the first two layers in ResNet and effectively enhance the representation of semantic features. Hence, we utilize the EDPM and the CSAM to fuse bitemporal features from the first two and last two layers, separately. These fused features are progressively decoded through three MLIFD, generating the final feature map representing semantic changes. The final feature map is used to calculate the loss with semantic change GTs for network training. In addition, features fused via EDPM and CSAM are upsampled to the original image size and fed into their respective classifiers for deep supervision. The features of EDPM are used to calculate loss with edge-change GT, while those of CSAM are used to calculate loss with binary-change GT. This design enables selective learning of distinct characteristics among different feature levels, leveraging rich spatial detail information of low-level features and rich semantic

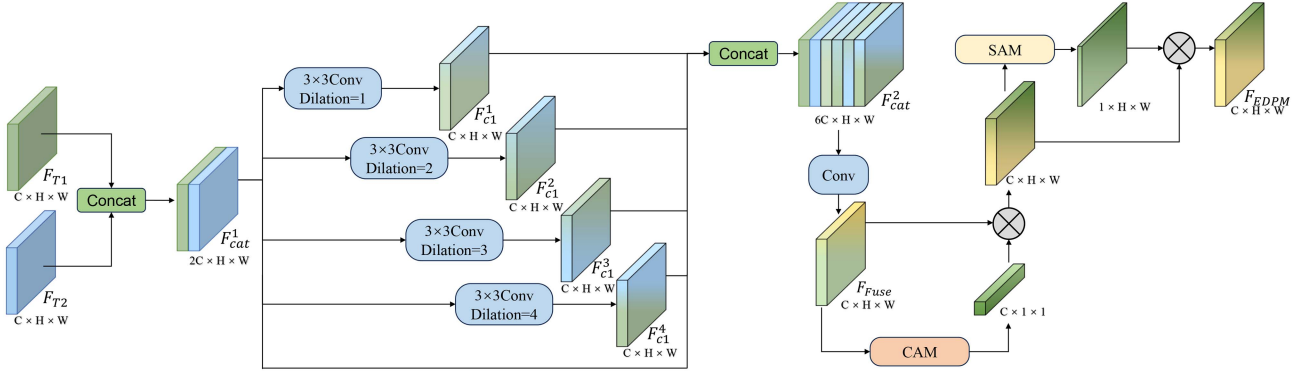


Fig. 3. Edge detail preservation module.

categories information of high-level features to their full extent. It prevents inaccurate prediction of change categories caused by ambiguous semantic information of low-level features and imprecise prediction of change edges caused by low-resolution spatial information of high-level features.

B. Edge Detail Preservation Module

In the change detection tasks, variations in lighting conditions, capture angles, cloud and fog occlusion, and seasonal changes may lead to significant spectral differences within the same land cover type between bitemporal images or even within different regions of the same image. Such differences can introduce misleading and incomplete information in change detection tasks, particularly for low-level features. Therefore, while the spatial information from low-level features can accurately indicate the edge of landcovers, the semantic information they provide often lacks precision in identifying the category of landcovers. Given this, prioritizing and effectively leveraging the precise and detailed spatial context information is more crucial during low-level feature fusion than focusing on semantic information. By moderately extracting comprehensive land cover semantic information while strengthening the utilization of spatial information in low-level features, the model improves its effectiveness and robustness. This tradeoff between the usage of spatial and semantic in low-level features contributes to overcoming the spectral differences in the unchanged regions and enhancing the model's accuracy in change detection.

Considering the problems mentioned above, we propose the EDPM, depicted in Fig. 3. The EDPM takes channel-concatenated bitemporal features as input. Subsequently, four parallel 3×3 convolution operations are adopted to simultaneously convolve the concatenated features. These convolved features are then concatenated with the original bitemporal features along the channel dimension, enriching them with spatial contextual information. Subsequent dimensionality reduction is performed through a convolutional operation. The four parallel convolutional layers use dilation rates ranging from 1 to 4, ensuring ample receptive fields and producing a structured sparse representation in the spatial dimension without significantly increasing computational cost or parameters. This operation

resembles the regularization concept used in traditional methods for extracting normalized feature representations [20], [56]. The group convolution operation effectively exploits the local connectivity property of convolutions, constructing a feature representation paradigm similar to a feature pyramid, which excellently integrates low-level features with a focus on their spatial detail information. The formulations of the above operations are as follows:

$$F_{cat}^1 = F_{T1} \oplus F_{T2} \quad (1)$$

$$F_{c1}^i = Conv1^{d=i}(F_{cat}^1), i = 1, 2, 3, 4 \quad (2)$$

$$F_{Fuse} = Fuse(F_{cat}^1, F_{c1}^1 \oplus F_{c1}^2 \oplus F_{c1}^3 \oplus F_{c1}^4) \quad (3)$$

here \oplus denotes the concatenation of features along the channel dimension.

The fused feature is then refined and purified through the attention mechanism to enhance the network's focus on key regions. A channel attention module (CAM) is first adopted to enhance the feature. CAM aggregates spatial information from features using average pooling (F_{avg}^c) and max-pooling (F_{max}^c) layers. After pooling the features, a weight-shared MLP with a hidden layer receiving them to produce the channel attention maps $Map_c \in \mathbb{R}^{C \times 1 \times 1}$. Besides, the size of the hidden activations in the weight-shared MLP is $\frac{C}{r} \times 1 \times 1$, with r representing the reduction ratio aims at minimizing parameter overhead. Finally, the resulting feature can be obtained by performing an elementwise addition on the output from each pooled feature of the shared MLP. The CAM can be expressed as follows:

$$M_c(F) = \sigma(W_1(W_0(F_{avg}^c)) + W_1(W_0(F_{max}^c))) \quad (4)$$

here, σ is the sigmoid function, $W_0 \in \mathbb{R}^{\frac{C}{r} \times C}$, $W_1 \in \mathbb{R}^{C \times \frac{C}{r}}$. The MLP weights W_0 and W_1 are shared for both inputs, followed by ReLU activation after W_0 .

After producing the CAM output, it is refined in spatial aspects by passing it through a spatial attention module (SAM). The SAM applies two channel dimension pooling layers to integrate the channel information, producing average pooling (F_{avg}^s) and max pooling (F_{max}^s) features. A 7×7 convolutional layer then receives the channel-concatenated feature produced by them to generate the final output. The computation of SAM can be

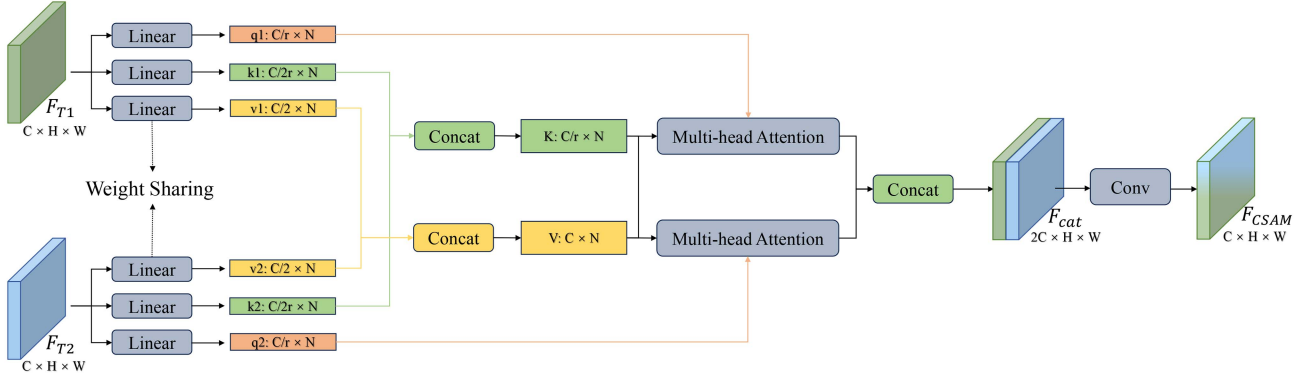


Fig. 4. Cross-temporal semantic attention module.

expressed as follows:

$$\mathcal{M}_s(F) = \sigma(\text{Conv}2(F_{\text{avg}}^s \oplus F_{\text{max}}^s)) \quad (5)$$

here, σ denotes the sigmoid function. Therefore, the EDPM module's output can be given by

$$F_{\text{EDPM}} = \mathcal{M}_s(F_{\text{CAM}}) * F_{\text{CAM}} \quad (6)$$

$$F_{\text{CAM}} = \mathcal{M}_c(F_{\text{Fuse}}) * F_{\text{Fuse}}. \quad (7)$$

C. Cross-Temporal Semantic Attention Module

Unlike low-level feature fusion, the fusion for high-level features should prioritize the integration of global semantic information rather than the spatial details information, due to the richer semantic context present in deeper layers and ambiguous spatial context representation. Therefore, we propose the CSAM for the fusion of high-level features, as illustrate in Fig. 4. In the CSAM, we employ three steps to fuse and constrain the features to focus on the semantic aspect.

