Hyperspectral Marine Oil Spill Detection Network With Enhanced Superpixel Segmentation and Attention Mechanisms

Yuqing Zhang, Zhijing Ye, Chengyong Zheng, Bing Yang, Jiangtao Peng, *Senior Member, IEEE*, and Weiwei Sun, *Senior Member, IEEE*

Abstract— In recent years, marine oil spills have occurred frequently, causing serious damage to the marine ecological environment. Hyperspectral images (HSIs) can provide rich spectral and spatial information, and have broad development prospects in marine oil spill detection. This article proposes a hyperspectral marine oil spill detection network, HMOSDN, that integrates improved superpixel segmentation and a mixed attention mechanism (MAM). First, to deal with the extensive clutter and diverse morphology of marine oil spill areas in HSIs, we propose a new superpixel segmentation algorithm based on improved simple linear iterative clustering (ISLIC), which achieves preliminary extraction of spatial features and reduces spatial noise via a Gaussian filter and a pixel intensity smoothing technique (PIST). Then, to further fuse spectral and spatial features and strengthen the feature mining and utilization of fused information, we design a spectral–spatial feature extraction network with an MAM, MAM-SSFEN, which adds a spectral attention module and a spatial attention module, further improving the performance of the deep feature extraction network for oil spill detection. Experiments on the hyperspectral oil spill database (HOSD) demonstrate that our proposed method, HMOSDN, outperforms several other detection techniques regarding area under the curve (AUC) and recall evaluation metrics.

Index Terms— Feature extraction network, hyperspectral oil spill detection (HOSD), improved simple linear iterative clustering (ISLIC), mixed attention module.

I. INTRODUCTION

WITH the continuous development of the marine transportation industry, marine oil spills have occurred frequently around the world, causing serious damage to the marine environment and marine life [1], [2]. On April 20, 2010, a major oil spill occurred in the Gulf of Mexico, where an oil drilling platform called Deepwater Horizon exploded, leaking approximately 3.2 million barrels of oil and contaminating at least 2500 km^2 of seawater [3], [4]. On April 27, 2021, two cargo ships collided near Qingdao, Shandong, China, resulting in a leakage of approximately 9400 tons of cargo oil. The contaminated area reached 4360 km^2, causing serious damage to the local fishing industry and ecological environment [5]. Therefore, for countries around the world, how to conduct timely and effective monitoring when oil spills occur is an urgent issue [6].

Remote sensing is a long-distance detection technique that uses sensors to collect and process electromagnetic wave information radiated and reflected by targets. Even for oil spills in remote areas, remote sensing technology can achieve monitoring [7]. Common sensors include infrared sensors, visible light sensors, and synthetic aperture radar (SAR) [8], [9]. Infrared and visible light sensors are the earliest remote sensing technologies used for detecting marine oil spills. However, infrared and visible data have many drawbacks, such as susceptibility to adverse weather conditions and low separability between seawater and oil [10]. SAR is a high-resolution imaging radar that is not affected by weather conditions and can achieve all-weather monitoring in detecting marine oil spills [11]. However, SAR also has some drawbacks, such as its reliance on surface roughness changes, long imaging time, high cost, and susceptibility to interference from other natural phenomena, which limits its application in oil spill detection.

Compared to traditional remote sensing technologies, hyperspectral imaging offers superior spectral resolution (spectra) and provides abundant spatial information (image), effectively integrating imaging and spectral data [12]. The 2-D images within HSIs capture the shape and structural features of oil spill areas. In addition, the 1-D spectral data in HSIs reveal the chemical and physical properties of oil [13]. By exploiting the distinct spectral signatures of oil and its interaction with water, HSIs enable highly precise identification of oil spills, particularly when the spectral resolution is very high.

They provide nearly continuous spectral curves, allowing for the detailed depiction of oil's spectral characteristics and facilitating accurate detection of oil spill zones in marine environments [14]. In addition, HSIs facilitate nondestructive, long-distance detection, making them well-suited for complex ocean environments. However, hyperspectral imaging can be affected by weather conditions. Under good weather conditions, hyperspectral imaging can capture high-quality images that combine spectral and spatial information. In addition, due to the extensive and intricate nature of hyperspectral data, effective data processing techniques are essential for subsequent oil spill detection tasks [15]. The existing hyperspectral oil spill detection methods can be roughly divided into two categories: traditional machine learning algorithms and deep learning algorithms [16].

Traditional machine learning techniques can achieve shallow feature extraction. For example, Dongmei et al. [17] proposed a classification algorithm based on wavelet transform, which selects the sensitive bands of different thicknesses of oil spill films by analyzing the spectral curves of different substances and singularity of high-frequency wavelet coefficient curves. Liu et al. [18] proposed a method for selecting reference spectral features and used the adaptive cosine estimator algorithm to detect oil spills. However, these methods only consider spectral information and ignore the rich spatial information in HSIs, resulting in insufficient feature extraction. HSIs have rich spatial information, which can significantly improve the effect of oil spill detection. Such as, Duan et al. [19] proposed a texture-aware total variation method, which uses SVM to achieve oil spill classification. Xu et al. [20] presented an oil spill detection method by using the LBP texture feature and K-means algorithm for oil spill monitoring. However, these methods only extract shallow features of spectral and spatial information, often requiring expert judgment and manual adjustment of parameters to achieve relatively good detection results. This means that current technologies still have certain limitations and rely on manual intervention. In order to further improve the accuracy and efficiency of detection, future research may need to explore deeper feature extraction methods and reduce reliance on manual adjustments to achieve more automated and intelligent hyperspectral oil spill detection.

With the rapid development of deep learning, many models based on deep learning have also been applied to the field of hyperspectral image (HSI) classification and detection. For example, Hu et al. [21] applied 1D-CNN to HSI classification and achieved better classification results compared to SVM, but it still only considers spectral features and ignores spatial features. Ji et al. [22] proposed a domain adaptive and interactive differential attention network for HSI change detection based on a CNN backbone. Zhu et al. [23] proposed a detection method that combines spatial information with a stacked autoencoder network, achieving the classification of marine oil spills. Zhao and Du [24] used 2DCNN to extract spatial features from reduced dimensional images based on PCA. Luo et al. [25] presented a diff-feature contrast enhancement network for semi-supervised HSI change detection using a

limited number of labeled samples. Wang et al. [26] proposed a hyperspectral oil spill detection framework that uses 1-D and 2-D convolutional neural networks to extract spectral–spatial features, then performs feature fusion, and finally inputs them into the classifier for classification. Duan et al. [27] proposed a novel structure-preserved hypergraph convolutional network based on hypergraph convolution and high-order structure preservation for HSI classification. Roy et al. [28] proposed the HybridSN model for HSI classification, which combines 2-D convolution and 3-D convolution. In addition, the high dimensionality and large data volume of HSIs have always been a big challenge. In [29], a multiple feature-based superpixel-level decision fusion (MFSuDF) method was proposed. In order to avoid excessive computational costs, superpixel-guided kernel principal component analysis was designed to reduce the dimensions of HSIs and the noise impact. In [30], a multiscale superpixel-level group-clustering framework (MSGCF) was proposed for hyperspectral band selection. In order to better group and cluster spectral bands, a new superpixel-level distance measure is utilized, which jointly considers spectral–spatial information. Zhao et al. [31] proposed a convolutional transformer network HSI classification method that effectively combines convolution and transformer structures. This also indicates that the transformer architecture has great potential in the field of HSI classification. In [32], a spectral–spatial feature tokenization transformer (SSFTT) method was proposed. This method organically integrated the backbone CNN and transformer structure, effectively improving classification performance. To resolve the challenge of Transformer learning a large number of parameters when processing hyperspectral data, Zhao et al. [33] proposed a lightweight ViT network called groupwise separable convolutional ViT (GSC-ViT). In [34], a multiscale super token transformer (MSSTT) model was proposed for HSI classification, which can reduce the computational overhead and redundancy. Ravikumar et al. [35] proposed a novel method based on the concept of matrix capsules with an Expectation Maximization routing algorithm and the effectiveness of the method was validated on three HSI datasets. Liu et al. [36] proposed an HSI classification method based on the self-supervised learning of spectral masking (SSLSM) and achieved good classification results. Due to the difficulty of labeling HSIs, especially in complex scenes such as the ocean, research on unsupervised HSI classification methods is also very popular. In [37], an unsupervised oil spill detection method based on isolation forest (iForest) was proposed for HSIs. In [38], a superpixel-based and spatially regularized diffusion learning (S2DL) was proposed to process HSIs and demonstrate impressive clustering performance effectively. In [39], a sharpening-aware graph network (SAGN) was developed for HSI change detection. In [40], a fine–coarse–fine progressive graph framework with distribution alignment (FCFDA) was proposed to learn progressive features across multilevel graphs for HSI change detection. In [41], a spectral Kolmogorov-Arnold network (KAN) for HSIs change detection was proposed, which uses a KAN encoder to improve the computational efficiency. In summary,
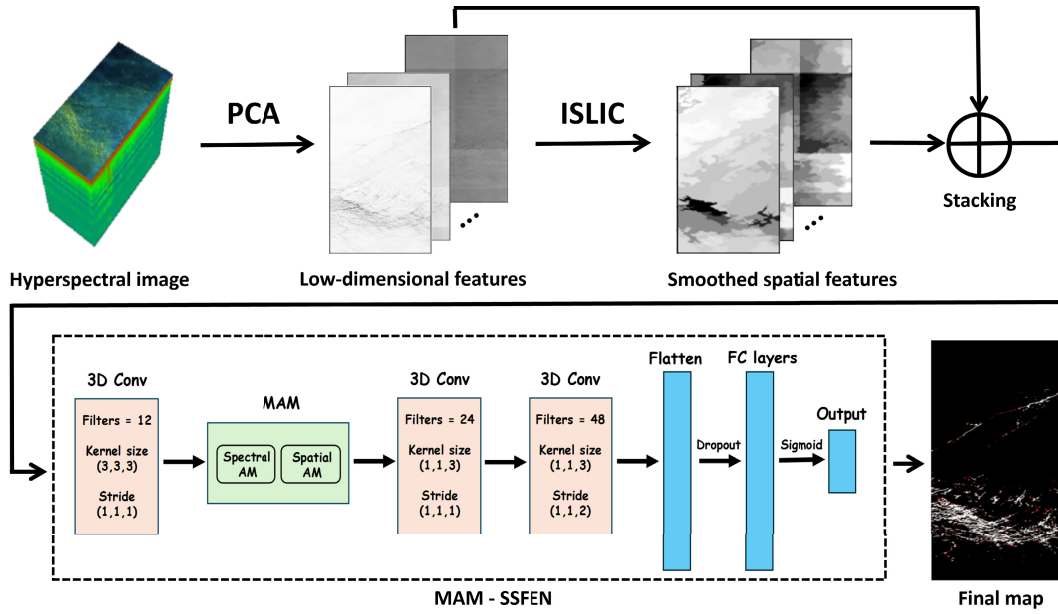
Fig. 1. Flowchart of our proposed marine oil spill detection network HMOSDN.

