STMNet: Single-Temporal Mask-Based Network for Self-Supervised Hyperspectral Change Detection

Tianyuan Zhou, *Student Member, IEEE*, Fulin Luo, *Senior Member, IEEE*, Chuan Fu, *Member, IEEE*, Tan Guo, *Member, IEEE*, Xiaopan Wang, Bo Du, *Senior Member, IEEE*, and Xinbo Gao, *Fellow, IEEE*

***Abstract*— Multitemporal hyperspectral images (HSIs) have been widely applied in change detection (CD) of different land covers for their rich spectral features and image details. However, alignment and labeling pairs of bitemporal HSIs are labor-intensive. In this article, we propose a single-temporal mask-based network (STMNet) for self-supervised HSI CD from a new perspective of detecting masks as changes. STMNet implements self-supervised by treating artificially constructed masks attached to single-temporal HSI as changed regions. To this end, we design a multiscale mask change simulation (MMCS) strategy to generate pseudo-second-temporal HSI closer to the real case. Meanwhile, a global–local feature aggregation network is proposed to enhance long-distance and local spatial–spectral feature extraction. To the best of our knowledge, this is the first work in the field of HSI CD that uses single-temporal HSIs and eliminates the need for labeling and pairing samples, alleviating the problem of difficult multitemporal HSI annotation. The visual and quantitative experimental results on three HSI datasets show that the proposed STMNet outperforms the compared state-of-the-art methods for HSI CD. Codes are available at https://github.com/Zhoutya/ChangeDetection-STMNet.**

***Index Terms*— Change detection (CD), hyperspectral image (HSI), mask, multiscale feature, single temporal.**

I. Introduction

CHANGE detection (CD) detects changes by observing remote sensing images of the same area at different times and is widely used in urban planning [1], environmental monitoring [2], agricultural surveys [3], and disaster assessment [4]. Numerous satellites provide abundant hyperspectral images (HSIs) for studying surface changes but also face great bitemporal labeling challenges. Therefore, it is important to develop effective algorithms to understand the variations in multitemporal HSIs and reduce the dependence on labeling [5].

Based on multitemporal HSIs, many classic CD methods have been developed by using spectral information to construct a certain algebraic operation [6], such as image difference, image ratio, image regression, absolute distance, and change vector analysis (CVA) [7]. Other transformation-based methods project HSIs into a low-dimensional feature space that reveals the changed properties, which mainly includes principal component analysis (PCA) [8], independent component analysis (ICA) [9], multivariate alteration detection (MAD) [10], and iterative slow feature analysis (ISFA) [11]. These CD methods are often based on the spectral differences between different temporal HSIs, which cannot fully exploit the inherent characteristics of complex HSIs [12].

In recent years, the development of deep learning techniques has brought new solutions to improve the efficiency and accuracy of CD. Initially, deep learning methods highly rely on supervised information for learning, and the main representative methods are ReCNN [13], BCNN [14], MSDFFN [15], CAST [16], and so on. The quality of labeled samples is closely related to the final detection effect for supervised learning, but the acquisition and labeling of samples are time-consuming and labor-intensive, forcing the emergence of a growing number of models that do not need labeled samples [17], [18]. Self-supervised learning (SSL) aims to improve the feature extraction ability of the model by designing proxy tasks to mine the data's representational properties as supervisory information for unlabeled data [19]. Introducing the SSL strategy to CD can alleviate the dependence on labeled data and make more effective use of existing unlabeled multitemporal HSI data [20].

Existing SSL HSI CD methods can be broadly classified into two categories: one generates high-confidence pseudolabels to serve as supervisory information, and the other constructs supervisory information between features based on contrastive learning. Pseudolabeling-based SSL methods can quickly obtain pseudolabels using existing traditional CD methods. For example, Li et al. [21] used structural similarity (SSIM) and CVA to generate plausible labels to

Received 3 September 2024; revised 20 November 2024; accepted 16 December 2024. Date of publication 30 December 2024; date of current version 22 January 2025. This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant 2024CDJYXTD-009; in part by the National Natural Science Foundation of China under Grant 62371076, Grant 62201109, and Grant 62071340; in part by the New Chongqing Youth Innovative Talents Project under Grant CSTB2024NSCQ-QCXMX0071; in part by the Natural Science Foundation of Chongqing under Grant CSTB2022NSCQ-MSX0452 and Grant CSTB2024NSCQ-MSX0393; and in part by Chongqing Performance Incentive of Research Institutions Program under Grant CSTB2023JXJL-YFX0036. *(Corresponding author: Fulin Luo.)*

Tianyuan Zhou, Fulin Luo, and Chuan Fu are with the College of Computer Science, Chongqing University, Chongqing 400044, China (e-mail: zhoutianyuan1016@163.com; luoflyn@163.com; fuchuan@cqu.edu.cn).

Tan Guo is with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: guot@cqupt.edu.cn).

Xiaopan Wang is with Chongqing Geomatics and Remote Sensing Application Center, Chongqing 401147, China (e-mail: wangxpcq@163.com).

Bo Du is with the School of Computer Science, Wuhan University, Wuhan 430072, China (e-mail: dubo@whu.edu.cn).

Xinbo Gao is with Chongqing Key Laboratory of Image Cognition, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: gaoxb@cqupt.edu.cn).

Digital Object Identifier 10.1109/TGRS.2024.3523541

guide training. Ou et al. [22] constructed a self-supervised comparative pretraining model and trained the downstream CD network using high-confidence pseudolabeled samples. Wu and Chen [23] alternated training between the classification branch and the correction branch to correct the initial pseudolabels by confident learning. Hu et al. [24] proposed a binary change-guided network named BCG-Net; it guides the unmixing process from the perspective of CD and iteratively optimizes the pseudolabels obtained from CD. Zhan et al. [25] constructed a superpixel-guided SSL network (S3Net); it classifies the results of the hyperpixel segmentation according to three overlap types, obtaining region-level labels for the changes in bitemporal phases. However, pseudolabeling-based methods are limited by the accuracy of the initial pseudolabels, and incorrect pseudolabels can mislead the training of the model.

With the widespread use of contrastive learning, to avoid the limitations of the initial pseudolabeling, some contrastive SSL methods train networks by establishing similarity constraints between different features. Hu et al. [26] constructed a contrast learning network named HyperNet by constraining the consistency pairs of invariant pixels between the bitemporal images to learn invariant features. Zhang et al. [27] designed a 3-D depthwise separable autoencoders (DSConvAEs), which trains the network by minimizing the loss of reconstructed feature vectors at the pixel level and the patch level. Huang et al. [28] proposed a contrastive two-domain residual attention network named TRAMNet which designed a random data augmentation pool and a soft contrastive loss function. Jian et al. [29] presented an uncertainty-aware graph SSL (UA-GSSL) for HSI CD, and it constructed node- and edge-level graph contrastive learning to achieve self-supervised representation. However, contrastive SSL methods require the construction of a large number of negative sample pairs during training; some methods still require a small number of labeled samples for downstream fine-tuning. Moreover, both of the above SSL methods require preprocessing the paired images from the same geographic location during training.

When labeling real HSIs, not only operations such as radiometric correction and geometric image alignment need to be performed but also manual field visits are required to ensure the accuracy of labeling. When crop growth in the same area needs to be monitored for a long period, multiple time series of images need to be aligned and labeled at the same time, which is very time-consuming and labor-intensive. With the rapid development of remote sensing imaging technology, we can monitor the Earth around the clock. Every day, we have access to a large amount of unprocessed unpaired remote sensing image data, which provides a new research scenario for CD. In addition, when real-time processing in satellite orbit is required, the labeling and fine-tuning of new images is also very tedious due to the limited arithmetic power, which will greatly hinder the level of automation and intelligence.

In recent years, in the field of high spatial resolution (HSR) remote sensing image CD, some work based on single-temporal images has attempted to address the image pairing issues. Zheng et al. [30] cleverly bypassed the problem of collecting bitemporal samples by using pseudo-non-paired single-temporal interimage variations as a supervisory signal. This method is named ChangeStar and utilizes labeled single-temporal images and samples from different geographic regions to form nonpaired temporal sample pairs. Chen et al. [31] proposed a single-temporal CD framework based on intraimage and interimage patch exchange (I3PE), and it generates pseudo-bitemporal image pairs and corresponding change labels by swapping patches. Then, a small number of samples were used for fine-tuning in downstream tasks.

Same as the HSRs, unpaired unlabeled single temporary HSIs are easier and less costly to acquire than paired labeled multitemporal HSIs. Especially when facing multiple-temporal series CD application scenarios, utilizing single-temporal HSIs is a more efficient solution. However, the above single-temporal-based methods still require partially labeled samples to provide supervisory information and do not completely get rid of the reliance on labeled samples. To reduce human and material resources more efficiently, we would like to implement an algorithm that utilizes only single-temporal phase images in the training phase, without labeling samples, and without fine-tuning to obtain an effective change detector.