First, bitemporal features $F_{T1}, F_{T2} \in \mathbb{R}^{C \times H \times W}$ are performed feature clustering with a weight-shared network, respectively. We use three MLP layers to form this network. After passing the features through this network, Queries $q1, q2 \in \mathbb{R}^{\frac{C}{r} \times N}$, Keys $k1, k2 \in \mathbb{R}^{\frac{C}{2r} \times N}$, and Values $v1, v2 \in \mathbb{R}^{\frac{C}{2} \times N}$ of bitemporal features are generated, where $N = H \times W$, and r represent the dimensional reduction rate, which is set to 8 in our case. Each Query, Key, and Value pair is clustered with the same MLP layers in the weight-shared network. Subsequently, Keys and Values from both temporal images are concatenated to form a change dictionary, while Queries remain separate. Finally, we obtain two Queries ($q1, q2$), one change dictionary (K, V) for the following cross-temporal semantic attention mechanism. These processes can be represented as the following equations:

$$q1 = \text{MLP}_{\text{query}}(F_{T1}), \quad q2 = \text{MLP}_{\text{query}}(F_{T2}) \quad (8)$$

$$k1 = \text{MLP}_{\text{key}}(F_{T1}), \quad k2 = \text{MLP}_{\text{key}}(F_{T2}) \quad (9)$$

$$v1 = \text{MLP}_{\text{value}}(F_{T1}), \quad v2 = \text{MLP}_{\text{value}}(F_{T2}) \quad (10)$$

$$K = k1 \oplus k2, \quad V = v1 \oplus v2. \quad (11)$$

The change dictionary consists of key-value pairs containing semantic information that reflects the differences between the

bitemporal images. Next, we separately use $q1$ and $q2$ to compute the multihead attention with the change dictionary (K, V).

The multihead attention mechanism (MHA) enables the model to distribute varying degrees of attention to different input regions. It processes different representation subspaces of the input in parallel and merges them at the end. This allows the model to gain the capacity to concentrate on information from different positions. Previous CD methods often apply MHA on single-temporal or concatenated bitemporal features [24], [25], [54]. They tend to suppress the differences between bitemporal features by learning the correlation of unchanged regions between bitemporal features, enhancing the changed features indirectly. In our case, the MHA mechanism computes attention between each Query ($q1, q2$) and the change dictionary (K, V). By directly comparing single-temporal features with “change” features, it can more intuitively identify regions in each temporal feature that are relevant to “change,” thereby enhancing the representation of change semantics. The computation of MHA can be represented by the following equations:

$$\text{MHA}(Q, K, V) = (\text{head}_1 \oplus \dots \oplus \text{head}_h) \cdot W^O \quad (12)$$

$$\text{head}_i = \text{Attention}(Q \cdot W_i^Q, K \cdot W_i^K, V \cdot W_i^V) \quad (13)$$

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \quad (14)$$

here, Q represent Queries, K represent Keys, and V represent Values. The Keys have a dimensionality of d_k . W_i^Q, W_i^K, W_i^V , and W^O are the parameter matrixes need learned by the model.

Finally, we reshape and concatenate the two obtained MHA outputs for each temporal back to the size of input features $\mathbb{R}^{C \times H \times W}$, resulting in F_{cat} with the size of $F_{\text{cat}} \in \mathbb{R}^{2C \times H \times W}$. Afterward, we apply a 3×3 convolution with batch normalization (BN) and ReLU activation to obtain the final fused feature of CSAM ($F_{\text{CSAM}} \in \mathbb{R}^{C \times H \times W}$). The formulation can be expressed as follows:

$$F_{\text{CSAM}} = \text{ReLU}(\text{BN}(\text{Conv}1(F_{\text{cat}}))) \quad (15)$$

$$F_{\text{cat}} = \text{MHA}(q1, K, V) \oplus \text{MHA}(q2, K, V). \quad (16)$$

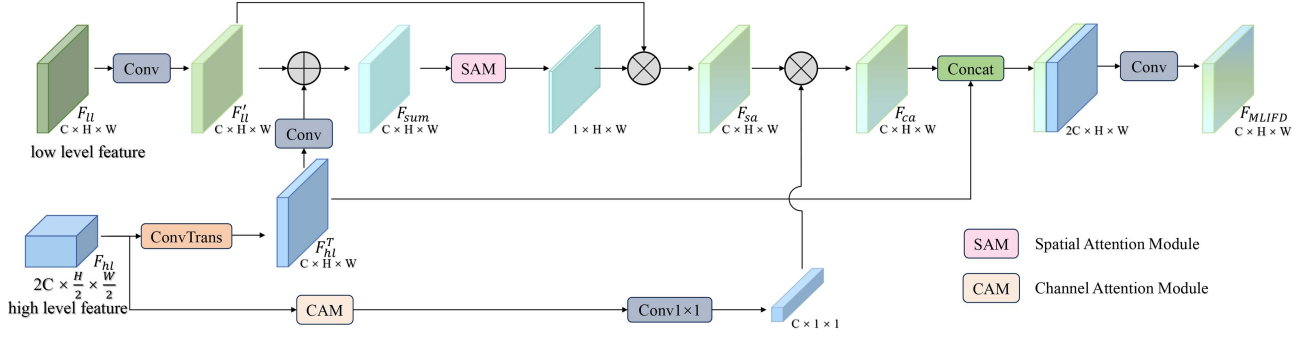


Fig. 5. Multilevel interactive fusion decoder.

D. Multilevel Interactive Fusion Decoder

Due to the success of UNet [18], change detection networks frequently use skip connections in decoding modules to decode multilevel features jointly. This method effectively preserves the detailed spatial information of the low-level features, while using the semantic cues of high-level features to guide the classification of pixels whose object categories are not clearly identified in the shallow layers [18], [19], [57]. As a result, it improves the change detection result. However, these methods often rely on simple upsampling operations for fusing high-level and low-level features during decoding. This may lead to the loss of precise information and confusion during the upsampling process, resulting in the blurring of low-level features' spatial context detail and a decrease in high-level features' semantic accuracy. In light of the differences in the representation of features at different levels, it is essential to consider their respective characteristics not only during the feature fusion stage but also during the feature decoding stage.

Based on the abovementioned issues, we propose the MLIFD, the detail structure illustrate in Fig. 5. Each decoding unit of this decoder simultaneously receives a high-level feature $F_{hl} \in \mathbb{R}^{2C \times \frac{H}{2} \times \frac{W}{2}}$ and a low-level feature $F_{ll} \in \mathbb{R}^{C \times H \times W}$. The high-level feature is passed to three streams: the first stream starts with performing the channel dimension reduction and upsampling operation through a transposed convolution layer, resulting in $F'_{hl} \in \mathbb{R}^{C \times H \times W}$. Then, the two sets of different level features F'_{hl}, F_{ll} are convolved separately and added together to obtain the preliminary aggregated feature $F_{sum} \in \mathbb{R}^{C \times H \times W}$, which is used to generate the spatial attention map of size $\mathbb{R}^{1 \times H \times W}$ through the SAM to refine the convolved low-level feature F'_{ll} . After elementwise multiplying the F'_{ll} with the attention map, the first fused feature F_{sa} can be obtained. The formulation for the first stream of high-level features is as follows:

$$F_{sa} = \mathcal{M}_s(F_{sum}) \cdot Conv1(F_{ll}) \quad (17)$$

$$F_{sum} = Conv1(Conv3(F_{hl})) + Conv1(F_{ll}). \quad (18)$$

The second stream starts with the channel reduction operation through a CAM and a 1×1 convolution layer, obtaining a channel attention weight map of size $\mathbb{R}^{C \times 1 \times 1}$. The second fused feature F_{ca} is obtained by elementwise multiplication of this channel weight attention map and the first fused feature F_{sa} to allocate the same channel information weights across all pixel

positions. The formulation for the second stream is as follows:

$$F_{ca} = Conv0(\mathcal{M}_c(F_{hl})) \cdot F_{sa}. \quad (19)$$

Through the first and second streams, the features at different levels are progressively fused. Note that the fused feature is produced in a gentle and gradual process, where the high-level feature acts as a guide for the low-level feature, rather than being directly added or concatenated with it. This approach preserves the detailed spatial context of the low-level features while leveraging the rich semantic information of high-level features. It effectively avoids information interference caused by the low-resolution spatial context representation of high-level features and addresses potential decreases in spatial resolution during feature fusion. The third stream of high-level features aims to avoid information loss. We concatenate the upsampled high-level feature F'_{hl} obtained through transposed convolution with the fused feature F_{ca} based on the guidance of the high-level features and perform fusion through convolution to obtain the final result $F_{MLIFD} \in \mathbb{R}^{C \times H \times W}$. The expression for the final output of this module is as follows:

$$F_{MLIFD} = ReLU(BN(Conv1(Conv3(F_{hl}) \oplus F_{ca}))). \quad (20)$$

E. Deep Supervision and Loss Function

The difficulty of classifying whether a pixel belongs to a change category in remote sensing change detection tasks is closely related to its spatial position within its cluster. In a complex bitemporal image pair, pixels near the edges of change clusters are more likely to be misclassified, while those closer to the center of clusters tend to have higher classification accuracy owing to their higher intraclass consistency. Therefore, during network training, rather than treating all pixels equally, we need to pay more attention to the edge pixels. Therefore, we deconstruct the original GTs into binary change GTs and edge change GTs to apply multitask learning with level-specific constraints. For binary change GTs, we categorize all change classes as change, represented by 1 in the GT, while 0 indicates unchanged land cover. Edge change GTs are further derived by extracting the edges from the binary change GTs. In the GT edge change, 1 denotes the edges of the changed areas, while 0 represents all the other regions.