exploring new deep-learning methods to intensively analyze both spectral and spatial features of HSIs for marine oil spill detection is a burgeoning research area with significant potential for future development in the field of hyperspectral oil spill detection.

Although deep-learning methods can improve classification performance, hyperspectral oil spill detection methods still face some challenges, such as the complex distribution of marine oil spills. This article designs a hyperspectral oil spill detection network HMOSDN that integrates improved superpixel segmentation and a mixed attention mechanism (MAM). Initially, we employ PCA to reduce the dimensionality of the original high-dimensional data, extracting key spectral features. We further refine the SLIC algorithm by implementing Gaussian filters for image smoothing and introducing pixel intensity smoothing to reduce noise in the spatial information of the images. This refined method aids in the preliminary extraction of spatial features from HSIs. Following this, we combine the spectral features condensed by PCA with the spatial features derived from the improved simple linear iterative clustering (ISLIC) algorithm to assemble the spectral–spatial features. These features are then input into the network. An MAM is incorporated following the first convolutional layer to aid the network classification. By embedding both spectral and spatial attention, this mechanism heightens the network's focus on crucial features, thus significantly improving the network's oil spill detection capabilities. The hyperspectral oil spill detection network HMOSDN proposed in this article is presented in Fig. 1. The main contributions are summarized as follows.

1) A new superpixel segmentation algorithm based on ISLIC is proposed through the application of Gaussian filters for smoothing and pixel intensity techniques to curtail spatial noise, optimizing the preliminary extraction of spatial features.

2) A novel feature extraction framework, MAM-SSFEN, is developed by integrating both spectral and spatial data from HSIs. This is achieved through the implementation of an MAM in the network architecture, including both spectral and spatial attention modules. This design enhances the network's ability to focus on relevant features for more accurate oil spill detection.

3) We propose a new oil spill detection model HMOSDN that combines improved superpixel segmentation and an MAM for marine oil spill detection. Compared with other detection methods, it has achieved better detection performance in terms of AUC and recall evaluation metrics.

The rest of this article is organized as follows. Section II provides details of the proposed method, including PCA, ISLIC, SSFEN, and MAM. Section III gives detailed experimental results and analysis. Section IV gives the conclusion.

II. METHODOLOGY

High dimensionality is a hallmark characteristic of HSI data. To efficiently manage and reduce the spectral data's dimensionality, we initially employ PCA to distill key information from the HSI along the spectral dimension, resulting in a reduced set of feature images. Following this, the ISLIC is designed to perform superpixel segmentation on these feature images, enabling the identification and mapping of oil spill areas within the images. Subsequently, the spectral features extracted via PCA are merged with the spatial features obtained through ISLIC. This fused data is then input into the feature extraction framework. Within this framework, a specifically designed MAM—incorporating both spectral and spatial components—is employed to pinpoint more significant features. This refined focus enhances the accurate detection of oil spill areas. This section offers an in-depth discussion of the methods deployed in the proposed detection framework, including PCA, ISLIC, the spectral–spatial feature extraction

network, and the MAM, providing clear insights into their functionality and integration.

A. Superpixel Segmentation Based on Improved SLIC

To extract the spatial features from HSI data, after dimensionality reduction by PCA, the ISLIC method is proposed, which is utilized to perform superpixel segmentation on the feature images and can curtail spatial noise.

1) Principal Components Analysis (PCA): PCA is a classic feature extraction algorithm widely applied in HSIs. The PCA algorithm projects high-dimensional raw data onto a low-dimensional space through multivariate linear transformation to reduce the dimensionality of the data. First, centralize the HSI and obtain a data matrix X. The purpose of centralization is to eliminate the influence of dimensionality on covariance and ensure that each feature has the same importance in subsequent calculations. Then, calculate the covariance matrix C of matrix X, and obtain the eigenvalues and corresponding eigenvectors of C as follows:

$$C = VEV^T \tag{1}$$

where V means the eigenvectors, which represent the main direction of the data, and E means the eigenvalues, which represent the importance of the direction. Finally, the raw data is projected into the space of the eigenvectors, and the first k principal components are selected based on the eigenvalues to construct image features. The first k principal components F are calculated as follows:

$$F = V_k X \tag{2}$$

where V_k means the eigenvector matrix, composed of the eigenvectors corresponding to the first k eigenvalues.

2) ISLIC: HSIs generally contain tens of thousands of pixels, which require significant processing costs. Superpixels are image blocks with similar spectral information and other features, as proposed by Ren and Malik [42]. By grouping pixels based on feature similarity, a large number of adjacent pixels can be replaced by a small number of superpixels, greatly simplifying HSI processing tasks. Nowadays, the superpixel segmentation algorithm has become a key technology in the field of HSI processing.

Simple linear iterative clustering (SLIC) [43] is an image segmentation algorithm proposed in 2012. The SLIC algorithm can generate uniform and neat superpixels, and it is one of the most widely used superpixel segmentation algorithms. Compared to other superpixel segmentation algorithms, the SLIC algorithm is relatively simple, runs fast, and can maintain objects' contours well. The SLIC algorithm first evenly allocates the initial centers of superpixels, also known as seed points, on the image. In practical operation, it is usually necessary to first divide the image into rectangular grids of equal size and then set the center point of each grid as the position of the seed point, which directly affects the size and shape of the generated superpixels. Assuming there are a total of N pixels in the image F_N that need to be divided into K superpixels of the same size, each superpixel has a size of about $\lfloor (N/K) \rfloor$. The initial position of the seed point can

be obtained through uniform sampling, and the distance S between adjacent seed points is calculated as follows:

$$S = \sqrt{\frac{N}{K}}. \tag{3}$$

Then, each pixel is assigned a class label. Specifically, we consider a region around each seed point, calculate the distance of each pixel in the region to the seed point, and assign it to the nearest seed point. The distance used in SLIC combines the color distance and spatial distance. The distance $D(p, q)$ between pixel point p and seed point q is calculated as follows:

$$D_c(p, q) = ||F_p - F_q||_2 \tag{4}$$

$$D_s(p, q) = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2} \tag{5}$$

$$D(p, q) = \sqrt{\left(\frac{D_s(p, q)}{S}\right)^2 + \left(\frac{D_c(p, q)}{m}\right)^2} \tag{6}$$

where $D_c(p, q)$ denotes the feature difference between the pixel point p and the seed point q. F_p and F_q represent the spectral features of the pixel point p and the seed point q from F, respectively. $D_s(p, q)$ denotes the spatial distance between the pixel point p and the seed point q. (x_p, y_p) and (x_q, y_q) represent the coordinates of the pixel point p and the seed point q, respectively. m is a fixed constant, which is used to control the compactness of superpixels. Then, we update the cluster center based on the average position of the assigned pixels. Finally, the above operation is repeated and the cluster center is updated until the cluster center no longer changes.