Motivated by these observations and insights, we propose a single-temporal mask-based network (STMNet) for self-supervised hyperspectral CD, as shown in Fig. 1. According to the best of our knowledge, this is the first work in the field of HSI CD to detect change based on a single-temporal HSI. STMNet can simulate the changed region using the multiscale mask to learn effective changed features from single-temporal images and get an efficient change detector. The implementation of the STMNet process consists of three parts: multiscale mask change simulation (MMCS) strategy, global–local feature aggregation codecs, and self-supervised training. First, a multiscale masking strategy based on the original spectra is designed to simulate the pseudo-second-temporary changed image. A global–local multiscale feature extraction aggregation network is designed for detecting changes from both single-temporal HSI and masked HSI. The main contributions of this article include the following.

1) To simulate pseudo-second-temporary HSI realistically, we propose the MMCS strategy. MMCS strategy leverages original spectral information with multiscale masks to enable adaptive learning of features across different scales.
2) To extract discriminative features, we propose a global–local feature aggregation network to enhance long-distance feature interaction and local spatial–spectral feature aggregation.
3) To solve the difficult dual-time sample collection problem, we propose an end-to-end STMNet that achieves CD based on single-temporary HSI by constructing change masks, offering a novel approach for HSI CD in self-supervised scenarios. The experimental results on three HSI datasets show its significant performance.

The rest of the article is organized as follows. Section II introduces the details of the proposed STMNet. Section III presents the experiments. In the end, Section IV draws some conclusions of this article and suggestions for future work.

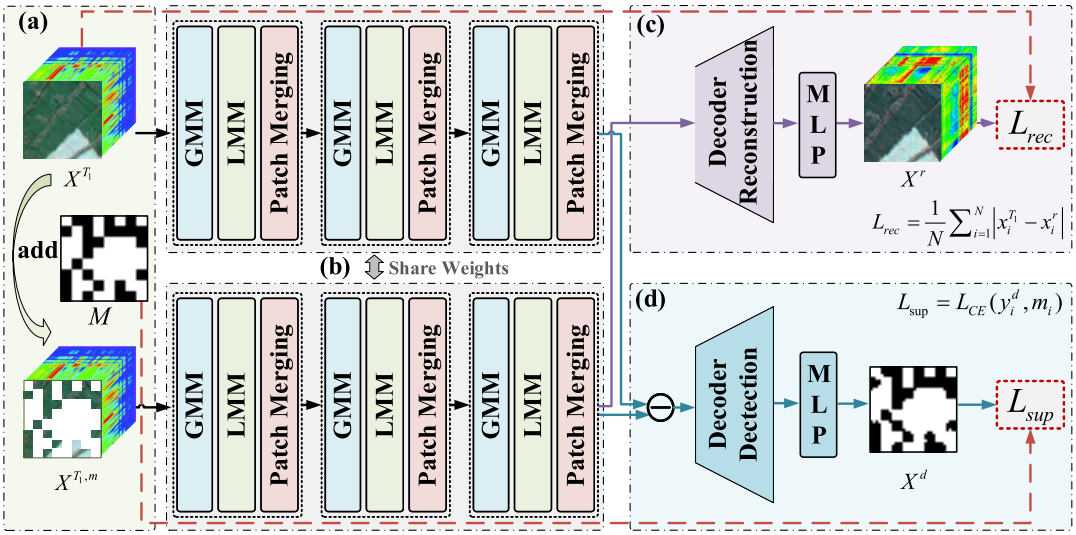


Fig. 1. Structure of the proposed STMNet for HSI CD. (a) MMCS strategy, where a mask is added to the single-temporal HSI to generate the pseudo-second-temporary HSI; the obtained two HSIs are fed into (b): the global–local feature aggregation encoder for feature extracting, which embeds a GMM and an LMM to fully extract the spatial–spectral information; the pseudo-second-temporary branch outputs are fed into (c): reconstruction decoder to obtain the reconstructed feature map; and the multilevel difference features of the two branch outputs are fed into (d): detection decoder to obtain the final CD results.

II. Proposed Method

In this section, we introduce the proposed STMNet for the self-supervised HSI CD task in Fig. 1, which is composed of mask construction, feature extraction, and SSL. The single-temporal HSI patches and the corresponding HSI patches after adding the mask are passed through the global–local feature aggregation network to get the reconstruction feature map and differential feature map. In the following, we will explain each module of the network in detail.

A. Overall Architecture

First, let $X^{T_1} \in {}^{C\times H\times W}$ and $X^{T_2} \in {}^{C\times H\times W}$ represent bitemporal HSIs of the same size, where H, W, and C denote the height, width, and number of spectral bands, respectively. A sliding window sampling of the original image with size P yields a series of patch samples as $x_i^{T_1} \in {}^{C\times P\times P}, i = 1, 2, \ldots, N$, where N denotes the number of samples. Assuming that the generated mask is $m_i \in {}^{C\times P\times P}, i = 1, 2, \ldots, N$, the mask is added to the given single-temporal HSI and obtaining masked HSI $x_i^{T_1,m} \in {}^{C\times P\times P}$. In our model, the masked HSI is considered as another temporal HSI, and CD is achieved by detecting the changed regions (i.e., the areas that have been masked) between the bitemporal HSIs. With the designed multiscale masking strategy and global–local feature aggregation network, the extraction of changed regions in general scenes can be achieved with certain generalization performance.

The encoder–decoder network [32] is widely used in CD tasks due to its excellent feature extraction ability. In the encoder stage of STMNet, we construct two encoders with shared weights for bitemporal HSIs. As shown in Fig. 1, the single-temporal HSI patches and the masked HSI patches are fed to the upper and lower temporal encoder branches to obtain rich multiscale features. Subsequently, the different feature maps are obtained by subtracting the same scale feature maps from the dual-time feature maps. These multiscale difference features are then connected to the decoder via skipping connections for achieving cross-layer connectivity of features and reducing the loss of details.

The encoding phase consists of three downsampling units, each of which contains a global mixing module (GMM), a local mixing module (LMM), and a patch merging operation. GMM and LMM are used to extract and aggregate global local information from feature maps for adequate feature extraction. The encoder stage of the bitemporal branches can be summarized as follows:

$$E_l^{T_1} = \left(f_c{}^{3\times3}\left(\mathrm{LMM}\left(\mathrm{GMM}\left(E_{l-1}^{T_1}\right)\right)\right)\right), l = 1, 2, 3$$
$$E_l^{T_1,m} = \left(f_c{}^{3\times3}\left(\mathrm{LMM}\left(\mathrm{GMM}\left(E_{l-1}^{T_1,m}\right)\right)\right)\right), l = 1, 2, 3 \quad (1)$$

where $E_l^{T_1}$ and $E_l^{T_1,m}$ represent the feature maps generated by the lth downsampling layer for the input patch and $f_c{}^{3\times3}$ represents a convolution operation with a kernel size of 3×3. LMM$(\cdot)$ and GMM$(\cdot)$ denote the LMM and the GMM. All convolution layers are followed by a batch normalization layer and a leaky rectified linear unit (LeakyReLU) layer.

Next, the obtained feature maps of the two branches are subjected to a differential operation to obtain the differential feature map, and the feature map is subjected to a multilayer upsampling detection decoder to restore the feature map to the original input size. The decoder stage can be also summarized as follows:

$$D_l = f_c{}^{1\times1}\left[f_{\mathrm{dc}}{}^{3\times3}(D_{l-1}); \left(E_{l-1}^{T_1} - E_{l-1}^{T_1,m}\right)\right], l = 1, 2, 3 \quad (2)$$

where D_l represents the output feature map after the lth deconvolution operation, $f_{\mathrm{dc}}{}^{3\times3}$ represents a deconvolution operation with a kernel size of 3×3, and $f_c{}^{1\times1}$ represents a convolution operation with a kernel size of 1×1.

[;] represents the stacking operation along the channel dimension, which achieves the skip connections of feature maps.

At the same time, we put the output of the masked HSI branch into the reconstruction decoder to obtain a reconstruction of the original single-temporal HSI map. The decoder stage can be also summarized as follows:

$$R_l = f_c^{1\times1}\left[f_{\mathrm{dc}}^{3\times3}(R_{l-1}); E_{l-1}^{T_1}\right], \quad l = 1, 2, 3 \tag{3}$$

where R_l represents the reconstruction feature map after the lth deconvolution operation. By incorporating skip connections between multilayer downsampling and upsampling paths, we can mitigate the loss of fine-grained details.

After that, we project the obtained feature maps to another space with multilayer perceptron (MLP) that does not share weights

$$X^d = M^d(D_3),\ X^r = M^r(R_3) \tag{4}$$

where $M^d(\cdot)$ and $M^r(\cdot)$ represent the MLP projectors. Ultimately, the reconstruction map X^r and the detection map X^d are constrained with the initial single-temporal HSI X^{T_1} and the designed mask M, respectively.