We compute loss functions for the network's final output with the semantic change GTs: if they are not provided, binary

change GTs are employed. Binary and edge change GTs are used through deep supervision to calculate their respective losses. As illustrated in Fig. 2, MLCNet adopts four deep supervision branches. Each branch upsamples the fused bitemporal features to the original image size. These upsampled features are then passed to their respective classifiers to generate prediction results. EDPM-linked classifiers compute losses with edge change GTs, while CSAM-linked classifiers utilize binary change GTs for loss calculation. Therefore, three different GTs are used to calculate the final training loss \mathcal{L} , which can be defined as the combination of the following three loss functions:

$$\mathcal{L} = \alpha_1 \mathcal{L}_{\text{semantic}} + \alpha_2 \sum_{i=1}^2 \mathcal{L}_{\text{binary}}^i + \alpha_3 \sum_{i=1}^2 \mathcal{L}_{\text{edge}}^i \quad (21)$$

here, the final loss \mathcal{L} is the combination of semantic (L_{semantic}), binary (L_{binary}), and edge (L_{edge}) loss functions. α_1 , α_2 , α_3 are the corresponding weights that control the tradeoff between three loss functions. We set $\{\alpha_1, \alpha_2, \alpha_3\}$ to $\{1.0, 0.2, 0.2\}$ for training MLCNet.

Since there are considerable differences in the foreground and background quantities between the binary and edge change GTs, the binary cross entropy loss and the Dice loss are used for the computation of $\mathcal{L}_{\text{binary}}$ and $\mathcal{L}_{\text{edge}}$. The formulas are as follows:

$$\mathcal{L}_{CE}(P, G) = w_{\text{pos}} G \log(P) + w_{\text{neg}} (1 - G) \log(1 - P) \quad (22)$$

$$w_{\text{pos}} = n_{\text{pos}} / n_{\text{sum}} \quad (23)$$

$$w_{\text{neg}} = n_{\text{neg}} / n_{\text{sum}} \quad (24)$$

$$\mathcal{L}_{\text{Dice}}(P, G) = \frac{2 \cdot P \cdot G}{P^2 + G^2} \quad (25)$$

here P represents the predicted map generated by the classifiers of the four deep supervision branches, and G is the corresponding GT. In addition, to consider the global context as much as possible when calculating the final loss, the cross-entropy loss and the Lovász-softmax loss [58] are used to compute the corresponding loss in $\mathcal{L}_{\text{semantic}}$.

IV. EXPERIMENTS

To validate the performance of our proposed method, experiments were conducted on three public CD datasets: LEVIR building change detection (LEVIR-CD) dataset [21], side-looking change detection dataset (S2Looking) [59], and the simulated multimodal aerial remote sensing dataset (SMARS) [60]. This section first introduces the datasets and comparison methods, followed by the evaluation metrics used in the experiments, and finally showcases the experiments and the analysis.

A. Dataset Introduction

LEVIR-CD [21]: LEVIR-CD is a dataset consisting of 637 pairs of VHR Google Earth image patches. Each patch measures 1024×1024 pixels and was captured over a period of 5 to 14 years. These bitemporal images show significant changes in land

use, especially in construction expansion. The dataset contains a wide range of building types, including villa residences, tall apartments, small garages, and large warehouses. LEVIR-CD is fully annotated, containing a total of 31 333 individual instances of changed buildings. We employed the original partitioning of the LEVIR-CD dataset for training and testing. The data are divided using a nonoverlapping approach with a resolution of 512×512 . In the end, we obtain 1780 pairs of data for network training, 128 pairs for validation, and 512 pairs for testing.

S2Looking [59]: The S2Looking dataset is a comprehensive bitemporal dataset for building change detection, comprising large-scale side-looking satellite images. These images are captured from varying off-nadir angles and cover rural areas worldwide. The dataset contains 5000 registered pairs of bitemporal images, each sized 1024×1024 pixels with a resolution ranging from 0.5 to 0.8 m per pixel. Annotated with over 65 920 instances of change, the dataset includes label maps indicating regions of newly constructed and demolished buildings for each bitemporal image pair. In the S2Looking dataset, we follow the original data partitioning, applying nonoverlapping 512×512 cropping to each image set. This yields 14 000 training pairs, 2000 validation pairs, and 4000 testing pairs. For ground truth (GT) processing, we explore two scenarios. The first scenario only uses binary change information, the second scenario adds specific semantic change information, where the change category in binary change GT is further split into newly constructed buildings and demolished buildings. This dual approach allows for a detailed analysis of both binary and semantic changes.

SMARS [60]: SMARS is a synthetic dataset featuring scenario pairs from simulated Paris and Venice urban changes, named SParis and SVenice, for training change detection applications. It includes ortho-images and digital surface models (DSMs) showcasing changes like increased buildings and reduced green spaces. Rendered at 30 and 50 cm ground sampling distances (GSDs), the 30 cm GSD is primarily used by our experiment. The dataset offers binary and ternary change GT for detection tasks; the categories of ternary change GT further split the change category in binary change GT into newly constructed buildings and demolished buildings. We use the original partition and the data are cropped into 256×256 patches, resulting in 1584 training pairs, 924 validation pairs, and 1474 testing pairs below 30 cm GSD.

B. Comparison Methods

To comprehensively evaluate our method, we compare it with the following state-of-the-art networks, including three classic fully CNNs: FC-EF [18], FC-Siam-conc [18], and FC-Siam-diff [18], three attention based networks: STANet-PAM [21], SNUNet [19], HANet [57], and five transformer-based networks: ChangeFormer [25], BIT [24], RSP-BIT [61], AMT-Net [53], and Changer [54] as comparison methods for our network.

C. Evaluation Metrics

To evaluate the congruence between predictions and the GT in remote sensing change detection applications, we employ six

TABLE II
PERFORMANCE OF MODELS IN BINARY CHANGE DETECTION

Method	LEVIR-CD(%)						S2Looking(%)						SMARS(%)					
	P	R	F1	IoU	Kappa	OA	P	R	F1	IoU	Kappa	OA	P	R	F1	IoU	Kappa	OA
FC-EF [18]	90.38	88.11	89.23	80.56	88.67	98.93	59.81	48.20	53.38	36.41	52.87	98.98	95.42	91.14	93.23	87.32	90.52	97.12
FC-Siam-conc [18]	90.83	89.67	90.25	82.23	89.73	99.01	73.46	54.13	62.33	45.28	61.94	99.21	<u>96.71</u>	94.99	95.84	92.02	94.93	98.51
FC-Siam-diff [18]	92.88	87.12	89.91	81.67	89.39	99.00	70.25	55.41	61.95	44.88	61.54	99.18	96.03	<u>95.99</u>	96.01	92.32	95.12	98.55
STANet [21]	91.29	87.28	89.24	80.58	88.68	98.91	44.81	31.39	36.92	22.64	35.28	95.35	96.52	92.84	94.64	89.83	93.48	97.92
SNUNet [19]	<u>93.34</u>	87.81	90.49	82.64	90.00	99.06	63.22	50.31	56.03	38.92	55.55	99.04	95.89	95.53	95.71	91.77	94.76	98.45
HANet [58]	93.01	<u>89.68</u>	<u>91.31</u>	<u>84.01</u>	<u>90.87</u>	<u>99.17</u>	68.79	62.05	<u>65.25</u>	<u>48.42</u>	<u>64.94</u>	99.20	96.56	95.63	96.10	92.49	95.24	98.59
ChangeFormer [25]	94.00	<u>87.02</u>	<u>90.37</u>	<u>82.43</u>	<u>89.88</u>	<u>99.06</u>	68.55	50.17	57.93	40.78	57.50	99.12	95.42	95.18	95.30	91.03	94.26	98.30
BIT [24]	91.99	87.13	89.49	80.98	88.94	98.96	74.04	50.02	59.71	42.56	59.31	99.18	96.19	94.95	95.57	91.51	94.60	98.40
RSP-BIT [63]	92.03	88.37	90.16	82.09	89.63	99.00	77.31	52.16	62.29	45.23	61.92	<u>99.24</u>	96.67	95.93	<u>96.30</u>	<u>92.87</u>	<u>95.50</u>	<u>98.67</u>
AMTNet [54]	92.44	89.36	90.87	83.27	90.39	99.09	65.62	56.30	60.60	43.48	60.17	99.13	96.78	95.48	96.13	92.54	95.21	98.51
Changer [55]	92.06	89.11	90.56	82.75	90.15	99.22	76.54	56.33	64.90	48.04	64.48	99.16	95.96	94.97	95.46	91.32	94.46	98.36
MLCNet	92.72	90.77	91.73	84.73	91.30	99.16	71.84	<u>61.56</u>	66.30	49.59	65.92	99.25	97.90	97.56	97.73	95.55	97.27	99.22

Bold indicate the highest score, underline indicate the second score.

well-established metrics that have been widely utilized, encompass precision (Pre.), recall (Rec.), F1-score (F1), intersection over union (IoU), Cohen’s Kappa (Kappa), and overall accuracy (OA). Each metric can be defined as follows:

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

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

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (28)$$

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

$$\text{Kappa} = \frac{p_o - p_e}{1 - p_e} \quad (30)$$

$$\text{OA} = \frac{TP + TN}{TP + TN + FP + FN} \quad (31)$$

where TP, TN, FP, and FN represent the pixel number of true positive, true negative, false positive, and false negative, respectively. p_o, p_e are defined as follows:

$$\begin{cases} p_o = \frac{TP+TN}{TP+TN+FP+FN} \\ p_e = \frac{(TP+FP)(TP+FN)+(TN+FN)(TN+FP)}{(TP+TN+FP+FN)^2} \end{cases} \quad (32)$$

All six metrics indicate better model performance as they increase.