However, the SLIC algorithm faces some challenges in hyperspectral marine oil spill detection tasks due to various factors inherent to the complexity of such scenarios. First, HSIs provide a rich set of spectral data that may exhibit subtle differences between oil spills and the surrounding water. The traditional SLIC algorithm, primarily designed for RGB images, might not effectively utilize the rich spectral information available. Second, marine environments are complex. Some factors such as waves, varying illumination, and dispersion of oil, complicate the segmentation process. Third, at the boundary of oil spills, pixels may contain mixed spectral signatures of both oil and water, potentially leading to misclassification. Finally, oil spills can vary widely in size, making it difficult for a fixed-size superpixel approach to adapt.

Therefore, this article proposes an ISLIC algorithm based on smoothing processing. First, we use a Gaussian filter (GF) to smooth the images. This helps to reduce noise and smoothens the image, making the segmentation process more robust against texture and noise variations. Gaussian filter is a linear filter based on Gaussian distribution, which can perform weighted averaging on each pixel point. The closer the pixel is to the center pixel, the greater the weight, which is determined by the Gaussian function. The formula for the 2-D Gaussian function is as follows:

$$G(X, Y) = \frac{1}{2\pi\sigma^2} e^{-\frac{X^2+Y^2}{2\sigma^2}} \tag{7}$$

where σ means the standard deviation of the Gaussian function, determining the degree of smoothness. Next, we apply

a Gaussian function to compute the weighted average of all pixels. For a certain pixel point (x_p, y_p), the formula for Gaussian smoothing is as follows:

$$\hat{\boldsymbol{f}}_p = \sum_{(x_q, y_q) \in \boldsymbol{U}_p} G(x_p - x_q, y_p - y_q) \boldsymbol{f}_q \qquad (8)$$

where $\boldsymbol{U}_p$ denotes the neighborhood centered around (x_p, y_p), $\hat{\boldsymbol{f}}_p$ denotes the pixel value of the smoothed image at this point (x_p, y_p), (x_q, y_q) denotes the coordinates of the pixel adjacent to the point (x_p, y_p), and $\boldsymbol{f}_q$ denotes the pixel value of the original image at the point (x_q, y_q). As the weight assigned by the Gaussian filter diminishes with increasing distance from the center, the intensity of the smoothing effect decreases toward the edges. This gradual reduction in smoothing near the periphery helps preserve the image's edge details.

For the original image $\boldsymbol{F}_N$, which contains N pixels, after segmentation, K superpixels are obtained, which satisfy the following formula:

$$\boldsymbol{P}_1 \cup \boldsymbol{P}_2 \cup \cdots \cup \boldsymbol{P}_K = \boldsymbol{F}_N \qquad (9)$$

$$\boldsymbol{P}_i \cap \boldsymbol{P}_j = \emptyset, \forall i, j \in \{1, 2, \ldots, K\}, i \neq j \qquad (10)$$

where $\boldsymbol{P}_i$ means the ith superpixel. Once the superpixels are formed, a pixel intensity smoothing technique (PIST) is applied within each superpixel to average out the pixel intensity variations further, while respecting each superpixel boundary. Although the SLIC algorithm can segment images into superpixels with similar characteristics to reduce the noise of spatial information, there may still be certain intensity differences between pixels due to factors such as lighting and reflection, which are not conducive to subsequent processing. To further reduce the intensity difference between pixels within the same superpixel and achieve deeper denoising, this article replaces the intensity value of each pixel with the average intensity value of the superpixel. In this way, the random noise changes of individual pixels are reduced, achieving smooth processing of pixel intensity. The average intensity M_i of the ith superpixel is calculated as follows:

$$M_i = \frac{1}{n_i} \sum_{j=1}^{n_i} P_{ij}, i = 1, 2, \ldots, K \qquad (11)$$

where n_i means the number of pixels within the ith superpixel, and P_{ij} means the intensity value of the jth pixel within the ith superpixel. After obtaining the average intensity of each superpixel, replace the intensity value of each pixel with the average intensity value of the superpixel it is located in. Although this smoothing process changes the intensity values of pixels, it is performed within each superpixel, which is determined based on spatial information in the image. Therefore, it still preserves important spatial information, which is beneficial for subsequent processing and classification tasks.

This ISLIC, incorporating Gaussian filtering and smooth pixel intensity processing, aims to provide more robust and spatially coherent superpixel segmentation for HSIs under ocean scenes. This proposed method is particularly effective in marine oil spill detection tasks where noise reduction and boundary preservation are crucial.

B. Spectral–Spatial Feature Extraction Network With Mixed Attention Mechanism

To extract the spectral–spatial features from complex hyperspectral data, MAM-SSFEN has been designed and consists of two modules, i.e., SSFEN and MAM. The MAM-SSFEN architecture used in this article is shown in Fig. 1.

1) SSFEN: Compared to 2-D convolution, 3-D convolution adds depth information that can simultaneously extract spectral and spatial features of HSIs, so the computational complexity of 3-D convolution is usually higher than that of 2-D convolution [44]. Feature extraction networks can include multiple convolutional layers, pooling layers, and fully connected layers. In convolutional layers, the feature extraction network uses 3-D convolutional kernels to extract features from input data, and the convolutional layer contains multiple convolutional kernels. The calculation formula for the convolutional layer of the feature extraction network is as follows:

$$v_{ij}^{\text{xyz}} = f\left(\sum_{k=1}^{K} \sum_{w=1}^{W_i} \sum_{h=1}^{H_i} \sum_{b=1}^{B_i} w_{ijk}^{\text{whb}} v_{(i-1)k}^{(x+w)(y+h)(z+b)} + b_{ij}\right) \quad (12)$$

where v_{ij}^{xyz} means the value of position (x, y, z) in the ith layer and jth feature map, f means the activation function, K means the number of feature maps in the previous layer, W_i, H_i, and B_i, respectively, represent the width, height of the convolutional kernel, and spectral depth, w_{ijk}^{whb} represents the value at position (w, h, b) of the convolution kernel connecting the jth convolution kernel in the ith layer and the kth feature map, $v_{(i-1)k}^{(x+w)(y+h)(z+b)}$ means the value of the kth feature map output by the $(i-1)$th layer at position $(x+w, y+h, z+b)$ and b_{ij} means the bias term corresponding to the jth feature map in the ith layer. The pooling layer is usually used after the convolutional layer to reduce the size of the feature map and improve the computational speed of the model. The fully connected layer is used to integrate local features and map them to the final classification result through the classifier of the output layer. In addition, 3DCNN generally adds an activation function after the convolutional layer to increase nonlinearity and further enhance the network's feature learning ability. Common activation functions include sigmoid, ReLU, Tanh, and so on.

Our architecture utilizes 3-D convolutional kernels to simultaneously extract spectral and spatial features. In the network, each pixel is treated as an $n \times n \times f$ volume. Among them, $n \times n$ represents the size of the spatial domain associated with the pixel, and f represents the number of spectral bands. Our proposed network architecture consists of three 3-D convolutional layers, one flattened layer, and two fully connected layers. Basically, the input data is 3-D voxels, and the 3-D convolutional layer is first used to extract spectral–spatial features simultaneously. To produce a pooling effect after each convolution operation, the stride of the last two convolutional layers alternates between 1 and 2 [45]. Next is a flattened layer, which converts the feature map output by the convolutional layer into the vector, achieving the transition from the convolutional layer to the fully connected layer. To prevent overfitting, dropout was added after the flattened layer to increase the robustness of the network. Finally, there

is a fully connected layer, where the size of the final layer is set to the number of categories and the sigmoid is used as the activation function where the sigmoid activation of the ith output unit is as

$$\text{Sigmoid}(\boldsymbol{x}) = \frac{1}{1 + e^{-x}} \qquad (13)$$

where $\boldsymbol{x}$ means the 1-D output vector of the convolutional layer. The loss function we adopt is binary cross-entropy, which calculates the logarithmic difference between the true label and the predicted probability. The binary cross-entropy is adopted as follows:

$$\text{Loss} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right] \qquad (14)$$

where N is the total number of samples, y_i is the ground truth, and $\hat{y}_i$ is the predicted label.

The optimizer we have chosen is Adam, which combines the advantages of AdaGrad and RMSProp algorithms. Adam can adaptively adjust the learning rate of parameters during the training process by calculating the first-order and second-order moments of gradients. In addition, Adam can accurately and quickly minimize the loss function, improving the training performance of deep neural networks, especially on large-scale datasets such as HSI datasets.