B. Multiscale Mask Change Simulation Strategy

Current applications of masking in SSL mainly follow the classical masked autoencoder model such as MAE [33] and SimMIM [34]. Such models train the encoder to gain feature extraction capability by reconstructing the masked part, and after pretraining, the encoder is migrated to a downstream task for fine-tuning. When applied in the field of HSI CD, there is a difference between the mask pretrained model that focuses on reconstructing the masked part and the change detection (CD) task that needs to detect the changed part. This leads to difficulty in extracting effective features related to the changes in the downstream task by the pretrained feature extractor and difficulty in fine-tuning the model. Therefore, we change our thinking by directly shifting the goal of the network from reconstructing the masked part to detecting the masked part, bypassing downstream fine-tuning, and achieving CD without labeled samples.

To obtain the HSI close to the real second temporary, we propose the MMCS strategy. Specifically, we utilize multiscale masks to simulate change regions. Different features have different scale characteristics; thus, we set up mask base units of sizes 2, 4, and 8. The changed and unchanged regions are often not uniformly distributed, so we set different mask ratios covering 20%–80%. To adapt to crop features at different scales, different mask block sizes and different mask ratios are set, and these independent multiscale masks are randomly mixed with equal probability to obtain the final mask m. Under the 32 × 32 block size, according to the 50% mask ratio, the different masks with the mask unit to 2, 4, and 8, respectively, are shown in Fig. 2.

Visualization of change simulation strategies is shown in Fig. 3. First, to be closer to the real acquired HSI, we consider the masked part as a changing region and the unmasked part as an unchanging region, and add random noise to simulate the sensor's sampling error. Second, we extract the center position

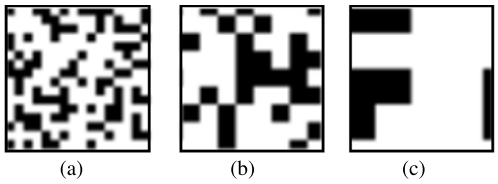


Fig. 2. Multiscale masks generated in the 32 × 32 patch. The mask unit size is (a) masksize=2, (b) masksize=4, and (c) masksize=8, respectively.

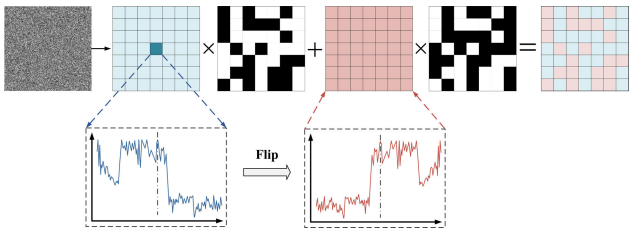


Fig. 3. MMCS strategy. The spectrum at the center pixel in the original patch is flipped horizontally as a change spectrum, the region to be masked is replaced with the change spectrum, and the rest of the regions are added with random noise to simulate imaging differences.

of a spectral band from the original single-temporary patch and perform a lateral flip operation on it as the spectrum of the changing pixel point. As a result of the lateral flip operation, the obtained spectrum will be different from the spectrum of any other pixel in that patch. Unlike random initialization or zero initialization, filling the masked region with this spectrum can simulate the real change sample. This process can be represented by the following equation:

$$x_i^{T_1,m} = x_i^{T_1,n}(1 - m_i) + x_i^{T_1,s} m_i \tag{5}$$

where $(1 - m_i)$ represents the unmasked regions, $x_i^{T_1,m}$ represents the HSI after adding mask, $x_i^{T_1,n}$ representing the HSI after adding Gaussian noise, and $x_i^{T_1,s}$ represents the spectral value after the flip.

C. Global–Local Feature Aggregation Modules

In the encoding stage, to achieve effective aggregation of the spatial–spectral information and obtain discriminative differential features, we embedded GMM and LMM to implement long-range information interaction and local information aggregation, respectively.

1) Global Mixing Module: To improve the interaction capability between long-range information, we propose GMM to realize token interaction in the row and column directions. The GMM makes the spatial distances of pixel points, which are originally far apart, close to each other by slicing and reorganizing the feature maps in different directions of row and column, and the specific structure is shown in Fig. 4.

Specifically, we group and rearrange the feature maps in H and W dimensions, apply convolution for feature extraction, later restore the features to their original shapes, and connect them to the input by skip connection. First, given input features $x \in \mathbb{R}^{c\times h\times w}$, the module divides the features into k blocks in the channel dimension and connects them along the column dimension and get $x \in \mathbb{R}^{\frac{c}{k}\times h\times kw}$. Apply 3 × 3 convolution to this feature map, and then, reduce it to the original dimensions.

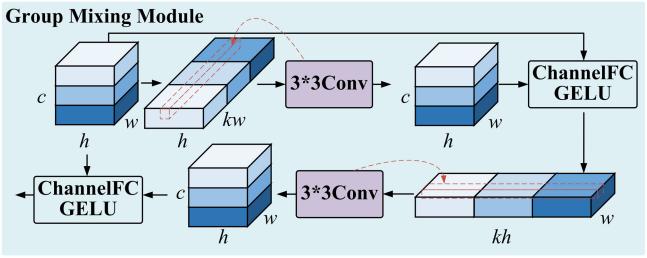


Fig. 4. Structure of the proposed GMM. The GMM slices and reorganizes the feature map along rows and columns, respectively, allowing tokens to interact and capturing long-range dependencies.

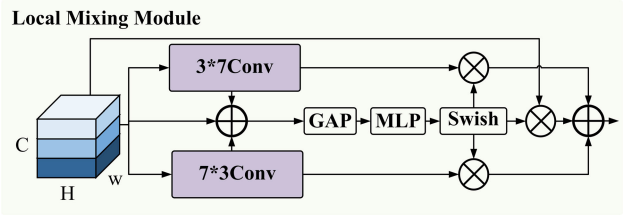


Fig. 5. Structure of the proposed local mixing module. The LMM utilizes convolution kernels of different sizes along the length and width to mine local features and then design an attention mechanism to fuse the spatial spectrum features.

Second, the reduced feature maps are fused with the initial feature maps by channel FC and GELU function, and the features are then sliced and joined together along the row dimension and get $x \in \mathbb{R}^{\frac{c}{k} \times kh \times w}$. The same 3×3 convolution is performed and reduced to the original dimension. The application of 3×3 convolution makes the information that is originally far apart in the patch interact.

2) Local Mixing Module: Due to the special 3-D structure of HSI, its spatial and spectral dimensions contain rich information. The idea of the attention mechanism is to learn a set of weighting coefficients autonomously through the network and dynamically emphasize regions of interest with the weights. We can introduce the attention mechanism to adaptively filter out redundant features and focus on the change information in the feature map. To further fuse the optimized features from both spatial and spectral dimensions and maintain the relevance of the HSI data structure, we propose a local mixing module called LMM. The structure of LMM is shown in Fig. 5.

First, apply 3×7 and 7×3 convolution operations to the input feature map x to extract local features in different directions

$$x_{\text{add}} = x + f_c{}^{3\times7}(x) + f_c{}^{7\times3}(x). \tag{6}$$

The extracted spatial features in different directions are summed with the initial feature maps, and the channel attention is applied to these fused feature maps. Specifically, the global average pooling (GAP) is first used to compress the spatial dimensions of the input feature maps, then the MLP is used to compress and extract the features, and the Swish function is used for activation

$$\text{Att}(x) = \text{Swish}(\text{MLP}(\text{GAP}(x_{\text{add}}))) \tag{7}$$

where $\text{Swish}(x) = x \cdot \text{Sigmoid}(x)$; unlike the commonly used sigmoid activation function, the Swish function has a constant derivative greater than 0. Its smoothness plays an important role in optimization and generalization.

The final output of LMM can be described as follows:

$$\text{LMM}(x) = x_{\text{add}} \cdot \text{Att}(x). \tag{8}$$

By multiplying the feature maps with the shared channel attention weights, the obtained feature maps highlight the useful regions and suppress the useless regions. With the designed LMM module, the local features can be aggregated efficiently, and the fusion of spatial spectrum attention can be achieved.

D. Loss Function

After the decoders, we obtain the reconstruction maps x_i^r and detection maps x_i^d, respectively. To implement SSL, we construct two loss functions to constrain the training of the network using the initial single-temporal HSI $x_i^{T_1}$ and the generated mask m_i.

First, the probability estimate obtained from the fully connected layers can be used to predict the final labels of the input patches. The final CD results can be described as follows:

$$y_i = \text{softmax}\left(\text{fc}\left(x_i^d\right)\right) \tag{9}$$

where y_i is the predicted probability result and fc denotes the full connection layers to extract the features and reduce dimension.