D. Training Details

Our models were developed using PyTorch 1.8.1 (CUDA 11.1) and were trained on four NVIDIA GeForce GPUs with 24 GB of memory. During training, we used the AdamW optimizer with a weight decay of $5e-4$ and a learning rate of $1e-3$, with a cosine annealing strategy for updating the learning rate, the annealing epoch and maximum training epoch are set to 20 and 80 separately, to minimize the loss. All models converged before reaching the maximum training epoch. For each model, we repeated the training process at least five times with a batch size of 32. The final comparison was based on the average of the top three highest accuracies achieved by each model.

E. Comparative Experiments

1) *Binary Change Detection Experiment*: During the binary change detection experiments, the LEVIR-CD [21], S2Looking, and SMARS datasets are used solely with binary change GT for supervising the network training. Table II summarizes the quantitative metric results of different methods. Data comparison from the table indicates that MLCNet generally outperforms most of the compared methods across a range of metrics and scenarios.

Taking the LEVIR-CD dataset as an example, MLCNet exhibits a 0.42% improvement in the F1 score over the second-best HANet. Notably, compared to other methods, the improvements are more significant, with increments of 2.50% (over FC-EF), 1.48% (over FC-Siam-conc), 1.82% (over FC-Siam-diff), 2.49% (over STANet), 1.24% (over SNUNet), 1.36% (over ChangeFormer), 2.24% (over BIT), 1.54% (over RSP-BIT), 0.86% (over AMTNet), and 1.17% (over Changer).

The significant improvements seen in MLCNet can be attributed to several factors. First, the network employs different specifically designed modules for high-level and low-level features’ fusion. This approach effectively constrains the network to appropriately prioritize the strengths associated with features at different levels. Second, the MLIFD decoding unit effectively preserves the respective advantages of multilevel features when decoding, avoiding mutual interference between information from different level features. As a result, it can simultaneously utilize the semantic information of high-level features and the spatial detail information of low-level features to generate detection results with both high intraclass consistency and well-defined edges. Furthermore, the network’s training process uses multitask learning with level-specific constraint strategies, which further enhances the network’s utilization of the advantages of features at different levels through rigorous constraints. These advantages of the network are demonstrated in subsequent ablation experiments.

In addition to the basic results for binary change detection, we also separately calculate the F1 and IoU scores for the edge precision of each network, as shown in Table III. Our MLCNet demonstrates a notable advantage in edge precision compared to other models. By comparing the differences between CNN-based and transformer-based methods in binary change detection accuracy and edge precision in Tables II and III, we find that although transformer-based methods generally have

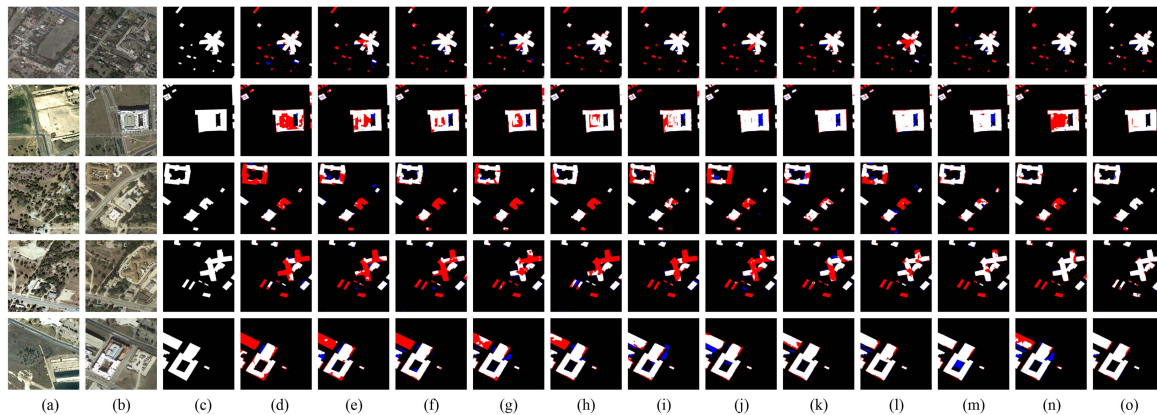


Fig. 6. Visualization result of our proposed methods and the comparative SOTA methods on LEVIR-CD. (a) T1 images. (b) T2 images. (c) GT. (d) FC-EF. (e) FC-conc. (f) FC-diff. (g) STANet. (h) SNUNet. (i) HANet. (j) BIT. (k) ChangeFormer. (l) RSP-BIT. (m) AMTNet. (n) Changer. (o) MLCNet. Black, white, red, and blue regions indicate the true negative, true positive, false negative, and false positive, respectively.

TABLE III
PERFORMANCE OF MODELS IN EDGE CHANGE DETECTION

Method	LEVIR-CD(%)		S2Looking(%)		SMARS(%)	
	F1	IoU	F1	IoU	F1	IoU
FC-EF [18]	60.78	43.66	22.93	12.95	47.42	31.08
FC-Siam-conc [18]	61.41	44.31	22.93	12.95	59.20	42.05
FC-Siam-diff [18]	<u>62.11</u>	<u>45.04</u>	<u>23.16</u>	<u>13.09</u>	59.28	42.13
STANet [21]	57.71	40.56	15.84	08.64	59.20	42.05
SNUNet [19]	62.02	44.95	17.47	09.57	58.12	40.96
HANet [58]	56.70	39.57	13.19	07.06	<u>61.30</u>	<u>44.20</u>
ChangeFormer [25]	59.07	41.92	15.56	08.43	53.84	36.84
BIT [24]	58.88	41.73	16.90	09.23	54.90	37.84
RSP-BIT [63]	59.15	41.99	17.59	09.65	54.79	37.73
AMTNet [54]	58.77	41.62	15.43	08.36	54.92	37.86
Changer [55]	56.68	39.54	20.72	11.56	53.48	36.50
MLCNet	62.99	45.97	27.95	16.25	70.61	54.58

Bold indicate the highest score, underline indicate the second score.

certain advantages in binary accuracy compared to CNN-based methods, CNN-based methods perform better in edge precision. This is because while transformer-based methods possess better global information interaction, they also tend to blur edge details, whereas CNNs excel in extracting and preserving edge details. Our proposed MLCNet effectively integrates the spatial detail preservation capability of CNNs with the global information interaction capability of attention mechanisms, resulting in significant improvements in both binary detection accuracy and edge precision.

Figs. 6–8 illustrate the visualization results for our methods and the comparative methods across the three datasets. To show how our model maintains the semantic consistency of change categories and edge detection, we chose some samples from the more complicated environments in our test set. To make our detection results easier to understand, we use blue and red to represent false positives and false negatives, respectively. Meanwhile, black represents true negatives and white represents true positives. It can be observed that transformer-based methods (BIT, ChangeFormer, RSP-BIT, AMTNet, Changer) exhibit fewer false negative regions in change regions and generally produce better visual results in detected changes when compared to CNN-based methods (FC-EF, FC-Siam-conc, FC-Siam-diff).

However, methods based on attention (SNUNet, HANet) and CNN demonstrate certain advantages in the accuracy of change edge. In addition, our proposed MLCNet not only exhibits lower void rates within change categories but also performs well at change category boundaries, yielding results closest to GT overall. Taking the second row of visualization results in Fig. 6 as an example, compared to other transformer-based methods, MLCNet not only better preserves overall structural changes in buildings, but also accurately identifies small nonchanging areas within significant change regions. This allows for precise change edges to be established. Furthermore, as illustrated in the first and third rows of Fig. 6, MLCNet consistently outperforms other comparative methods in accurately delineating complex building edge regions. Overall, our proposed MLCNet architecture design shows strong effectiveness by presenting lower void rates within change categories and generating results closest to the GT. The results clearly indicate the efficiency of the MLCNet architecture design that we proposed.

2) *Semantic Change Detection Experiment*: To further explore MLCNet’s detection capabilities in semantic change detection environments, we train and test our network with three classes of semantic labels from the S2Looking and SMARS datasets. As mentioned in Section IV-A, for the three semantic labels (unchanged, newly constructed, demolished buildings), we further deconstruct them into edge change labels and binary change labels. A schematic diagram of the refined dataset is shown in Fig. 9. In Fig. 9(d), the red and cyan regions indicate the areas where buildings were demolished or newly constructed, respectively.