The Adam optimizer adjusts the learning rate of each parameter by weighting the average of two indices. The first index-weighted average is the index-weighted average of the gradient, and the second is the index-weighted average of the square of the gradient. The updated rules for the Adam optimizer are as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \qquad (15)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \qquad (16)$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \qquad (17)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \qquad (18)$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \qquad (19)$$

where g_t means gradient of the parameter, β_1 and β_2 are the decay rate, m_t and v_t are estimates of first-order and second-order moments, $\hat{m}_t$ and $\hat{v}_t$ are the first and second moment estimates after deviation correction, θ_t is the parameter that needs to be optimized, θ_{t+1} is the updated parameter, and η means learning rate and ϵ is a tiny constant.

2) MAM: As the complexity of hyperspectral oil spill detection tasks increases, the need for more sophisticated mechanisms to enhance feature extraction becomes apparent. Attention mechanisms have emerged as a powerful solution, allowing networks to dynamically focus on relevant parts of the input data. The MAM designed in this study serves as a lightweight yet effective enhancement that can be embedded into any convolutional architecture, thereby broadening the applicability of attention mechanisms in various domains. The MAM used in this article is added after the first 3-D convolutional layer. This placement is critical, as it allows the MAM to refine the feature maps produced by the convolutional
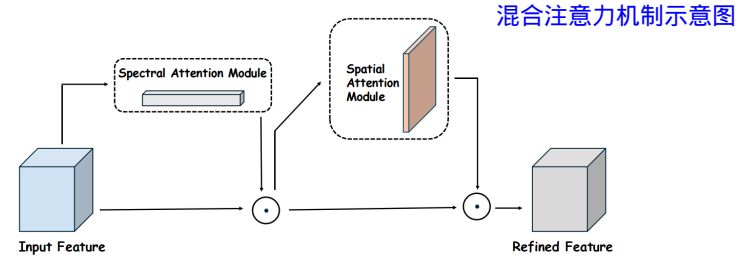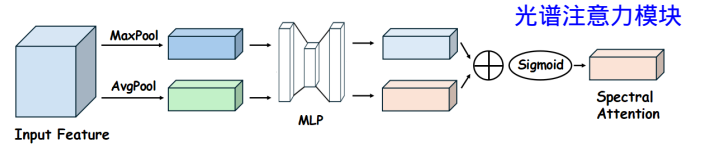

Fig. 2. Overview of MAM.

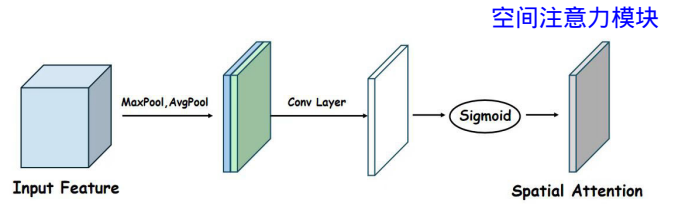
Fig. 3. Diagram of spectral attention module.

Fig. 4. Diagram of spatial attention module.

layers before they are passed on to subsequent layers for further processing.

MAM consists of spectral and spatial attention modules, and the overview of MAM is presented in Fig. 2. MAM utilizes spectral and spatial attention modules to enhance the input feature map, and the resulting enhanced features can be used as inputs for subsequent network layers to improve the feature expression ability of convolutional neural networks. By first applying spectral attention and then spatial attention, the model performs better as suggested in [46].

The diagram of spectral attention modules is presented in Fig. 3. The spectral attention module is before the spatial attention module. For the input feature map, first perform global max pooling and average pooling operations on each spectral band, and input the pooled features into a shared MLP to learn the attention weights for each spectral band. Next, add the two output feature maps and use the sigmoid activation function to limit the attention weights to between 0 and 1, resulting in the final spectral attention $M_{\text{spe}}(F)$. This process can be represented as follows:

$$M_{\text{spe}}(F) = \sigma(\text{MLP}(\text{AvgPool}(F)) + \text{MLP}(\text{MaxPool}(F)))$$
$$(20)$$

where F means the input feature map, and σ means sigmoid activation function. Finally, multiply the spectral attention weights with the original feature map to obtain the attention-weighted spectral feature map F'.

The diagram of spatial attention modules is presented in Fig. 4. The spatial attention module is located after the spectral

attention module. For the obtained spectral feature map F', global max pooling and average pooling are performed in the spectral dimension. Next, the two obtained feature maps are concatenated along the spectral dimension, and a convolutional layer is used to generate spatial attention weights. The weights are limited to 0–1 using the sigmoid activation function to obtain the final spatial attention $M_{spa}(F')$. This process can be represented as follows:

$$M_{\mathrm{spa}}(F') = \sigma\left(g\left(\left[\mathrm{AvgPool}(F'); \mathrm{MaxPool}(F')\right]\right)\right) \quad (21)$$

where g means convolution operation, and σ means sigmoid activation function. Finally, the weights of spatial attention $M_{spa}(F')$ are multiplied with the spectral feature map F' to obtain the final spectral–spatial feature map F''. The complete process can be represented as follows:

$$F' = M_{\mathrm{spe}}(F) \odot F \quad (22)$$
$$F'' = M_{\mathrm{spa}}(F') \odot F' \quad (23)$$

where $\odot$ means element-wise multiplication.

III. EXPERIMENTAL RESULTS AND ANALYSIS

This section provides a detailed description of the dataset used in the experiment, parameter analysis, parameter settings and comparison methods, detection results, and ablation experiments. To demonstrate the advantages of the proposed method, we employed ten comparative methods, including the following.

1) Traditional machine learning methods: Logistic regression, SVM, and random forest.
2) Convolutional neural network-Based Methods: 1DCNN [21], 2DCNN [24], and HybridSN [28].
3) Convolutional transformer network (CTN) [31].
4) An HSI classification method based on deep matrix capsules (DMC) [35].
5) A lightweight ViT network known as groupwise separable convolutional ViT (GSC-ViT) [33].
6) An MSSTT [34].

In order to objectively evaluate the detection performance of different methods, this article adopts two widely used evaluation metrics: area under the curve (AUC) [47] and recall [48].

1) AUC: AUC is the area under the ROC curve, and the larger its value, the better the classification performance, the AUC is defined as

$$\mathrm{AUC} = \int_{\infty}^{-\infty} \mathrm{TPR}(H)\mathrm{FPR}'(H)\mathrm{dH} \quad (24)$$

where $\mathrm{TPR}(H)$ means the true positive rate, which measures the proportion of true positive samples among all positive samples when the threshold is H, and $\mathrm{FPR}(H)$ means the false positive rate, which measures the proportion of false positive samples among all negative samples when the threshold is H.

2) Recall: Recall measures the proportion of true positive samples among all positive samples, which is defined as

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad (25)$$

where TP means true positives and FN means false negatives.

TABLE I
SOME FEATURES OF THE FIRST 8 HSIS IN THE HOSD

Database	Spatial size	Band	Resolution	Fight time
GM1	1200*633	224	7.6	5/17/2010
GM2	1881*693	224	7.6	5/17/2010
GM3	1430*691	224	7.6	5/17/2010
GM4	1700*691	224	7.6	5/17/2010
GM5	2042*673	224	7.6	5/17/2010
GM6	2128*689	224	8.1	5/18/2010
GM7	2302*479	224	3.3	7/09/2010
GM8	1668*550	224	3.3	7/09/2010

A. HOSD Database

The hyperspectral oil spill database (HOSD) was established by Duan et al. [37], and is a public oil spill detection database. HOSD was created based on the 2010 Gulf of Mexico oil spill using the airborne visible/infrared imaging spectrometer (AVIRIS) sensor. HOSD covers a wide range and has a large amount of data, consisting of 18 HSIs. This article selected the first 8 HSIs in HOSD as experimental data. The first 8 HSIs already include almost all types of oil spill distributions, including concentrated oil spills, dispersed oil spills, strip-shaped oil spills, and so on. The main work of this article focuses on the development of effective deep-learning algorithms for the detection of marine oil spill areas. Based on the labeling of the oil spill areas and the seawater areas in the HOSD, our analysis is focused on a binary classification problem—distinguishing between oil spills and no oil spills. Table I lists some indicators of the first 8 HSIs in the HOSD. The different spatial resolutions of images are due to the varying heights of different routes.

For the feasibility analysis of pattern recognition, we plot the spectral reflectance curves of the oil and seawater pixels, as shown in Fig. 5. From the overall trend, the spectral reflectance of oil and seawater shows a rapid increase followed by a rapid decrease and then oscillates within a certain range. Furthermore, it can be seen that the spectral reflectance of oil is significantly higher than that of seawater within the band range of 30–60, 70–80, 90–110, 120–150, and 170–200. There is a significant difference between the two spectral reflectances. By analyzing these spectral reflectance curves, pattern recognition algorithms can be trained to identify oil spills and no oil spills. This information can then be used to develop effective and accurate pattern recognition systems for monitoring and detecting oil spills in marine environments.