CD task can be considered as a binary classification task, and each pixel of bitemporary HSIs is divided into two categories, i.e., change and unchanged. Therefore, we employ the designed multiscale mask in Section II-B to supervise the CD results. We use the cross-entropy loss function, and the loss function is calculated as follows:

$$L_{\text{sup}} = -\frac{1}{N}\sum_{i=1}^{N}\left(m_i \log y_i + (1 - m_i)\log(1 - y_i)\right) \tag{10}$$

where n denotes the number of samples and m_i is the mask label of the given sample.

To also give the network the ability to learn real HSIs, we designed the reconstruction loss for the output of the masked HSI, and the reconstruction loss function is calculated as follows:

$$L_{\text{rec}} = L1\left(X^{T_1}, X^r\right) = \frac{1}{N}\sum_{i=1}^{N}|x_i^{T_1} - x_i^r|. \tag{11}$$

The total loss function consists of the above loss functions as follows:

$$L_{\text{total}} = L_{\text{sup}} + L_{\text{rec}}. \tag{12}$$

In the training stage, the model is optimized by minimizing the loss function using gradient backpropagation. In the testing stage, the output of the detection decoder is used for testing. The pseudocode is given in Algorithm 1.

Algorithm 1 Pseudocode for the STMNet

Input: Bi-temporary HSIs $x_i^{T_1}$, $x_i^{T_2}$; Encoder $E(\cdot)$, Reconstruction Decoder $D^r(\cdot)$, Detection Decoder $D^d(\cdot)$, MLP projector $M^r(\cdot)$ and $M^d(\cdot)$, classifier $fc^d(\cdot)$; batch size B, epochs T, Samples N.

Output: Binary change detection map.

// Initialize training dataset
$D^{tra} = \{x_i^{T_1}, i = 1, 2, \ldots, N\}$;
for each $t \in [1, T]$
  Split and shuffle D into $\frac{|D^{tra}|}{B}$ batches;
  for each $b \in [1, B]$
    Get multi-scale masks m_i;
    Get pseudo second-temporary HSI $x_i^{T_1,m}$ by Eq. (5);
    Obtain multi-scale features by Eq. (1);
    Obtain the detection map x_i^d by Eq. (2) and Eq. (4);
    Obtain the reconstructed x_i^r by Eq. (3) and Eq. (4);
    Compute overall loss by Eq. (12);
    Update the parameters of overall network;
  end for
end for
// Initialize testing dataset
$D^{te} = \{(x_i^{T_1}, x_i^{T_2}), i = 1, 2, \ldots, N\}$;
Split and shuffle D^{te} into $\frac{|D^{te}|}{B}$ batches;
for each $b \in [1, B]$
    Obtain multi-scale features by Eq. (1);
    Obtain the detection map x_i^d by Eq. (2) and Eq. (4);
    Get the final binary change map y_i via $fc^d(\cdot)$;
end for

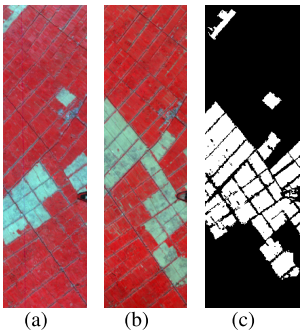
III. Experiment Results and Analysis

In this section, we first describe the HSI-CD datasets and the evaluation metrics to validate the effectiveness of the model. Then, we briefly describe the corresponding comparison algorithms and specific experimental details. Then, a series of comparison experiments are provided to validate the model effects, as well as ablation experiments to verify the effectiveness of each module.

A. Datasets and Evaluation Measures

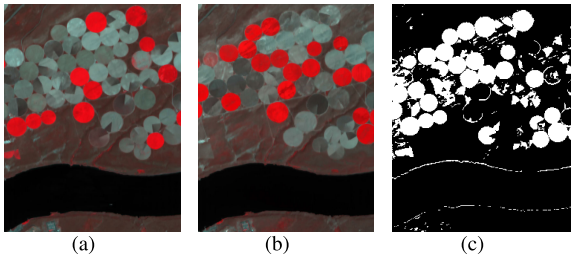
1) Datasets: The first dataset, named "Farmland," belongs to a farmland near the city of Yancheng, Jiangsu province, China, which was acquired by Earth Observing-1 (EO-1) Hyperion sensor on May 3, 2006, and April 23, 2007, respectively. The dataset has 242 bands in the range of 0.4-2.5 m with a spatial resolution of 30 m, as shown in Fig. 6. After removing noise and water absorption bands, it contains 155 spectral bands for experiments, and the spatial size of each image is 450 × 140 pixels. The main change areas are farmland.

The second dataset, named "Hermiston," as shown in Fig. 7, belongs to an irrigated farmland from the Hermiston City area in the Unites States, which was acquired in 2013 and 2014. This dataset was obtained by the Hyperion sensor mounted on the EO-1 satellite. The spatial size of each image is 307 × 241 pixels including 154 spectral bands after eliminating noise. The main change is farmland cover.



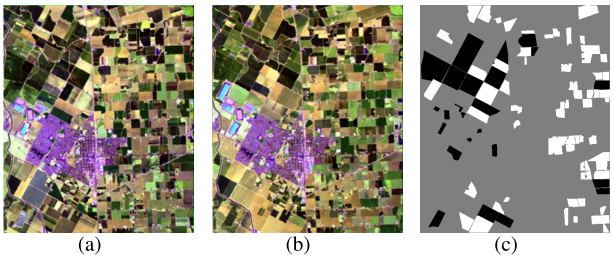
(a) (b) (c)

Fig. 6. Farmland dataset. (a) Image acquired on May 3, 2006. (b) Image acquired on April 23, 2007. (c) Ground truth.



(a) (b) (c)

Fig. 7. Hermiston dataset. (a) Image acquired in 2013. (b) Image acquired in 2014. (c) Ground truth.



(a) (b) (c)

Fig. 8. Bay dataset. (a) Image acquired in 2013. (b) Image acquired in 2015. (c) Ground truth.

The third dataset, named "Bay," as shown in Fig. 8, was taken in 2013 and 2015 with the AVIRIS sensor surrounding the city of Patterson (California). The Bay dataset has a large spatial size of 600 × 500 pixels and 224 spectral bands. The main change areas are covered by farmlands and buildings. Note that there are a large number of unknown regions, and only the labeled changed and unchanged areas are adopted for training and assessment.

2) Evaluation Measures: To better quantify the performance of the proposed method, we mainly used the overall accuracy (OA) and kappa coefficient (KC) as metrics, precision (Pr), recall (Re), and $F1$-score ($F1$), were introduced as an auxiliary evaluation. When the proportion of samples from different categories is very unbalanced, the categories with large proportions often have a large influence on OA to evaluate the effectiveness of the model. Thus, we introduced Pr, Re, and their harmonic mean $F1$ as synthetic evaluation.

The metrics are defined as follows:

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

$$\text{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{14}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{15}$$

$$F1 = \frac{2\text{PR}}{P + R} \tag{16}$$

$$\text{Kappa} = \frac{\text{OA} - p_e}{1 - p_e} \tag{17}$$

$$p_e = \frac{(\text{TP} \times \text{FN}) + (\text{TP} \times \text{FP}) + (\text{TN} \times \text{FN}) + (\text{TN} \times \text{FP})}{N^2} \tag{18}$$

where true positive (TP) indicates the number of pixels correctly classified as changed regions, true negative (TN) denotes the number of pixels correctly classified as unchanged regions, false positive (FP) represents the number of pixels misclassified as changed regions, and false negative (FN) is the number of pixels misclassified as unchanged regions. The larger value of these evaluation metrics indicates better detection performance. To show the experimental results more prominently, TP, TN, FP, and FN are shown in white, red, green, and black in the visualization maps, respectively.

B. Compared Methods and Experimental Details

To evaluate the performance of the proposed architecture, we further compared our method with some excellent CD methods, including MSCD [35], HyperNet [26], S3Net [25], BCG-Net [24], DSConvAEs [27], SNTS [36], and UA-GSSL [29]. MSCD trains a network in a self-supervised fashion by using deep clustering and contrastive learning. HyperNet constructed a contrast learning without negative sample pairs to learn invariant features from bitemporary images. S3Net uses super-pixel guidance to generate image objects and then SSL of object-level spatial change features. BCG-Net combines unmixing with detecting change to eliminate cumulative error through iterative optimization. DSConvAE autonomously learns spatial-spectral features through autoencoders and constrains the learning process through reconstruction loss. SNTS alternatively and iteratively performs change-aware spectral unmixing and subpixel-guided clustering for change recognition. UA-GSSL achieves CD by graph-based node- and edge-level contrastive learning.