For comparison, we choose classification-based methods from the aforementioned techniques. Adjustments are applied exclusively to the final layer outputs of these comparison methods, originally single-channel or dual-channel outputs, which are transformed into three-channel outputs corresponding to the categories of unchanged objects, newly constructed buildings, and demolished buildings. The training approach for MLCNet remains consistent with the earlier description. Table IV presents the accuracies of different categories and the mean accuracy on the S2Looking and SMARS datasets. The four metrics in

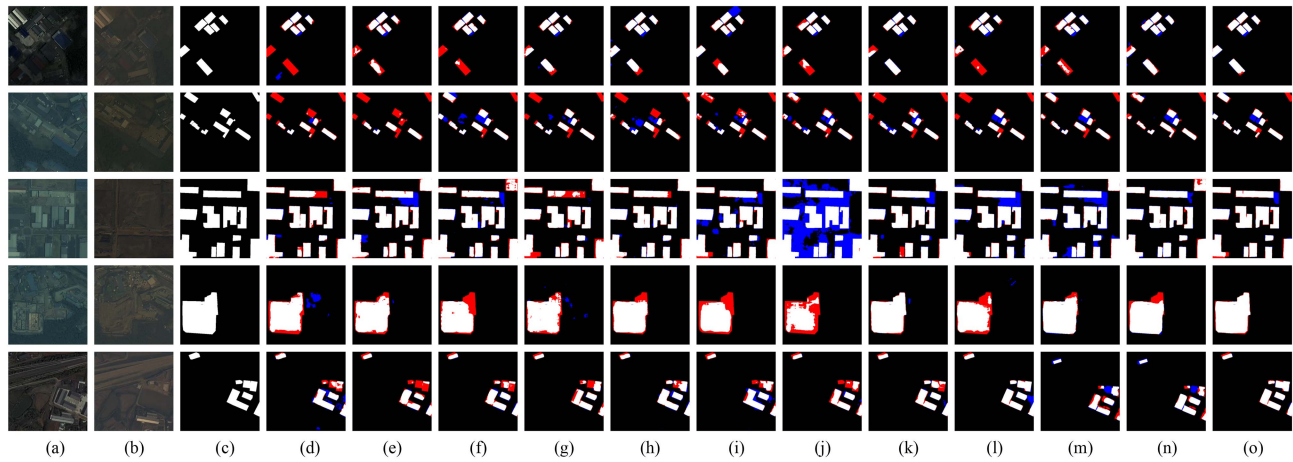


Fig. 7. Visualization result of our proposed methods and the comparative SOTA methods on S2Looking. (a) T1 images. (b) T2 images. (c) GT. (d) FC-EF. (e) FC-conc. (f) FC-diff. (g) STANet. (h) SNUNet. (i) HANet. (j) BIT. (k) ChangeFormer. (l) RSP-BIT. (m) AMTNet. (n) Changer. (o) MLCNet. Black, white, red, and blue regions indicate the true negative, true positive, false negative, and false positive, respectively.

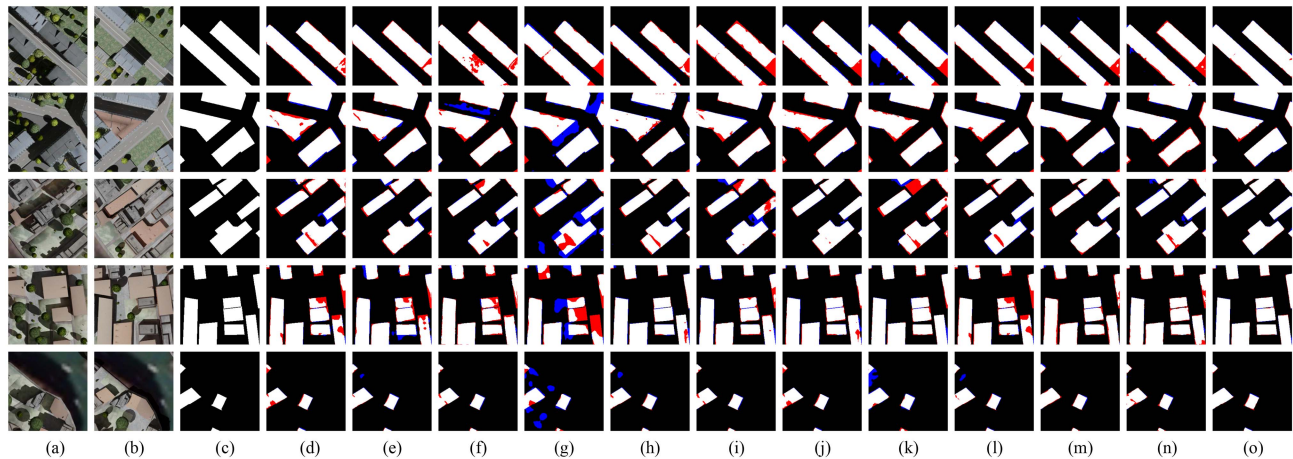


Fig. 8. Visualization result of our proposed methods and the comparative SOTA methods on SMARS. (a) T1 images. (b) T2 images. (c) GT. (d) FC-EF. (e) FC-conc. (f) FC-diff. (g) STANet. (h) SNUNet. (i) HANet. (j) BIT. (k) ChangeFormer. (l) RSP-BIT. (m) AMTNet. (n) Changer. (o) MLCNet. Black, white, red, and blue regions indicate the true negative, true positive, false negative, and false positive, respectively.

TABLE IV
PERFORMANCE OF MODELS IN SEMANTIC CHANGE DETECTION

Method	S2Looking(F1%)				SMARS(F1%)			
	Unchanged	Newly Constructed	Demolish	Mean	Unchanged	Newly Constructed	Demolish	Mean
FC-EF [18]	99.48	58.50	36.64	64.88	98.54	82.48	75.25	85.42
FC-Siam-conc [18]	99.60	66.56	49.29	71.82	99.09	84.94	76.75	86.93
FC-Siam-diff [18]	99.58	65.66	50.16	71.80	99.12	84.99	77.11	87.07
SNUNet [19]	99.52	60.45	42.33	67.43	99.05	93.60	93.35	95.33
HANet [58]	99.45	53.53	37.26	63.41	99.14	<u>93.62</u>	93.29	95.35
ChangeFormer [25]	99.55	61.96	39.43	66.98	98.96	84.07	80.09	87.71
BIT [24]	99.59	64.06	48.27	70.64	99.03	85.14	82.89	89.02
RSP-BIT [63]	<u>99.61</u>	66.37	50.86	72.28	<u>99.24</u>	92.13	92.68	94.68
AMTNET [54]	99.55	65.42	45.72	70.23	<u>99.08</u>	93.35	<u>93.85</u>	<u>95.43</u>
Changer [55]	99.58	68.76	<u>53.97</u>	<u>74.10</u>	99.00	92.47	92.28	94.59
MLCNet	99.62	70.20	55.21	75.01	99.39	94.78	94.28	96.15

Bold indicate the highest score, underline indicate the second score.

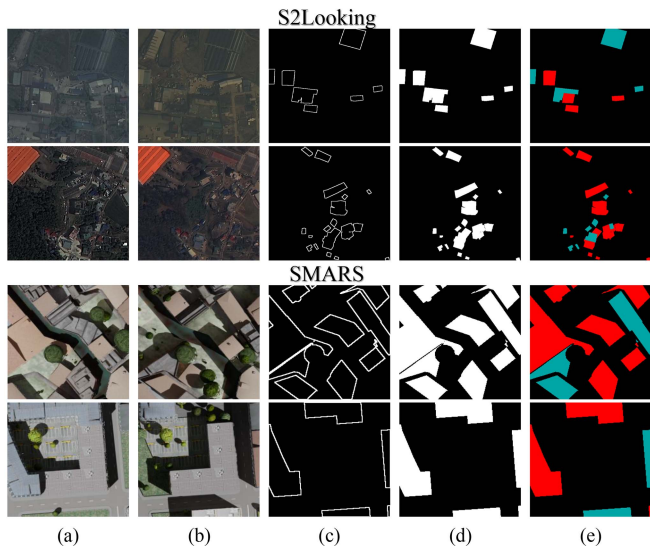


Fig. 9. Examples of two refined datasets. (a) T1 images. (b) T2 images. (c) Edge change GT. (d) Binary change GT. (e) Semantic change GT. Edge GT: **White** = changed edges, **black** = other areas. Binary GT: **White** = changed regions, **black** = unchanged regions. Semantic GT: **Cyan** = newly constructed buildings, **Red** = demolished buildings, **black** = unchanged regions.

the table represent the F1 scores for unchanged areas, newly constructed buildings, demolished buildings, and the average F1 score, respectively. It can be observed that in the scenario of using semantic labels, our model exhibits significantly better accuracy compared to other methods. The visualization result is shown in Fig. 10. Red, cyan, and black, respectively, represent newly constructed buildings, demolished buildings, and unchanged areas. Through visualization results, we can observe that MLCNet demonstrates significant advantages over other methods, especially in the aspect of keeping intra-class consistency and edge delineation. In the semantic change detection task, distinguishing similar categories becomes more challenging. Therefore, it can be observed that while some other methods may yield acceptable results from a binary change detection perspective, they produce unclear boundaries for various change categories and exhibit low intraclass consistency in semantic change detection scenarios. This further underscores the unique advantage of MLCNet’s design in maintaining boundary clarity and intraclass consistency.

3) *Model Complexity and Efficiency Analysis*: To evaluate the complexity and efficiency of our proposed MLCNet, we present the number of trainable parameters, the number of arithmetic operations [i.e., floating point operations (FLOPs)], and the frames per second (FPS) of MLCNet and other comparative methods on the LEVIR-CD [21] dataset in Table V. We report the speed (FPS) with an input size of 512×512 on a single NVIDIA GTX 3090 GPU. According to Table V, MLCNet has the highest number of learnable parameters, reaching 30.328 M, indicating a higher model complexity of our MLCNet. However, it is important to note that despite the higher model complexity, our model’s FLOPs remain at a moderate level compared to all the comparative methods. Although MLCNet has a larger number of parameters, its computational efficiency is still higher

TABLE V
MODEL PARAMETER COUNT AND COMPUTATIONAL EFFICIENCY STATISTICS

Method	Params(M)	FLOPs(G)	FPS
FC-EF [18]	1.349	57.036	79.48
FC-Siam-conc [18]	<u>1.545</u>	84.958	64.66
FC-Siam-diff [18]	1.349	<u>75.294</u>	63.58
STANet [21]	12.176	196.436	45.85
SNUNet [19]	2.612	709.389	43.11
HANet [58]	10.198	281.231	37.82
ChangeFormer [25]	24.271	134.338	17.06
BIT [24]	3.006	124.768	59.12
AMTNet [54]	24.664	343.635	15.54
Changer [55]	11.372	339.573	42.62
MLCNet	30.328	224.475	41.24

Bold indicate the minimum or fastest score, underline indicate the second score.