B. Parameter Analysis

In line with common practices in deep learning for HSI processing, we employ PCA for dimensionality reduction to derive a condensed set of feature images in our study. We carried out a series of experiments to specifically assess the impact of PCA on the preservation of oil spill feature bands, as depicted in Fig. 6. Our results reveal that as we increase the number of principal components, the detection accuracy improves initially across all datasets and then decreases after reaching a peak. Optimal detection performance, evidenced by the highest AUC and recall scores, is achieved when
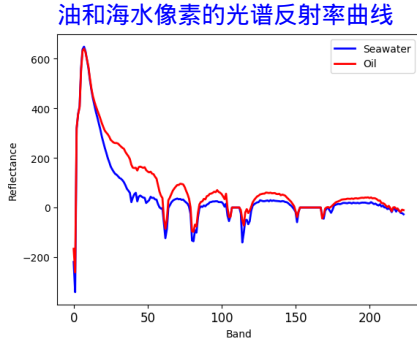
Fig. 5.   Spectral reflectance curves of seawater and oil.
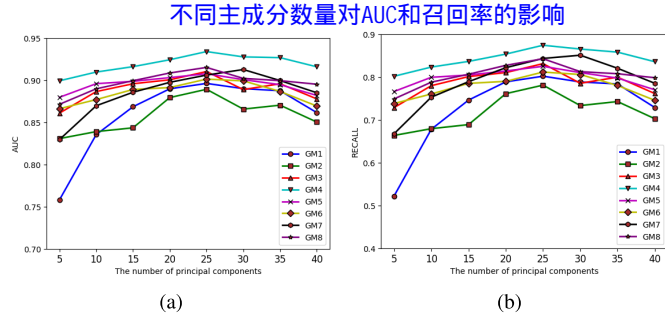

(a)   (b)

Fig. 6.   AUC and recall of the proposed method with different numbers of principal components.

the number of principal components is set to 25. While PCA's dimensionality reduction does result in some loss of information pertaining to oil spill features, it simultaneously removes noise and redundant data, thus mitigating the risk of overfitting. It also brings efficiency improvement. Furthermore, when the principal components exceed 25, no additional benefits are observed in terms of AUC and recall. In addition, due to the limited computing memory, we cannot directly process the whole feature band without PCA. On the other hand, despite these informational losses, ISLIC strengthens the effect of feature extraction from the spectral dimension to the spatial dimension. The incorporation of ISLIC features significantly boosts the performance of our proposed model, effectively countering the potential negative impacts of the PCA. This detailed analysis underlines the effectiveness of our combined feature extraction methodology in enhancing oil spill detection.

C. Parameter Settings

All experiments are implemented on a computer with NVIDIA A100 Tensor Core GPU. Deep learning models are implemented on the TensorFlow platform. The optimal parameter settings of comparison methods are set as follows.

1) *Logistic regression:* The penalty is set to l_1. The C is set to 10. The solver is set to liblinear.
2) *SVM:* The kernel is set to rbf. The C is set to 100. The gamma parameter is set to 0.1.
3) Random forest: The number of estimators is set to 200. The maximum depth is set to 20. The minimum number of samples contained in each leaf node is set to 2. The minimum number of samples specified for each nonleaf node is set to 2.

4) *1DCNN:* 1DCNN consists of three 1-D convolutional layers. The filters for these three layers are set to 32, 48, and 120. The kernel size is set to 1. The batch size of 1DCNN is set to 64. The learning rate and epoch are set to 0.001 and 300.
5) *2DCNN:* 2DCNN consists of three 2-D convolutional layers. The filters for these three layers are set to 32, 48, and 120. The kernel sizes are set to (2, 2). The batch size and patch size of 2DCNN are set to 64 and 5. The learning rate and epoch are set to 0.001 and 300.
6) *HybridSN:* HybridSN consists of three 3-D convolutional layers and one 2-D convolutional layer. The kernel sizes of these layers are set to (3, 3, 7), (3, 3, 5), (3, 3, 3), and (3, 3), respectively. The batch size of HybridSN is set to 256. The learning rate and epoch are set to 0.001 and 100. The weight parameters are optimized by Adam.
7) *CTN:* The batch size and patch size of CTN are set to 32 and 15. The learning rate and epoch are set to 0.001 and 300. The weight parameters are optimized by Adam.
8) *DMC:* DMC consists of the input convolution layer, the primary capsule layer, two convolutional capsule layers, and the class capsule layer. The kernel sizes of these layers are set to (5, 5), (1, 1), (3, 3), (3, 3), and (1, 1), respectively. The strides are set to 2, 1, 2, 1, and 1, respectively. The learning rate and epoch of DMC are set to 0.001 and 50.
9) *GSC-ViT:* The heads, depth, spatial size of groups, and number of groups of GSC-ViT are set to (3, 3, 3), (1, 1, 1), [3, 3, 3], and [16, 16, 16]. The batch size and patch size are set to 128 and 9. The learning rate and epoch are set to 0.001 and 200.
10) *MSSTT:* The split ratio of MSSTT is 1:9. The neighborhood sampling size are 3×3 and 7×7, respectively. The batch size and patch size are set to 64 and 5. The learning rate and epoch are set to 0.001 and 100.

The parameter settings of our method are as follows.

1) *PCA:* The number of dimensions after dimensionality reduction is set to 25.
2) *ISLIC:* The number of segments is set to 100. The standard deviation of the Gaussian filter is set to 5.
3) *SSFEN:* The filters for three 3-D convolutional layers are set to 12,24, and 48, the kernel sizes are set to (3,3,3), (1,1,3), and (1,1,3), and the strides are set to (1, 1, 1), (1, 1, 1), and (1, 1, 2). The dropout rate is 0.5. The patch size, batch size, epoch, and learning rate are set to 5, 64, 300, and 0.001, respectively.
4) *MAM:* The number of convolution kernels is set to 9. The kernel size is set to (5, 5, 5).

On each HSI, only 1% of the labeled samples are randomly selected as the training set, and the remaining samples are for testing. In the conducted experiments, each trial was executed a total of ten times. Subsequently, the mean values of the evaluation metrics (AUC and recall) are calculated and reported, thereby facilitating a comprehensive assessment of the detection performance.

TABLE II
AUC RESULTS OBTAINED BY ALL DETECTION METHODS

AUC	LR	SVM	RF	1DCNN	2DCNN	HybridSN	CTN	DMC	GSC-ViT	MSSTT	Ours
GM1	0.7882	0.8104	0.8025	0.8398	0.8620	0.8683	0.8350	0.8720	0.8429	0.8616	**0.8963**
GM2	0.7914	0.8019	0.7853	0.8242	0.8502	0.8386	0.7689	0.8863	0.8356	0.8436	**0.8893**
GM3	0.8677	0.8631	0.8588	0.8605	0.8795	0.8837	0.8118	0.8780	0.8555	0.8728	**0.9104**
GM4	0.8846	0.8928	0.8821	0.8996	0.9076	0.8988	0.8552	0.8859	0.8997	0.8984	**0.9341**
GM5	0.8414	0.8476	0.8561	0.8558	0.8797	0.8757	0.7952	0.9035	0.8834	0.8737	**0.9078**
GM6	0.8454	0.8569	0.8612	0.8553	0.8748	0.8721	0.8069	0.8635	0.8700	0.8725	**0.9015**
GM7	0.8343	0.8335	0.8129	0.8257	0.8562	0.9110	0.8890	**0.9207**	0.8547	0.8519	0.9062
GM8	0.8732	0.8802	0.8799	0.8802	0.8842	0.8849	0.8544	0.8947	0.8603	0.8799	**0.9153**
Mean	0.8408	0.8483	0.8424	0.8551	0.8743	0.8791	0.8270	0.8881	0.8628	0.8693	**0.9076**

TABLE III
RECALL RESULTS OBTAINED BY ALL DETECTION METHODS

Recall	LR	SVM	RF	1DCNN	2DCNN	HybridSN	CTN	DMC	GSC-ViT	MSSTT	Ours
GM1	0.5804	0.6251	0.6094	0.6887	0.7305	0.7447	0.6794	0.7496	0.6950	0.7317	**0.8022**
GM2	0.5843	0.6052	0.5719	0.6513	0.7026	0.6794	0.5401	0.7750	0.6739	0.6892	**0.7813**
GM3	0.7412	0.7314	0.7225	0.7312	0.7669	0.7756	0.6329	0.7636	0.7191	0.7538	**0.8315**
GM4	0.7732	0.7897	0.7681	0.8066	0.8203	0.8021	0.7171	0.7769	0.8048	0.8016	**0.8745**
GM5	0.6879	0.6998	0.7183	0.7205	0.7674	0.7593	0.5972	0.8172	0.7758	0.7553	**0.8253**
GM6	0.6946	0.7180	0.7268	0.7189	0.7562	0.7512	0.6234	0.7328	0.7480	0.7532	**0.8119**
GM7	0.6847	0.6833	0.6390	0.6813	0.7373	0.8394	0.7969	**0.8586**	0.7302	0.7293	0.8436
GM8	0.7526	0.7671	0.7674	0.7728	0.7775	0.7787	0.7185	0.7991	0.7300	0.7694	**0.8430**
Mean	0.6873	0.7025	0.6904	0.7214	0.7573	0.7663	0.6632	0.7841	0.7346	0.7479	**0.8266**