In addition, we compare our STMNet with other unsupervised, supervised, and semisupervised methods, including CVA [7], PCAKM [8], ISFA [11], ReCNN [13], BCNN [14], MSDFFN [15], D2AGCN [37], CSA-Net [38], EMS-Net [39], GlobalMind [40], MCS4CD [41], DLIEG [42], and DCENet [43]. Among them, CVA, PCAKM, and ISFA are classic CD algorithms. CVA is the most commonly used traditional method, which can detect changes by changing intensity and change direction. PCAKM uses the PCA method to project the original data into a new lower-dimensional feature space, and the CD is achieved by k-means clustering. ISFA can extract slow-varying features from time series. ReCNN, BCNN, and MSDFFN are fully supervised methods; here, we follow the experiments in [43] to use 0.2% of the samples. D2AGCN, CSA-Net, EMS-Net, GlobalMind, MCS4CD, DLIEG, and DCENet are semisupervised methods that use smaller sample sizes compared with fully supervised methods, and here, we compare the accuracies by quoting directly from the original article.

In our network, comprehensively considering the complexity of calculation and spatial–spectral information, we chose the input patch size as 32 × 32. Our network was trained and tested on an NVIDIA GeForce 3090 GPU with 24-GB memory using the PyTorch [44] framework. For the comparison methods with published codes, we reproduced the codes of these comparison algorithms based on the original article. For the comparison methods with unpublished codes, we directly compare them with the experimental results from the original article.

In experiments, we selected 20% unlabeled samples from the datasets as the training samples and the rest as the testing samples. We selected the training samples by using one as a random seed. In the stage of training, we used the SGD optimizer with a weight decay of $5e^{-3}$. The initial learning rate was designed to be $1e^{-4}$ and decayed by a factor of 0.1 at every 10 epochs. The number of total epochs was 50; the batch size of training samples was set to 128. In the stage of testing, real bitemporary HSIs are fed into the network model, which undergoes the global–local feature aggregation encoder and the detection decoder to obtain an output binary CD map.

C. Experimental Results

1) Experimental Results on the Farmland Dataset: Table I shows the results of each model on the Farmland dataset. Supervised learning methods tend to have better accuracy due to the guidance of labeled samples compared with traditional CVA and PCAKM methods. However, supervised learning methods can be constrained by the sample size and lack effective utilization of unlabeled samples. Their performance also deteriorates when limited labeled samples are used. For example, the accuracy of BCNN and MSDFFN decreases to 94.30% and 95.57% with 0.2% samples. Semisupervised methods such as EMS-Net and DCENet reduce the dependence on labeled samples by enhancing the consistency of invariant features and strengthening the changing features through contrast learning, respectively. They can achieve better accuracy performance but still need the guidance of limited labeled samples. In SSL methods, by designing effective self-supervision strategies, compared with semisupervised methods, they not only can completely get rid of the dependence on labeled samples but also can utilize unlabeled samples more efficiently. The simple CVA algorithm can also obtain an accuracy of 95.25% on the Farmland dataset, which indicates that the dataset itself is easy to segment. In this case, compared with all the methods, the proposed STMNet has the best performance and can even exceed that of ReCNN, a supervised method that uses 20% of the samples for training. This demonstrates the validity of our idea of implementing CD based on a single temporary utilizing a masking strategy.

Fig. 9 shows the visualization results of the experiments on the Farmland datasets. From the visual observations, compared with the other methods, our proposed STMNet presents the fewest FP pixels, thus achieving the best visual performance. The unsupervised CVA and PCAKM methods show a large number of misclassification pixels around the

TABLE I
COMPARISONS BETWEEN STMNET AND VARIOUS METHODS ON THE FARMLAND DATASETS

	Methods	Ratio%	OA%	KC	F1%	Pr%	Re%
Un-Sup	CVA[7]	0	95.25	0.8860	91.97	90.33	93.66
	PCAKM[8]	0	95.14	0.8837	91.82	89.78	93.96
	ISFA[11]	0	95.75	0.8996	/	/	/
Sup	ReCNN[13]	20	97.30	0.9346	95.36	95.72	95.02
	BCNN[14]	0.2	94.30	0.8632	90.36	88.70	92.09
	MSDFFN[15]	0.2	95.57	0.8949	92.65	89.33	**96.24**
Semi-Sup	D2AGCN[37]	1.59	93.74	0.8568	/	/	/
	CSA-Net[38]	0.32	94.22	0.8641	90.56	86.09	95.52
	EMS-Net[39]	0.32	95.60	0.8961	92.76	**97.28**	88.65
	DCENet[43]	0.2	96.56	0.9156	93.97	95.72	92.27
Self-Sup	MSCD[35]	0	87.79	0.7051	87.79	93.38	68.15
	HyperNet[26]	0	89.31	0.7596	83.91	78.86	89.64
	S3Net[25]	0	91.86	0.8132	/	/	/
	BCG-Net[24]	0	94.19	0.8548	89.47	84.89	94.55
	DSConvAEs[27]	0	96.28	0.9095	93.57	93.28	93.86
	SNTS[36]	0	96.58	0.9083	/	/	/
	UA-GSSL[29]	0	96.67	0.9233	/	/	/
	STMNet	0	**97.47**	**0.9387**	**95.66**	96.25	95.07

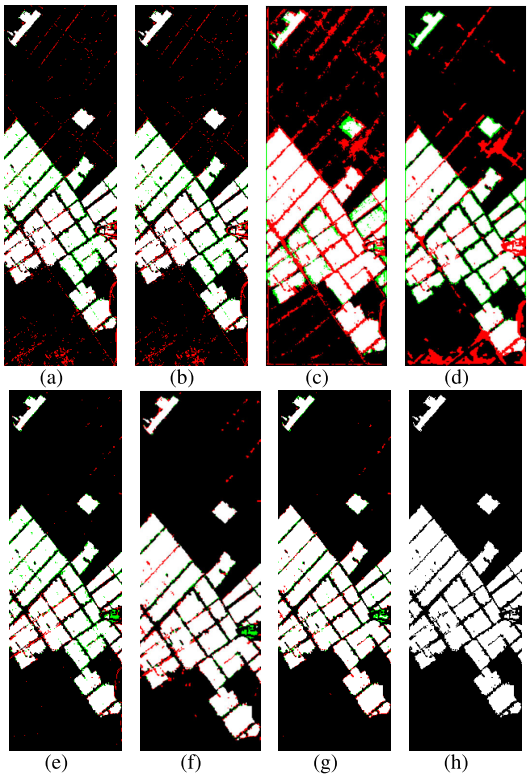


Fig. 9. Visualized results of different methods on the Farmland dataset. (a) CVA. (b) PCAKM. (c) MSCD. (d) HyperNet. (e) BCGNet. (f) UAGSSL. (g) STMNet. (h) Ground truth.

edges of the changed areas and small targets because they lack enough learning of spatial–spectral information. The SSL MSCD and HyperNet exhibit more misclassified pixels in the unchanged areas (black regions) because the MSCD was originally applied to multisensor CD and did not take into account the effect of the rich spectral information of the HSIs, and HyperNet lacks the extraction and learning of long-range connections. The STMNet has fewer misclassification points

TABLE II
COMPARISONS BETWEEN STMNET AND VARIOUS METHODS ON THE HERMISTON DATASETS

	Methods	Ratio%	OA%	KC	F1%	Pr%	Re%
Un-Sup	CVA[7]	0	92.02	0.7416	78.85	97.90	66.01
	PCAKM[8]	0	92.01	0.7413	78.83	97.90	65.98
	ISFA[11]	0	90.23	0.6716	72.62	**98.52**	57.50
Sup	ReCNN[13]	20	89.74	0.6664	72.56	90.97	61.65
	BCNN[14]	0.2	89.85	0.6896	75.26	83.53	68.49
	MSDFFN[15]	0.2	92.86	0.7834	82.80	90.59	76.25
Semi-Sup	GlobalMind[40]	1.35	**95.56**	**0.8781**	**90.71**	96.20	85.82
	MCS4CD[41]	0.27	94.46	0.8458	/	/	/
	DCENet[43]	0.2	94.62	0.8389	87.29	93.4	81.93
Self-Sup	MSCD[35]	0	78.51	0.4788	62.01	51.54	77.82
	HyperNet[26]	0	92.06	0.7613	81.12	87.40	75.69
	BCG-Net[24]	0	94.99	0.8572	88.96	89.52	88.40
	DSConvAEs[27]	0	90.47	0.6905	74.63	62.17	**93.33**
	STMNet	0	95.14	0.8571	88.81	85.64	92.23

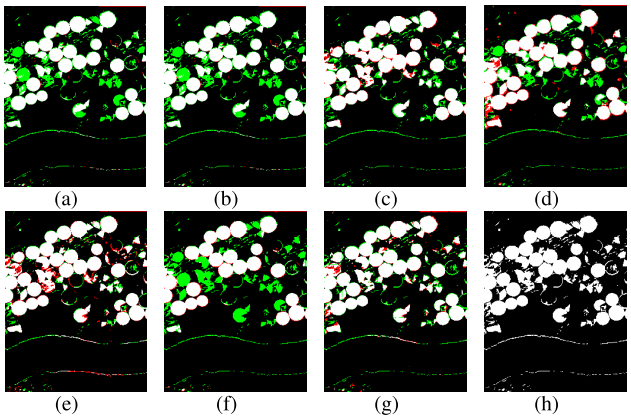


Fig. 10. Visualized results of different methods on the Hermiston dataset. (a) CVA. (b) PCAKM. (c) DCENet. (d) HyperNet. (e) BCGNet. (f) DSConvAEs. (g) STMNet. (h) Ground truth.

and better CD details than BCG-Net and UA-GSSL, mainly shown in the middle right small target areas of the image. This demonstrates the effectiveness of the proposed multiscale masking strategy and global local feature aggregation modules.