(FLOPs lower) than some models with fewer parameters, such as Changer [54], AMTNet [53], and HANet [57]. In addition, MLCNet’s FPS also remains at a moderate level, indicating good computational efficiency that can meet practical application needs.

F. Ablation Studies

The excellent performance of MLCNet on the three datasets demonstrates the rationality of our design approach, which emphasizes the different advantages of multilevel features by applying different constraints and different modules to them. To further validate the effectiveness of MLCNet’s module design and training strategies, we conducted the following ablation experiments.

1) *Ablation Study of Fusion Modules*: To further explore the optimal ratio of EDPM to CSAM, we conducted the following experiments: Dynamically adjusting the numbers of EDPM and CSAM modules within the range of 0 to 4 (the total sum is always equal to 4) in the network and testing the final accuracy. The results are presented in Table VI, where the col of “EDPM num” and “CSAM num” indicates the number of EDPM and CSAM used in the model separately. In all tested networks, the EDPM modules are consistently positioned above the CSAM modules, intended for processing lower level features. We denote the network with the ratio of EDPM number to CSAM number equal to 4:0 as “EDPNet” and the network with the ratio equal to 0:4 as “CSANet.” According to Table VI, networks that apply different fusion modules for features at different levels demonstrate higher accuracy compared to those using a single fusion module. Moreover, when the EDPM and CSAM modules are evenly distributed with a ratio of 2:2, the overall model accuracy reaches its optimum. Networks with more EDPM modules in the same ratio (1:3 versus 3:1, or 0:4 versus 4:0) demonstrate a certain advantage in edge accuracy. This further validates their ability to preserve spatial details and enhance edge accuracy. Although CSAM has more parameters, networks with more CSAM modules in the same ratio demonstrate lower accuracy in the SMARS dataset. The reason probably is that the illumination of buildings in the SMARS is more uniform

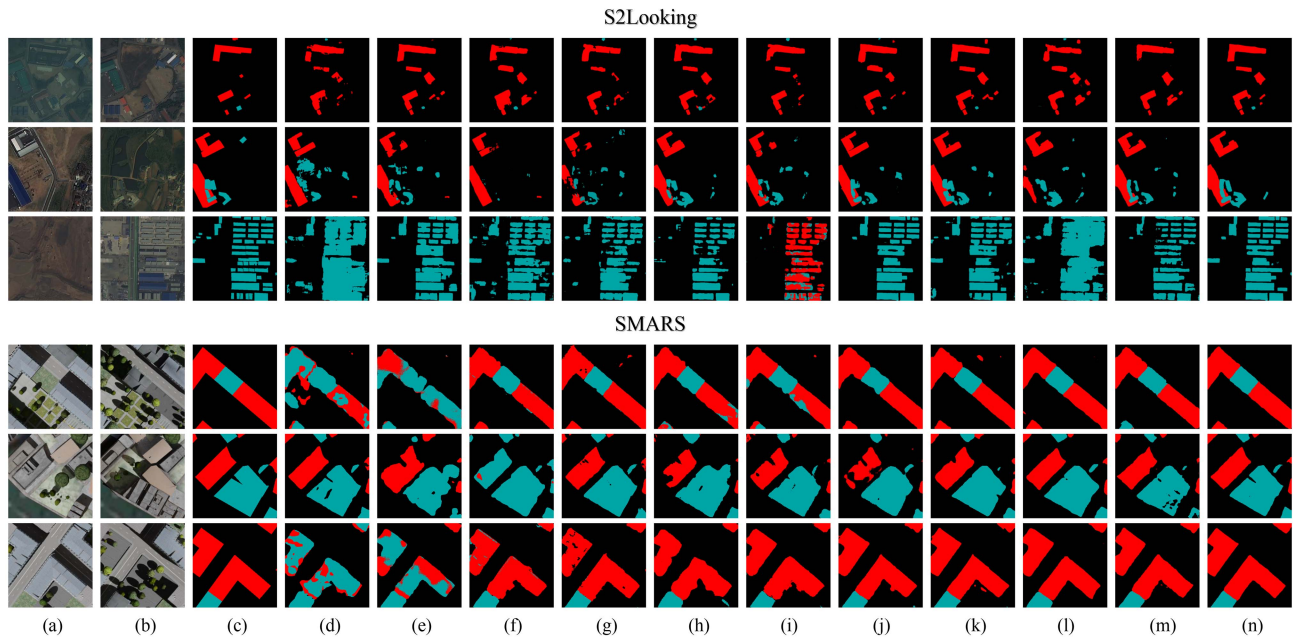


Fig. 10. Visual comparisons of the proposed method and the SOTA approaches on the semantic change detection scenario. (a) T1 images. (b) T2 images. (c) GT. (d) FC-EF. (e) FC-conc. (f) FC-diff. (g) SNUNet. (h) HANet. (i) BIT. (j) ChangeFormer. (k) RSPBIT. (l) AMTNet. (m) Changer. (n) MLCNet.

TABLE VI
ABLATION STUDY OF THE MODEL WITH DIFFERENT FUSION MODULES

EDPM num	CSAM num	S2Looking(%)								SMARS(%)							
		Change						Edge		Change						Edge	
		P	R	F1	IoU	Kappa	OA	F1	IoU	P	R	F1	IoU	Kappa	OA	F1	IoU
4	0	76.03	55.14	63.92	46.98	63.39	99.08	26.91	15.54	97.19	95.44	96.31	92.88	95.50	98.67	58.75	41.60
3	1	74.79	55.95	64.01	47.07	63.62	99.20	26.42	14.82	97.47	97.02	97.25	94.64	96.63	99.00	66.27	48.91
2	2	71.97	61.62	66.40	49.70	66.05	99.27	27.86	16.19	97.63	97.07	97.35	94.84	96.81	99.10	70.29	54.58
1	3	71.38	61.48	66.06	49.33	65.69	99.25	22.72	13.30	97.60	96.66	97.13	94.42	96.59	99.09	61.93	45.27
0	4	74.84	58.48	65.65	48.87	65.25	99.25	20.31	11.30	97.31	95.24	96.26	92.80	95.50	98.74	57.54	40.39

Bold indicate the highest score.

and stable, and the background is less complex, in this situation, edge accuracy is more important since the intraclass consistency is high enough. Meanwhile, in the more complex S2Looking dataset, the network with more CSAM modules in the same ratio demonstrates higher accuracy.

To investigate the feature extraction ability of these models, three pairs of bitemporal RS images are randomly selected from SMARS, and the final feature maps of MLCNet, EDPNet, and CSANet are visualized in Fig. 11. By comparing the feature maps, it can be observed that EDPNet exhibits richer spatial details but lacks intraclass consistency among different objects, whereas CSANet shows the opposite characteristics. As shown in Fig. 11(d), it can be observed that the EDPNet based solely on the EDPM exhibits high-intensity features at change edges. However, its semantic consistency within change categories is low, leading to issues such as holes shown in Fig. 11's third row. Moreover, the variance in features across various change categories is minimal, potentially causing semantic ambiguity. While this may not be prominent in binary change scenarios, it becomes particularly evident in semantic change tasks. The CSAM based CSANet produces relatively high semantic consistency but generates rounded edges for changed buildings and lower intensity boundary features. In contrast, MLCNet effectively preserves both rich spatial details and intraclass

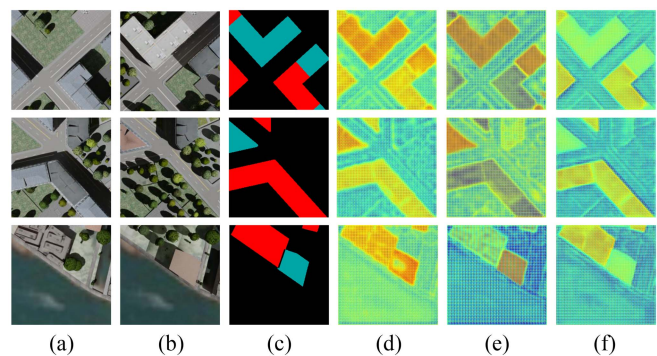


Fig. 11. Feature maps visualization result obtained by MLCNet and its variants. (a) T1 image. (b) T2 image. (c) Semantic GT. (d) EDPNet. (e) CSANet. (f) MLCNet.

consistency among change regions. The visualization result of features maps, along with the accuracy metrics presented in Table VI, collectively demonstrates the effectiveness of our proposed approach.

2) *Ablation Study of Auxiliary Loss*: The second ablation experiment further validates the effectiveness of employing edge change labels and binary change labels for deep supervision in MLCNet. The tested models include:

TABLE VII
ABLATION STUDY OF THE MODEL WITH DIFFERENT AUXILIARY LOSS

Method	LEVIR-CD(%)						S2Looking(%)						SMARS(%)					
	P	R	F1	IoU	Kappa	OA	P	R	F1	IoU	Kappa	OA	P	R	F1	IoU	Kappa	OA
Base	91.91	89.94	90.91	82.52	90.48	99.19	74.59	57.09	64.67	47.29	64.25	99.15	97.69	96.12	96.90	93.98	96.52	98.73
+ Edge GT	92.78	89.19	90.95	82.79	90.53	99.21	74.85	57.44	65.00	47.82	64.60	99.17	97.85	96.17	97.00	94.18	96.60	98.94
+ Binary GT	92.66	89.92	91.27	83.22	90.84	99.19	74.93	57.91	65.33	48.30	64.94	99.20	97.21	97.55	97.38	94.89	96.93	99.03
+ E/B GT	92.72	90.77	91.73	84.73	91.30	99.16	71.84	61.56	66.30	49.59	65.92	99.25	97.90	97.56	97.73	95.55	97.27	99.22

Bold indicate the highest score.