D. Experimental Results

Tables II and III present the AUC and recall results for all detection methods. It can be clearly seen that LR achieves the worst objective results because LR is prone to underfitting and has insufficient adaptability to complex data. For the other two traditional machine learning algorithms, SVM and RF, their detection performance is slightly better than LR, but still not good enough. This is because traditional machine learning algorithms only utilize spectral information and ignore spatial information, and can only extract shallow features. Compared to traditional machine learning algorithms, deep learning algorithms can obtain more discriminative deep features. It can be clearly seen that 1DCNN has achieved objective results superior to LR, SVM, and RF on all datasets. However, 1DCNN also fails to utilize spatial information, which reduces its detection performance. 2DCNN can extract features in local domains and effectively extract spatial information of HSIs. The experimental results indicate that 2DCNN outperforms 1DCNN in both AUC and recall evaluation metrics. However, deep networks based solely on 1-D or 2-D cannot simultaneously extract spectral and spatial information, while HybridSN combines 3-D and 2-D convolutions to fully extract spectral–spatial features. Especially on GM7, HybridSN achieved very good results, and in terms of average AUC and Recall, HybridSN is slightly better than 2DCNN. Compared with HybridSN, this article chooses to use 3-D convolution alone, combined with improved superpixel segmentation and MAM, and obtains the best results on seven images. Compared with CTN, DMC, GSC-ViT, and MSSTT, our method is also superior to these four new methods in terms of average AUC and recall, only inferior to the DMC method in GM7. Specifically, on GM1, GM3, GM4, and GM6, compared to the other ten methods, our method improved at least 2.4% and 5% in AUC and recall evaluation metrics, respectively.

By comparing the final results in Fig. 7, it can be found that traditional machine learning detection methods cannot effectively identify oil spill areas, even in scenarios where the oil spill distribution is relatively simple, such as GM2 and GM6. This is because they only extracted shallow features and did not utilize the rich spatial information in HSIs. In contrast, 1DCNN has slightly improved the detection performance, but there are still many undetected oil spills in some images, especially in scenes with complex oil spill distributions such as GM1, GM5, and GM7. The detection performance of 2DCNN and HybridSN on all images is superior to that of 1DCNN, and the detection results are also quite similar because they fully utilize the spatial information of HSIs. Although CTN combines convolution and transformer structures, its detection performance on the dataset in this article is not very good, especially on GM2, GM3, GM5, and GM6, where there are many undetected oil spill areas. The detection performance of GSC-ViT is better than CTN, but there are still many undetected oil spill areas on GM1, GM2, and GM3. MSSTT achieves acceptable detection results, with better detection performance than CTN and GSC-ViT. Compared with other comparison methods, DMC has better detection performance and can identify oil spill areas well, especially on GM7. However, overall, DMC is still inferior to our proposed method, especially in GM3, GM4, and GM6, where DMC has significantly more undetected oil spill areas.

It can be seen that compared with other detection methods, our method can effectively identify oil spill areas in all images, regardless of whether it is scenes with relatively simple oil spill distributions such as GM2, GM4, and GM6 or complex scenes such as GM1, GM3, GM5, and GM8, achieving the
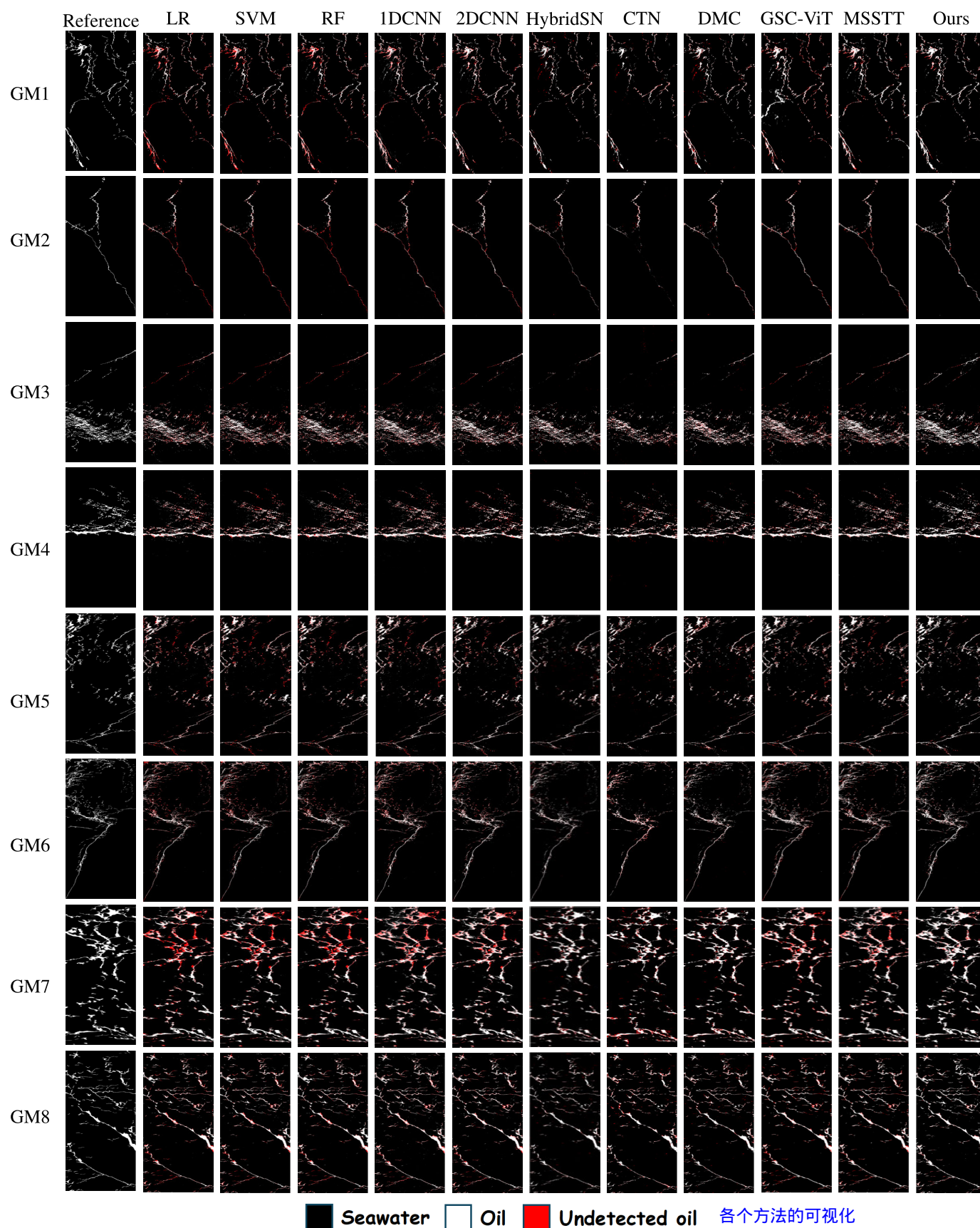
Fig. 7.   Reference maps and the detection maps of all detection methods on eight HSIs from GM1 to GM8.

TABLE IV
RUNNING TIME OF ALL DETECTION METHODS

Time(s)	LR	SVM	RF	1DCNN	2DCNN	HybridSN	CTN	DMC	GSC-ViT	MSSTT	Ours
GM1	**0.24**	19.00	19.43	175.93	119.88	59.11	251.32	995.12	186.80	165.81	199.10
GM2	**0.33**	12.25	25.66	267.98	175.39	25.54	487.93	2663.41	274.23	244.77	308.69
GM3	**0.31**	20.78	20.66	205.21	136.34	50.68	386.79	3453.66	211.73	184.76	233.78
GM4	**0.35**	14.94	20.25	210.72	146.63	46.78	338.82	819.51	218.17	192.42	236.45
GM5	**0.36**	47.79	35.20	302.01	207.68	33.32	553.53	1404.88	312.24	293.51	336.01
GM6	**0.41**	44.35	30.79	306.05	207.02	38.51	566.80	2897.56	314.98	301.12	355.69
GM7	**0.30**	59.42	23.31	201.58	145.50	34.70	442.21	775.61	212.69	195.16	238.29
GM8	**0.28**	18.70	18.69	184.50	130.39	21.97	285.27	4551.22	187.36	170.87	210.14
Mean	**0.32**	29.65	24.25	231.75	158.61	38.83	414.08	2195.12	239.78	218.55	264.77

best detection results. The good detection performance of our method is mainly due to the following three reasons: First, we have improved the SLIC algorithm by smoothing to reduce noise in the image and achieve preliminary extraction of spatial feature information. Second, SSFEN is used as the detection network to extract both spectral and spatial information from HSIs. Finally, spectral attention and spatial attention modules are added to enhance the network's attention to important features and further improve its detection performance.