2) Experimental Results on the Hermiston Dataset: Table II shows the results of each model on the Hermiston dataset. Compared with the Farmland dataset, the Hermiston dataset has a more complex edge and is more difficult to classify. Compared with unsupervised and supervised models with limited samples, semisupervised models can obtain higher accuracy due to the special network design for limited samples. Our proposed STMNet is lower than the GlobalMind that employs 1.35% samples but still obtains higher accuracy than MCS4CD and DCENet and achieves better performance than some semisupervised methods without any labeled samples. In the SSL model, our method achieves OA exceeding that of BCG-Net and suboptimal values in KC coefficients, while outperforming other SSL methods such as MSCD, HyperNet, and DSConvAEs.

Fig. 10 shows the visualization results of the experiments on the Hermiston datasets. On the Hermiston dataset, the surface targets appear as partially overlapping and dispersed circular

TABLE III
COMPARISONS BETWEEN STMNET AND VARIOUS METHODS ON THE BAY DATASETS

	Methods	Ratio%	OA%	KC	F1%	Pr%	Re%
Un-Sup	CVA[7]	0	82.55	0.6558	83.00	94.16	74.20
	PCAKM[8]	0	82.81	0.6606	83.32	94.05	74.79
	ISFA[11]	0	89.17	0.7848	89.05	96.95	82.34
Sup	BCNN[14]	0.2	92.74	0.8519	93.64	94.24	93.04
	MSDFFN[15]	0.2	95.01	0.8985	95.58	**97.14**	94.08
Semi-Sup	DLIEG[42]	0.5	**96.05**	0.9087	93.76	/	/
	CSA-Net[38]	0.3	95.68	**0.9135**	**95.88**	93.95	**97.89**
	DCENet[43]	0.2	95.58	0.9099	96.10	97.3	94.93
Self-Sup	MSCD[35]	0	75.95	0.5214	73.84	66.11	83.62
	HyperNet[26]	0	90.79	0.8152	91.29	92.24	90.37
	S3Net[25]	0	85.13	0.7045	/	/	/
	SNTS[36]	0	81.27	0.3930	/	/	/
	UA-GSSL[29]	0	91.16	0.8224	/	/	/
	STMNet	0	91.38	0.8256	92.25	89.40	95.28

areas prone to loss of entire areas. Compared with other methods, our proposed STMNet presents the least number of false-positive pixels and, thus, achieves almost optimal visual performance. CVA and PCAKM lost a large number of change regions in their detection results due to their lack of mining complex spectral and spatial information. The semisupervised DCENet has a lot of false-positive pixels in the edge regions, and the self-supervised HyperNet loses some large independently existing regions. BCG-Net and DCENet have fewer FN and FP pixels and show better detail in CD compared with other self-supervised methods.

3) Experimental Results on the Bay Dataset: Table III shows the results of each model on the Bay dataset. This dataset has a large number of unlabeled regions; the traditional CVA and PCAKM methods have lower accuracy performance due to the interference of unlabeled regions. At the same time, due to the large number of unlabeled regions, which makes the labeled regions account for a lower percentage of the whole image, the introduction of labeled samples can guide the model to focus more on learning the spatial and spectral features of the regions to be detected. Combining the CD results of each method, we find that supervision using a small number of samples gives better accuracy on the Bay dataset. Also, supervised and SSL methods have a significant accuracy gap on this dataset. Compared with other SSL methods, our proposed STMNet obtains the best accuracy performance. This suggests that STMNet learns the more discriminative features of variation as much as possible without any labeled samples.

Fig. 11 shows the visualization results of the experiments on the Bay dataset. This dataset contains a large number of unlabeled samples, shown in gray in the figure. CVA and PCAKM have more misclassified pixel points compared with other methods, such as the green areas in the figure, which means misclassifying changed pixels as unchanged pixels. In the SSL methods, MSCD has more misclassified pixel points, and HyperNet and UA-GSSL have fewer misclassified pixel points, but some large regions still face the loss of region edges. Compared with them, our proposed STMNet can detect all large regions and has better visualization performance.

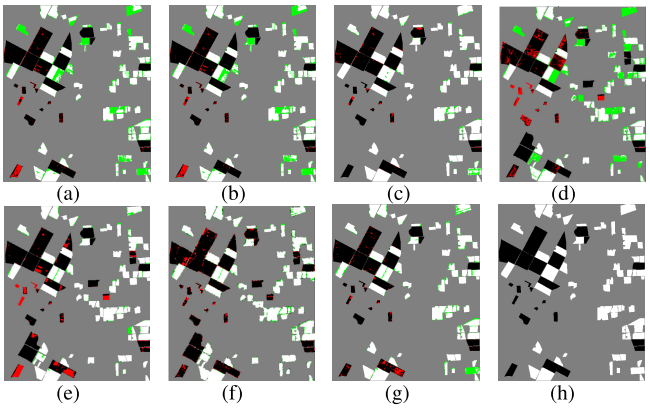


Fig. 11. Visualized results of different methods on the Bay dataset. (a) CVA. (b) PCAKM. (c) DCENet. (d) MSCD. (e) HyperNet. (f) UA-GSSL. (g) STMNet. (h) Ground truth.

TABLE IV
COMPARISONS ABLATION OF EACH MODULE ON THREE DATASETS

Dataset	Model						
	LMM	×	×	✓	✓	✓	✓
	GMM	×	✓	×	✓	✓	✓
	MLP	✓	✓	✓	×	✓	✓
	L_{rec}	✓	✓	✓	✓	×	✓
Farmland	OA%	97.24	97.36	97.34	97.26	97.02	**97.44**
	KC	0.9328	0.9358	0.9352	0.9338	0.9276	**0.9382**
	F1%	95.23	95.44	95.39	95.32	94.85	**95.62**
	Pr%	94.96	95.35	94.86	96.24	94.55	**96.28**
	Re%	95.50	95.52	**95.92**	94.42	95.16	94.97
Hermiston	OA%	94.73	94.84	95.05	94.74	94.91	**95.14**
	KC	0.8414	0.8471	0.8535	0.8404	0.8492	**0.8571**
	F1%	87.45	87.98	88.49	87.33	88.15	**88.81**
	Pr%	81.44	83.87	84.32	80.41	84.08	**85.64**
	Re%	94.41	92.52	93.09	**95.55**	92.64	92.23
Bayarea	OA%	90.34	91.25	90.99	90.38	90.70	**91.38**
	KC	0.8041	0.8225	0.8173	0.8055	0.8124	**0.8256**
	F1%	91.37	92.19	91.96	91.36	91.55	**92.25**
	Pr%	89.16	**89.98**	89.77	88.50	87.83	89.40
	Re%	93.71	94.52	94.26	94.40	**95.60**	95.28

D. Ablation Study

1) Ablation of Each Module: In this article, we proposed some modules to improve the performance of change feature learning. To more clearly show the effectiveness of each module, we conducted the ablation experiments for each module, including LMM, GMM, MLP projector, and L_{rec} on the three datasets. Specifically, we selectively censored the modules to be tested and designed six experiments. Table IV shows the experimental results.