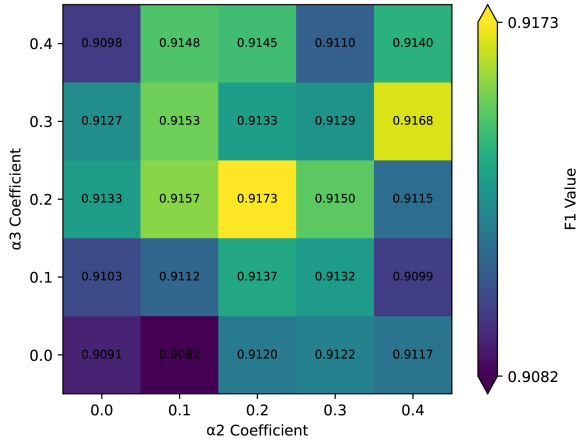


Fig. 12. Ablation study of weights setting to loss functions.

- 1) *Base*: The MLCNet is trained without deep supervision.
- 2) *+ Edge GT*: The MLCNet's four deep supervision branches all use edge labels for training.
- 3) *+ Binary GT*: The MLCNet's four deep supervision branches all use binary labels for training.
- 4) *+ E/B GT*: The MLCNet is trained with edge labels and binary labels on the deep supervision branches connected to EDPM and CSAM, respectively.

Table VII shows the accuracy of these models. It can be observed that using edge GT and binary GT separately for auxiliary loss calculation leads to a certain improvement in the network's accuracy. However, the improvement in accuracy by solely using edge GT is relatively insignificant, with only a 0.04%, 0.33%, and 0.1% increase on the three datasets, respectively. Conversely, the F1 accuracy improvement by solely adding binary GT is 0.36%, 0.66%, and 0.48%, respectively. Moreover, with targeted supervision using edge labels for shallow layers and binary labels for deep layers, the accuracy improvement becomes more pronounced, reaching 0.82%, 1.63%, and 0.83% on the three datasets, respectively. This demonstrates the feasibility of the multitask level-specific constraint strategy.

3) *Ablation Study of Joint Loss Function*: As illustrated in (21), our loss function \mathcal{L} is made up of three components: semantic loss $\mathcal{L}_{\text{semantic}}$, binary loss $\sum_{i=1}^2 \mathcal{L}_{\text{binary}}^i$, and edge loss $\sum_{i=1}^2 \mathcal{L}_{\text{edge}}^i$. Corresponding to the three loss functions, we set three weights α_1 , α_2 , α_3 to control the tradeoff. In this experiment, we set α_1 as 1, and separately change the α_2 , α_3 for model training. The result is shown in Fig. 12. It can be observed that max accuracy is achieved at $\alpha_2 = 0.2$, and $\alpha_3 = 0.2$. Meanwhile, we find that increasing α_2 and α_3 beyond 0.2 results in a slight decrease in accuracy. This suggests that giving more

TABLE VIII
ABLATION STUDY OF THE MODEL WITH DIFFERENT COMPOUND MODE OF SAM AND CAM

Method	LEVIR-CD(%)					
	P	R	F1	IoU	Kappa	OA
CAM to SAM	92.72	90.77	91.73	84.73	91.30	99.16
SAM to CAM	91.79	89.57	90.67	82.93	90.26	99.23
SAM plus CAM	92.07	89.93	90.99	83.46	90.60	99.25

Bold indicate the highest score.

weight to binary and edge losses may lead to overfitting or model instability. Conversely, decreasing α_2 and α_3 below 0.2 also leads to a decrease in accuracy, indicating the importance of balancing the contributions of semantic, binary, and edge losses in the overall optimization process.

4) *Ablation Study of Different Compound Mode of SAM and CAM*: In our EDPM module, we utilize two attention modules, the SAM and the CAM, to enhance the sparse hierarchical representation created through various dilation parallel convolution. These modules can be combined in two serial ways and one parallel way. In EDPM, a serial approach is employed, first using CAM followed by SAM. The mathematical expressions for this can be referenced in (6) and (7). Two other combinations of modules are expressed as follows:

$$F_{\text{sc}} = \mathcal{M}_c(\mathcal{M}_s(F) * F) * \mathcal{M}_s(F) * F \quad (33)$$

$$F_{\text{parallel}} = \mathcal{M}_s(F) * F + \mathcal{M}_c(F) * F \quad (34)$$

here, $\mathcal{M}_c(\cdot)$ and $\mathcal{M}_s(\cdot)$ represent CAM and SAM, respectively. F denotes the input feature, F_{sc} is the output feature of the serial combined mode (SAM followed by CAM), and F_{parallel} is the output feature of the parallel combined mode. We evaluated the other two combination modes on the LEVIR-CD dataset [21] and present the final accuracy in Table VIII. It can be observed that the original configuration of MLCNet, using CAM first followed by SAM, achieves the best performance. Since we have already captured constructed sparse representation features through the different dilation rate parallel convolution operations, using CAM first to guide the model to learn “what” is important is probably more suitable than using SAM first to tell the model “where” to focus. After the model learns “what” we want it to focus on, SAM can better refine the attention of features on spatial dimension. About the parallel mode, when spatial and channel attention are applied in parallel, the combined effect might dilute the individual contributions of each attention mechanism. The additive approach might lead to an averaging effect, where the specific benefits of focusing on important channels and spatial regions are not fully realized, leading to a slight drop in accuracy [62].

V. CONCLUSION

Our study focuses on the development of a change detection network for high-resolution remote sensing imagery that effectively leverages the different strengths aspect of features at different levels. To achieve this, we have designed distinct operations for different level features. Two different fusion modules are utilized to merge features from two different temporal phases at the same level, each module tailored to enhance the advantage aspect and decrease the influence of the disadvantage aspect of corresponding level features. Besides, three different supervision strategies for low level, high level, and final output features are designed to optimize the network. Our experimental findings suggest that our proposed approach surpasses the current SOTA methods. Comprehensive ablation experiments have been conducted, affirming the efficacy of our proposed multitask level-specific constraint strategy. While our method demonstrates effectiveness in applying different fusion methods to low-level and high-level features, the rigid partitioning of multilevel features remains a limitation. Future research could explore the development of a more automated module, akin to LSTM's gate control mechanism, to adaptively select focus directions for different feature levels. This would maximize the utilization of each level's strengths.