Table IV shows the running time of all detection methods. It can be observed that LR performs the fastest. This is because the principle of the LR algorithm is relatively simple and easy to implement. Relatively speaking, the running time of traditional methods is faster than that of deep learning methods. Compared with other deep learning methods, our method runs slower than 1DNN and 2DCNN, and HybridSN also runs faster than our method, because the use of hybrid CNNs reduces the complexity of the model [35]. GSC-ViT and MSSTT also run slightly faster than our proposed method. Compared with CTN and DMC, our method has a faster running time. Especially for DMC, its running time is the slowest. Overall, the running time of our proposed method is acceptable. Reducing the running time of the model is also an important topic, but it is not the focus of our research.

E. Ablation Experiments

To verify the effectiveness of GF and PIST in ISLIC, we conducted the ablation experiment. Table V shows the AUC and recall results of the ablation experiment of GF and PIST. It can be seen that both GF and PIST can improve the detection performance of the model. Specifically, after adding GF, the average AUC and average recall of the model increased by 0.81% and 1.67%, respectively, while after adding PIST, the average AUC and average recall increased by 2.41% and 5.01%, respectively. It can be found that the improvement effect of adding PIST alone is better than GF. Finally, after adding both GF and PIST, the average AUC and average recall of the detection network achieve their highest values in most images, which verifies the effectiveness of GF and PIST.

In addition, to verify the effectiveness of ISLIC and MAM in the proposed detection method, we conducted another ablation experiment. Table VI shows the AUC and recall results of the ablation experiment of ISLIC and MAM. It can be seen that the addition of ISLIC improves the detection performance on all images, with an increase of 1.47% and 2.97% on average

TABLE V
ABLATION STUDY OF GF AND PIST

	GF	×	✓	×	✓
	PIST	×	×	✓	✓
GM1	AUC	0.8681	0.8804	0.8948	**0.8963**
	Recall	0.7436	0.7686	0.7998	**0.8022**
GM2	AUC	0.8565	0.8641	0.8844	**0.8893**
	Recall	0.7153	0.7302	0.7713	**0.7813**
GM3	AUC	0.8827	0.8881	0.9074	**0.9104**
	Recall	0.7747	0.7846	0.8244	**0.8315**
GM4	AUC	0.9086	0.9145	0.9312	**0.9341**
	Recall	0.8222	0.8336	0.8681	**0.8745**
GM5	AUC	0.8838	0.8953	0.9069	**0.9078**
	Recall	0.7752	0.7999	0.8224	**0.8253**
GM6	AUC	0.8599	0.8830	0.8931	**0.9015**
	Recall	0.7256	0.7741	0.7935	**0.8119**
GM7	AUC	0.8891	0.8780	**0.9080**	0.9062
	Recall	0.7986	0.7756	**0.8449**	0.8436
GM8	AUC	0.8992	0.9091	0.9149	**0.9153**
	Recall	0.8070	0.8294	0.8391	**0.8430**
Mean	AUC	0.8810	0.8891	0.9051	**0.9076**
	Recall	0.7703	0.7870	0.8204	**0.8266**

TABLE VI
ABLATION STUDY OF ISLIC AND MAM

	ISLIC	×	✓	×	✓
	MAM	×	×	✓	✓
GM1	AUC	0.8750	0.8861	0.8823	**0.8963**
	Recall	0.7584	0.7814	0.7739	**0.8022**
GM2	AUC	0.8503	0.8734	0.8664	**0.8893**
	Recall	0.7028	0.7493	0.7359	**0.7813**
GM3	AUC	0.8863	0.9012	0.8991	**0.9104**
	Recall	0.7809	0.8121	0.8087	**0.8315**
GM4	AUC	0.9048	0.9221	0.9198	**0.9341**
	Recall	0.8143	0.8495	0.8454	**0.8745**
GM5	AUC	0.8795	0.8930	0.8913	**0.9078**
	Recall	0.7664	0.7942	0.7910	**0.8253**
GM6	AUC	0.8835	0.8933	0.8901	**0.9015**
	Recall	0.7746	0.7949	0.7883	**0.8119**
GM7	AUC	0.8821	0.8958	0.8926	**0.9062**
	Recall	0.7961	0.8203	0.8203	**0.8436**
GM8	AUC	0.8915	0.9057	0.8998	**0.9153**
	Recall	0.7930	0.8226	0.8105	**0.8430**
Mean	AUC	0.8816	0.8963	0.8927	**0.9076**
	Recall	0.7733	0.8030	0.7968	**0.8266**

AUC and average recall, respectively, proving the effectiveness of ISLIC. Similarly, after adding the MAM, the average AUC and average recall increased by 1.11% and 2.35%, respectively, which also proves the effectiveness of MAM. Comparing

ISLIC	×	✓	×	✓
MAM	×	×	✓	✓
GM1	**87.07**	89.10	189.05	199.10
GM2	**129.77**	131.49	291.20	308.69
GM3	**99.36**	101.35	221.55	233.78
GM4	**105.82**	108.09	232.40	236.45
GM5	**142.26**	144.31	322.29	336.01
GM6	**143.83**	146.07	325.26	355.69
GM7	**101.09**	103.34	222.86	238.29
GM8	**91.67**	92.12	249.70	210.14
Mean	**112.61**	114.48	256.79	264.77

ISLIC and MAM separately, it can be found that ISLIC has a greater improvement effect on model detection performance. Finally, it can be seen that the detection effect of adding ISLIC and MAM simultaneously is better than that of ISLIC or MAM alone. Compared with the benchmark model, after adding ISLIC and MAM, the average AUC and average recall have been improved by 2.60% and 5.33%, respectively, which fully demonstrates the effectiveness of ISLIC and MAM.

Table VII shows the running time of ablation experiments. It can be seen that adding ISLIC or MAM increases the running time of the model. ISLIC increased the running time by very little, averaging less than 2 s, while MAM increased the running time by more than twice because MAM increases the complexity of the model. Considering the improvement of detection performance by these two modules, we think that the increased running time is acceptable.

IV. CONCLUSION

In this article, we present an efficient oil spill detection method HMOSDN integrating an MAM and superpixel segmentation. Remarkably, this method demonstrates excellent detection results using only 1% of training samples. The approach offers several distinct advantages. First, the proposed method initially extracts the spatial features from images by utilizing a superpixel segmentation algorithm. Second, Gaussian filtering is applied to smooth the images, and a PIST is utilized to diminish noise, enhancing the quality of the input features. Third, Post the first convolutional layer, we incorporate spectral and spatial attention modules in the network. These modules are specifically designed to refine the convolutional features, thereby significantly boosting the detection performance of the system. The experimental results show that our proposed detection method outperforms other comparative methods overall regarding AUC and recall evaluation metrics, even if the number of training samples is very small. In future work, we hope to further improve the oil spill detection framework, such as adding timing characteristics of spectral information and adopting the detection framework based on the new network.

REFERENCES

[1] P. Li, Q. Cai, W. Lin, B. Chen, and B. Zhang, "Offshore oil spill response practices and emerging challenges," *Mar. Pollut. Bull.*, vol. 110, no. 1, pp. 6–27, Sep. 2016.

[2] J. Chen, W. Zhang, Z. Wan, S. Li, T. Huang, and Y. Fei, "Oil spills from global tankers: Status review and future governance," *J. Cleaner Prod.*, vol. 227, pp. 20–32, Aug. 2019.

[3] Y. Li, H. Lu, Z. Zhang, and P. Liu, "A novel nonlinear hyperspectral unmixing approach for images of oil spills at sea," *Int. J. Remote Sens.*, vol. 41, no. 12, pp. 4684–4701, Jun. 2020.

[4] P. Duan, Z. Xie, X. Kang, and S. Li, "Self-supervised learning-based oil spill detection of hyperspectral images," *Sci. China Technolog. Sci.*, vol. 65, no. 4, pp. 793–801, Apr. 2022.

[5] X. Kang, B. Deng, P. Duan, X. Wei, and S. Li, "Self-supervised spectral–spatial transformer network for hyperspectral oil spill mapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5507410.

[6] U. Bhangale, S. S. Durbha, R. L. King, N. H. Younan, and R. Vatsavai, "High performance GPU computing based approaches for oil spill detection from multi-temporal remote sensing data," *Remote Sens. Environ.*, vol. 202, pp. 28–44, Dec. 2017.