The effectiveness of the added module is demonstrated by analyzing the experimental results on the three datasets where attempts to remove the proposed module result in a decrease in accuracy compared with the full model. Higher accuracies are obtained with the addition of either LMM or GMM alone compared with the complete removal of LMM and GMM, which indicates that both the LMM and GMM modules play a role in aggregating local information to achieve fusion of spatial–spectral features and enable telematic interactions. The model accuracy decreases after removing the MLP projection header, which indicates that adding MLP helps to project the

TABLE V
COMPARISONS ABLATION OF THE MULTISCALE MASKS ON THREE DATASETS

Dataset	Mask	OA%	KC	F1%	Pr%	Re%
Farmland	w/o noise-adding	97.38	0.9361	95.45	94.67	**96.24**
	w/o multi-scale masks	97.27	0.9343	95.36	**96.98**	93.80
	0 Initialisation	96.65	0.9189	94.25	94.82	93.70
	Random Initialisation	96.01	0.9031	93.12	92.97	93.27
	ours	**97.47**	**0.9387**	**95.66**	96.25	95.07
Hermiston	w/o noise-adding	95.10	0.8552	88.69	84.59	**93.22**
	w/o multi-scale masks	95.12	0.8558	88.69	84.96	92.76
	0 Initialisation	94.37	0.8380	87.43	**86.97**	87.90
	Random Initialisation	94.44	0.8397	87.55	86.72	88.39
	ours	**95.14**	**0.8571**	**88.81**	85.64	92.23
Bay	w/o noise-adding	91.31	0.8241	92.21	89.56	95.02
	w/o multi-scale masks	91.25	0.8233	92.08	88.64	**95.80**
	0 Initialisation	90.83	0.8138	91.86	**90.14**	93.65
	Random Initialisation	91.06	0.8190	91.99	89.46	94.68
	ours	**91.38**	**0.8256**	**92.25**	89.40	95.28

features to another space, which facilitates the subsequent reconstruction or CD. The model accuracy decreases when removing the reconstruction loss part, which is because the reconstruction part gives the model the ability to learn features from real HSIs. In addition, on the Farmland dataset, the difference in accuracy due to the addition or subtraction of individual modules is not as pronounced as on the other datasets, which is because the Farmland dataset itself is easy to discriminate the changes, and it is easy to get better results for CD. Overall, on the three datasets, the designed modules can improve CD performance.

2) Ablation of Multiscale Masks: Next, we test the details of the proposed multiscale masking strategy on the three datasets to demonstrate the effectiveness of the proposed strategy. We designed a series of experiments to compare the performance of the model under four scenarios, namely, no spectral additive noise, no multiscale masking, and zero initialization and random initialization for the spectra of the masked regions. Experimental results are shown in Table V. To ensure the fairness of the comparison, we used the same settings in all the experiments.

According to the results, first, compared with not adding noise, there is a slight increase in the accuracy of the model after noise addition to the spectrum, which suggests that noise addition can simulate imaging differences in real situations. Second, the final accuracy is also affected by using only a single-scale mask, which indicates that the designed multiscale mask effectively matches the characteristics of different scales. Third, both zero initialization and random initialization of the spectra make it difficult to achieve the best accuracy, which means that it is crucial to flip the existing spectra to fill the masked regions, which can make the masked image closer to the real HSI of the second temporal phase. In conclusion, the proposed multiscale masking strategy can effectively simulate the variations generated by the real second temporal HSI, which is the key to making the proposed single-temporal-based CD model effective.

TABLE VI
DISCUSSION OF THE MASK RATIO ON THREE DATASETS

Dataset	Ratio	OA%	KC	F1%	Pr%	Re%
Farmland	0.2	94.90	0.8785	91.48	94.27	88.85
	0.4	97.07	0.9288	94.95	95.02	94.87
	0.5	97.38	0.9365	95.50	95.91	95.10
	0.6	**97.47**	**0.9387**	**95.66**	**96.25**	95.07
	0.8	97.35	0.9354	95.41	94.87	**95.94**
Hermiston	0.2	93.90	0.8199	85.87	82.17	89.91
	0.4	95.09	0.8550	88.62	84.88	92.70
	0.5	95.10	0.8558	88.67	84.50	**93.26**
	0.6	**95.14**	**0.8571**	**88.81**	85.64	92.23
	0.8	91.45	0.7743	83.06	**93.05**	75.01
Bay	0.2	87.45	0.7405	89.42	**92.38**	86.65
	0.4	90.47	0.8073	91.43	88.54	94.52
	0.5	90.90	0.8156	91.85	89.35	94.49
	0.6	91.17	0.8222	91.96	87.93	**96.37**
	0.8	**91.38**	**0.8256**	**92.25**	89.40	95.28

E. Discussion

1) Discussion of the Mask Ratio: For different datasets, the shape features of the embedded features are rich and diverse, which requires different optimal masking rates for different datasets. To fully investigate the effect of mask rate size on the detection results, we tested the detection results under different mask rates on three datasets. We set the mask rate size to 0.2, 0.4, 0.5, 0.6, and 0.8, respectively. The OA, KC, and other evaluation metrics are shown in Table VI.

As the masking ratio increases, the network accuracy fluctuates to varying degrees. Based on the experimental results, we selected 0.6, 0.6, and 0.8 as the masking ratio for the Farmland, Hermiston, and Bay datasets, respectively. Observing the pseudocolor images of the bitemporary HSIs with the naked eye, it can be found that the changing areas mainly include the change of the whole field and the change of some fine edges. When the mask ratio is too small, the simulated change areas become very scattered. When the neighboring fields in the real scenario have all changed, that is, a large adjacent area has changed, it is difficult for the model to adapt to this scenario. When the mask ratio is too large, the simulated change areas will be too dense, and when there is a large unchanged area, the model will be easy to mistakenly detect the unchanged areas as change areas.

2) Discussion of the Training Sample Size: To comprehensively investigate the influence of the training sample size on the detection results, we tested the detection results under different training samples on the three datasets. We set the training sample size as 10%, 20%, 30%, 40%, and 50%. We presented the experimental results in a curve line chart. The OAs, KCs, and other evaluation indicators are shown in Fig. 12.

For the Farmland dataset, once the training samples reach 20%, the improvement in model accuracy is very limited by increasing the sample size. For the Hermiston and Bay datasets, continuously increasing the training samples instead

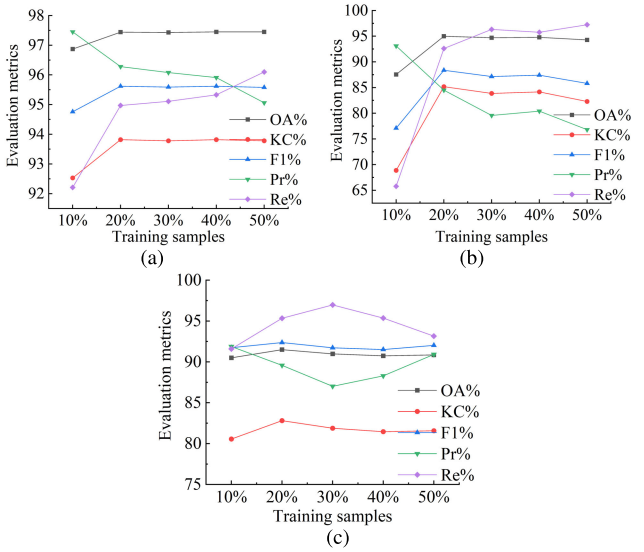


Fig. 12. CD results under different unlabeled sample sizes on the datasets. (a) Farmland dataset. (b) Hermiston dataset. (c) Bay dataset.

TABLE VII
RUNTIME COST OF SSL METHODS ON THE FARMLAND DATASET

Method	PCAKM	HyperNet	BCG-Net	DSConvAEs	STMNet
Runtime(s)	0.21	572.35	1664.43	1142.63	2031.95

leads to a decrease in OA and KC. Based on the experimental results, we selected 20% as a uniform training sample size for the three datasets. Prior experience with supervised learning methods has taught us that in general, the accuracy of the network improves with increasing training samples. Current experiments show that suitable samples help the model learn richer features and improve model performance. However, increasing the sample size does not always improve accuracy. This is because increasing the sample size has less impact on the network when most of the sample types are already covered by training samples.

3) Discussion of the Computational Cost: We tested the computational cost of all the compared methods on the Farmland dataset, as shown in Table VII. It can be seen that the traditional methods have a shorter running time, but the detection accuracy is poor, and it is difficult to detect the details of the changes. Deep learning methods have higher detection accuracy but require more training time. Compared to other algorithms, STMNet needs to spend more time in the training stage due to the large number of unlabeled samples combined with multiple masks for training. In the testing stage, we cut the whole HSI into multiple patches and then tested them. Its testing time is shorter, around 1.43 s. Subsequently, more efficient training strategies will be explored.

IV. CONCLUSION

In this article, we propose an end-to-end framework named STMNet, including a multiscale masking strategy and a global local feature enhancement module to detect the changed regions based on a single temporary HSI. STMNet can detect changes using only a single-temporary HSI, effectively bypassing the difficulty of collecting dual-time corresponding temporary images. Experimental results on three HSI datasets show that the proposed method can produce more accurate CD results using unlabeled samples more efficiently than other comparative methods. In future work, we plan to further develop the unsupervised model to enhance the masking and realize more flexible masking strategies to match rich and diverse feature shapes.

REFERENCES

[1] R. Jaturapitpornchai, M. Matsuoka, N. Kanemoto, S. Kuzuoka, R. Ito, and R. Nakamura, "Newly built construction detection in SAR images using deep learning," *Remote Sens.*, vol. 11, no. 12, p. 1444, Jun. 2019.