REFERENCES

- [1] A. Singh, "Review article digital change detection techniques using remotely-sensed data," *Int. J. Remote Sens.*, vol. 10, no. 6, pp. 989–1003, 1989.
- [2] D. Wen, X. Huang, F. Bovolo, J. Li, and J. A. Benediktsson, "Change detection from very-high-spatial-resolution optical remote sensing images: Methods, applications, and future directions," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 4, pp. 68–101, Dec. 2021.
- [3] C. Wu, H. Chen, B. Du, and L. Zhang, "Unsupervised change detection in multitemporal VHR images based on deep kernel PCA convolutional mapping network," *IEEE Trans. Cybern.*, vol. 52, no. 11, pp. 12084–12098, Nov. 2022.
- [4] Z. Li, W. He, M. Cheng, J. Hu, G. Yang, and H. Zhang, "Sinolc-1: The first 1 m resolution national-scale land-cover map of China created with a deep learning framework and open-access data," *Earth Syst. Sci. Data*, vol. 15, no. 11, pp. 4749–4780, 2023.
- [5] Z. Li, H. Zhang, F. Lu, R. Xue, G. Yang, and L. Zhang, "Breaking the resolution barrier: A low-to-high network for large-scale high-resolution land-cover mapping using low-resolution labels," *ISPRS J. Photogrammetry Remote Sens.*, vol. 192, pp. 244–267, 2022.
- [6] M. S. Rana and S. Sarkar, "Prediction of urban expansion by using land cover change detection approach," *Heliyon*, vol. 7, no. 11, 2021.
- [7] M. Pirmazar, K. Ostad-Ali-Askari, and S. Eslamian, "Change detection of urban land use and urban expansion using GIS and RS, case study: Zanjan province, Iran," *Int. J. Constructive Res. Civil Eng.*, vol. 4, no. 10, pp. 2454–8693, 2018.
- [8] D. Lu, P. Mausel, E. Brondizio, and E. Moran, "Change detection techniques," *Int. J. Remote Sens.*, vol. 25, no. 12, pp. 2365–2401, 2004.
- [9] L. J. Jansen and A. Di Gregorio, "Parametric land cover and land-use classifications as tools for environmental change detection," *Agriculture, Ecosyst. Environ.*, vol. 91, no. 1-3, pp. 89–100, 2002.
- [10] Z. Zheng, Y. Zhong, J. Wang, A. Ma, and L. Zhang, "Building damage assessment for rapid disaster response with a deep object-based semantic change detection framework: From natural disasters to man-made disasters," *Remote Sens. Environ.*, vol. 265, 2021, Art. no. 112636.
- [11] Z. Li, C. Tang, X. Li, W. Xie, K. Sun, and X. Zhu, "Towards accurate and reliable change detection of remote sensing images via knowledge review and online uncertainty estimation," 2023, *arXiv:2305.19513*.
- [12] Z. Li et al., "The outcome of the 2021 IEEE GRSS data fusion contest—track MSD: Multitemporal semantic change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1643–1655, 2022.
- [13] R. D. Johnson and E. S. Kasischke, "Change vector analysis: A technique for the multispectral monitoring of land cover and condition," *Int. J. Remote Sens.*, vol. 19, no. 3, pp. 411–426, 1998.
- [14] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 2, no. 1-3, pp. 37–52, 1987.
- [15] 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.
- [16] 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.
- [17] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Interv.*, 2015, pp. 234–241.
- [18] R. C. Daudt, B. Le Saux, and A. Boulch, "Fully convolutional siamese networks for change detection," in *Proc. 25th IEEE Int. Conf. Image Process.*, 2018, pp. 4063–4067.
- [19] 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, 2022, Art. no. 8007805.
- [20] Q. Guo, J. Zhang, S. Zhu, C. Zhong, and Y. Zhang, "Deep multiscale Siamese network with parallel convolutional structure and self-attention for change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5406512.
- [21] 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.
- [22] Z. Li et al., "Lightweight remote sensing change detection with progressive feature aggregation and supervised attention," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5602812.
- [23] X. Zhao et al., "Fractional Fourier image transformer for multimodal remote sensing data classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 2, pp. 2314–2326, Feb. 2022.
- [24] 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.
- [25] W. G. C. Bandara and V. M. Patel, "A transformer-based siamese network for change detection," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 207–210.
- [26] J. Ma, J. Duan, X. Tang, X. Zhang, and L. Jiao, "EATDer: Edge-assisted adaptive transformer detector for remote sensing change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2023, Art. no. 5602015.
- [27] P. Zhu, H. Xu, and X. Luo, "Mdaformer: Multi-level difference aggregation transformer for change detection of VHR optical imagery," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 118, 2023, Art. no. 103256.
- [28] M. Liu, Z. Chai, H. Deng, and R. Liu, "A CNN-transformer network with multiscale context aggregation for fine-grained cropland change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4297–4306, 2022.
- [29] 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.
- [30] M. Yin, Z. Chen, and C. Zhang, "A CNN-transformer network combining CBAM for change detection in high-resolution remote sensing images," *Remote Sens.*, vol. 15, no. 9, 2023, Art. no. 2406.
- [31] 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.
- [32] J. Pan et al., "Mapsnet: Multi-level feature constraint and fusion network for change detection," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 108, 2022, Art. no. 102676.
- [33] F. Ghazouani, I. R. Farah, and B. Solaiman, "A multi-level semantic scene interpretation strategy for change interpretation in remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8775–8795, Nov. 2019.
- [34] Y. Feng, J. Jiang, H. Xu, and J. Zheng, "Change detection on remote sensing images using dual-branch multilevel intertemporal network," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4401015.
- [35] Y. Xiang, X. Tian, Y. Xu, X. Guan, and Z. Chen, "EGMT-CD: Edge-guided multimodal transformers change detection from satellite and aerial images," *Remote Sens.*, vol. 16, no. 1, p. 86, 2023, doi: [10.3390/rs16010086](https://doi.org/10.3390/rs16010086).
- [36] X. Wang et al., "Double u-net (w-net): A change detection network with two heads for remote sensing imagery," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 122, 2023, Art. no. 103456.

- [37] L. Xia, J. Chen, J. Luo, J. Zhang, D. Yang, and Z. Shen, "Building change detection based on an edge-guided convolutional neural network combined with a transformer," *Remote Sens.*, vol. 14, no. 18, 2022, Art. no. 4524.
- [38] T. Wang et al., "Detect globally, refine locally: A novel approach to saliency detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3127–3135.
- [39] J. Li, W. He, W. Cao, L. Zhang, and H. Zhang, "UANet: An uncertainty-aware network for building extraction from remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5608513.
- [40] Y. Zhan, K. Fu, M. Yan, X. Sun, H. Wang, and X. Qiu, "Change detection based on deep Siamese convolutional network for optical aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 10, pp. 1845–1849, Oct. 2017.
- [41] J. Liu, M. Gong, K. Qin, and P. Zhang, "A deep convolutional coupling network for change detection based on heterogeneous optical and radar images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 3, pp. 545–559, Mar. 2018.
- [42] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet : A nested u-net architecture for medical image segmentation," in *Proc. Deep Learn. Med. Image Anal. Multimodal Learn. Clin. Decis. Support, 4th Int. Workshop, DLMIA, 8th Int. Workshop, ML-CDS 2018, Held Conjunction*, 2018, pp. 3–11.
- [43] J. Lei, Y. Gu, W. Xie, Y. Li, and Q. Du, "Boundary extraction constrained siamese network for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5621613.
- [44] Y. Chen, W. Lai, W. He, X.-L. Zhao, and J. Zeng, "Hyperspectral compressive snapshot reconstruction via coupled low-rank subspace representation and self-supervised deep network," *IEEE Trans. Image Process.*, vol. 33, pp. 926–941, 2024.
- [45] C. Zhang et al., "A deeply supervised image fusion network for change detection in high resolution bi-temporal remote sensing images," *ISPRS J. Photogrammetry Remote Sens.*, vol. 166, pp. 183–200, 2020.
- [46] H. Zhang, M. Lin, G. Yang, and L. Zhang, "ESCNet: An end-to-end superpixel-enhanced change detection network for very-high-resolution remote sensing images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 1, pp. 28–42, Jan. 2023.
- [47] M. Wang, X. Li, K. Tan, J. Mango, C. Pan, and D. Zhang, "Position-aware graph-cnn fusion network: An integrated approach combining geospatial information and graph attention network for multi-class change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 4402016.
- [48] H. Zheng et al., "HFA-Net: High frequency attention siamese network for building change detection in VHR remote sensing images," *Pattern Recognit.*, vol. 129, 2022, Art. no. 108717.
- [49] X. Wang et al., "A high-resolution feature difference attention network for the application of building change detection," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 112, 2022, Art. no. 102950.
- [50] 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.
- [51] W. He et al., "Non-local meets global: An iterative paradigm for hyperspectral image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 4, pp. 2089–2107, Apr. 2022.
- [52] M. Chen et al., "A full-scale connected CNN–transformer network for remote sensing image change detection," *Remote Sens.*, vol. 15, no. 22, 2023, Art. no. 5383.
- [53] W. Liu, Y. Lin, W. Liu, Y. Yu, and J. Li, "An attention-based multiscale transformer network for remote sensing image change detection," *ISPRS J. Photogrammetry Remote Sens.*, vol. 202, pp. 599–609, 2023.
- [54] S. Fang, K. Li, and Z. Li, "Changer: Feature interaction is what you need for change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5610111.
- [55] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 770–778.
- [56] Y. Zhang, J. Liu, W. Yang, and Z. Guo, "Image super-resolution based on structure-modulated sparse representation," *IEEE Trans. Image Process.*, vol. 24, no. 9, pp. 2797–2810, Sep. 2015.
- [57] C. Han, C. Wu, H. Guo, M. Hu, and H. Chen, "HANet: A hierarchical attention network for change detection with bitemporal very-high-resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 3867–3878, 2023.
- [58] M. Berman, A. R. Triki, and M. B. Blaschko, "The Lovász-Softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4413–4421.
- [59] L. Shen et al., "S2looking: A satellite side-looking dataset for building change detection," *Remote Sens.*, vol. 13, no. 24, 2021, Art. no. 5094.
- [60] M. Fuentes Reyes et al., "A 2D/3D multimodal data simulation approach with applications on urban semantic segmentation, building extraction and change detection," *ISPRS J. Photogrammetry Remote Sens.*, vol. 205, pp. 74–97, 2023.
- [61] D. Wang, J. Zhang, B. Du, G.-S. Xia, and D. Tao, "An empirical study of remote sensing pretraining," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5608020.
- [62] 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.



Taoyuan Liu received the B.S. degree in engineering of surveying and mapping from the School of Geodesy and Geomatics, Wuhan University, Wuhan, China, in 2021. He is currently working toward the M.A. degree in photogrammetry and remote sensing with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University.

His research interests include deep learning, remote sensing change detection, disaster assessment.



Jiepan Li (Graduate Student Member, IEEE) received the B.S. degree in measurement and control technology from the School of Electronic Information, Wuhan University, Wuhan, China, in 2020. He is currently working toward the Ph.D. degree in photogrammetry and remote sensing with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University.

His research interests include deep learning, building extraction, change detection, salient object detection, and camouflaged object detection.



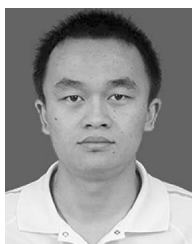
Weinan Cao (Student Member, IEEE) received the B.S. degree from Northeastern University, Shenyang, China, in 2020. He is currently working toward the M.A. degree in photogrammetry and remote sensing with the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan, China.

His main research interests include hyperspectral anomaly detection and unsupervised object detection.



Minghao Tang received the B.S. degree in geological engineering from China University of Geosciences, Wuhan, China, in 2016 and the Ph.D. degree in mining engineering from Chongqing University, Chongqing, China, in 2021.

He is currently with the Zhejiang Academy of Emergency Management Science, Beijing, China. His research interests include disaster prevention, mitigation, response, and emergency management. His research interests include road network extraction, semantic segmentation, and deep learning.



Guangyi Yang received the Ph.D. degree in electronic engineering from Wuhan University, Wuhan, China, in 2018.

He is currently a Senior Engineer with the School of Electronic Information, Wuhan University. His research interests include machine vision and image processing