[7] L. Bing, Q.-G. Xing, X. Liu, and N.-N. Zou, "Spatial distribution characteristics of oil spills in the Bohai Sea based on satellite remote sensing and GIS," *J. Coastal Res.*, vol. 90, pp. 164–170, Sep. 2019.

[8] M. Fingas and C. Brown, "Review of oil spill remote sensing," *Marine Pollution Bull.*, vol. 83, no. 1, pp. 9–23, 2014.

[9] F. Bandiera, A. Masciullo, and G. Ricci, "A Bayesian approach to oil slicks edge detection based on SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 5, pp. 2901–2909, May 2014.

[10] G. Cheng, G. Wang, and J. Han, "ISNet: Towards improving separability for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5623811.

[11] A. Buono, F. Nunziata, M. Migliaccio, and X. Li, "Polarimetric analysis of compact-polarimetry SAR architectures for sea oil slick observation," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 5862–5874, Oct. 2016.

[12] K. Ren, W. Sun, G. Yang, X. Meng, J. Peng, and H. Li, "Multistage hybrid denoising network for satellite hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5534717.

[13] L. Gao, D. Wang, L. Zhuang, X. Sun, M. Huang, and A. Plaza, "BS³LNet: A new blind-spot self-supervised learning network for hyperspectral anomaly detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5504218.

[14] L. Dabbiru, S. Samiappan, R. A. A. Nobrega, J. A. Aanstoos, N. H. Younan, and R. J. Moorhead, "Fusion of synthetic aperture radar and hyperspectral imagery to detect impacts of oil spill in Gulf of Mexico," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 1901–1904.

[15] B. Rasti, P. Ghamisi, and J. Chanussot, "Mixed noise reduction in hyperspectral imagery," in *Proc. 10th Workshop Hyperspectral Imag. Signal Process., Evol. Remote Sens. (WHISPERS)*, Sep. 2019, pp. 1–4.

[16] X. Kang, Z. Wang, P. Duan, and X. Wei, "The potential of hyperspectral image classification for oil spill mapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5538415.

[17] S. Dongmei et al., "Classification of the different thickness of the oil film based on wavelet transform spectrum information," *Aquatic Proc.*, vol. 3, pp. 133–143, Mar. 2015.

[18] D. Liu, J. Zhang, and X. Wang, "Reference spectral signature selection using density-based cluster for automatic oil spill detection in hyperspectral images," *Opt. Exp.*, vol. 24, no. 7, pp. 7411–7425, 2016.

[19] P. Duan, J. Lai, J. Kang, X. Kang, P. Ghamisi, and S. Li, "Texture-aware total variation-based removal of sun glint in hyperspectral images," *ISPRS J. Photogramm. Remote Sens.*, vol. 166, pp. 359–372, Aug. 2020.

[20] J. Xu, X. Pan, B. Jia, X. Wu, P. Liu, and B. Li, "Oil spill detection using LBP feature and K-means clustering in shipborne radar image," *J. Mar. Sci. Eng.*, vol. 9, no. 1, p. 65, Jan. 2021.

[21] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, no. 1, pp. 1–12, 2015.

[22] Y. Ji et al., "Domain adaptive and interactive differential attention network for remote sensing image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5616316.

[23] X. Zhu, Y. Li, Q. Zhang, and B. Liu, "Oil film classification using deep learning-based hyperspectral remote sensing technology," *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 4, p. 181, Apr. 2019.

[24] W. Zhao and S. Du, "Spectral–spatial feature extraction for hyperspectral image classification: A dimension reduction and deep learning approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 8, pp. 4544–4554, Aug. 2016.

[25] 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.

[26] B. Wang et al., "A spectral–spatial features integrated network for hyperspectral detection of marine oil spill," *Remote Sens.*, vol. 13, no. 8, p. 1568, Apr. 2021.

[27] Y. Duan, F. Luo, M. Fu, Y. Niu, and X. Gong, "Classification via structure-preserved hypergraph convolution network for hyperspectral image," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5507113.

[28] S. K. Roy, G. Krishna, S. R. Dubey, and B. B. Chaudhuri, "HybridSN: Exploring 3-D–2-D CNN feature hierarchy for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 2, pp. 277–281, Feb. 2019.

[29] S. Jia et al., "Multiple feature-based superpixel-level decision fusion for hyperspectral and LiDAR data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 2, pp. 1437–1452, Feb. 2021.

[30] S. Jia et al., "A multiscale superpixel-level group clustering framework for hyperspectral band selection," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5523418.

[31] Z. Zhao, D. Hu, H. Wang, and X. Yu, "Convolutional transformer network for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[32] L. Sun, G. Zhao, Y. Zheng, and Z. Wu, "Spectral–spatial feature tokenization transformer for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5522214.

[33] Z. Zhao, X. Xu, S. Li, and A. Plaza, "Hyperspectral image classification using groupwise separable convolutional vision transformer network," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5511817.

[34] Z. Meng, T. Zhang, F. Zhao, G. Chen, and M. Liang, "Multiscale super token transformer for hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 21, pp. 1–5, 2024.

[35] A. Ravikumar, P. N. Rohit, M. K. Nair, and V. Bhatia, "Hyperspectral image classification using deep matrix capsules," in *Proc. Int. Conf. Data Sci., Agents Artif. Intell. (ICDSAAI)*, vol. 1, Dec. 2022, pp. 1–7.

[36] W. Liu et al., "Self-supervised feature learning based on spectral masking for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 4407715.

[37] P. Duan, X. Kang, P. Ghamisi, and S. Li, "Hyperspectral remote sensing benchmark database for oil spill detection with an isolation forest-guided unsupervised detector," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5509711.

[38] K. Cui et al., "Superpixel-based and spatially regularized diffusion learning for unsupervised hyperspectral image clustering," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 4405818.

[39] B. Yang, W. Sun, and J. Peng, "SAGN: Sharpening-aware graph network for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5518812.

[40] B. Yang, S. Pan, W. Sun, Z. Guo, Z. Ye, and J. Peng, "FCFDA: Fine–coarse–fine progressive graph framework with distribution alignment for hyperspectral image change detection," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, 2024, Art. no. 5538714.

[41] Y. Wang et al., "SpectralKAN: Kolmogorov–Arnold network for hyperspectral images change detection," 2024, *arXiv:2407.00949*.

[42] X. Ren and J. Malik, "Learning a classification model for segmentation," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, vol. 1, Oct. 2003, pp. 10–17.

[43] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[44] Y. Li, H. Zhang, and Q. Shen, "Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.*, vol. 9, no. 1, p. 67, Jan. 2017.

[45] X. Liu, Q. Sun, Y. Meng, C. Wang, and M. Fu, "Feature extraction and classification of hyperspectral image based on 3D- convolution neural network," in *Proc. IEEE 7th Data Driven Control Learn. Syst. Conf. (DDCLS)*, May 2018, pp. 918–922.

[46] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 3–19.

[47] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2005.

[48] N. Chinchor, "MUC-4 evaluation metrics," in *Proc. 4th Conf. Message Understand.*, 1992, p. 22.

Yuqing Zhang is currently pursuing the Ph.D. degree in science with Macau University of Science and Technology, Macau, China.

His research interests include deep learning and hyperspectral image processing.

Zhijing Ye received the Ph.D. degree in mathematics from Huazhong University of Science and Technology, Wuhan, China, in 2016.

He is currently an Assistant Professor with the Faculty of Innovation Engineering, Macau University of Science and Technology, Macau, China. His research interests include machine learning, pattern recognition, and hyperspectral image processing.
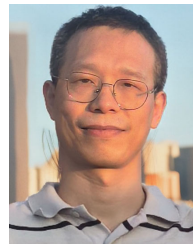

Chengyong Zheng received the Ph.D. degree in control science and engineering from Huazhong University of Science and Technology, Wuhan, China, in 2015.

He is currently an Associate Professor with Wuyi University, Jiangmen, China. His research interests include hyperspectral image analysis, computer vision, and pattern recognition.

Bing Yang received the Ph.D. degree in statistics from Huazhong University of Science and Technology, Wuhan, China, in 2021.

He is currently a Lecturer with the College of Sciences, China Jiliang University, Hangzhou, China. His research interests include hyperspectral image processing and deep learning.

Jiangtao Peng (Senior Member, IEEE) received the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2011.

He is currently a Professor with the Faculty of Mathematics and Statistics, Hubei University. His research interests include machine learning and hyperspectral image processing.
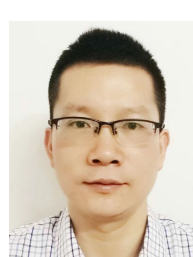
Weiwei Sun (Senior Member, IEEE) received the Ph.D. degree in cartography and geographic information engineering from Tongji University, Shanghai, China, in 2013.

He is currently a Full Professor with Ningbo University, Ningbo, Zhejiang, China. His research interests include hyperspectral image processing, anomaly detection, and target recognition.