[2] S. H. Khan, X. He, F. Porikli, and M. Bennamoun, "Forest change detection in incomplete satellite images with deep neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5407–5423, Sep. 2017.

[3] F. Li, F. Zhou, G. Zhang, J. Xiao, and P. Zeng, "HSAA-CD: A hierarchical semantic aggregation mechanism and attention module for non-agricultural change detection in cultivated land," *Remote Sens.*, vol. 16, no. 8, p. 1372, Apr. 2024.

[4] M. Ji, L. Liu, R. Du, and M. F. Buchroithner, "A comparative study of texture and convolutional neural network features for detecting collapsed buildings after earthquakes using pre- and post-event satellite imagery," *Remote Sens.*, vol. 11, no. 10, p. 1202, 2019.

[5] S. Liu, D. Marinelli, L. Bruzzone, and F. Bovolo, "A review of change detection in multitemporal hyperspectral images: Current techniques, applications, and challenges," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 2, pp. 140–158, Jun. 2019.

[6] M. Hasanlou and S. T. Seydi, "Hyperspectral change detection: An experimental comparative study," *Int. J. Remote Sens.*, vol. 39, no. 20, pp. 7029–7083, Oct. 2018.

[7] F. Bovolo and L. Bruzzone, "A theoretical framework for unsupervised change detection based on change vector analysis in the polar domain," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 1, pp. 218–236, Jan. 2007.

[8] 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.

[9] A. Hyvärinen and E. Oja, "Independent component analysis: Algorithms and applications," *Neural Netw.*, vol. 13, nos. 4–5, pp. 411–430, Jun. 2000.

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

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

[12] A. Shafique, G. Cao, Z. Khan, M. Asad, and M. Aslam, "Deep learning-based change detection in remote sensing images: A review," *Remote Sens.*, vol. 14, no. 4, p. 871, Feb. 2022.

[13] L. Mou, L. Bruzzone, and X. X. Zhu, "Learning spectral–spatial–temporal features via a recurrent convolutional neural network for change detection in multispectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 924–935, Feb. 2019.

[14] Y. Lin, S. Li, L. Fang, and P. Ghamisi, "Multispectral change detection with bilinear convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 10, pp. 1757–1761, Oct. 2020.

[15] F. Luo, T. Zhou, J. Liu, T. Guo, X. Gong, and J. Ren, "Multiscale diff-changed feature fusion network for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5502713.

[16] X. Zhang, S. Tian, G. Wang, X. Tang, J. Feng, and L. Jiao, "CAST: A cascade spectral-aware transformer for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5520714.

[17] F. Luo, Y. Liu, X. Gong, Z. Nan, and T. Guo, "EMVCC: Enhanced multi-view contrastive clustering for hyperspectral images," in *Proc. 32nd ACM Int. Conf. Multimedia*, Oct. 2024, pp. 6288–6296.

[18] G. Wang et al., "Negative deterministic information-based multiple instance learning for weakly supervised object detection and segmentation," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 15, 2024, doi: 10.1109/TNNLS.2024.3395751.

[19] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 44–53, Jan. 2018.

[20] Y. Wang, C. M. Albrecht, N. A. A. Braham, L. Mou, and X. X. Zhu, "Self-supervised learning in remote sensing: A review," 2022, *arXiv:2206.13188*.

[21] Q. Li et al., "Unsupervised hyperspectral image change detection via deep learning self-generated credible labels," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 9012–9024, 2021.

[22] X. Ou, L. Liu, S. Tan, G. Zhang, W. Li, and B. Tu, "A hyperspectral image change detection framework with self-supervised contrastive learning pretrained model," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 7724–7740, 2022.

[23] H. Wu and Z. Chen, "Self-supervised confident learning for hyperspectral image change detection," in *Proc. 12th Workshop Hyperspectral Imag. Signal Process., Evol. Remote Sens. (WHISPERS)*, Sep. 2022, pp. 1–4.

[24] M. Hu, C. Wu, B. Du, and L. Zhang, "Binary change guided hyperspectral multiclass change detection," *IEEE Trans. Image Process.*, vol. 32, pp. 791–806, 2023.

[25] T. Zhan, M. Gong, X. Jiang, and E. Zhang, "S^3Net: Superpixel-guided self-supervised learning network for multitemporal image change detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.

[26] M. Hu, C. Wu, and L. Zhang, "HyperNet: Self-supervised hyperspectral spatial–spectral feature understanding network for hyperspectral change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5543017.

[27] Y. Zhang et al., "Depthwise separable convolutional autoencoders for hyperspectral image change detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.

[28] Y. Huang, L. Zhang, W. Qi, C. Huang, and R. Song, "Contrastive self-supervised two-domain residual attention network with random augmentation pool for hyperspectral change detection," *Remote Sens.*, vol. 15, no. 15, p. 3739, Jul. 2023.

[29] P. Jian, Y. Ou, and K. Chen, "Uncertainty-aware graph self-supervised learning for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5509019.

[30] Z. Zheng, A. Ma, L. Zhang, and Y. Zhong, "Change is everywhere: Single-temporal supervised object change detection in remote sensing imagery," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15173–15182.

[31] H. Chen, J. Song, C. Wu, B. Du, and N. Yokoya, "Exchange means change: An unsupervised single-temporal change detection framework based on intra- and inter-image patch exchange," *ISPRS J. Photogramm. Remote Sens.*, vol. 206, pp. 87–105, Dec. 2023.

[32] Z. Lv, H. Huang, W. Sun, T. Lei, J. A. Benediktsson, and J. Li, "Novel enhanced UNet for change detection using multimodal remote sensing image," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, pp. 1–5, 2023.

[33] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15979–15988.

[34] Z. Xie et al., "SimMIM: A simple framework for masked image modeling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9643–9653.

[35] S. Saha, P. Ebel, and X. X. Zhu, "Self-supervised multisensor change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4405710.

[36] H. Wu and Z. Chen, "Self-supervised change detection with nonlocal tensor train and subpixel signature guidance," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5529617.

[37] J. Qu, Y. Xu, W. Dong, Y. Li, and Q. Du, "Dual-branch difference amplification graph convolutional network for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5519912.

[38] R. Song, W. Ni, W. Cheng, and X. Wang, "CSANet: Cross-temporal interaction symmetric attention network for hyperspectral image change detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[39] M. Hu, C. Wu, and B. Du, "EMS-Net: Efficient multi-temporal self-attention for hyperspectral change detection," 2023, *arXiv:2303.13753*.

[40] M. Hu, C. Wu, and L. Zhang, "GlobalMind: Global multi-head interactive self-attention network for hyperspectral change detection," *ISPRS J. Photogramm. Remote Sens.*, vol. 211, pp. 465–483, May 2024.

[41] L. Liu, D. Hong, L. Ni, and L. Gao, "Multilayer cascade screening strategy for semi-supervised change detection in hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 1926–1940, 2022.

[42] W. Dong, Y. Yang, J. Qu, S. Xiao, and Y. Li, "Local information-enhanced graph-transformer for hyperspectral image change detection with limited training samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5519912.

[43] F. Luo, T. Zhou, J. Liu, T. Guo, X. Gong, and X. Gao, "DCENet: Diff-feature contrast enhancement network for semi-supervised hyperspectral change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5511514.

[44] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, Jan. 2019, pp. 1–11.

Tianyuan Zhou (Student Member, IEEE) received the B.S. degree in electronic information engineering from the University of Shanghai for Science and Technology, Shanghai, China, in 2021. She is currently pursuing the Ph.D. degree with the College of Computer Science, Chongqing University, Chongqing, China.

Her research interests include hyperspectral image change detection, remote sensing image processing, and machine learning.

Fulin Luo (Senior Member, IEEE) received the B.E. degree in mechanical engineering and automation from Southwest Petroleum University, Chengdu, China, in 2011, and the M.E. and Ph.D. degrees in instrument science and technology from Chongqing University, Chongqing, China, in 2013 and 2016, respectively.

He was an Associate Researcher and a Post-Doctoral Researcher with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University, Wuhan, China, from 2017 to 2021. He was a Research Fellow with Nanyang Technological University, Singapore, from 2020 to 2021. He has been a Professor with the College of Computer Science, Chongqing University, since 2022. His research interests include remote sensing processing, computer vision, and biomedical analysis.

Chuan Fu (Member, IEEE), photograph and biography not available at the time of publication.

Tan Guo (Member, IEEE), photograph and biography not available at the time of publication.

Xiaopan Wang, photograph and biography not available at the time of publication.

Bo Du (Senior Member, IEEE), photograph and biography not available at the time of publication.

Xinbo Gao (Fellow, IEEE), photograph and biography not available at the time of publication.